HEWLETT PACKARD

File Server Reference Guide

for NS-ARPA/1000 and ARPA/1000

Software Services and Technology Division
11000 Wolfe Road
Cupertino, CA 95014-9804

Manual Part No. 91790-90054
E0495

Printed in U.S.A. April 1995
First Edition

NOTICE

The information contained in this document is subject to change without notice.

HEWLETT-PACKARD MAKES NO WARRANTY OF ANY KIND WITH REGARD TO THE MATERIAL, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. Hewlett-Packard shall not be liable for errors contained herein or for incidental or consequential damages in connection with the furnishing, performance, or use of this material.

Hewlett-Packard assumes no responsibility for the use or reliability of its software on equipment that is not furnished by Hewlett-Packard.

RESTRICTED RIGHTS LEGEND

Use, duplication, or disclosure by the Government is subject to restrictions as set forth in subparagraph (c) (1) (ii) of the Rights in Technical Data and Computer Software clause at DFARs 252.227.7013.

UNIX ® is a registered trademark in the United States and other countries, licensed exclusively through X/Open Company Limited.

NFS is a trademark of Sun Microsystems, Inc.

Printing History

The Printing History below identifies the edition of this manual and any updates that are included. Periodically, update packages are distributed which contain replacement pages to be merged into the manual, including an updated copy of this printing history page. Also, the update may contain write-in instructions.

Each reprinting of this manual will incorporate all past updates; however, no new information will be added. Thus, the reprinted copy will be identical in content to prior printings of the same edition with its user-inserted update information. New editions of this manual will contain new information, as well as all updates.

To determine what manual edition and update is compatible with your current software revision code, refer to the Manual Numbering File. (The Manual Numbering File is included with your software. It consists of an "M" followed by a five digit product number.)

First Edition . . . . . April 1995 . . . . . Rev. 6200 (Software Update 6.2)

Preface

This documentation describes the Hewlett-Packard *file server* functionality that is part of the NS-ARPA/1000 (HP 91790A) and ARPA/1000 (HP 98170A) products. This functionality allows the HP 1000 to service file requests made by a remote client node that has NFS (Network File System) installed. An *NFS client* node may be, for example, an HP 9000, or, another vendor or platform with NFS client capability. The HP 1000 can only service these NFS clients, it cannot be a client itself.

Note If you will be accessing HP 1000 files from an HP 9000 client, prior to reading this documentation you should be familiar with HP-UX and have access to *HP-UX Reference* manuals, specifically the following:

HP 9000, Using NFS Services

HP 9000, Installing and Administering NFS Services

Audience

This documentation is intended for the general user, and for those with the responsibility of installing, configuring, and maintaining the HP 1000 file server.

Assumptions

This documentation assumes a basic understanding of the RTE-A file system, general networking concepts and terminology.

Benefits of the File Server

Some of the benefits of the HP 1000 file server are that it:

- Provides RTE-A program developers access to program development tools on the HP 9000, such as vi or emacs.
- Enables analysis and reporting utilities running on the HP 9000 to access data files residing on an RTE-A system.
- Complements the RTE-A to HP-UX Migration Tools product when migrating RTE-A applications to the HP 9000.

Organization

Chapter 1 *Overview*

Introduces general concepts, terms, and gives an overview of the file server on the HP 1000.

Chapter 2 *Accessing the File Server*

Describes the user interface from a client's perspective, discusses restrictions, and provides examples of usage.

Chapter 3 *File Server Options and Considerations*

Describes the file server program (FSRV), runstring parameters, and special considerations.

Chapter 5 *Installing the File Server*

Describes the steps and files involved in the file server installation, configuration, and initialization.

Chapter 6 *Troubleshooting*

Describes the types of client requests serviced by the file server, trouble-shooting guidelines, and error messages.

Appendix A *Setting the Local Time*

Describes the RDATE utility that can be used to synchronize the system times of the file server and its clients.

Glossary

Table of Contents

Chapter 5
Troubleshooting

Appendix A
Setting the Local Time

Glossary

List of Illustrations

Tables

1

Overview

This *File Server Reference Guide* focuses on the HP 1000 file server capability of NS-ARPA/1000 and ARPA/1000. In the generic sense, a "server" is a system (or program) that provides a service and a "client" is a system (or program) that requests a service of the server. The HP 1000 file server allows files that reside on the HP 1000 to be shared in a networked environment of computer systems with different operating systems. Users (or "clients") of the file server can access remote files as if they are local, eliminating unnecessary remote file transfers to the local node.

Definition of Terms

The following terms have explicit meaning within this documentation and are defined below for clarity:

HP 1000 file server — refers to the functionality residing on the HP 1000 that allows *NFS client* nodes to access HP 1000 RTE-A files; within the context of this documentation, may also be referred to as the "file server" or FSRV, the file server program on the HP 1000. The HP 1000 file server functionality is part of the NS-ARPA/1000 (HP 91790A) and ARPA/1000 (HP 98170A) products.

NFS client — refers to a system (other than an HP 1000) which has NFS (Network File System) client software installed; within the context of this documentation may also be referred to as the "client". An *NFS client* is the only type of client whose file requests the HP 1000 file server can service.

File Server Operation

The main function of the HP 1000 file server is to service file system access requests received from a client. Additionally, it handles the initial request from a client to make a file system available for access, and provides information on the protocols and port numbers that the file server uses. (See Chapter 5 for further discussion of these types of requests). Prior to the startup of the file server program FSRV (see Chapter 3), user mappings and access rights must be specified in the files `/etc/ux_users`, `/etc/ux_groups`, and `/etc/exports` (see Chapter 4). The file server program uses these files to verify a user before fulfilling a file access request. Once validated, the file server inherits the capability of the requesting user for the purposes of completing the request, and returns the results to the client.

2

Accessing the File Server

RTE-A file systems to be made available by the HP 1000 file server are specified in the `/etc/exports` file by the HP 1000 system administrator. The HP 1000 file server program (FSRV) reads `/etc/exports` to determine which RTE-A directories to make accessible to a particular client. Before a client can gain file access to RTE-A file systems designated in the `/etc/exports` on the HP 1000, the system administrator (or superuser) for the client must import (or mount) the desired RTE-A file systems via the mount command. (See the documentation for your client system for more information on the mount command.)

Once the file server and client are set up for access, files on the HP 1000 may be accessed from the client as if they were local files. However, the use of a mounted RTE-A file system is not completely transparent when you access it from a client. Many aspects of the RTE-A file system are different from the typical UNIX file system. These differences result in some restrictions on the use of the RTE-A file system via the HP 1000 file server, especially in the area of file names and file types.

File Naming Restrictions

When accessing an RTE-A file volume from a client, file names are subject to certain restrictions imposed by the host RTE-A file system in regard to case sensitivity and the use of special characters.

Case Reporting

The RTE-A file system is not case sensitive and shifts file names to uppercase when they are saved. However, when accessing an RTE-A file from a client, regardless of whether you specify file names in uppercase, lowercase, or a combination, RTE file names will always be reported in lowercase to the client. By default, the HP 1000 file server ignores case which means a client is able to specify an RTE file by names that are normally unique on the client system, but are not unique on an RTE-A volume. For example, if you specify the following file names from a client, in actuality they all reference the same file (stored as README on the RTE-A file volume) and will be reported by the HP 1000 file server as a file named `readme`:

```
NFS_client> ls ReadMe
readme

NFS_client> ls README
readme

NFS_client> ls readME
readme
```

Care should be taken that you do not inadvertently overwrite an existing file due to RTE case insensitivity. For example, if you use the vi editor on the client to create an RTE file named “ReadME”, the file will actually be saved as “README” and will overwrite any other file of the same name (regardless of case) if it currently exists.

You can use the −c runstring option for the HP 1000 file server program (FSRV) to inhibit attempts to access any file names that contain any uppercase character. Such attempts will result in an error if the −c runstring option is in effect. (See Chapter 3, “File Server Options and Considerations” for more information on runstring options for the FSRV file server program.)

Character Mapping

A legal file name on RTE-A is composed of a maximum 16 characters for the file name and four characters for the file type extension. The name and type extension are separated by a period, for example,

```
filename.ext
```

Because the period “.” character is used as a separator character, it cannot be used in the file name or the file type extension itself. Additionally, the slash “/”, colon “:”, left bracket “[”, space “ ”, comma “,” and the greater than character “>” are also special characters that cannot be used in legal file names on RTE-A.

To allow you a greater range of legal file names from a client, the HP 1000 file server maps certain RTE special characters (see Table 2-1) when reporting or accessing files. All commas “,” in file names are converted to left parenthesis “(” by the HP 1000 file server. Colons “:” are converted to vertical lines “|” and periods “.” are converted to right parenthesis “)”. Periods that are in legal position for an RTE type extension are not converted.

Table 2-1. File Server Character Mapping

RTE Reserved Character	Mapped Character
,	(
:	|
.	)
/	not mapped
[	not mapped
<space>	not mapped
>	not mapped
Note: Periods in legal position for an RTE file extension are not mapped.	

For example, on the client:

```
NFS_client%  touch foo:foo
```

Create a file.

```
NFS_client%  ls foo:foo
foo:foo
```

When the file is listed on the client, it is listed as the created name.

On the server:

```
RTE_server> dl foo@
foo|foo
```

If the file is listed locally on the RTE-A system, it is listed as `foo|foo` due to the conversion of the “:” colon to a “|” vertical bar.

If character mapping is in effect, your existing RTE file names that use left or right parentheses "()" or vertical lines "|" will have a different name when accessed from a client. For example, an existing RTE file on the server named `)test` is mapped to the file name `.test`, making it a hidden file when accessed by the NFS client:

```
RTE_server> touch )foo

RTE_server> dl )foo
)foo

NFS_client% ls -a .foo
.foo
```

Also note that mapped characters used in file names created from a client are mapped to the corresponding RTE reserved character when accessed from a client. When listed locally on the HP 1000 file server, the RTE reserved characters are converted to the corresponding mapped character. For example,

```
NFS_client% touch foo\|foo
```
(Quote the vertical bar to indicate file named `foo|foo`.)

```
NFS_client% ls foo*
foo:foo
```
(File is listed on client as `foo:foo` due to character mapping)

```
RTE_server> dl foo|foo
foo|foo
```
(File is listed on server as `foo|foo` due to character mapping)

You can use the −m runstring option for the HP 1000 file server program (FSRV), to override character mapping. (See Chapter 3, "File Server Options and Considerations" for more information on runstring options for the FSRV file server program.) If you override character mapping, RTE reserved characters that would normally be mapped are no longer allowed in files created from a client. For example:

```
NFS_client%  touch foo:foo
touch : foo: cannot create
```

Directory Names

The names of directories created from a client cannot contain the period character (.) except in the first position. (Note that when character mapping is disabled, no periods are allowed in directory names.) The ".DIR" type extension is automatically appended to directory names when directories are referenced from a client. This leads to a potential conflict between, for example, the file "NAME" and a directory "NAME.DIR" when both reside in the same directory. From a client, both of these files are referenced as "name". Results in this case are unpredictable; therefore the HP 1000 file server does not support using files and directories with the same name.

Type 12 Files

The optimal file type for files created by a client is a UNIX-style byte stream file. Beginning with revision 6.2, RTE-A file type 12 specifies a byte stream file. Any RTE-A file that is created by a client, other than a directory or a symbolic link, is created as a type 12 byte stream file.

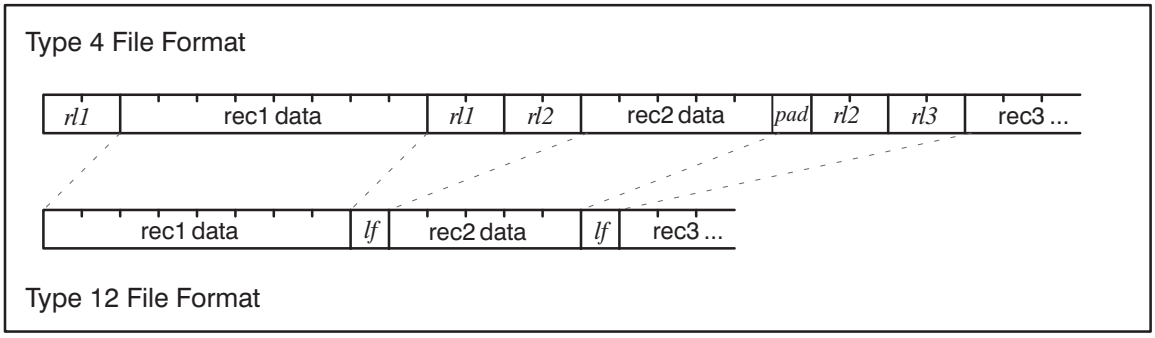
Caution Note that BOOTEX cannot read type 12 files. Take care that your boot command file does not get converted to a type 12 file. See Chapter 4, "File Server Installation" for a precautionary measure to take.

Translation to Type 12 File Format

All files other than type 4 files are transferred in binary mode when accessed from a client. You can use the −a runstring option in the HP 1000 file server program (FSRV) to specify additional RTE variable record length ASCII file types that should be translated to byte streams for file reads requested from a client. (See Chapter 3, "File Server Options and Considerations" for more information on runstring options for the FSRV file server program.)

When reading RTE variable record length ASCII files in ASCII mode, the HP 1000 file server strips out the preamble and postamble, indicated by "rl" (record length) in the figure below, and inserts a line feed (lf) after every record. This enables RTE ASCII files to look like UNIX ASCII files to the client. From a client application expecting an ASCII file, an RTE ASCII file looks like a byte stream file.

Figure 1-1. Translation to a Type 12 File Format



RTE ASCII files can also be edited on the client using vi and emacs. The HP 1000 file server does not support the writing of variable record length ASCII file, but you are able to edit them from a client due to how vi and emacs operate. These utilities read the entire file and then create new files when replacing files. The existing files are either renamed or removed. This causes variable length files to be converted to type 12 files if they are edited from a client.

To perform the variable record length ASCII translation, the file server must calculate the byte count of the data. When the status for an RTE ASCII file is requested for the first time, the file

server must count the number of bytes. If the file cannot be opened or read for any reason, the EOF position will be reported as zero bytes. To increase the performance, the file server stores a byte offset in word 32 of the RTE directory entry that can be used to calculate the number of bytes in the translated file to the RTE EOF position. Whenever an RTE ASCII file is modified on RTE, word 32 is cleared by D.RTR. This will cause a recalculation on the first status call from a client.

The first time an "ls" command is performed from the client on a mounted RTE directory, the file server will read every variable record length ASCII file in the directory for file types defined by the −a runstring option of the FSRV file server program (default: type 4 files). As a result, the initial access of a directory that contains files that result in a large amount of byte counting, will be relatively slow. Files will only need to be recounted if they are subsequently modified by an RTE utility. This is not required for type 12 files because they have their byte count in their directory entry.

Note that you should not translate files that contain new-line (line-feed) characters because records will be split at the new line when transferred back.

Note The HP 1000 file server does not support the writing of variable record length ASCII files.

Directory Entry for Type 12 Files

The directory entry for a type 12 file has the same definitions as all other RTE files with the exception of the ^EOF. Normally, the ^EOF is the number of words in a file. The ^EOF pointer in the directory entry for type 12 files is defined as the number of bytes in the file, rotated right by one bit. For type 12 files, the number of records and the record length fields are undefined.

Changing File Types

As of revision 6.2, the type 12 file type is a reserved file format type used when RTE-A files are created from a client. If you have a pre-existing user-defined file type 12, you must change these user-defined type 12 files to a different type. You can use the RN command from CI to change the type of a file. For example:

```
CI> rn filename:::12 filename:::4
```

$UFMP Library

The $UFMP library includes support of FMP with symbolic links and type 12 files. You will need to choose this library during the installation process if you plan to utilize the HP 1000 file server functionality. See the “Effect on Existing RTE-A Applications” section for more information on determining when your applications need to be linked with the $UFMP library.

FMP Code Growth Considerations

At revision 6.2, due to the code growth of the $UFMP library versus the $FMP library, some non-CDS applications will need modifications before they can be relinked. The level of modifications and effort depends upon the application.

For programs performing only reads and writes, the code growth is around 800 words. For programs that need to read, write, and position files the code growth is approximately 1000 words. For most CDS applications, this is not a problem. This may be a problem for non-CDS applications. Only CDS versions of HP-supplied utilities on RTE-A are supported with type 12 files.

Effect on Existing RTE-A Applications

The HP 1000 file server does not impose any restrictions to the client when reading an existing RTE-A file. Therefore applications that are only collecting data for export to a client do not need to be modified. Other types of applications may require modification or action as indicated in Table 2-2.

Table 2-2. Effect on Existing RTE-A Applications

Application Type	Action Required
Application that only collects data for export to a client.	*Action may not be required. May require relinking with $FMP or $SFMP if code growth is a problem (and application does not need to read type 12 files).*
Application that needs to read and write files that are written from a client.	*Relink application with the $UFMP library. (FMP code growth considerations apply.)*
Application that makes calls to FmpPosition or FmpSetPosition and needs to position type 12 files.	*Modify application to support the positioning of type 12 files; relink with $UFMP library. FMP code growth considerations apply. See Note below.*
Application that only needs to read files written from an NFS client on a very limited basis.	*Consider converting the files on the RTE-A system to type 4 files before the application accesses them. This is a last resort solution for any non-CDS application that requires access to data generated by an NFS client.*

Note See the appendix called "Converting Programs for Type 12 Support" in the *RTE-A Programmer's Reference Manual*, part number 92077-90007, for information on how to modify existing programs to position type 12 files.

3

File Server Options and Considerations

The system administrator should start up the HP 1000 file server according to the needs of users and with the special considerations described in Table 3-1 in mind. To start up the HP 1000 file server, the system administrator executes the program called FSRV with the desired options. Networking must already be initialized and enabled before starting up FSRV.

FSRV Runstring Options

Only one copy of FSRV is allowed to run at one time, and is typically scheduled in the welcome file. Runstring options are described below:

```
fsrv [-cemnuv] [-a type] [-d seconds] [-s blks] [-t timezone]
```

where,

`-a type` (ASCII conversion) Variable length RTE files of the *type* specified are transferred in ASCII mode. A line-feed character is added at the end of every record in the file. FSRV always converts type 4 files. Note that file types used for binary data should not be specified.

`-c` (Case folding) Inhibit the case folding of all file names. When you select this option, file names specified with any uppercase characters are rejected. (See Chapter 2 for more information on case reporting.)

`-d seconds` (Directory timeout) Use the *seconds* argument as the timeout for the server's directory buffer cache. This timeout specifies the maximum length of time that a directory buffer can remain in cache on the server. A timeout of "0" disables the directory buffer cache. (Default: 10 seconds)

`-e` (EOF cache) Disable EOF caching on the file server. By default, the file server caches the EOF information for files being written by a client. See the entry regarding ^EOF posting in "FSRV Special Considerations" (Table 3-1) for more information.

`-m` (File name mapping) Inhibit the reserved character mapping. When you specify this option, all of the RTE file name restrictions are in effect when accessing or creating files from a client. (See Chapter 2 for more information on character mapping.)

`-n` (Mount table file) Specifies not to use the mount table file to restore previous mount requests and not to update the mount table file after mount and umount requests from a client. (See the documentation for your client system for more information on the mount and umount commands.)

`-s` *`blks`* (Default file size) Use the *`blks`* argument as the default file size for files created from a client. Performance considerations apply when using this option. See "FSRV Special Considerations" for more information. (Default: 128 blocks)

`-t` *`timezone`* (Time Zone) Use the information in `/etc/tztab` for the time zone specified in the *`timezone`* argument. If the −t option is not used, the local RTE system is assumed to be running in UTC (Coordinated Universal Time).

`-u` (Update) The update mode causes the modification time for all directories to always be returned as the current time. See "FSRV Special Considerations" for more information.

`-v` (Verbose) The verbose mode is provided for diagnostic purposes only. Usage of this option is not recommended under normal conditions. Setting the break flag on the FSRV server program also toggles the state of the verbose flag. The verbose mode messages are written to the NS-ARPA/1000 or ARPA/1000 event logger (EVMON) as protocol-specific information (P bit, class 1). The LOGCHG utility can be used to change the log mask that EVMON uses. (For more information on the EVMON and LOGCHG utilities, see the *NS-ARPA/1000 Maintenance of Principles and Operation Manual*, part number 91790-90031, or the *ARPA/1000 Node Manager's Manual*, part number 98170-90001.)

Example

The example below starts up the FSRV server with the directory cache disabled. In addition to type 4 files, type 3 and type 32 files will be treated as ASCII files. The time zone is set to Pacific Standard Time, Pacific Daylight Time.

```
CI> fsrv -d 0 -a 3 -a 32 -t PST8PDT
```

FSRV Special Considerations

Table 3-1. FSRV Special Considerations

Subject	Comments
Casefolding	FSRV case folds all file names referenced from a client.
Changing File Ownership	The RTE file system only supports ownership of directories and not files. Files are owned by the owner of the directory in which they reside. Attempts to change the ownership of a file from a client are ignored.
Directory Caching	FSRV keeps a cache of RTE directory tracks. The cache is only used for reading directory information. By default, a directory track can remain in the cache for a maximum of 10 seconds. This timeout can be changed with the −d runstring option. (See “Directory Cache Timeout” below for more information on the effects of changing the FSRV directory cache timeout.)
Directory Cache Timeout	The −d option can be used to set the timeout for the directory track cache. For systems whose exported RTE directories are simultaneously being used by local RTE applications, you may want to decrease or eliminate the timeout for the cache. This will cause a decrease in performance, but, it will also decrease the probability of obtaining stale directory information on the client.
^EOF Posting	By default, FSRV does not post the ^EOF pointer to the disk each time a write request is serviced. The ^EOF pointer is posted to the disk when the file being written is closed or after a request to read the same file is serviced. This is done for performance reasons; however, doing this means that FSRV is no longer stateless. If anything causes FSRV to abort before the ^EOF is posted, the file will be corrupted. The −e runstring option causes the ^EOF pointer to be updated on the disk each time a write request extends a file.
File Access Mode	The access mode of a file on an NFS client contains bits that have no meaning on the RTE file system. The bits designating character special files, block special files, named sockets, “set user id”, “set group id”, and “save swapped text after use” can never be set on an RTE file from an NFS client. The execute bits can be set and used by a client but they have no meaning on the local RTE file system. These bits are for use by NFS clients only.
Performance	All files created by FSRV are created as type 12 (byte stream) files. The default file creation size is 128 blocks. This value can be changed with the −s runstring option. If the default file size is too small, the performance of the FSRV monitor will be degraded significantly due to the file extent creation and access.
RTE Directory Specification	When specified by a client, RTE directories cannot contain a period ‘.’ in the name.

Table 3-1. FSRV Special Considerations - continued

Subject	Comments
RTE File Access	Files are accessed by FSRV in a shared open mode. FSRV cannot access files that are already open exclusively to other RTE processes.
Type 12 Byte Stream Files	All files created by FSRV are created as RTE type 12 (byte stream) files. The FSRV server program does not support the writing of variable length ASCII files. Variable length ASCII files can be read from a client. When data is read from a variable length ASCII file from a client, the FSRV server program translates the data into an ASCII byte stream by inserting line feeds after each record.
UNIX Hard Links	Hard links cannot be made on RTE systems. Attempts to create links from a client result in errors. Note that some UNIX applications cannot be used with a mounted RTE disk because of this. For example, RCS on HP-UX uses hard links.

For a listing of error messages returned by the FSRV file server program, please refer to Chapter 5.

4

Installing the File Server

This chapter covers the installation, configuration, and initialization of the HP 1000 file server. File server functionality should be installed by the system or network administrator.

Requirements

The HP 1000 file server software is shipped as part of the NS-ARPA/1000 (HP 91790A) and ARPA/1000 (HP 98170A) products (revision 6.2 or greater), but is configured separately from these products. Prior to configuring your HP 1000 node as a file server, the following must be installed and operational:

- RTE-A (HP 92077A) with VC+ (HP 92078A), revision 6.2 or greater.
- $UFMP, the FMP library that supports type 12 files.
- NS-ARPA/1000 (HP 91790A) or ARPA/1000 (HP 98170A), revision 6.2 or greater.
- LAN/1000 (HP 12076A) Link Interface

Installation

The HP 1000 file server software is installed when you install the NS-ARPA/1000 or ARPA/1000 products. Files specific to the file server are installed in the directories listed below.

/PROGRAMS/	**/HELP/**	**/ETC/**
FSRV.RUN	EXPORTS	TZTAB
RDATE.RUN	FSRV	
	RDATE	

These files will be placed in the directories indicated above if you install the file server using the installation command file for NS-ARPA/1000, `INSTALL_NS1000.CMD`, or the installation command file for ARPA/1000, `INSTALL_ARPA.CMD`. (The `INSTALL_NS1000.CMD` and `INSTALL_ARPA.CMD` installation command files, can be called automatically by the general RTE-A and subsystem installation command file, `RTE_INSTALL.CMD`. See the *RTE-A System Generation and Installation Manual*, part number 92077-90034, for more information on the `RTE_INSTALL.CMD` installation process.)

In addition, the file server expects the following files in the /ETC directory. You will need to create these files for your environment and place them in the /ETC directory:

```
/ETC/
  EXPORTS
  UX_USERS
  UX_GROUPS
  HOSTS
```

Configuration and Initialization Files

The HP 1000 file server uses the files described below.

/etc/exports	Contains the export information for directories that can be exported to NFS clients.
/etc/hosts	Contains the association of Internet addresses to official host names and aliases. All NFS clients specified in the /etc/exports file must also have an entry in the /etc/hosts file.
/etc/ux_users	Contains user access identification information.
/etc/ux_groups	Contains group access identification information.
/etc/fsrv.mnt	File created or read by the file server program (FSRV) at initialization to restore its internal mount tables.
/etc/tztab	Describes the differences between UTC (Coordinated Universal Time) and local time.

/etc/exports

The file /etc/exports contains the export information for directories that can be exported to NFS clients. When a mount request is received from a client, the export information for the directory specified is obtained from this file. If the file server cannot access this file for any reason when a remote mount request is received, the mount request is rejected. All client names referred to in this file must also have an entry in the /etc/hosts file.

An entry in /etc/exports for a directory consists of a line in the following form:

directory -*option*[,*option*]...

where *directory* is the path name of a directory and *options* can have any of the following values and forms:

ro	Export the directory read-only. If not specified, the directory is exported read-write.

`rw=`*`hostname`*`[:`*`hostname`*`]`

Export the directory read-mostly. Read-mostly means read-only to most machines, but read-write to those specified. If not specified, the directory is exported read-write to all.

`anon=`*`uid`*

If a request comes from an unknown user, use *`uid`* as the effective user ID. Note: Root users (*`uid`* of 0) are always considered "unknown" by the file server unless they are included in the `root` option below. The anonymous user must be a valid member of NOGROUP. (If the anonymous *`uid`* does not correspond to a valid RTE user, the anonymous user will have the same access rights as a non-owner, non-superuser, but, no directories or files may be owned by the anonymous user.) Setting `anon=0` or `anon=3` allows anonymous users to have superuser access.

The default value for this option is 32767. Setting `anon` to a value greater than 32767 disables anonymous access.

`root=`*`hostname`*`[:`*`hostname`*`]`

Give manager access only to the root users from a specified *`hostname`*. The default is for no hosts to be granted manager access.

`access=`*`client`*`[:`*`client`*`]`

Give mount access to each *`client`* listed. A *`client`* can be a *`hostname`*. A directory name with no accompanying name list allows any machine to mount the given directory.

If no *`options`* other than `access` will be specified, the entry in the `/etc/exports` file may also be in the following form:

```
directory   client client
```

`#`

A `#` character anywhere in the file indicates a comment that extends to the end of the line.

Note Changing the information in `/etc/exports` does not immediately change any of the attributes for currently mounted directories. The client must issue a umount and mount request to take on the new attributes defined in this file; the `/etc/exports` file is only read when a mount request is received.

In the `/etc/exports` file, the file system names are left-justified and followed by a list of client names separated by spaces. The client names are searched for in `/etc/hosts`. A file system name with no accompanying name list means the file system is available to everyone.

The root directory on an RTE disk volume can be exported by specifying the exported directory name as `/.lu`. For example, `/.16` refers to the root directory on the file system volume on LU 16. FMGR cartridges cannot be exported.

Note that a root directory can only contain entries for global directories. Regular files cannot be created on root directories.

The following are example entries in an `/etc/exports` file:

```
/.16            vending washing             # export lu 16 to only these machines
/user           -access=clients             # export to my clients
/scratch                                    # export to everyone
/usr2           -access=bucko:snardra       # export to only these machines
/system         -root=canal:beer            # root access only to these machines
/usr/new        -anon=0                     # give all machines root access
/usr/temp       -rw=ram:zebra               # export read-write only to these
/system         -ro                         # export read-only to everyone
/usr/stuff      -access=bear,anon=12,ro     # several options on one line
```

Given the above example `/etc/exports` file, to mount LU 16 for file server access, the NFS client would create a mount point and issue the mount command to mount the file system LU:

```
client_superuser% mkdir /LU16                                  (Create mount point)
client_superuser% mount RTE_servername:/.16  /LU16             (Mount LU 16 volume)
```

(See the documentation for your client system for more information on creating mount points and the mount command.)

Note You cannot export either a parent directory or a subdirectory of an exported directory that resides within the same disk volume. For instance, you should not export both /usr and /usr/local if both directories reside on the same disk volume.

/etc/hosts

For every NFS client specified in `/etc/exports` there must be a corresponding entry for the client node in `/etc/hosts`. The `/etc/hosts` file contains a list of other nodes in the network with which your local system can communicate. Each entry in the file must be in the following format:

```
IP_address host [aliases]
```

where:

IP_address The IP address that uniquely identifies the node. *IP_address* must be in Internet "dot" notation, for example, `192.6.1.1`. Refer to the *NS-ARPA/1000 User/Programmer Reference Manual*, part number 91790-90020, or the *ARPA/1000 User's Manual*, part number 98170-90002, for more information on IP Addresses.

host Name of the host. Host name can contain any printable character except space, new-line, or the comment character (`#`). *Naming Convention*: The first nine characters should be unique for each network host.

aliases Common name or names for the node. An *alias* is a substitute for *host*. *Aliases* are optional and are separated by spaces. *Naming Convention*: The first nine characters should be unique for each network host.

For example:

```
15.10.56.1 system1 labdev           # host1 and its alias
```

The /etc/hosts file is also used by the BSD IPC services. For more information on how /etc/hosts is used with the BSD IPC services, refer to Appendix B of the *BSD IPC Reference Manual for NS-ARPA/1000 and ARPA/1000*, part number 91790-90060.

/etc/ux_users

Every file access request has a UID (User Identification) associated with it. The HP 1000 file server adopts the capability of the RTE user associated with the UID. The /etc/ux_users file contains the mappings of RTE user names to client UIDs. The HP 1000 file server program, FSRV, reads the /etc/ux_users file upon startup of FSRV.

Entries in the /etc/ux_users file are in the following format:

```
rte_username client_uid
```

where:

rte_username Valid RTE user logon name.

client_uid Valid client user identification number (UID) for the corresponding *rte_username*. The *client_uid* is the UID field from the /etc/passwd file on the client.

Example entries:

```
garyg 492
marie 236
brian 638
```

Requests from an NFS client that contain a UID that is not found in the /etc/ux_users file are considered anonymous. The access rights for anonymous users are defined in the /etc/exports file.

Note If new entries or changes are made to the `/etc/ux_users` file, you must restart the FSRV file server program for the changes to take effect. (See the "Restarting FSRV" section later in this chapter.)

/etc/ux_groups

The FSRV file server program also reads the `/etc/ux_groups` file upon startup. This file contains the mappings of RTE group names to client group identification numbers (GIDs) in the following format:

```
rte_groupname client_gid
```

where:

rte_groupname Valid RTE group name.

client_gid Valid client group identification number (GID) for the corresponding *rte_groupname*. The *client_gid* is the GID field from the `/etc/group` file on the client.

Example entries:

```
lab 23
support 32
field 34
admin 100
```

Note If new entries or changes are made to the `/etc/ux_group` file, you must restart the FSRV file server program for the changes to take effect. (See the "Restarting FSRV" section later in this chapter.)

/etc/fsrv.mnt

At initialization, FSRV reads `/etc/fsrv.mnt` to restore its internal mount table to its previous state. This file is created if it does not exist. After every successful mount or umount client request is processed, FSRV updates this file. If this functionality is disabled with the FSRV −n option, or if the file is purged before FSRV is restarted, clients that mounted a directory prior to a restart will need to issue another mount request.

/etc/tztab

The `/etc/tztab` file describes the differences between Coordinated Universal Time (UTC) and local time. Several local areas can be represented simultaneously with historical detail.

The `/etc/tztab` file consists of one or more time zone adjustment entries. The first line of the entry contains a unique string that is matched with the value of the string provided to FSRV with the −t option.

The format of the first line of the entry is

```
tznamediffdstzname
```

where,

tzname	Time zone name or abbreviation. The *tzname* must begin with an alphabetic character.
diff	Difference in hours from UTC. Fractional values of *diff* are expressed in minutes preceded by a colon.
dstzname	Name or abbreviation of the “Daylight Savings” time zone.

The second and subsequent lines of each entry detail the time zone adjustments for that time zone. The lines contain seven fields each. The fields are separated by spaces or tabs. The first six fields specify the first minute in which the time zone adjustment, specified in the seventh field, applies.

```
field1 field2 field3 field4 field5 field6 field7
```

The first six fields are integer patterns that specify:

field1	minute (0-59)
field2	hour (0-23)
field3	day of the month (1-31)
field4	month of the year (1-12)
field5	year (1970-2038)
field6	day of the week (0-6, with 0=Sunday)

The minute, hour, and month of the year must contain a number in the (respective) range indicated above. The day of the month, year, and day of the week can contain a number as above or two numbers separated by a minus (indicating an inclusive range). Either the day of the month or the day of the week field must be a range, the other must be a simple number.

The seventh field is a string that describes the time zone adjustment in its simplest form.

field7	*tznamediff*

where,

tzname — An alphabetic string giving the time zone name or abbreviation. *tzname* must match either the `tzname` field or the `dstzname` field in the first line of the time zone adjustment entry.

diff — The difference in hours from UTC. Any fractional *diff* is shown in minutes preceded by a colon.

Comments begin with a # in the first column, and include all characters up to a new-line. Comments are ignored.

Examples

The time zone adjustment table for the Eastern Time Zone in the United States is:

```
EST5EDT
0 3 6     1  1974      0-6 EDT4
0 3 22-28 2  1975      0   EDT4
0 3 24-30 4  1976-1986 0   EDT4
0 3 1-7   4  1987-2038 0   EDT4
0 1 24-30 11 1974      0   EST5
0 1 25-31 10 1975-2038 0   EST5
```

Normally (as indicated in the first line) Eastern Standard Time is five hours earlier than UTC. During Daylight Savings time, it changes to a 4-hour difference. The first time Daylight Savings Time took effect (second line) was on January 6, 1974 at 3:00 a.m., EDT. Note that the minute before was 1:59 a.m., EST. The change back to standard time took effect (sixth line) on the last Sunday in November of the same year. At that point, the time went from 1:59 a.m., EDT to 1:00 a.m., EST. The transition to Daylight Savings Time since then has gone from the last Sunday in February (third line) to the last Sunday in April (fourth line) to the first Sunday in April (fifth line). The return to standard time for the same period has remained at the last Sunday in October (seventh line).

The table for the Central Time Zone in Australia shows adjustments where the difference from UTC includes a fractional part of an hour.

```
CST-9:30CDT
0 3 25-31 10 1971-2038 0   CDT-10:30
0 1 27    2  1972      0-6 CST-9:30
0 1 1-7   3  1973-2038 0   CST-9:30
```

Starting Up FSRV

Prior to starting the FSRV file server program, the RDATE utility should be called if you need to synchronize the system time with that of a remote client system. RDATE sets the system time via communication with a remote system; its use helps to maintain consistent file access times between systems. The SYSTZ utility must be called prior to running RDATE to set the system time zone offset and daylight savings time flag ($SYSTZ) used by RDATE. (For more information on the RDATE utility, see Appendix A. For more information on the SYSTZ utility, see the *RTE-A User's Manual*, part number 92077-90002.)

FSRV obtains its time zone information from the `/etc/tztab` file when you run it with the −t option.

Following is an example of the steps to start up the FSRV file server:

`SYSTZ 8`	Sets System Time for Pacific Standard Time; offset from Coordinated Universal Time is +08:00 (hours:minutes). Daylight Savings Time is not in effect. Sets $SYSTZ for use by the RDATE utility.
`RDATE timemaster`	Sets system time from remote system named timemaster. Remote host can also be specified by IP address.
`FSRV -d 0 -a 3 -c -t PST8PDT`	Starts up the FSRV file server with the directory cache disabled. In addition to type 4 files, type 3 files will be treated as ASCII files. Inhibit casefolding option is on causing file names specified with any uppercase characters to be rejected. In this example, FSRV will match the "PST8PDT" string entered with the −t option to the corresponding entry in the `/etc/tztab` file for Pacific Standard Time, Pacific Daylight Time (PST8PDT) time zone information. For more information on the FSRV runstring options, see Chapter 3, "Runstring Options and Considerations".

If you want the FSRV file server program to automatically execute at boot up, customize the above commands for your environment and place them in your welcome file.

Caution Do not MPACK a disk while it has directories mounted to a client. Doing so may render FSRV unable to locate accurate file handle information.

Verifying FSRV Startup

Once you have started FSRV, you can quickly verify that it is running by using the program status (PS) command on the RTE-A system: For example:

```
RTE_A> ps fsrv
FSRV  PR(    50) PC(27671) M   RN( -GLOB )
```

If the program status command indicates FSRV is not running, you can check the end of EVMON's event log file to help determine the cause of the FSRV startup failure. (For more information on the EVMON utility, see the *NS-ARPA/1000 Maintenance of Principles and Operation Manual*, part number 91790-90031, or the *ARPA/1000 Node Manager's Manual*, part number 98170-90001.)

Restarting FSRV

To restart the FSRV file server program, first make sure that no clients are currently using the server. You can then terminate FSRV with the command:

```
OF,FSRV,ID
```

After FSRV has successfully terminated, it may be restarted. When you restart FSRV, changes to the `/etc/ux_group`, `/etc/ux_users`, and `/etc/exports` files will take effect.

Evaluating User Applications

In some cases RTE user applications that will access files created by the FSRV file server will need to be modified and/or relinked. (See Chapter 2, section called "Effect on RTE-A User Applications" to determine any impact to your applications.)

Verifying Client Access

After starting the FSRV file server program on the HP 1000, you can validate that it is operational from a client by running the `rpcinfo` command on the client system. The example below queries the HP 1000 system and returns information on the ports and over which protocols the HP 1000 file server can receive requests from the client.

```
NFS_client > /usr/etc/rpcinfo -p rtemachine
   program vers proto   port
    100005    1   udp  33118  mountd
    100005    1   tcp  33117  mountd
    100003    2   udp   2049  nfs
```

You can also run `rpcinfo` on the client to test the loopback functions of the various services using the UDP or TCP port numbers returned in the original query. For example:

```
NFS_client > /usr/etc/rpcinfo -n 2049 -u rtemachine nfs 2
program 100003 version 2 ready and waiting

NFS_client > /usr/etc/rpcinfo -n 33117 -t rtemachine mountd 1
program 100005 version 1 ready and waiting

NFS_client > /usr/etc/rpcinfo -n 33118 -u rtemachine mountd 1
program 100005 version 1 ready and waiting
```

Successful `rpcinfo` loopbacks also verify that the network is operational between the server and the client.

Verify that the HP 1000 file server is operational by mounting an RTE file system from a client and accessing an RTE mounted file from a client.

Changing the Boot Command File Type

Caution BOOTEX uses a non-CDS D.RTR and cannot read type 12 files. Make sure that your boot command file does not get converted to a type 12 file. As a precautionary measure, it is recommended that you change your boot command file to a file type other than type 4 (or any other type specified in the −a runstring option for FSRV) to avoid inadvertently editing and saving your boot command file from a client as a type 12 file. You can use the RN command to change a file's type. For example:

```
RN BOOT.CMD BOOT.CMD:::33
```

File Server Summary Checklist

Prior to configuring the HP 1000 file server, the following should be installed and operational:

- Revision 6.2 or greater, RTE-A/VC+ with $UFMP library.
- Revision 6.2 or greater, NS-ARPA/1000, or ARPA/1000.

Continue with file server installation, configuration, and initialization:

- Verify file server files exist in the appropriate directories.
- Create or edit `/etc/hosts` as needed for clients that will access the file server.
- Create `/etc/exports`.
- Create `/etc/ux_groups`.
- Create `/etc/ux_users`.
- Run SYSTZ prior to running RDATE to set $SYSTZ, the system's time zone offset and daylight savings time flag (if necessary).
- Run RDATE to set the system time from a remote client system (if necessary).
- Start up the FSRV file server program with options desired for your operating environment.
- Verify the program status of thr FSRV program to be sure it is running.
- Add SYSTZ and RDATE (if necessary), and FSRV runstrings to your Welcome file if you desire automatic execution at boot up in the future.
- Determine if user application programs that will use type 12 files created by the file server need to be modified.

On NFS clients that will be accessing the HP 1000 file server:

- Verify that the HP 1000 file server is operational by use of the `rpcinfo` command, successfully mounting an RTE file system volume, and accessing an RTE mounted file from the client.

5

Troubleshooting

This chapter describes the types of client requests the HP 1000 file server can handle, guidelines for troubleshooting, and error messages that may be returned. Problems related to power up and connectivity to the network are not addressed.

Client Requests

The HP 1000 file server consists of one main program called FSRV that responds to requests from an NFS client. Requests from clients can be of the types specified in the following table.

Table 5-1. Client Request Types

Client Request Type	Server Response
portmapper request (1)	The file server will return to the client the protocols and port numbers on which the server will listen for future mount and file system access requests from the client. A portmapper request is generated by a mount request.
mount request (1)	The file server will return to the client the file handle that allows file access, after validating that access is allowed to the requesting client.
file system request (1)	The file server responds by performing the requested local file operation, and returning the results back to the client.

(1) The HP 1000 file server can service these types of requests *only* if the NFS client is using the following protocol versions:

Port Mapper Service – Version 2
Mount Service – Version 1
NFS Service – Version 2

The HP 1000 file server cannot respond to certain procedures/services requested by an NFS client. These are listed in Table 5-2.

Table 5-2. Unserviceable Client Requests

Procedure/Service Type	Description	Comment or Impact
NFSPROC_LINK	File system access request to create a hard link to a file.	RTE-A does not support links to a file.
MOUNTPROC_DUMP	Mount request to return the server's mount entries.	Not implemented.
PMAPPROC_SET	Portmapper request to register an RPC program with the portmapper.	Not implemented. Users cannot register their own programs.
PMAPPROC_UNSET	Portmapper request to unregister an RPC program with the portmapper.	Not implemented. Users cannot register their own programs.
PMAPPROC_CALLIT	Remote Procedure Call of any registered RPC program.	Not implemented. Users cannot register their own programs.

Troubleshooting Guidelines

From an NFS client perspective, the most basic problems that could be encountered when attempting to access the HP 1000 file server are: a mount request fails, the server does not respond to a file system request, file access is restricted, or programs hang.

This section provides troubleshooting guidelines if you experience these problems between the HP 1000 file server and an NFS client. Table 5-3 shows possible causes and solutions for these conditions. If the NFS client is an HP 9000 system, for additional details, also see the "Troubleshooting " chapter of the *HP 9000 Installing and Administering NFS Services* manual.

Table 5-3. Troubleshooting Problem Conditions Experienced by Client

Condition	Explanation / Recommended Action
Access is restricted.	*User is not validated by the /etc/ux_users and /etc/ux_groups files.* *Access is restricted in the /etc/exports file.*
Cannot access/write files if the owner of the directory is in the group NOGROUP.	*Users from an NFS client may not have group access on directories owned by a member of NOGROUP on RTE. The group NOGROUP on RTE is handled in a special way. The group protection bits are ignored for RTE users with the group NOGROUP.* *RTE will not display the status of the group bits if the owner of the directory is in the group NOGROUP. However, the protections are still reported (and used) by the NFS client.* *The group permissions can still be set by the PROT command on RTE, but, PROT will not show you the settings of the group permissions (nor will the LL command on RTE).* *Use the PROT command to set the group bits to the desired value. Examination of the group protection bits can be done from the NFS client.* *For more information about the group NOGROUP, see the discussion on "protection displayed" under the DL command, and information on groups under the OWNER command in the RTE-A User's Manual, part number 92077-90002.*

Table 5-3. Troubleshooting Problem Conditions Experienced by Client (continued)

Condition	Explanation / Recommended Action
Mount request to server fails.	*Server may be down; check that the FSRV server is running.* *Check that client is listed in server's /etc/export and /etc/hosts files.*
Mount table /etc/fsrv.mnt cannot be stored or restored.	*FSRV does not have enough free space to store or restore the mount table.* *Increase the HEAP area of FSRV.* *If the HEAP area cannot be increased, reduce the size of the /etc/tztab file by eliminating the tztab rule in your time zone for future years and/or past years.*
Multiple client retries on a mount request.	*Adjust timeout specified in the mount request.*
Programs hang.	*Server may be down; check that the FSRV server is running.*
Server file is busy; cannot open file.	*File is already exclusively open to another user.*
Server is not responding.	*Server may be down; check that the FSRV server is running.* *Check that client is listed in server's /etc/export and /etc/hosts files.* *Check that the server's address is correct in the client's /etc/hosts file.* *Adjust timeout specified in the mount request.*
Time zone table /etc/tztab cannot be restored.	*FSRV does not have enough free space to store the time zone table.* *Increase the HEAP area of FSRV.* *If the HEAP area cannot be increased, reduce the size of the /etc/tztab file by eliminating the tztab rule in your time zone for future years and/or past years.*

RTE in the File Server Environment

The following table describes information that should be kept in mind when using RTE in the file server environment, and may be useful in problem prevention and troubleshooting:

Table 5-4. RTE in the File Server Environment

Subject	Comments / Caveats
BOOTEX cannot read type 12 files.	*BOOTEX uses a non-CDS D.RTR and does not read type 12 files. Make sure that the boot command file does not get converted to a type 12 file. As a precautionary measure it is recommended that you change the BOOT.CMD file type to a file type that was not included in the FSRV runstring (for example, 24).*
DS Transparency constraint.	*Files created from an NFS client (type 12 files) cannot be read over DS transparency from systems that do not support type 12 files. This includes all RTE-A revisions prior to 6.2.* *Note that if you will be using DS transparency from your local system to read type 12 files on another system, your local system must have been built with the appropriate FMP libraries necessary for type 12 support – even if you will not be using the file server on the local system.*
FTP transfer of type 12 ASCII files to RTE-A systems that do not support type 12 files.	*If the transfer type is set to ASCII, the file is translated from type 12 to "FTP" ASCII by the FTP server. The FTP client translates the "FTP" ASCII to a type 4 file.* *Note that ASCII mode is only required if the destination system is an RTE-A system that does not support type 12 files.*
RTE does not allow the time stamps of root directories to be modified.	*Attempts to change the time stamps of root directories are always replied as successful even though RTE does not allow this function.*
RTE does not automatically update the time stamps for directories.	*Since HP-UX uses the time stamps to determine if a directory, that is already in its cache, needs to be re-read, sometimes, the client may not notice that a file has been added from the server side until the directory cache (on the client) times out. By default, on HP-UX, the attribute cache is held between 30 and 60 seconds.*

Table 5-4. RTE in the File Server Environment (continued)

Subject	Comments / Caveats
RTE does not have execute permission bits.	*The execute permission bits that are used from a client are not accessible from any RTE application. The bits are stored in a file's directory entry and they can only be set or displayed from the NFS client.*
RTE only has ownership of directories and not files.	*Attempts to change the ownership of files are always replied as successful even though RTE does not have ownership information for normal files.*
Symbolic links made from an NFS client may refer to file names that are illegal from RTE.	*The name stored in a symbolic link is not subject to the file name translation process. When a client tries to make a symbolic link, the server cannot assume that the contents of the link refer to a file that is local to the server (for example, "../../homedir/.profile").*
TF does not back up type 12 files.	*TF will not back up type 12 files; FST must be used.*

FSRV Error Messages

The following error messages are returned by the FSRV file server program. An explanation of the error and suggested action are indicated as appropriate. When these errors occur they are reported to the event logger for NS-ARPA/1000 or ARPA/1000. Event logging is enabled with the EVMON utility. (For more information on the EVMON utility, see the *NS-ARPA/1000 Maintenance of Principles and Operation Manual*, part number 91790-90031, or the *ARPA/1000 Node Manager's Manual*, part number 98170-90001.)

The actual error message returned to the NFS client under these conditions will depend upon the error handling by the client.

fsrv: atach error : <#>

FSRV was unable to attach to the new session. See the ATACH utility information in the *RTE-A Relocatables Reference Manual*, part number 92077-90037, for an explanation of error numbers returned by ATACH.

fsrv: Cannot find the time zone information for <TIMEZONE>

The time zone specified in the −t runstring option could not be found in the `/etc/tztab` file.

fsrv: clgon error : <#>

FSRV was unable to logon. (Requires LOGON Rev.6200.) See the CLGON utility information in the *RTE-A Relocatables Reference Manual*, part number 92077-90037, for an explanation of error numbers returned by CLGON.

fsrv: Error decoding /etc/tztab.

The `/etc/tztab` file cannot be decoded. Verify that the file entries are in the correct format.

fsrv: Invalid file type specified with −a option.

The −a option can only be used with variable record length file types.

fsrv: MOUNT procedure: DUMP is not implemented.

MOUNTPROC_DUMP procedure to return server's mount entries is not implemented.

fsrv: Multiuser session not enabled.

Multiuser session must be enabled FSRV to operate.

fsrv: NFS procedure: LINK is not implemented.

NFSPROC_LINK procedure to create a link to a file is not implemented; links to a file are not supported on RTE-A.

fsrv: No more room in mount table.

There is insufficient free space in the program to store the mount table. Increase the HEAP area of the FSRV program. If the HEAP area cannot be increased, the size of the time zone table may possibly be reduced. This can be achieved by eliminating the tztab rule in your time zone for future years and/or past years.

fsrv: Not enough room to restore the mount table from
fsrv: /etc/fsrv.mnt. Only mounting <#> directories.

There is insufficient free space in the program to read the prior mount table. Increase the HEAP area of the FSRV program. If the HEAP area cannot be increased, the size of the time zone table may possibly be reduced. This can be achieved by eliminating the tztab rule in your time zone for future years and/or past years.

fsrv: Not enough room to store the time zone table.

There is insufficient free space in the FSRV program to store the time zone table. The time zone table is stored in the HEAP area of the FSRV program. If the HEAP area cannot be increased, the size of the time zone table must be decreased. This can be achieved by eliminating the tztab rule in your time zone for future years and/or past years.

fsrv: PORTMAP procedure: CALLIT is not implemented.

PMAPPROC_CALLIT procedure to allow remote procedure (RPC) calls to any registered RPC program is not implemented.

fsrv: PORTMAP procedure: SET is not implemented.

PMAPPROC_SET procedure to allow users to register an RPC program with the portmapper is not implemented. RTE users are unable to register their own applications with FSRV.

fsrv: PORTMAP procedure: UNSET is not implemented.

PMAPPROC_UNSET procedure to allow users to unregister an RPC program with the portmapper is not implemented. RTE users are unable to register their own applications with FSRV.

fsrv: retry request from <000.000.000.000>
fsrv: prog=<PROG NAME> proc=<PROC NAME>

Client's request timed out. FSRV detected the retry.

fsrv: RPC authentication error.

Only the AUTH_UNIX and AUTH_NONE authentication protocols are supported.

fsrv: RPC version mismatch.

Only RPC version 2 is supported.

fsrv: Syntax error in /etc/hosts.

Review the entries in the `/etc/hosts` file for syntax errors.

fsrv: Unable to get a session number.

GETSN failed; the system is out of session numbers.

fsrv: Unable to obtain RTE group ID for <GROUPNAME>

A group name was found in the `/etc/ux_groups` file that is not a valid group on the RTE system.

fsrv: Unable to obtain RTE user ID for <USERNAME>

A user name was found in the `/etc/ux_users` file that is not a valid user on the RTE system.

fsrv: Unknown host <HOSTNAME>

A `HOSTNAME` specified in `/etc/exports` could not be found in the `/etc/hosts` file.

fsrv: Unknown option in /etc/exports.

Review the entries in the `/etc/exports` file for syntax errors.

BSD IPC Errors Returned by FSRV

The following errors are reported by FSRV when any of the networking calls used by FSRV report an error. See the *BSD IPC Reference Manual for NS-ARPA/1000 and ARPA/1000*, part number 91790-90060, for information on the specific BSD IPC error message number that is returned by FSRV.

fsrv:socket error : <#>

fsrv:bind error : <#>

fsrv:setsockopt error : <#>

fsrv:getsockname error : <#>

fsrv:listen error : <#>

fsrv:select error : <#>

fsrv:accept error : <#>

fsrv:recv error : <#>

fsrv:recvfrom error : <#>

fsrv:send error : <#>

fsrv:sendto error : <#>

fsrv:shutdown error : <#>

fsrv:getpeername error : <#>

A

Setting the Local Time

The RDATE utility sets the local RTE system time by obtaining it from a remote system. The RDATE utility can be used to set the time on the HP 1000 to that of the NFS client. The $SYSTZ value in the system must already be set prior to using RDATE. The $SYSTZ value contains the local system's time zone offset and is set by the SYSTZ utility. (See the *RTE-A User's Manual*, part number 92077-90002, for more information on the SYSTZ utility.)

If the RDATE utility is used it is typically called in the Welcome file after SYSTZ runs, but before the FSRV file server is scheduled. In order to keep your system time accurate over time, you may want to run RDATE hourly using cron. (See the *RTE-A User's Manual*, part number 92077-90002, for more information on the cron utility.) The syntax for RDATE is as follows:

```
RDATE [-q] host
```

where:

`-q`	Quiet option; inhibits all messages.
`host`	Specifies the remote system by IP address or the host name. If the host name is specified, the host's IP address must be specified in the `/etc/hosts` file.

Return values:

$return1	: 0	system time was successfully set.
	: 1	$SYSTZ has not been set on the local system.
	: 2	host name not found in `/etc/hosts`.
	: 3	could not connect to remote system.
	: 4	host returned an unreasonable time value.
	: 5	could not set the system time.

When RDATE is successful, the following values are also returned:

$return2	:	hour
$return3	:	minute
$return4	:	second
$return5	:	day of the year
$return_s	:	13 character string in the form: *`yymmdd.hhmmss`*

Example 1: Set date from remote host named keymaster.

```
CI> rdate keymaster
Fri Jul 1, 1994  2:12:04 pm
CI> echo $return1,$return2,$return3,$return4,$return5,$return_s
0,14,12,4,182,940701.141204
```

Example 2: Run RDATE with quiet option using remote host IP address.

```
CI> rdate -q 15.0.255.25
CI> echo $return1,$return2,$return3,$return4,$return5,$return_s
0,14,12,5,182,940701.141205
```

Glossary

anonymous user

A user unknown to RTE. The effective user id becomes that assigned by the `anon` option in the `/etc/exports` file.

client

Generically speaking, a system or program that requests a service from a server system or program. *See also* NFS client.

/etc/exports

File that describes the directories that can be exported to NFS clients.

/etc/fsrv.mnt

File read at initialization by the FSRV server to restore its internal mount table to its previous state.

/etc/hosts

File that associates Internet addresses with official host names and aliases. All NFS client names indicated in the `/etc/exports` file must also have an entry in the `/etc/hosts` file.

/etc/tztab

File that describes the differences between Coordinated Universal Time (UTC) and local time.

/etc/ux_groups

File read at initialization that describes valid client group identification (gid) to RTE group mappings.

/etc/ux_users

File read at initialization that describes valid client user identification (uid) to RTE user mappings.

export

To make a file system available to a remote NFS client node.

file server

See HP 1000 file server and FSRV.

file system

A organized directory structure of files.

FSRV

HP 1000 file server program that services NFS client requests. *See also* HP 1000 file server.

HP 1000 file server

Also referred to as file server in this documentation. Refers to the functionality residing on the HP 1000 that allows NFS client nodes to access HP 1000 RTE-A files. *See also* FSRV.

import

To obtain access to a remote file system.

IP address

Internet Protocol address.

mount

Command issued by an NFS client to obtain access to (or to import) a remote file system or directory.

mount point

The name of the directory on the client to which a file system is mounted.

NFS

The Network File System (NFS) is a protocol developed by Sun Microsystems that allows nodes in a computer network to access files at other nodes. (NFS is a trademark of Sun Microsystems, Inc.)

NFS client

A system (other than an HP 1000) that has Network File System (NFS) client software installed.

node

A computer system in a network.

RDATE

Command that allows setting the system time from a remote system.

server

Generically speaking, a system or program that responds to requests of client systems or programs. *See also* HP 1000 file server.

$SYSTZ

The $SYSTZ value must be set with the SYSTZ utility. This value contains the Coordinated Universal Time (UTC) offset for the local time zone. $SYSTZ is used by the RDATE utility.

TCP

Transmission Control Protocol.

type 12 files

The byte stream RTE file type created when files are created from an NFS client while mounted to the HP 1000 file server.

UDP

User Datagram Protocol.

umount command

Command issued by an NFS client that unmounts a file system, thereby preventing future access to the server file system.

UTC

Coordinated Universal Time.

Index