Installing and Configuring Pascal Workstation Clients

1. Introduction

This manual tells how to install the Pascal Workstation System on any HP 9000 Series 200 or 300 computer to be a client on an SRM/UX system. It also describes how to establish contact with an SRM/UX server from the client workstation.

A major part of Pascal Workstation System start-up involves creating an appropriate directory structure on the SRM/UX file system. Much of this is done automatically by the SRM/UX installation process when you load the SRM/UX tape, but you must manipulate certain files to customize your SRM/UX Pascal Workstation System clients.

How you approach the client workstation start-up procedure depends on how tasks are divided between the SRM/UX System Administrator and the individual client workstation user, as well as the degree to which each client system will be customized. System customization involves modifying system boot files; modifying the TABLE and INITLIB files; and/or creating individual autostart files for each workstation.

Before You Start

Before you begin the Pascal Workstation system start-up:

- ensure all of the hardware components of your SRM/UX system are properly installed
- ensure that the SRM/UX system administrator has properly installed, configured, and started running all of the software components of the SRM/UX server
- ensure that the SRM/UX system administrator has made the appropriate entries in the server's /etc/srmdconf file for your particular workstation. See chapter 4 of this manual for server configuration details.

What You Need for Start-up

- the media which holds the Pascal Workstation system software
- the Pascal Workstation system manual set, for more detailed information about the start-up options mentioned in this chapter

2. Booting Pascal and Establishing Access to the SRM/UX System

For complete instructions on booting the Pascal Workstation system, refer to the manuals: *Pascal 3.X Workstation system Volume 1 and 2* for whichever version (for example: 3.2 replaces 3.X)you are installing. There you will learn how to install the Pascal Workstation system from the flexible discs shipped with the system.

The search rules for these files are NOT the same as for HP BASIC. The path searched first is /WORKSTATIONS/SYSTEMnnnnnn, where nnnnnn is the last 6 hex digits of the Pascal Workstation system client LAN card Link address.

Note

Any alpha characters MUST be capitalized in the name of this directory; for example: /WORKSTATIONS/SYSTEM012ABC is correct, but WORKSTATIONS/SYSTEM012abc is not.

If /WORKSTATIONS/SYSTEMnnnnnn does not exist, the path /WORKSTATIONS/SYSTEM is searched for the three extension boot files. If neither of these paths exist, or if the INITLIB file is not found, the boot will fail with an error: -10. If the STARTUP or TABLE files are not found, booting will be completed, but the Pascal Workstation system client system will be practically unusable. Both of these paths are relative to the root directory for volume 8, as described in the /etc/srmdconf section in chapter 4.

The names of the INITLIB, STARTUP and TABLE programs should follow the standard rules, based on the name of the SYSTEM_P file. For example, if the system file has the name SYSP33, the other files should be named INITP33, STARTP33, and TABLEP33.

Once booting is complete and TABLE has executed, rbootd is no longer used by the Pascal Workstation system client. Srmd on HP-UX and the SRM driver in the Pascal Workstation system INITLIB file take over. If !!srmd!!(1M) is not running or the proper modules are not in INITLIB, the client Pascal Workstation system will be unusable.

To establish access to the SRM/UX system from a running Pascal Workstation system, that did NOT boot from the SRM/UX server, do the following:

- execute the required driver files: DATA_COMM and SRM (for the SRM connection) or DATA_COMM, LAN, and SRM (for the LAN connection)

- manually reexecute the Pascal Workstation system's auto-configuration program: TABLE

Once the proper driver modules have been executed, the auto-configuration program can recognize the SRM/UX system.

Note

The following instructions assume that the client workstation has its own flexible-disc drive.

Table 7-1. To Boot the Pascal Workstation system:

Step	Action	Result	Explanation or Note
1	Ensure that the workstation is off		
2	If the workstation has an attached disc drive, turn it on		
3	If the workstation is a Series 200, insert the disc labeled BOOT: into the drive with the lowest unit number. If the workstation is a Series 300, insert the disc labeled BOOT2: into the drive with the lowest unit number.		To determine which drive has the lowest unit number, refer to the disc drive's operator's guide.
4	If the workstation has a dual lexible disc drive, insert the disc labeled SYSVOL: into the other available drive. (The SYSVOL: disc must be write-enabled)		
5	Turn on the workstation	The workstation loads the Pascal Workstation system (if booting from the server unattended; if you use attended booting, you should select the SYSTEM_P boot file from the Boot ROM menu)and then displays the prompt: **New system date?** If, instead, you get the prompt: **Please put SYSVOL in unit #3 and press the x key...** then replace the boot disc with the disc labeled SYSVOL: and press the [x] key. (The SYSVOL: disc must be write-enabled)	
6	Enter the correct date and time by typing over the displayed date and time. Then press [Return]	The Pascal Workstation system displays the Main Command Level prompt line	Note that the time is recorded as 24-hour clock time (i.e., 1:00 p.m. is 13:00, 2:00 p.m. is 14:00, etc.)

Once you have booted the Pascal system, follow these steps to establish access to the SRM/UX system:

TABLE 7-2. To Establish Access to the SRM/UX system:

Step	Action	Result	Explanation or Note
1	If your Pascal Workstation system is on double-sided microdiskettes,replace the SYSVOL: disc with the disc labeled ACCESS:. Otherwise, replace the SYSVOL: disc with the disc labeled CONFIG:.		The ACCESS: or CONFIG: disc contains the driver modules required to access the SRM/UX system
2	Press [x]	The Pascal Workstation system displays: **Execute what file?**	To execute a file
3	If you are running over the SRM connection, do step 3 (this step)and skip step 4. If you have double-sided microdiskettes, type (including the trailing period): **ACCESS:DATA_COMM.** **[Return]** Otherwise, type (including the trailing period): **CONFIG:DATA_COMM.** **[Return]**		To install the first required driver module, DATA_COMM. The required driver modules are programs that install themselves automatically when you execute them.
4	If you have double-sided microdiskettes, type (including the trailing period): **ACCESS:LAN.** **[Return]**		To install the required driver modules IOMPX and LANDVR.
5	Press [x]	The Pascal Workstation system displays: **Execute what file?**	To execute a file.
6	If you have double-sided microdiskettes, type (including the trailing period): **ACCESS:SRM.** **[Return]** Otherwise, type (including the trailing period): **CONFIG:SRM.** **[Return]**		To install the remaining required driver module, SRM.

7	If the workstation has only a single disc drive, replace the disc currently in the drive with the BOOT: or BOOT2: disc. Otherwise, ensure that the BOOT: or BOOT2: disc is still in one of the workstation's disc drives.		
8	Press [x]	The Pascal Workstation system displays: **Execute what file?**	To execute a file.
9	If the workstation is a Series 300, type (including the trailing period): **BOOT2:TABLE.** **[Return]** Otherwise, type (including the trailing period): **BOOT:TABLE.** **[Return]**		To manually execute the Pascal Workstation system's auto-configuration program, TABLE.
10	Replace the BOOT2: or BOOT: disc with the disc labeled ACCESS:.		
11	Press [F]	The Pascal Workstation system displays the Filer's command line.	To enter the Filer
12	Press [V]	The Filer displays a list of the volumes that the Pascal Workstation System recognizes. (See the following example display)	To see if the Pascal Workstation System recognizes the SRM/UX volume 8 at volume #5.
13	Continue with the next section on "Creating Directories on the SRM/UX System Disc.		

Example Pascal Filer Volumes Listing

```
      Volumes on-line:
        1   CONSOLE:
        2   SYSTERM:
        3 # ACCESS:
        5 # PRIMARY:
        6   PRINTER:
      Prefix is - PRIMARY:
```

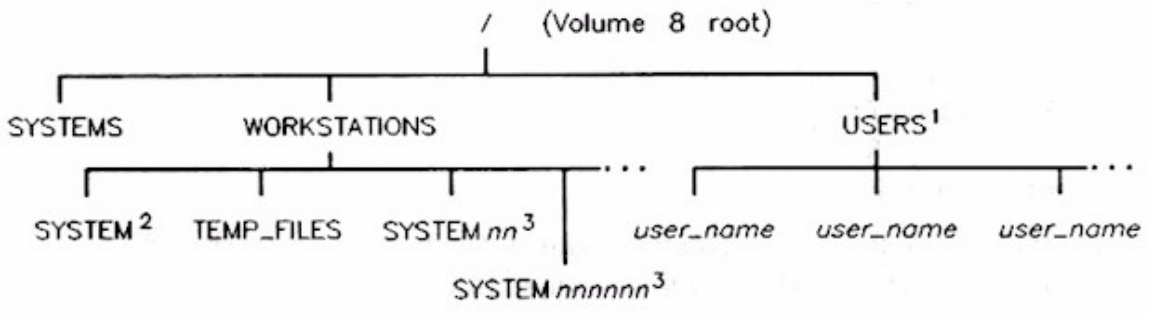
Notice that in the listing for Volume 5 and in the last line, you should see either PRIMARY or the "Volume Name" that the SRM/UX system administrator has assigned to your client in the server's /etc/srmdconf file.

TABLE 7-3. Troubleshooting Hints

If:	Then:
the volume name of your SRM/UX volume 8 is not in the display	try reexecuting the DATA_COMM, LAN, SRM, and TABLE programs
the Filer's Volumes command still does not recognize volume #5	check to see whether the SRM/UX hardware is properly configured and installed. For instance, the (unmodified) TABLE program expects the select code on the workstation's SRM interface to be at its factory setting: 21. Refer to "special config" for other select codes and host nodes.
neither of the two suggestions above work	Refer to the troubleshooting appendix (Appendix B) of this manual and/or ask your SRM/UX System Administrator for help.

The following illustration shows a directory structure suitable to support Pascal Workstation Systems on an SRM/UX system:

This structure should be placed below the Volume 8 Root directory as defined in the SRM/UX server's /etc/srmdconf file. If this structure is created by the Pascal Workstation system client user, it will get placed under the Volume 8 Root automatically.



[1] The USERS directory is optional but highly recommended.

[2] The SYSTEM directory is the default system directory, used by client workstations for which no dedicated system directory exists.

[3] *nn* and *nnnnnn* represent the two digit Node Address on SRM and the six hex character Link Level Address of the client workstation on LAN

Figure 7-1. SRM/UX Pascal Workstation System Directory Structure

TABLE 7-4. Create the Required Directory Structure on the Server's File System:

Step	Action	Result	Explanation or Note
1	Create the /WORKSTATIONS directory as follows:		
1a	Press [M]	The Filer displays: **Make file or directory? (F/D)**	To make a directory
1b	Press [D]	The Filer prompts: **Make what directory?**	To make a directory
1c	Type: **#5:/WORKSTATIONS [Return]**	The Filer prompts: **Directory is 'WORKSTATIONS' correct ? (Y/N)**	To specify the /WORKSTATIONS directory on volume #5 of the SRM/UX system disc
1d	Check the directory name and if it is correct, press [Y] If the directory name is wrong, press [[N]]and start over again with step 1a.	If you pressed [Y] the Filer creates the new directory and displays: **Directory WORKSTATIONS made** If you pressed [N] the Filer displays its command line and does *not* create the directory.	[Y] accepts the new directory. [N] rejects the new directory.
2	Make /WORKSTATIONS the default directory as follows:		To reduce the amount of typing involved in the next steps
2a	Press [P]	The Filer prompts: **Prefix to what directory?**	To specify a prefix directory path. The Filer prefixes every directory path with the current prefix (default) directory path.

2b	Type: **#5:/WORKSTATIONS [Return]**	The Filer displays: **Prefix is WORKSTATIONS:**	To specify the /WORKSTATIONS directory on volume #5 of the SRM/UX system disc
3	Within /WORKSTATIONS, the default directory, create the following directories (with names exactly as listed): **SYSTEM** **TEMP_FILES**		See Step 1 of this table for a detailed example of how to create a directory.
4	Press [P] and type: /	The Filer displays the root directory's name	To return to the root directory
5	Create the directory: **/USERS** Specify the directory name as you choose.		To create a /USERS directory on the SRM/UX file system. Within this directory, create a personal working directory for each workstation user.
6	Make the default directory (as specified in Step 5): **/USERS**		To reduce the amount of typing involved in the next step. See Step 2 for a detailed example of how to specify a prefix (default) directory.
7	Create a personal working directory for every workstation user. For example: john_public		Valid directory names are 1 to 14 characters long (1 to 16 characters long if the server administrator has converted to long file names-- see chapter 4 for details), and may have upper and lower case letters, digits, underscores (_), and periods (.).
8	Change to the default directory, i.e., the Volume 8 Root		
9	Continue creating additional directories below the SRM/UX root, as needed, e.g., UTILS		To create an organized directory structure on the SRM/UX server's file system
10	Continue with the next section on "Placing Pascal system files in the SYSTEM Directory."		

3. Placing Pascal Workstation System Files in the SYSTEM Directory

To have workstations:

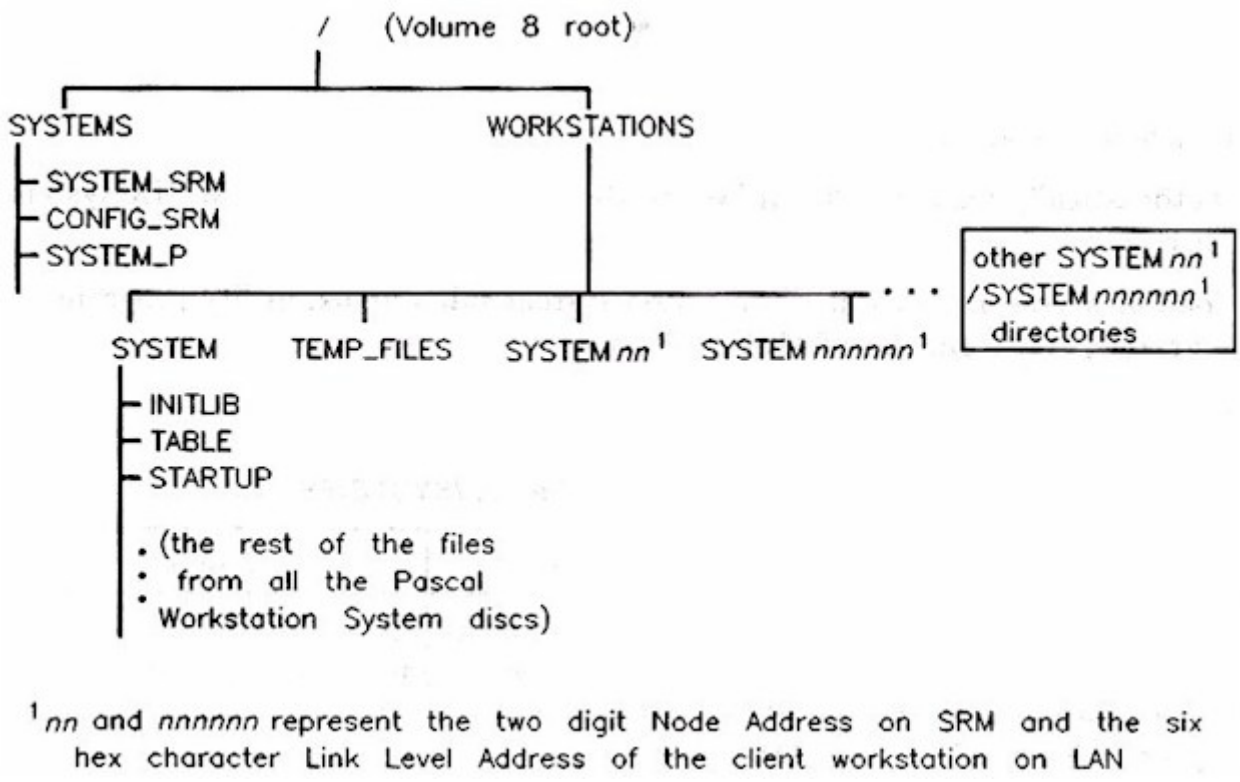
- automatically load the Pascal Workstation system from the SRM/UX system disc, and
-
- load or access any Pascal Workstation system subsystems, utility programs, libraries, etc. from the SRM/UX file system,

use one of the following methods:

TABLE 7-5. Place system files in /SYSTEMS

Method	Requirement(s)	Result
To boot in unattended mode, let each workstation's BootROM automatically boot the first system file in the /SYSTEMS directory.	• The Pascal system boot file (SYSTEM_P) must be the first bootable system file[1] in /etc/boottab. • The rest of the Pascal system files must be in the /WORKSTATIONS/SYSTEM directory.	All workstations automatically boot the Pascal system.
To boot in attended mode, repeatedly depress or hold down the [space] key after a SYSBOOT or cycle power until you get the BootROM menu of bootable system files to appear on the right-hand side of the screen. Then enter the characters to select the SYSTEM_P system boot file.	The Pascal system boot file (SYSTEM_P) must be in the /WORKSTATIONS/SYSTEM directory, along with the rest of the Pascal system files.	If each workstation user selects the same SYSTEM_P file, then they will all be configured with the same Pascal system.

1. A bootable system file is a system file whose name begins with SYS.



[1] *nn* and *nnnnnn* represent the two digit Node Address on SRM and the six hex character Link Level Address of the client workstation on LAN

Figure 7-2. Required Location of Pascal Workstation System Files

To place Pascal system files in the /WORKSTATIONS/SYSTEM directory on the SRM/UX file system from the client, follow these steps:

TABLE 7-6. System Files in /WORKSTATIONS/SYSTEM on Server's Volume 8:

Step	Action	Result	Explanation or Note
1	Make /WORKSTATIONS/SYSTEM the default directory as follows:		To reduce the amount of typing involved in the next steps
1a	Press **[P]**	The Filer prompts: **Prefix to what directory ?**	To specify a prefix directory path. The Filer prefixes every directory path with the current prefix (default) directory path.
1b	Type: **/WORKSTATIONS/SYSTEM [Return]**	The Filer displays: **Prefix is SYSTEM:**	To specify the /WORKSTATIONS/SYSTEM directory on the SRM/UX volume 8
2	Replace the ACCESS: disc in the workstation's disc drive with the disc labeled BOOT:.		All of the files on the BOOT: disc are the same as those on the BOOT2: disc, except for the INITLIB file.
3	Copy all of the BOOT: disc's files except SYSTEM_P and INITLIB to the default directory /WORKSTATIONS/SYSTEM on the SRM/UX volume 8 as follows:		Later you will be copying SYSTEM_P to the /SYSTEMS directory on the SRM/UX volume 8, and modifying the INITLIB files from the BOOT: and BOOT2: discs as you place them on the SRM/UX volume 8.
3a	Press **[F]**	The Filer prompts: **filecopy what file ?**	To copy one or more files
3b	Type: **BOOT:?,$ [Return]**	The Filer lists the first file on the disc and asks if you want to copy it	The ? is a wildcard that tells the Filer to individually list each file and ask you if you want to copy it from the disc. The $ is a wildcard that tells the Filer to give the copy of the file the same name as the original. The comma separates the source file specification from the destination file specification.
3c	When the Filer displays the file names SYSTEM_P and INITLIB, press **[N]**. For every other file, press **[Y]**.	The Filer reads and then writes each file you respond "yes" to.	To respond "yes" to every file-copy except SYSTEM_P and INITLIB.

4	Copy SYSTEM_P to the /SYSTEMS directory on the SRM/UX volume 8 as follows:		SYSTEM_P is the Pascal Workstation system boot file
4a	Press **[F]**	The Filer prompts: **filecopy what file ?**	To copy a file
4b	Type: **BOOT:SYSTEM_P,/SYSTEMS/$ [Return]**	The Filer copies the file	The $ is a wildcard that tells the Filer to give the copy of the file the same name as the original. The comma separates the source file specification from the destination file specification.
5	Replace the disc in the workstation's disc drive with a Pascal Workstation system disc other than BOOT: and BOOT2:		
6	Copy all of the files on the new disc to the default directory, /WORKSTATIONS/SYSTEM on the SRM/UX volume 8, as follows:		
6a	Press **[F]**	The Filer prompts: **filecopy what file ?**	To copy one or more files
6b	Type: **disc_name:=,$ [Return]** where disc_name is the name of the new disc (including the colon after the name).	The Filer copies the files on the new disc one after the other.	The = is a wildcard that stands for any combination of characters and it's use here causes the Filer to copy every file on the new disc, without prompting. **Example** If you inserted the disc labeled ACCESS: into the workstation's disc drive, you would type: **ACCESS:=,$ [Return]**
7	Repeat steps 5 and 6 for every remaining Pascal Workstation system disc (for example, SYSVOL: , CMP: , ASM: ,etc.)		To copy all of the Pascal Workstation system files to the /WORKSTATIONS/SYSTEM directory on the SRM/UX volume 8
8	**[P]** /WORKSTATIONS		To prefix back to /WORKSTATIONS

For convenience, designate the /WORKSTATIONS/SYSTEM directory on the SRM/UX file system as your workstation's system volume. This allows you to access the SRM/UX system's copies of the Pascal Workstation system's subsystems by pressing keys, such as **[E]** for Editor, **[F]** for Filer, etc. To do this, follow the steps in the next table.

Table 7-7 To Designate Your Workstation's system Volume:

Step	Action	Result	Explanation or Note
1	Press **[Q]**	The Pascal Workstation system displays the Main Command Level prompt line	To exit from (or quit) the Filer
2	Press **[X]**	The Pascal Workstation system prompts: **Execute what file?**	To execute a program
3	Type (including the trailing period): **TABLE. [Return]**	The TABLE auto-configuration program recognizes the SRM/UX server's /WORKSTATIONS/SYSTEM directory on volume 8 as volume #45	To re-execute the TABLE auto-configuration program
4	Press **[W]** Press **[S]**	The Pascal Workstation system prompts: **What new system unit number?**	To designate a new system volume
5	Type: **#45: [Return]**	The Pascal Workstation system makes volume #45 (the /WORKSTATIONS/SYSTEM directory) the system volume	
6	Press [Q]	The What command terminates	To quit the What command. Continue on with the next section on "Giving Each Workstation Access to system files.

```
Assembler Compiler Editor Filer Librarian
library system volume Default volume Quit

ASSEMBLER:   SYSTEM:ASSEMBLER
COMPILER:    SYSTEMCMP:COMPILER
EDITOR:      SYSTEM:EDITOR
FILER:       SYSTEM:FILER
LIBRARIAN:   SYSTEM:LIBRARIAN

* system volume:  SYSTEM:
: Default volume: WORKSTATIONS:
```

4. Giving Each Workstation Access to System Files

To make the Pascal Workstation system files available in every workstation's own system directory, create links from those directories to the Pascal Workstation system files.

The illustration below shows the SRM/UX file system's directory structure with one possible set of links. The links are from one client workstation's system directory to the server's Pascal Workstation system files.

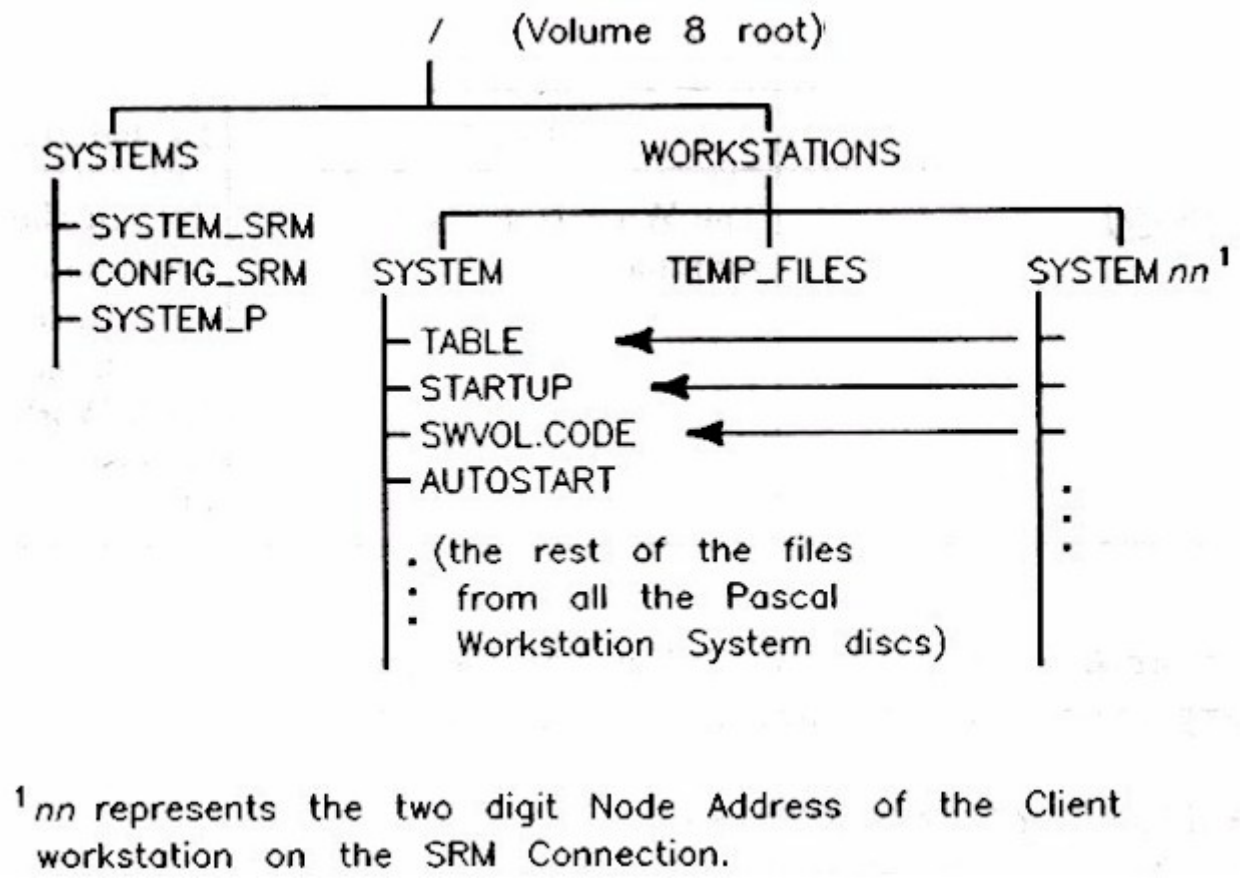


Figure 7-3. Access to Pascal Workstation System Files Through Links

Note:

The two character client Node Address, that is specified for clients on the LAN connection in the server's /etc/srmdconf LAN-CLIENTS table, is meaningless to Pascal. Pascal Workstation system recognizes only the six-hex-digit Link Level Address.

Table 7-8. Create Links to the Pascal System Files

Step	Action	Result	Explanation or Note
1	Press **[F]**	The Pascal Workstation system displays the Filer prompt line	To enter the Filer
1a	Press **[P]**	The Filer prompts: "Prefix to what directory?"	To prefix to a directory
1b	Type: **/WORKSTATIONS**	The Filer displays: "Prefix is WORKSTATIONS:"	To specify /WORKSTATIONS as the default directory
1c	Within /WORKSTATIONS, create a dedicated system directory for every workstation on your SRM/UX system (See the following examples).		To dedicate a system directory to a workstation, append the workstation's two (SRM card) or six character (LAN interface) node address to the directory name SYSTEM. (If a workstation's node address is only a single digit, precede it with a 0 (zero) to make it two digits long.) Every workstation's node address is either readable from an SRM card or is the last 6 digits of the workstation's LAN address.

Examples:

If a client workstation's node address is 8 (running over the SRM connection), create the directory:

SYSTEM08

If a client workstation's Link Level Address is 0x80009012345 then its client Node Address is 012345 (running over the LAN connection), so create the directory:

SYSTEM012345

Table 7-8.

Step	Action	Result	Explanation or Note
1d	Press **[D]**	The Filer prompts: **Duplicate or Move? (D/M)**	To create a duplicate link to a file
2	Press **[D]**	The Filer prompts: **Dup_link what file?**	To tell the Filer that you want to duplicate, not move, a file.
3	Type: **SYSTEM/?,SYSTEMnn/$ [Return]** OR **SYSTEM/?,SYSTEMnnnnnn/$ [Return]** where nn is the client workstation's two-digit node address, and nnnnnn is the client workstation's six-hex-digit emulated node address.	The Filer lists the first file in the directory, /WORKSTATIONS/SYSTEM, and asks if you want to create a duplicate link to it	Remember to use a leading 0 (zero) with single-digit node address. The ? is a wildcard that tells the Filer to display one file name at a time and allow you to select only those files you want to create a link to.
4	Press **[N]** when the Filer displays the file name AUTOSTART. Press **[Y]** when every other file name appears.	As the Filer creates a duplicate link to each file you respond "yes" to, the Filer displays each file name.	To respond "yes" to every duplicate link except for AUTOSTART.
5	Repeat steps 1 through 5 as many times as necessary to create a set of links from every workstation's own system directory to the Pascal Workstation system files in the /WORKSTATIONS/SYSTEM directory.	The Pascal Workstation system files are then accessible from every workstation's own system directory, and can be modified independently in each dedicated directory if so desired.	
6	Press **[Q]**	The Pascal Workstation system displays its Main Command Level prompt line.	To exit from (quit) the Filer
7	Continue with the next section on "Adding Required Driver Modules to the Initialization Library."		

5. Adding Required Driver Modules to the Initialization Library

Upon bootup, the Pascal Workstation System automatically loads the driver modules required to access the SRM/UX system, if you include them in the system's initialization library (INITLIB file). The driver modules that provide access to the SRM/UX system are: DATA_COMM, IOMPX, LANDVR, and SRM. LANDVR and IOMPX are not needed if you are running over the SRM connection only. Both IOMPX and LANDVR are contained in the file "LAN" on the ACCESS: disc. The Pascal Workstation System's Librarian makes it easy for you to add these modules to those already contained in the system's original INITLIB file.

The INITLIB file on the Pascal Workstation System's BOOT: disc (for Series 200 computers), differs slightly from the INITLIB file on the BOOT2: disc (for Series 300 computers). To each of these two INITLIB files, add the DATA_COMM and SRM driver modules (if on SRM) or DATA_COMM, IOMPX, LANDVR, and SRM (if on LAN), as well as any other drivers you would normally want. (e.g., GPIO) Create two new initialization libraries: one for client workstations that are Series 200 computers and one for client workstations that are Series 300 computers. If you will be running over the LAN connection, add the LAN driver module to INITLIB as well.

Both new initialization libraries will go into the SYSTEM directory of the SRM/UX volume 8. So that you can easily distinguish between the two initialization libraries, name one INIT_200 and the other INIT_300. Of course, you are free to name the libraries as you wish; these names are not required by the Pascal Workstation system.

The following diagram illustrates the process described above:

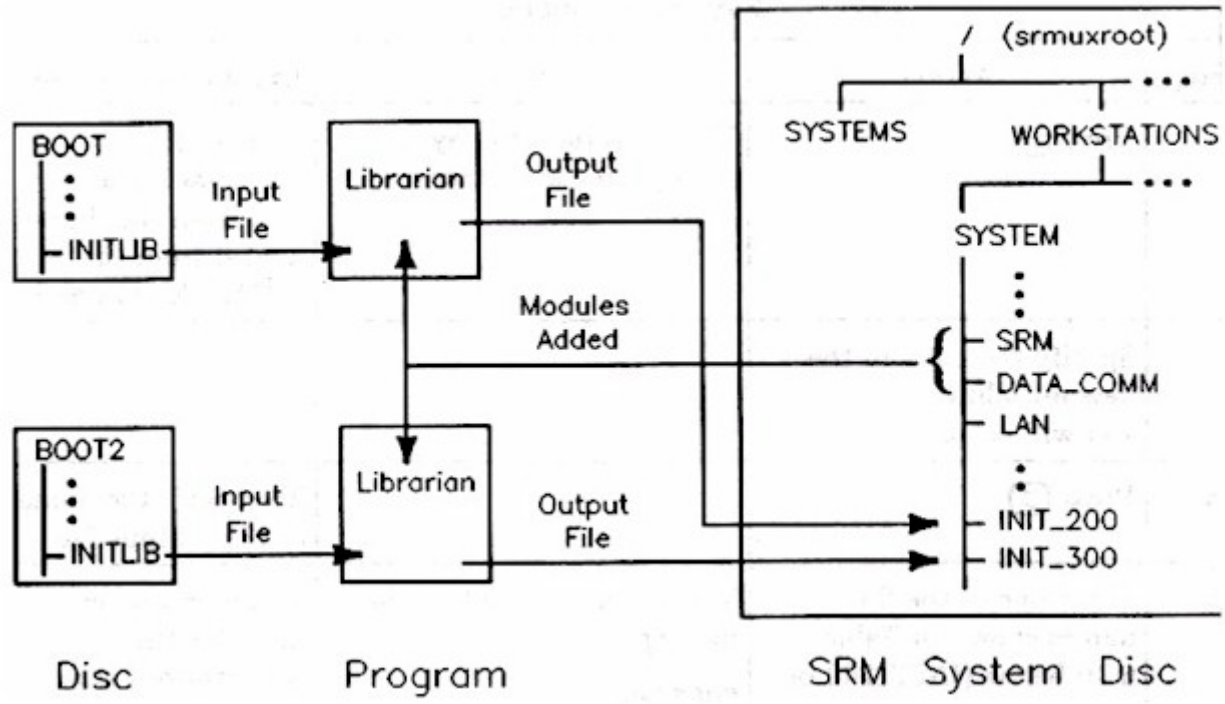


Figure 7-4. Adding SRM/UX Driver Modules to the Initialization Library

Note:

To create an initialization library for Series 200 workstations AND Series 300 workstations, you must perform the following procedure TWICE (once for each type of workstation).

Table 7-9.
Create a system Initialization Library
for Series 200 OR Series 300 Workstations:

Step	Action	Result	Explanation or Note
1	Press **[L]**	The Librarian displays its prompt line and status	To load the Librarian. The Librarian should be loaded from the SRM/UX volume 8.
2	Specify the path to the new initialization library you will create,as follows:		
2a	Press **[O]**		To specify the name of the Output file
2b	Enter one of the file names shown in Table 6-10 below (INIT_200 or INIT_300)	The Librarian displays the message: **COPYING**	To name the output file after the workstations it is intended for. The output file will be created in default directory, /WORKSTATIONS, where all of the other Pascal Workstation system files are.

Table 7-10.
INIT_200 and INIT_300

If you are creating an initialization library for:	Then type (including the trailing period):
Series 200 workstations	**INIT_200. [RETURN]**
Series 300 workstations	**INIT_300. [RETURN]**

Table 7-10.

Step	Action	Result	Explanation or Note
3	If you are creating an initialization library for Series 200 workstations, insert the BOOT: disc into the local disc drive. If you are creating an initialization library for Series 300 workstations, insert the BOOT2: disc into the local disc drive.		BOOT: contains the INITLIB file for Series 200 and BOOT2: contains the INITLIB file for Series 300 workstations
4	Specify the name of the source INITLIB file that you will add modules to, as follows:		
4a	Press **[I]**		To specify the name of the input file
4b	Enter one of the file names shown in Table 6-11 below.	The Librarian shows INITLIB as the name of the input file and displays the line: **M input Module: KERNEL**	The Input file is the INITLIB on the BOOT: or BOOT2: disc.

Table 7-11. BOOT:INITLIB and BOOT2:INITLIB	
If you are creating an initialization library for:	**Then type (including the trailing period):**
Series 200 workstations	**BOOT:INITLIB. [RETURN]**
Series 300 workstations	**BOOT2:INITLIB. [RETURN]**

Table 7-10.			
Step	**Action**	**Result**	**Explanation or Note**
5	Press **[T]**	After a few moments, the name of a new module (KBD) appears.	To transfer (copy) the module KERNEL to the output file

Note:
Read the instructions for the next step before you proceed.
Each time you transfer a module to the output file, a new module name appears. You will continue to copy modules from **INITLIB** to the output file UNTIL the name of the module **LAST** appears. **Do NOT transfer the module LAST at that time.**

Table 7-10.			
Step	**Action**	**Result**	**Explanation or Note**
6	Repeat step 5 until you see the line: **M input Module: LAST** near the bottom of the display. **Do not copy the module LAST yet!**	A new module name appears after you transfer the current module to the output file.	To transfer all of the INITLIB modules except LAST to the output file.
7	Get the required SRM, LAN, and DATA_COMM drivers and include them in the output file, as follows:		The SRM, LAN, and DATA_COMM drivers allow the Pascal Workstation system to access an SRM/UX system.
7a	Press **[I]**	The Librarian clears the current input file name from the display	To change the input file.
7b	Type (including the trailing period): **/WORKSTATIONS /SYSTEM/DATA_COMM. [Return]**	The module name **DATA_COMM** appears near the bottom of the display	To specify the file containing the DATA_COMM driver. This file is in the default directory, /WORKSTATIONS/SYSTEM , on the SRM/UX volume 8 root.
7c	Press **[A]**		To tell the Librarian to transfer all modules from the input file, DATA_COMM, to the output file. (In this case, there is only one module in the input file.)
7d	Press **[I]**	The Librarian clears the current input file from the display	To change the Input file
7e	If the clients using INITLIB will never access the server over the LAN connection, proceed to step 7i.		To avoid putting unnecessary modules in the INITLIB file.

7f	Type (including the trailing period): **/WORKSTATIONS /SYSTEM/LAN. [Return]**	The module name IOMPX appears near the bottom of the display.	To specify the file containing the LAN driver. This file is in the default directory, /WORKSTATIONS/SYSTEM on the SRM/UX volume 8.
7g	Press **[A]**		To tell the Librarian to transfer all modules from the input file, LAN, to the output file. (There are two modules in the input file: IOMPX and LANDVR.)
7h	Press **[I]**	The Librarian clears the current input file from the display	To change the Input file
7i	Type (including the trailing period): **/WORKSTATIONS /SYSTEM/SRM. [Return]**	The module name **SRM** appears near the bottom of the display.	To specify the file containing the SRM driver. This file is in the default directory, /WORKSTATIONS/SYSTEM on the SRM/UX volume 8 root.
7j	Press **[A]**		To tell the Librarian to transfer all modules from the input file, SRM, to the output file. (There is only one module in the input file.)
8	Now transfer the module LAST to the output file, as follows:		
8a	Press **[I]**	The Librarian clears the current input file name from the display.	To change the Input file
8b	Enter one of the file names shown in Table 6-11 below.	The module name **KERNEL** appears near the bottom of the screen.	The INITLIB file on the BOOT: or BOOT2: disc contains the module LAST.

Table 7-11.
BOOT and BOOT2: - LAST

If you are creating an initialization library for:	**Then type (including the trailing period):**
Series 200 workstations	**BOOT:INITLIB. [RETURN]**
Series 300 workstations	**BOOT2:INITLIB. [RETURN]**

Table 7-10.

Step	Action	Result	Explanation or Note
8c	Press **[M]**		To directly request a different module
8d	Type: **LAST [Return]**		To select module LAST
8e	Press **[T]**	Now all of the modules required to access SRM/UX system resources are in the new initialization library.	To transfer the module LAST to the output file.
			[continued]

9	Type **[K]**	The Librarian saves the output file INIT_200 or INIT_300 in the default directory, /WORKSTATIONS/SYSTEM , on the SRM/UX volume 8 root.	To "keep" the output file
10	Press **[Q]**	The Pascal Workstation system displays its Main Command Level prompt line.	
11	If you just created an initialization library for Series 200 workstations and you want to create one for Series 300 workstations, or the other way around, repeat steps 1 through 10. If you are finished creating the initialization library or libraries, continue with the next section on "Giving Each Workstation Access to the Correct Initialization Library."		

6. Giving Each Workstation Access to the Correct Initialization Library

For a workstation to boot the Pascal Workstation system from the SRM/UX server's file system, an initialization library must be available in the workstation's own system directory. If you have followed this procedure linearly from the beginning to this point, the client workstations don't yet contain an INITLIB in their system directories.

The /WORKSTATIONS/SYSTEM directory, however, contains the initialization libraries you just created for the client workstations. The files INIT_200 and INIT_300 contain modules required by Series 200 and Series 300 workstations, respectively.

To make the correct initialization library available in every workstation's own system directory, create links to INIT_200 and INIT_300. For example, the file INIT_300 might be called INITLIB (as required by the Pascal Workstation system) in a client workstation's own system directory.

To create a link from every client workstation's system directory to the correct initialization library, either INIT_200 or INIT_300, follow these steps:

Table 7-11.
Create a Link to Either INIT_200 or INIT_300:

Step	Action	Result	Explanation or Note
1	Press **[F]**	The Filer displays its prompt line	To enter the Filer
2	Press **[D]**	The Filer prompts: **Duplicate or Move ? (D/M)**	To create a duplicate link to a file
3	Press **[D]**	The Filer prompts: Dup_link what file?	To tell the Filer that you want to duplicate, not move, a file.
4	On your hardware configuration worksheet(s) from chapter 3, look up the node address of a client workstation whose system directory does not yet have a link to the proper initialization library. Start with the first workstation listed on the worksheet		To obtain the value you need for the next step
5	Enter one of the responses shown in Table 6-12 below.	The Filer creates a link to INIT_200 (named INITLIB there) from the client workstation's own system directory.	Remember to use a leading 0 (zero) with single-digit node addresses

Table 7-12. Series 200 and 300 /SYSTEMnn/INITLIB	
If the node address belongs to:	**Then type:**
Series 200 workstations	**INIT_200,../SYSTEMnn/INITLIB [Return]** where nn is the interface's two digit node address
Series 300 workstations	**INIT_300,../SYSTEMnn/INITLIB [Return]** where nn is the interface's two or six digit node address

Table 7-18.			
Step	**Action**	**Result**	**Explanation or Note**
6	Repeat steps 2 through 5 as many times as necessary to create a link from every client workstation's system directory to the correct initialization library.	This makes the correct initialization library available in every workstation's system directory	

If a workstation does not have its own system directory, then the necessary Pascal Workstation system files, including INITLIB, must be in the /WORKSTATIONS/SYSTEM directory on the server. At this point in the configuration process the /WORKSTATIONS/SYSTEM directory contains all of the system files, except INITLIB from the Pascal Workstation system discs.

The /WORKSTATIONS/SYSTEM directory, however, does contain the initialization libraries INIT_200 and INIT_300 with the driver modules required to access the SRM/UX system. The Pascal Workstation system can't find and load either INIT_200 or INIT_300 because it looks for the INITLIB file.

By creating a link from INIT_200 or INIT_300 to the file INITLIB, you make the initialization library available in the /WORKSTATIONS/SYSTEM directory.

CAUTION

INIT_200 may not work for Series 300 workstations, and INIT_300 will not work for Series 200 workstations. Any workstation without its own, dedicated system directory will try to use the INITLIB file in the /WORKSTATIONS/SYSTEM directory. Therefore, make the default INITLIB file whichever file (either INIT_200 or INIT_300) would successfully serve a majority of the client workstations on your SRM/UX system, or any new system you may attach before creating its own dedicated system directory.

Table 7-19.
Create a Link from INIT_200 or INIT_300 to INITLIB:

Step	Action	Result	Explanation or Note
1	Press **[D]**	The Filer prompts: **Duplicate or Move? (D/M)**	To create a duplicate link to a file
2	Press **[D]**	The Filer prompts: **Dup_link what file?**	To tell the Filer that you want to duplicate, not move, a file
3	If you expect to use INITLIB for Series 200 computers, then type: **INIT_200,INITLIB [Return]** If you expect to use INITLIB for Series 300 computers, then type: **INIT_300,INITLIB [Return]**	The Filer creates a link from INIT_200 or INIT_300 to the file INITLIB in the default directory /WORKSTATIONS/SYSTEM	
4	Press **[Q]**	The Pascal Workstation system displays its Main Command Level prompt line	To exit from (quit) the Filer

After booting the Pascal Workstation System from the SRM/UX server, client workstation users will be able to access the contents of the SRM/UX volume(s) for which they are given access in the server's /etc/srmdconf file and for which TABLE has provided access. Client workstation users will not be allowed access to the SRM/UX server's HP-UX file system anywhere above the root directories of any of the volumes they have been given access to. However, the SRM/UX System Administrator can allow clients to have access to as many volumes as they need under the HP-UX root directory structure.

Unless you want to configure your SRM/UX system so that the Pascal Workstation system clients can access multiple volumes, you are done installing the Pascal clients at this point.

7. Giving Pascal Clients Access to Multiple SRM/UX Volumes

This section assumes that your SRM/UX System Administrator has made the necessary entries in the /etc/srmdconf file on the server to associate particular Volume numbers and Volume names with particular directories.

If your SRM/UX system has directories not contained in the Volume 8 subtree to which users need access, then you must modify the Pascal Workstation system's auto-configuration program, TABLE, and give clients access to the new TABLE program.

The TABLE auto-configuration program assigns a unit number to each mass storage device available to a workstation when it runs. The system directory and default on Volume 8 directory are the only SRM/UX directories that the unmodified TABLE program can recognize and assign a unit number to. To have TABLE recognize and assign unit numbers to other SRM/UX directories, you must edit and recompile TABLE's source program: CTABLE. This section describes how to perform such a configuration change.

Modifying the CTABLE Program

The Pascal Workstation System's CTABLE file already contains source code that assigns unit #5 to the SRM/UX volume 8 root and tries to assign unit #45 to the system directory from which the client booted.

Table 7-20.
Allocate New Unit Numbers To Other SRM/UX Volumes

Step	Action	Result	Explanation or Note
1	Get the address number associated with each volume you want to automatically bring on-line. This information is available in the VOLUME-TABLE section of the file /etc/srmdconf in the server's file system. Your SRM/UX system administrator should provide these numbers to you.		
2	Record the select code of the card over which you are talking to the server, (the select code is probably 21)		
3	Return to the client workstation from which you began and Press **[E]** (for Editor)	The Editor displays the prompt: **file?**	To enter the Editor
4	Type: ***CTABLE [Return]**	The Editor displays the source code	To edit the CTABLE source program in the default system directory on the SRM/UX file system.
5	Near the beginning of the source program, you will see the note shown below. Using the arrow keys, move the cursor to the beginning of the first blank line below the "box".		To prepare to insert a required compiler directive in the source program

```
(*********************************************)
(*                                           *)
(*  Note: You will need to use the following *)
(*  compiler directive if the 'INTERFACE'    *)
(*  file is not in your current LIBRARY.     *)
(*  Modify the volume name as necessary for  *)
(*  your configuration.                      *)
(*                                           *)
(*  $search  'CONFIG:INTERFACE. '$           *)
(*                                           *)
(*********************************************)
```

Table 7-22.
Allocate New Unit Numbers To Other SRM/UX Volumes

Step	Action	Result	Explanation or Note
6	Press **[I]**	The Editor's command line disappears	To insert text
7	Type the compiler directive shown below		

Type (including the period after INTERFACE):

$search '#5:/WORKSTATIONS/SYSTEM/INTERFACE. '$

Table 7-22.

Step	Action	Result	Explanation or Note
8	Press **[Select]** or **[EXECUTE]**	The Editor's command line appears again	To accept the new line
9	Press **[J]** and then **[E]**	The Editor displays the end of the source program	
10	Use the up arrow key to scroll upward through the program until you see the section of code shown below		This section of code assigns unit numbers to SRM/UX volumes. The statement *tea_srm(45,sc,ba,du);* tries to assign unit #45 to the system volume from which the client booted.

```
{ duplicate entries for prefixing down the SRM }

with SRM_dav do
  begin
   {  tea_srm( 46,sc,ba,du);  {free}
   tea_srm( 45,sc,ba,du);  {for possible use as the system unit}
  end;  {with}
```

Table 7-23. sc, ba, and du		
Variable	**Description**	**Default Value**
SC	The select code of the workstation's SRM or LAN interface. This should have been recorded in step 2 of this procedure.	**21**
ba	The node address of the server. Set this value to be what the SRM/UX system administrator has assigned and entered in the VOLUME-TABLE of the /etc/srmdconf file.	In a one server system, usually 0
du	The volume address of the server's SRM/UX directory. This was recorded in step 1.	usually 8

Table 7-22.			
Step	**Action**	**Result**	**Explanation or Note**
11	Use the Editor's Delete, Insert, and eXchange functions to modify the section of CTABLE shown above as required for your SRM/UX system configuration. **Read the example below before you proceed; use it as a guide.**		Remember to press **[Select]** or **[EXECUTE]** to accept the changes from one function and redisplay the Editor's command line to select another function.

Example:

To illustrate the first set of changes required to give workstations access to multiple SRM/UX volumes, let us assume you have the following SRM/UX volumes available:

```
                          A server at node address 0 with:

CASE ONE:                 a directory at volume address 8
CASE TWO:                 a directory at volume address 9
CASE THREE:               a directory at volume address 10
                          The workstation's SRM interface cabled to
                          this server is at select code 21.

                          Another server at node address 1 with:

CASE FOUR:                a disc at volume address 8
CASE FIVE:                a disc at volume address 9
                          The workstation has a second SRM interface
                          at select code 22 cabled to this server.
```

CTABLE contains the program line for CASE ONE (ba is assigned the value 0 and du is assigned the value 8, if no other value for them is specified). To allow access to the volumes in CASE TWO through CASE FIVE, assign a new unit number to each disc, by modifying CTABLE as shown:

```
      tea_srm( 46, sc, ba, 9);     {change du to 9 for CASE TWO}
      tea_srm( 47, sc, ba, 10);    {Add this line for CASE THREE}
      tea_srm( 48, 22, 1, 8);      {Add this line for CASE FOUR}
      tea_srm( 49, 22, 1, 9);      {Add this line for CASE FIVE}

      tea_srm( 45, sc, ba, du);    { This line is for CASE ONE}
                                     {for possible use as the system unit}
          end;  {with}
```

Table 7-22.			
Step	**Action**	**Result**	**Explanation or Note**
12	Use the down arrow key to scroll through the program until you see the section of code shown below.		This section of code recognizes the SRM/UX system (root) directory as unit #5. Unit #45 references the client workstation's own system directory.

```
{ prefix the primary and secondary SRM unit entries  }

if not unit_prefix_successful('#5:/') then
  {do nothing};  {tries to set up uvid for possible default unit assignment below}

{ if not unit_prefix_successful('#46:/?') then zap_assigned_unit(46);  {free}

if not unit_prefix_successful('#45:'+srmsysprefix+srmnode(unitable^[45].sc)) then
  if not unit_prefix_successful('#45:'+srmsysprefix) then
    zap_assigned_unit(45);
```

Table 7-22.			
Step	**Action**	**Result**	**Explanation or Note**
13	Use the Editor's Delete, Insert, and eXchange functions to modify the section of CTABLE shown above as required for your SRM/UX system configuration. **Read the example below before you proceed; use it as a guide.**		Remember to press **[Select]** or **[EXECUTE]** to accept the changes from one function and redisplay the Editor's command line to select another function.

Example:

To illustrate the changes required to give workstations access to multiple SRM/UX volumes, the example SRM/UX system configuration shown above in step 11 continues here.

For the Pascal Workstation System to recognize additional SRM/UX volumes (besides the SRM/UX system volume) by the new unit numbers you assigned to them, modify CTABLE as follows:

```
{  prefix the primary and secondary SRM unit entries }

if not unit_prefix_successful('#5:/") then
  {do nothing};  {tries to set up uvid for possible default unit assignment below}

  ###Remove the { comment delimiter and the ? for CASE TWO, as shown below.

if not unit_prefix_successful('#46:/') then zap_assigned_unit(46); {free}

  ###Add these lines for CASE THREE, CASE FOUR, and CASE FIVE, respectively:

if not unit_prefix_successful('#47:/') then zap_assigned_unit(47);  {free}
if not unit_prefix_successful('#48:/') then zap_assigned_unit(48);  {free}
if not unit_prefix_successful('#49:/') then zap_assigned_unit(49);  {free}

if not unit_prefix_successful('#45:'+srmsysprefix+srmnode(unitable^[45].sc)) then
```

```
If not unit_prefix_successful('#45:'+srmsysprefix) then
  zap_assigned_unit(45);
```

Table 7-22.

Step	Action	Result	Explanation or Note
14	From the Editor's command line, press **[Q]**	The Editor displays several exit options	To tell the Editor that you want to finish the editing session
15	Press **[W]**	The Editor prompts for the name of the new file	To write the edited CTABLE out to a new file, while keeping the original CTABLE file unchanged
16	Type: ***NEWCTABLE [Return]**	The Editor displays the message **Writing...** and then prompts: **Exit from or Return to the Editor ?**	To save the edited CTABLE as NEWCTABLE in the default system directory on the server's SRM/UX file system
17	Press **[E]**	The Pascal Workstation system displays its Main Command Level prompt line	To exit from the Editor
18	Continue with the next section on "Compiling, Running, and Verifying the Modified CTABLE Program."		To produce a new auto-configuration program that can recognize all of the available directories on your SRM/UX system

8. Compiling, Running, and Verifying the Modified CTABLE Program

To compile, run, and verify your NEWCTABLE auto-configuration program, follow these steps:

Table 7-23.
Compile, Run, and Verify your NEWCTABLE program

Step	Action	Result	Explanation or Note
1	Press **[C]**	The Compiler prompts: **Compile what text?**	To load the Compiler
2	Type: ***NEWCTABLE [Return]**	The Compiler prompts: **Printer Listing?**	To specify the modified CTABLE program you just created and saved to the default system directory
3	Press **[N]**		For no listing
4	Press **[Return]**	The Compiler compiles the source program and displays status messages as the compilation progresses and then finishes	To accept the displayed output file name of NEWCTABLE.CODE.
5	If your did not get any compilation errors, press **[R]**		To run the NEWCTABLE.CODE program
6	Press **[F]**	The Pascal system displays the Filer's command line	To enter the Filer
7	Press **[V]**	The Filer displays a list of the volumes that the workstation recognizes. The listing should show the new unit numbers (for example, 46,47,48, and so on) you assigned to your additional SRM/UX volumes	To verify the new auto-configuration program recognizes all of your SRM/UX volumes
8	Press **[Q]**	The Pascal Workstation system displays its Main Command Level prompt line	To exit from the Filer
9	Continue with the next section on "Giving Each Workstation Access to the New TABLE Program."		To make the new auto-configuration program available to all of the workstations on your SRM/UX system

9. Giving Each Workstation Access to the New TABLE Program

Your newly created auto-configuration program, NEWCTABLE.CODE, and the old, unmodified auto-configuration program, TABLE, are both in the /WORKSTATIONS/SYSTEMnnnnnn directory (where nnnnnn is your client's six hex-digit node address over the LAN connection) or the /WORKSTATIONS/SYSTEMnn directory (where nn is your client's two-digit node address on the SRM connection). Also, at this point in the configuration process, every client workstation's own system directory contains a link to the old TABLE program.

For workstations to boot the Pascal Workstation system from the SRM/UX server's file system, the new auto-configuration program NEWCTABLE.CODE:

- must be called TABLE,
- must be accessible from every client workstation's own system directory

You can accomplish both of these things by overwriting the old TABLE auto-configuration program with the new one. Overwriting changes the contents of a file while preserving its original name and its links.

To overwrite the old auto-configuration program, TABLE, with the new auto-configuration program, NEWCTABLE.CODE, follow these steps:

Table 7-24.
Overwrite TABLE with NEWCTABLE.CODE

Step	Action	Result	Explanation or Note
1	Press **[F]**	The Pascal Workstation system displays the Filer's command line	To enter the Filer
2	Press **[F]**	The Filer prompts: **filecopy what file ?**	To attempt to copy a file (or perform a filecopy
3	Type: **NEWCTABLE.CODE,TABLE [Return]**	The Filer displays: **Reading...** and then: **SYSTEM:TABLE exists... Remove/ Overwrite/Neither? (R/O/N)**	To copy the file NEWCTABLE.CODE to TABLE in the default directory, /WORKSTATIONS/SYSTEM
4	Press **[O]**	The Filer overwrites the file TABLE with the contents of NEWCTABLE.CODE	To overwrite the file
5	Press **[Q]**	The Pascal Workstation system displays its Main Command Level prompt line	To exit from the Filer

By booting the Pascal system from the SRM/UX server's file system, client workstation users get automatic access to all of the available volumes in the SRM/UX file system. At this point you have given workstation users complete access to SRM/UX system resources.

There are a few more things you may want to do to make booting-up the Pascal Workstation system more convenient for users. See the next section on "System Customization Options" for details.

10. System Customization Options

By customizing client-workstation's bootup schemes, you provide users with fast access to the particular SRM/UX volumes and system resources they use most often. For example, at bootup time the Pascal Workstation system can automatically:

- make the user's personal SRM/UX directory the default directory, or make whichever SRM/UX directory the user accesses most frequently the default directory
- load subsystems (FILER, EDITOR, COMPILER, etc.) and other frequently-run programs from an SRM/UX system into a workstation's internal memory so that they run quickly when invoked
- assign unit numbers to a set of frequently-accessed directories on an SRM/UX disc, so that the user need only specify short unit numbers rather than long paths to access those directories

This section explains how to provide workstation users with fast, easy access to essential SRM/UX resources each time they boot their Pascal Workstation system.

Bootup-Time Default SRM/UX Directory

If a workstation user works primarily with the files in a certain SRM/UX directory, there are two ways to configure the Pascal Workstation system to boot-up so that directory is the default directory:

- create an autostart file for the workstation
- modify the workstation's auto-configuration program: *TABLE

Creating an autostart file is the faster, easier way of making this type of configuration change, but it will make boot-up take a few seconds longer than if you modify the TABLE program. But, unless you have other reasons to modify the TABLE program, the slight savings of boot-up time probably doesn't warrant going to the trouble.

Faster Program Execution
To achieve faster program execution, configure the Pascal Workstation system to automatically "P-load" subsystems (FILER, EDITOR, COMPILER, etc.) and/or other frequently-run programs. To "P-load" means to permanently transfer a copy of the subsystem or program from the SRM/UX disc to the client workstation's internal memory. Once the subsystem or program is "P-loaded" in a workstation's internal memory, it runs immediately whenever the user invokes it.

To make this kind of configuration change, create an autostart file for the client workstation. See the instructions later in this section to create autostart files.

Note
Be sure that the combined size of the subsystems and/or programs to be loaded does not exceed the capacity of the client workstation's memory.

SRM/UX Directory Access by Unit Number, Rather Than by Path
If a workstation user often accesses a number of directories on the same SRM/UX file system, configure the Pascal Workstation system to assign unit numbers to those directories. Then, to access one of those directories, the user need specify only the assigned unit number, instead of the (often lengthy) path to the directory.

To make this type of configuration change, modify the workstation's auto-configuration program: TABLE. Also, your SRM/UX system administrator must make the proper entries in the server's /etc/srmdconf file to allow access to these directories (see chapter 4).

Creating an Autostart File for Each Client Workstation

Using the Pascal Workstation system Editor, create an autostart file for each client workstation on your SRM/UX system. The autostart file must be in the client's own system directory.

The illustration below shows the correct placement of a client workstation's autostart file within the SRM/UX server's volume 8 directory structure:

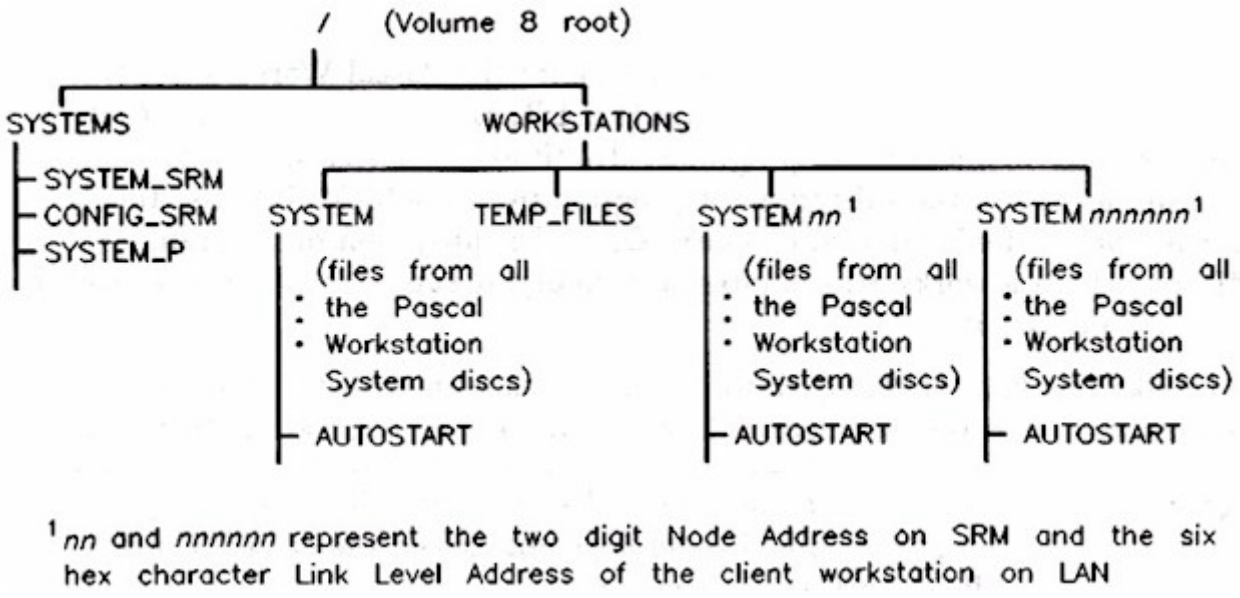


Figure 7-5. required Location of Pascal Workstation System Autostart File

The Pascal Workstation system treats the contents of an autostart file as if they came directly from a workstation's keyboard. Therefore, an autostart file may contain keystrokes that:

- invoke or exit from Pascal Workstation system subsystems
- initiate or terminate a command
- are a response to a system prompt

Immediately after a workstation boots the Pascal Workstation system, the system executes the contents of the workstation's autostart file (if one exists).

To create an autostart file for each client workstation on your SRM/UX system, follow these steps:

Table 7-25
Create an Autostart file for Each client

Step	Action	Result	Explanation or Note
1	From Main Command Level, press **[E]**	The Editor prompts: **file?**	To start the Editor
2	Press **[Return]**	The Editor displays its prompt line and clears the screen	This tells the Editor that you are going to create a new file
3	Press **[I]**	The Editor's command line disappears	To insert text

Note

Read the rest of this procedure without performing the tasks, to gain an understanding of the procedure flow. Examples for steps 4 through 9 show how to create a custom autostart file for a workstation user named Susan. Susan wants the Pascal Editor and Filer automatically "P-loaded" into her workstation's memory, to allow faster access to those subsystems. She wants the default directory at boot-up time to be her private directory on the SRM/UX file system. Finally, she wants the default directory on another available SRM/UX directory (unit #46:) to be /MARKETING/REPORTS.

Table 7-32

Step	Action	Result	Explanation or Note
4	Type in the contents of a particular client workstation's autostart file, pressing **[Return]** to start each line.		

Example

To satisfy Susan's configuration needs, create this autostart file:

```
Line 1:
Line 2:    [7]
Line 3:        P#45:EDITOR.
Line 4:        P#45:FILER.
Line 5:        FP#5:/USERS/SUSAN
Line 6:        U#46:/MARKETING/REPORTS
Line 7:        Q
```

Explanation

- **Line 1** is a blank line, representing the keystroke **[Return]** in response to the system's date prompt (the first prompt of the boot-up sequence). The Pascal Workstation system gets the correct date from the SRM/UX server, so there is no need to set the date here.

- **Line 2** has an optional 7 entered which sets the time zone for Mountain Standard time, in response to the system's time prompt. The Pascal Workstation system gets the correct time from the SRM/UX server Computer, so it is not necessary to set the time here.

- **Line 3**, **P** invokes the Permanent (P-load) command from Main Command Level. **#45:EDITOR.** represents the response required to load the Editor from the SRM/UX server's volume 8. Remember to include the trailing period after the subsystem name.

- **Line 4**, **P** invokes the Permanent command from Main Command Level. **#45:FILER.** represents the response required to load the Filer from the SRM/UX server's volume 8. Remember to include the trailing period after the subsystem name.

- **Line 5**, **F** invokes the Filer from Main Command Level. **P** invokes the Filer's Prefix command, and **#5:/USERS/SUSAN** represents the response required to set the default directory to Susan's personal SRM/UX directory.

- **Line 6**, **U** invokes the Filer's Unit directory command. **U#46:/MARKETING/REPORTS** represents the response required to set the default directory on unit #46: to REPORTS.

- **Line 7** has a **Q** to Quit (exit from) the Filer and return to Main Command Level.

Table 7-32			
Step	**Action**	**Result**	**Explanation or Note**
5	Press **[Select]**	The Editor redisplays its command line	To accept what you typed in
6	Press **[Q]**	The Editor displays several exit options	To tell the Editor that you want to finish the editing session
7	On your hardware configuration worksheet(s) from chapter 3, look up the node address of the client workstation whose autostart file you are creating.		
8	Press **[W]**	The Editor prompts for the name of the new file	To write the autostart instructions out to a new file
9	Enter the path shown below.		To store the client workstation's autostart file in the workstation's own system directory. You must name the file ***AUTOSTART**.

Type (including the trailing period):

#5:/WORKSTATIONS/SYSTEMnn/AUTOSTART. [Return]

where nn is the workstation's two (SRM) or six (LAN) digit node address. (Use a leading 0 (zero) for single-digit node addresses.)

Table 7-33. Store the Client's Autostart File	
Example Entry:	**Result:**
If the node address of Susan's workstation were 4, you would type: **#5:/WORKSTATIONS/SYSTEM04 /AUTOSTART. [Return]**	This stores Susan's autostart file in her workstation's own system directory.

Table 7-32

Step	Action	Result	Explanation or Note
10	Press **[E]**	The Pascal Workstation system displays its Main Command Level prompt line	To exit from the Editor
11	Repeat steps 1 through 10 as many times as necessary to create a unique autostart file for every client workstation. **Modify the examples in steps 4 and 9 as needed** to suit your SRM/UX user's needs and to accurately reflect the client workstation's node addresses on your SRM/UX system.		It may be easiest to do these steps from each client.

If you do not intend to modify the auto-configuration program for each workstation, you are finished with the start-up procedure for the client Pascal Workstation systems.

11. Modifying Each Client Workstation's Auto-Configuration Program

Each workstation's auto-configuration program, TABLE, has a source program called CTABLE. The program, CTABLE, is available in every workstation's own system directory.

You may have already modified the copy of CTABLE that is in the /WORKSTATIONS/SYSTEM directory, to give client workstations access to multiple to multiple SRM/UX directories. If so, the source program for each workstation's TABLE program is the modified version of CTABLE, which you named NEWCTABLE, and is now located in the /WORKSTATIONS/SYSTEM directory.

To customize each workstation's TABLE program, you must:

- edit the proper source program
- compile the edited source program
- make the new auto-configuration program available as TABLE in the client workstation's own system directory.

The earlier section of this chapter called: "Giving Workstations Access to Multiple SRM/UX Directories" has detailed instructions for customizing the TABLE program. Therefore, this section contains only an overview of how to customize each client-workstation's auto-configuration program. The differences are noted here, in detail, where the instructions differ from those in that section.

To customize a client workstation's auto-configuration program, TABLE, follow these steps:

Table 7-33
Customize a client's TABLE Program

Step	Action
1	If you have already modified the client workstation's TABLE program to allow access to multiple SRM/UX directories, edit the source program: **#5:/WORKSTATIONS/SYSTEM/NEWCTABLE.TEXT** Otherwise, edit the source program: **#5:/WORKSTATIONS/SYSTEMnn/CTABLE.TEXT** where nn stands for the workstation's two (SRM) or six (LAN) digit node address.
2	If you have not already modified the workstation's TABLE program to allow access to multiple SRM/UX directories, add the following compiler directive to the beginning of the TABLE program: **$search '#5:/WORKSTATIONS/SYSTEM/INTERFACE. '$** This directive belongs on the blank line just above the statement: **program {self-configuring} ctable;**

Note

Read the rest of this procedure without performing the tasks, to gain an understanding of the procedure flow. Steps 3 through 6 show how to create a customized TABLE program for a Pascal workstation user named Susan. Susan wants to access one of the SRM/UX system's spooler directories with a unit number (#47:), rather than a path (/PRINTER). She wants the default directory at boot-up time to be her private directory on the SRM/UX file system. Finally, she wants the default directory on another available SRM/UX directory (unit #46:) to be /MARKETING/REPORTS.

Table 7-37			
Step	**Action**	**Result**	**Explanation or Note**
3	Locate and modify the appropriate section(s) of code to make the Pascal Workstation system configuration most convenient for the user. **Use the following example as a guide.**		

Example:

To make the changes Susan requests, change parts of her auto-configuration source program, as follows:

```
{ duplicate entries for prefixing down the SRM }

(*************************************************************************)
(* NOTE: Additional duplicate SRM entries may be assigned here, then   *)
(*  prefixed down below after assigning the temp_unitable. However,    *)
(*  for correct behavior in assigning the system unit, specifically    *)
(*  if booting off the SRM/UX, unit #45 must be assigned AFTER all     *)
(*  the other SRM/UX units have been assigned!                         *)
(*************************************************************************)

with SRM_dav do
 begin
   #  changes made previously to give access to another SRM/UX directory
   tea_srm( 46, sc, ba, 1);  {free}

   #  Add this line to assign an additional unit number to the SRM/UX system,
   #  (also assigned unit #45:)
   tea_srm( 47, sc, ba, du);

   tea_srm( 45, sc, ba, du); {for possible use as the system unit}
  end; {with}

                  .
                  .
                  .
{ prefix the primary and secondary SRM/UX unit entries }

   # Auto-configuration program makes the directory referenced by unit #5:
   # the default directory. Add the correct path name to Susan's private
   # SRM/UX directory.

if not unit_prefix_successful('#5:/USERS/SUSAN')  then
 {do nothing};{tries to set up uvid for possible default unit assignment below}

   # Add the path name to access-time default directory for the other
   # available SRM/UX directory.

if not unit_prefix_successful('#46:/MARKETING/REPORTS')
         then zap_assigned_unit(46);
         {free}

   # Add the next line to allow the Pascal Workstation system to
   # recognize the additional unit number you assigned to
   # the SRM/UX volume 8 directory.
```

```
   # Also add the path name to of the directory which this unit
   # number is to reference.

if not unit_prefix_successful('#47:/PRINTER')
   then zap_assigned_unit(47);

if not unit_prefix_successful('#45:'+srmsysprefix+srmnode(unitable^[45].sc))
   then if not unit_prefix_successful('#45:'+srmsysprefix)
     then zap_assigned_unit(45);
```

Table 7-37

Step	Action
4	Choose a name other than CTABLE for the edited source program and write it out to the client workstation's own system directory. Then exit the Editor. *Example:* If you decided to call Susan's edited source program CTABLE.TEXT, and the node address of her workstation was 9, write her source program to the file: **#5:/WORKSTATIONS/SYSTEM09/CUSTOMCTABLE.TEXT**
5	Compile, run, and verify the new auto-configuration program. (The compiled program may also be in the workstation's own system directory.) *Example:* If you compiled Susan's auto-configuration source program, CUSTOMCTABLE.TEXT, the compiled version would be: **#5:/WORKSTATIONS/SYSTEM09/CUSTOMCTABLE.CODE**
6	Copy the compiled auto_configuration program to the file name TABLE in the client workstation's own system directory. (You must remove --not overwrite-- the workstation's original TABLE program at the Filer prompt, since this TABLE program is unique to the user.) *Example:* To specify the filecopy for Susan's customized auto-configuration program, type: **#5:/WORKSTATIONS/SYSTEM09/CUSTOMCTABLE.CODE,** **#5:/WORKSTATIONS/SYSTEM09/TABLE**
7	Repeat steps 1 through 6 as many times as necessary to customize every workstation's auto-configuration program: TABLE. Modify the examples in steps 2 through 5 as neededto suit your SRM/UX user's needs.

You are now finished with the start-up procedure for the Pascal Workstation systems on the SRM/UX system. Every client workstation with a BootROM version 3.0 or later (but not 3.0L) will have the ability to automatically boot the Pascal Workstation system from the SRM/UX server's file system. After the client workstations boot-up, they will customize their own configurations according to their resident auto_configuration programs and/or autostart files.