HP 9000 Networking

HP-UX SNAplus2 RJE User’s Guide

HP Part No. J2743-90001
Printed in U.S.A.
E0496

Edition 1

HEWLETT
PACKARD

Legal Notices

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

Hewlett-Packard makes no warranty of any kind with regard to this manual, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose. Hewlett-Packard shall not be held liable for errors contained herein or direct, indirect, special, incidental or consequential damages in connection with the furnishing, performance, or use of this material.

Warranty. A copy of the specific warranty terms applicable to your Hewlett-Packard product and replacement parts can be obtained from your local Sales and Service Office.

Restricted Rights Legend. Use, duplication or disclosure by the U.S. 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 for DOD agencies, and subparagraphs (c) (1) and (c) (2) of the Commercial Computer Software Restricted Rights clause at FAR 52.227-19 for other agencies.

HEWLETT-PACKARD COMPANY
3000 Hanover Street
Palo Alto, California 94304 U.S.A.

Use of this manual and flexible disk(s) or tape cartridge(s) supplied for this pack is restricted to this product only. Additional copies of the programs may be made for security and back-up purposes only. Resale of the programs in their present form or with alterations, is expressly prohibited.

This software is based in part on the Fourth Berkeley Software Distribution under license from the Regents of the University of California.

Printing History

New editions are complete revisions of the manual. The dates on the title page change only when a new edition or a new update is published.

Note that many product updates and fixes do not require manual changes and, conversely, manual corrections may be done without accompanying product changes. Therefore, do not expect a one-to-one correspondence between product updates and manual updates.

Edition 1	April 1996

Preface

The HP-UX SNAplus2 RJE User's Guide describes the features and functions of RJE (Remote Job Entry) for the HP 9000 Series 700 and 800 computers.

SNAplus2 RJE provides the functions of an IBM 3770 Communications terminal. It allows you to queue jobs locally before sending them to the host and to route host output to a disk file, a directory or a program. It also allows you to view host information and issue commands with the RJE console program.

The manual includes the following:

- An Overview of SNAplus2 RJE that defines its functions, describes its features and lists its typical uses.
- Details of the overall operation of SNAplus2 RJE
- Information on the function of each of the SNAplus2 RJE components
- The commands used to control SNAplus2 RJE
- How to use the console program
- The workstation style file and customization program

Audience

This manual is intended for use by anyone who uses SNAplus2 RJE to submit jobs to a host Job Entry Subsystem (JES) for processing, or to receive output from the host.

Related HP Documentation

The following publications are Hewlett-Packard manuals that are related to the HP-UX SNAplus2 RJE product:

- HP-UX SNAplus2 Installation Guide
- HP-UX SNAplus2 Migration Guide
- HP-UX SNAplus2 Administration Guide

Contents

1 Introducing SNAplus2 RJE

SNAplus2 RJE Features 15

Typical Uses of RJE 18

2 Getting Started

Using SNAplus2 RJE 20

Starting the Daemon 20
Specifying the Path to SNAplus2 Programs 20
Starting the RJE Workstation 21
Checking Workstation Status 22
Using the Console Program 23
Sending a Job 24
Stopping the Workstation 25

3 SNAplus2 RJE Components and Operation

Overall Structure of SNAplus2 RJE Components 28

Stages in Processing a Job 29

Contents of a Job File 30

Processing Input Data 32

Record Lengths 33
Sending Exchange Data 33
Sending Text or Binary Files 34
Example: 34
Compressing Job Files 35

Contents

Processing Output Data 36

Host Output Devices 36

Printers 37

Punches 38

Exchange Devices 38

Sending PDIR Information 39

Routing Output 42

Punch Padding 45

punch_padding Option 45

rjeusrpad Program 45

Using Output Programs 46

Program Output Without PDIRs 46

Program Output With PDIRs 46

General Guidelines for Output Programs 47

Routing Received Files with a usr Card 48

Usr Card Parameters 50

Including the usr Card in Your File 53

Example of usr Cards 54

Routing Received Files Without a usr Card 55

Taking Precautions When Receiving Files 56

Routing by Default 56

Routing by usr Card 56

Receiving Error Notification 56

Running the RJE Workstation 57

Workstation Directory Structure 57

Clearing a Workstation’s Directories 58

Specifying New Temporary Directory Paths 59

Setting logon_retry 59

Starting the Workstation 61

Displaying Workstation Status 62

Stopping the Workstation 63

Getting Information About RJE Jobs 64

Contents

Audit Log File 64
RJE Job Log File 68
QUED: Job Submitted to the Workstation 69
SENT: Job Sent to the Host 70
CANL: Job Cancelled 70
RECV: Output Received from the Host 70
ROUT: Output Routed by rjeusr/rjeusrpad 71
Other Sources of Information 71
Workstation Mail Messages 72

4 SNAplus2 RJE Commands

snaprjestart 76

snaprjestop 78

snaprjesend 80

Using Control Lines 84
Control Line Syntax 84
Control Line Commands 85
Control Line Examples 90

snaprjelst 94

snaprjecan 98

snaprjepur 100

snaprjestat 102

snaprjecmd 106

snaprjeclr 108

5 SNAplus2 RJE Console Program

RJE Console Program 112

Starting the Console Program 113

Understanding Status Line Information 115

Using the Console Program 117

Entering Host Commands 117
Viewing Console Information 118
Exiting Temporarily to the HP-UX Command Prompt 120
Ending the Console Program 120

6 SNAplus2 RJE Workstation Configuration

Overview of the Workstation Style File 123

RJE Style File Customization Program 124

Starting the Program - Style File for a Configured Workstation 125
Starting the Program - Setting Up a Style File Offline 125
Customizing the Workstation 126
Saving the Style File 127
Exiting the Program 128
Messages displayed by the program 128

Text File Format 130

Parameter Types 130
Default Parameter Values 132
Blank Space 132

RJE Style File Records 133

Contents

define_workstation_options 133
define_minor_options 135
define_forms_control 143
define_ws_device 145

Diagnostics 151

Configuring RJE Workstation with xsnapadmin 152
Workstation Configuration Information 152

A Program Output with PDIRs

Output Program Example 154

1

Introducing SNAplus2 RJE

SNAplus2 RJE (Remote Job Entry) provides the functions of an IBM 3770 terminal, allowing communications between SNAplus2 RJE and host Job Entry Subsystems such as the following:

- MVS/JES2 (Multiple Virtual Storage/Job Entry Subsystem 2)
- MVS/JES3 (Multiple Virtual Storage/Job Entry Subsystem 3)
- VSE/POWER (Virtual Storage Extended/Job Entry Subsystem)

Using SNAplus2 RJE, you can perform the following tasks:

- Submit jobs to the host for processing
- Receive output generated by the host
- Issue commands to the host Job Entry Subsystem, and receive output from these commands through the host console

This chapter introduces the features of SNAplus2 RJE and describes some typical uses.

SNAplus2 RJE Features

SNAplus2 RJE provides the following features, which are explained in more detail in Chapter 3 of this manual:

Multiple RJE Workstations

SNAplus2 SNAplus2RJE allows the use of multiple RJE workstations on a single SNAplus2 system. Each workstation can be configured for use by a group of users. This allows you to provide different RJE configurations to suit the requirements of different groups of users.

Local Job Queuing

Jobs submitted for a host are queued locally on a spool before being sent to the host. Spool commands are used to submit the jobs to an RJE workstation, to list the jobs spooled for the workstation, and to cancel jobs spooled for the workstation. SNAplus2 provides optional facilities to save status information on sent jobs (so that you can check that a job has been sent), and to purge the stored status information.

Multiple Printers and Punches

As many as nine printer and nine punch output devices may be configured for each RJE workstation. This configuration can be used in conjunction with the host configuration to handle different output for different types of data.

Disk and Program Output

Host output can be routed to a disk file or piped as standard input to a program or shell script for processing.

Host Console Program

A full-screen console program allows you to view host console information for an RJE workstation, and to issue commands to the host Job Entry Subsystem.

PDIRs

SNAplus2 RJE accepts PDIRs (Peripheral Data Information Records) from the host. These PDIRs can be stored in a file associated with the output or passed as parameters to a program that processes the output.

Code Conversion

Data can be translated from ASCII to EBCDIC when it is sent to the host, and from EBCDIC to ASCII when output is received from the host. It can also be sent or received as binary data without code conversion.

Exchange Data

SNAplus2 RJE supports sending and receiving exchange data.

Modifiable Data Record Size

In addition to the default record lengths of 80 bytes for punch data and 128 bytes for exchange data, SNAplus2 RJE allows data to be sent in records of any size from 80 bytes to 248 bytes. Note that the use of exchange data and record lengths other than 80 bytes depends on the host system's support for them.

Inbound and Outbound Compression

SNAplus2 RJE supports compression both on transferring data to the host and on receiving data from the host. File compression can improve the processing time for a job if you are submitting large files. Note that the use of compressed data depends on the host system's support for it.

Multiple LUs per Workstation

As many as five LUs on the same host connection may be configured for each RJE workstation.

usr card output routing

SNAplus2 RJE allows you to configure a device to send output to the rjeusr and rjeusrpad programs which route output according to usr card specifications.

Padding of Punch Records

SNAplus2 RJE allows you to configure a device to send output to the rjeusrpad program which routes output according to a usr card specification and pads punch output that is received in TEXT mode.

As an alternative to configuring an output program to do the padding, you can pad all punch output by setting the minor option **`punch_padding`** to the desired value.

Typical Uses of RJE

Listed below are some typical ways that you can use the features of RJE:

- Print host data at a local site

  You can use RJE to download host data sets (host files) from a remote host computer and print them on a local printer.

- Transfer, store and retrieve files

  You can use RJE to transfer a file from the HP-UX computer to a host data set (host file) for storage. The file can be downloaded later to an RJE file or retrieved by a user on a different computer. The file can be transferred without code conversion, or it may be translated to EBCDIC to allow it to be accessed with a host editor.

- Send jobs at convenient times

  RJE allows you to send jobs to the host and receive data from the host at a time that is convenient to you. For example, you can use a cron script to run the RJE workstation at night to send all the jobs submitted during the day. Alternatively, when submitting a job, you can use command-line options to start the workstation and to stop it after RJE activity has completed.

2

Getting Started

This chapter illustrates a simple use of an SNAplus2 RJE workstation. It contains general instructions and information on starting the control daemon, starting and stopping the workstation, sending a job, checking workstation and job status, and cancelling a job.

Using SNAplus2 RJE

Before you begin, you must configure an RJE workstation and start the SNAplus2 daemon. The SNAplus2 configuration file, which is set up and maintained by the system administrator, contains information that is required for RJE communications. For configuration information, see the *HP-UX SNAplus2 Administration Guide* and Chapter 6 of this manual.

Starting the Daemon

In this chapter, it is assumed that you have configured a workstation with at least one printer and one punch device. It is also assumed that you have started the SNAplus2 daemon with the following command:

```
snap start
```

Along with the daemon, it is necessary to activate the node, port and LS used by the RJE1 LUs. You can start the node, port and LS from the **xsnapadmin** program or from the command line program, **snapadmin**. (See the *HP-UX SNAplus2 Administration Guide* for details.) From the command line, type the following:

```
snapadmin init_node
```

```
snapadmin start_port, port_name=portname
```

```
snapadmin start_ls, ls_name=lsname
```

If you type the command, **snapman -c *lsname***, you should see the name of the activated link station.

Specifying the Path to SNAplus2 Programs

RJE executable programs are stored in the directory **/opt/sna/bin**. When you run the programs, you must specify the path to this directory. You can specify the path either by adding the directory to your PATH environment variable before you run the programs for the first time, or by including the directory name each time you run the programs.

If you add this directory to the definition of the PATH environment variable in your **.login** or **.profile** file, the programs will be located automatically. Alternatively, you can specify the directory name when you run the program, as in the following examples:

```
/opt/sna/bin/snaprjesend RJE1 myjob
```

```
/opt/sna/bin/snaprjecon RJE1
```

The sample command lines shown in this manual assume that you have added the directory to your PATH environment variable, and do not include the directory name.

Starting the RJE Workstation

The workstation name used in this chapter is RJE1.To start the workstation named RJE1, type the following command on the command line: (The workstation name must be in uppercase and must be only four characters long).

```
snaprjestart RJE1
```

SNAplus2 RJE displays the following message:

```
RJE Workstation RJE1 has started
```

You must be a member of the group that is configured for this workstation in order to issue commands for it. If any error messages display when you start the workstation, check the error log file (default is**/var/opt/sna/sna.err).** You can view this file with your favorite text editor.

You can also use the **tail** command to view the tail-end of a file. For example, type the following:

```
tail /var/opt/sna/sna.err
```

This command shows you the most recent error messages in the file.

Checking Workstation Status

Check to see that workstation RJE1 has established a connection with the host by typing the following command:

```
snaprjestat RJE1
```

The screen displays the status of the devices and LUs that you have configured for the workstation as in the following example:

```
Wkst   Device    Status      LU Name    Action
------------------------------------------------------------
RJE1   CONS      Inactive    *          *
RJE1   PUN4      Inactive    *          *
RJE1   PUN3      Inactive    *          *
RJE1   PUN2      Inactive    *          *
RJE1   PUN1      Inactive    *          *
RJE1   RDR1      Inactive    *          *
RJE1   PRT3      Inactive    *          *
RJE1   PRT2      Inactive    *          *
RJE1   PRT1      Inactive    *          *

Wkst   LU Name   Status      Device
------------------------------------------------------------
RJE1   RJELU09   Idle        *
RJE1   RJELU08   Idle        *
RJE1   RJELU07   Idle        *
RJE1   RJELU06   Idle        *
```

It may take a minute for the workstation to establish a session with the host. Continue checking on the workstation by reissuing the command until you see that the status of the LUs are either Idle, Sending or Receiving. If the LUs stay in the Pending state, check to see that the LUs are active on the host side. See your host administrator if the problem persists.

Using the Console Program

You can also check the status of the workstation with the RJE console program. Type the following command:

```
snaprjecon RJE1
```

The status line for the console appears at the bottom of the screen and shows the following:

```
F1 HELP     INSERT                 RJE1+     FOLLOW
```

Pressing f1 displays console help information.

INSERT means the editor is in insert mode. Text typed at the command line is inserted at the cursor position. **REPLACE** appears when the editor is in replace mode. Text typed will overwrite existing text.

RJE1+ indicates that the workstation has established a session with the host. **RJE1-** appears when the workstation has not established a session with the host.

FOLLOW indicates you are viewing new console messages. **BROWSE** appears when you are viewing earlier console messages.

You can issue host commands by typing them on the command line and pressing Return. The following JES2 command displays information about the remote host that the workstation is using. (RMT26 is an example remote name. Use the remote name that is configured for your workstation):

```
$DU,RMT26
```

To exit the console program, press f3 or CTRL-X.

Sending a Job

Try sending a job to the host. The following is a sample job file:

```
//IBM1 JOB (1111,AAA), 'Bill Smith',CLASS=A MSGCLASS=A
//STEP1 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSUT2 DD SYSOUT=B
//SYSIN DD DUMMY
//SYSUT1 DD *
~! cat datafile
/*
```

Note that the file named datafile will be included in the job before you send it to the host.

To send the job, type the following on the command line:

`snaprjesend RJE1` ***`filename`***

where *filename* is the name of your job file.

The screen displays the following message:

`Job file submitted to job spooler`

`Job spooled as SPL_abcdef`

`SPL_abcdef` is the Job's spool file name. It will be different for each job that you spool.

You can obtain information on the status of the job by viewing the RJE job log file (see "RJE Job Log File" in Chapter 3).

After the data has been received from the host, it will be sent to the location (directory, program, or file) specified in the device configuration. If the device was configured to send output to the program **`/opt/sna/bin/rjeusr`** or **`/opt/sna/bin/rjeusrpad`**, then output will be routed according to a usr card in the data, or it will be placed in the base directory for this workstation (**`/var/opt/sna/rje/RJE1/output`**). See chapter 3 of this manual for information on using the usr card.

Stopping the Workstation

To stop workstation RJE1, issue the following command:

```
snaprjestop RJE1
```

The workstation will stop after one minute of inactivity. If you want the workstation to stop immediately, issue the command with the **`-i`** option:

```
snaprjestop -i RJE1
```

This command stops the workstation immediately. However, before using the **`-i`** option, be sure that no data transmission is in progress.

You can submit jobs while the workstation is not running, and those jobs will be sent when the workstation is restarted. Try the following command after you stop the workstation:

```
snaprjesend RJE1 filename
```

Then type,

```
snaprjelst -al RJE1
```

Information will display about all of the jobs currently spooled for the workstation as in the example below:

```
SPL_irgjvf bsmith 11/02/92 20:00:43 WAITING 10 0 - 1 demo2B
```

After a job has been sent to the host, this information is deleted if you did not specify the **`list_sent_jobs`** option in the workstation style file. If you specify this option, you can use the **`snaprjelst`** command with the **`-s`** option (for sent jobs only) or the **`-b`** option (for both waiting and sent jobs) to list this information. See Chapter 4 for more information about this command.

If you want to delete this job from the spool before it is transmitted to the host, type the following command:

```
snaprjecan SPL_irgjvf
```

This command deletes a single job.

To delete all jobs that have been spooled with your user id, type the following:

```
snaprjecan RJE1 ALL
```

3

SNAplus2 RJE Components and Operation

This chapter explains in detail how to submit and process a job with RJE, and explains the function of each of the components.

Overall Structure of SNAplus2 RJE Components

The following diagram illustrates the various RJE components and the interactions between them. The main components of RJE are the spool commands, the RJE workstation, and the output from the host.

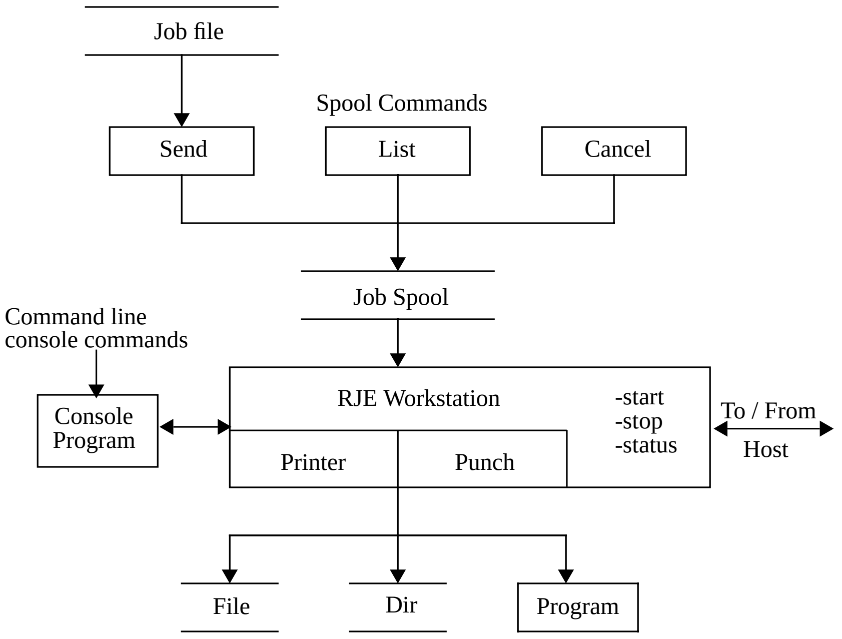


Figure 1 **Structure of SNAplus2 RJE Components**

Spool Commands

The “Send” spool command adds jobs to a local disk spool, the “List” command lists all the jobs that have been submitted to the spool, and the “Cancel” spool command removes jobs from the spool.

RJE Workstation

The RJE workstation, which runs independently of the spool commands, sends jobs to the host and receives the output.

Output

The output from the host routes to a directory, program, or file (defined by the configuration).

Stages in Processing a Job

To use RJE, all you need to do is start the RJE workstation and submit a job. The RJE workstation sends the job to the host and receives and routes the host output. Following are the stages involved in submitting and processing a job. These stages explain how RJE operates and shows all the facilities available to you.

1 Create the job file.

Create a job file that contains the instructions required by the host. This file may include all of the exact data required by the host, or it may contain control lines to modify the source data. The section, “Contents of a Job File”, which appears later in this chapter, explains the format of this file.

2 Add the job to the workstation's job spool.

Use the **`snaprjesend`** command to submit the job to a host for processing. This command does not immediately cause the job to be sent to the host, but adds it to a spool of jobs waiting to be sent by the RJE workstation.

For more information on this command, see “Processing Input Data”, later in this chapter and in chapter 4.

3 See whether the job has been sent.

While the job is waiting on the spool, you can use the **`snaprjelst`** command to list all the jobs you have submitted, or **`snaprjecan`** to cancel a queued job.

For more information on these commands, see “Processing Input Data”, later in this chapter and in chapter 4.

4 Run the RJE workstation.

Start the RJE workstation program if it is not already running when you submit the job.

Note that each RJE workstation should be configured to run on a particular SNAplus2 computer, which may be either a server or a client. You can start the workstation only on its configured computer, and you must issue any other RJE commands related to the workstation (for example, commands for stopping the workstation, spool commands, or the console program), on the same computer.

For more information on starting the workstation, see “RJE Workstation”, later in this chapter.

5 Send the job to the host.

The RJE workstation takes jobs from the spool in the order in which you submit them and sends them to the host for processing. The host then processes the jobs. If an error occurs during sending (for example, a link failure), the RJE workstation may retry sending the jobs when the session is established again. The workstation may make as many as 99 attempts for each job. However, this depends on the number of retry attempts you specified when you submitted the job.

For more information, see the **`snaprjesend`** command and “Using Control Lines with **`snaprjesend`**”, in chapter 4.

6 View host console information.

Once you submit the job to the host, use the console program to examine the status of jobs queued at the host, and view information returned by the host.

For more information, see “RJE Console Program”, later in this chapter and in chapter 5.

7 Receive the host output.

The host processes the job, and returns output to the RJE workstation.

8 Route the host output.

The RJE workstation sends the output from the host to the appropriate file, directory, or program, as specified in the configuration.

For more information, see “Processing Output Data” and “RJE Workstation”, later in this chapter.

The following sections provide more information on the RJE components and tasks associated with each of the above stages.

Contents of a Job File

The host operating system determines the syntax of the job file's contents. Check your host JES documentation for details. In general, the job file will consist of the following:

- Job Control Language (JCL), which provides user and password information and details of the processing required
- Data to be processed by the host
- JCL indicating the end of the data

A sample job file, which is intended for the host system JES2, appears below.

```
//XNKJA   JOB (05604Q,A,X),
//        MSGCLASS=A,
//        CLASS=E,
//        USER=XNKBJ,PASSWORD=DEMO
//SETUP   EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=*
//SYSIN    DD DUMMY
//SYSUT2   DD SYSOUT=*
//SYSUT1   DD *
TEST FILE LINE 1 THIS JOB WILL CAUSE THIS DATA TO BE SENT TO A
PRINTER DEVICE
TEST FILE LINE 2
TEST FILE LINE 3
TEST FILE LINE 4
TEST FILE LINE 5
TEST FILE LINE 6
TEST FILE LINE 7
TEST FILE LINE 8
TEST FILE LINE 9
TEST FILE LINE 10
TEST FILE LINE 11 THE END
/*
```

In this example, the first nine lines are JCL. The JCL causes the input data to be copied to a class A device, typically configured as printer output. The following eleven lines are the data to be processed by the host. The final line, /*, indicates the end of the data.

Processing Input Data

After you submit the file for processing. This section explains how to send jobs to an RJE workstation and the options that are available for sending different types of job files.

The commands below are spool control commands. Use these commands to access the job spool directly. A job spool exists for each RJE workstation. It is created the first time the workstation runs or the first time you send a job. When you send a job, the send command adds it to the spool for an RJE workstation. The workstation then takes the job from the spool and sends it to the host for processing.

To send a job to the job spool for processing, use the following command:

```
snaprjesend WKST jobfilename
```

(WKST is the name of your workstation).

The list command shows a list of all the jobs that are waiting on the spool for a particular workstation. This list contains all jobs that have been submitted to that workstation but have not yet been successfully sent to the host. To see this list, use the following command:

```
snaprjelst WKST
```

The cancel spool command cancels a job that is in the spool queue waiting to be sent to the host. To cancel a job from the spool, use the following command:

```
snaprjecan WKST spool-id
```

NOTE: There is a maximum of 1000 jobs that can exist at any one time in a workstation’s job spool.

```
Example: snaprjecan WKS1 SPL_fjghld
```

You can use these commands at any time as long as you have started the SNAplus2 control daemon. The spool is maintained independently of the workstation.

When the RJE workstation runs, it accesses the job spool. If any jobs are in the queue, the workstation sends them to the host for processing in the order in which they are spooled.

Once the workstation has successfully sent a job to the host, it removes this job from the spool. You can use the RJE console program to view messages from the host relating to the job and to issue host commands if necessary. You can use the console program at any time while the SNAplus2 control daemon is running. For more information, see “RJE Console Program,” later in this chapter and in chapter 5.

Record Lengths

Host Job Entry Subsystems normally handle data as punch data in 80-byte default record lengths. Some host systems can handle exchange data, which is normally in 128-byte default record lengths. Other host systems can accept either punch data or exchange data with record lengths other than these defaults.

SNAplus2 RJE provides a command-line option with the **`snaprjesend`** command to specify whether the data is punch or exchange data and to override the default record lengths. The **`-m`*n*** option specifies that the record length of the job file can be n bytes. The range is 80 - 248.

Example:

```
snaprjesend -m100 RJE1 myjob
```

This command sends the file **`myjob`** as punch data with a maximum record length of 100 bytes.

Sending Exchange Data

The host systems that handle exchange data (normally in 128-byte record lengths), require that such data be marked as “exchange data” to distinguish it from punch data, (normally in 80-byte records.)

When submitting jobs as exchange data, use the **`-e`** option with the **`snaprjesend`** command to indicate that you are submitting exchange data. Use the **`-mn`** option to indicate the record length if the record length is different from the default.

The following examples show the commands for sending exchange data:

`snaprjesend -e RJE1 myjob`

This sends the file as exchange data with the default record length of 128 bytes.

`snaprjesend -e -m150 RJE1 myjob`

This command sends the file as exchange data with a maximum record length of 150 bytes. (See the description of the **`snaprjesend`** command in chapter 4 for more information.)

Sending Text or Binary Files

Most jobs sent with RJE will be text files in ASCII. By default, SNAplus2 RJE converts the supplied data into EBCDIC before sending it to the host, so that it can be accessed with a host editor.

In some cases, you may need to send binary data to the host (submit data without code conversion). To send binary data, do the following:

1 Put the binary data into a separate file.

2 Include this file within the main job file using the ~! -b cat command.

Example:

```
//XNKJA    JOB (05604Q,A,X),
//         MSGCLASS=A,
//         CLASS=E,
//         USER=XNKBJ,PASSWORD=DEMO
//SETUP    EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=*
//SYSIN     DD DUMMY
//SYSUT2    DD SYSOUT=*
//SYSUT1    DD *
TEST FILE LINE 1
TEST FILE LINE 2
TEST FILE LINE 3
TEST FILE LINE 4
~! -b cat datafile
/*
```

This means that SNAplus2 RJE will code convert the JCL within the main job file (to enable the host to use it), but it will send the included data as binary data. For more information, see “Using Control Lines with snaprjesend”.

SNAplus2 RJE divides the supplied data into records in different ways for text and binary data as follows:

- It separates text data into records at ASCII new-line characters, and sends each line of the text as a separate record. If a line is longer than the specified record length (see previous section), SNAplus2 RJE splits it into two (or more) records as necessary. For example, if a line 100 bytes long is sent using the default record length of 80 bytes, it will split the line into an 80-byte record and a 20-byte record.
- SNAplus2 RJE regards binary data as a continuous sequence of bytes, and splits it into the maximum record size without scanning for new-line characters.

Compressing Job Files

Job file compression can improve the overall processing time for a job if you are sending large files to the host containing sequences of more than four of the same character (including the space or null characters) in succession. The RJE workstation configuration specifies whether job files submitted to a workstation will be compressed.

The use of compression depends on whether the host supports it. Check with your host personnel if necessary. If the host configuration specifies that it supports compression, the workstation will send the job compressed.

This option does not affect output received from the host. SNAplus2 always accepts compressed files sent from the host or files that are not compressed.

Processing Output Data

This section explains how an RJE workstation handles the output returned to it by the host and the options that are available for processing and routing host output.

The SNAplus2 RJE configuration determines some of the options for controlling output for a particular RJE workstation. This section explains those options and the use of the usr card. See your System Administrator if you are not sure how your RJE workstation handles output data, or if you want to change the configuration.

Host Output Devices

SNAplus2 RJE supports three types of host output devices:

- Printers
- Punches
- Exchange devices

The host uses different types of devices for different types of output. Printers are used for data formatted by the host (with control information such as horizontal and vertical tabs); punches and exchange devices are used for unformatted data (such as data files), transferred as records.

The host configuration for an RJE workstation includes a number of printer devices, punch devices, and/or exchange devices. A number identifies different devices within each device type:

- 1 to *n* printers
- 1 to *n* punches
- 1 to *n* exchange devices

When sending output, the host specifies (by number and device type) a particular printer, punch, or exchange device to which it is sending the output.

The SNAplus2 RJE configuration for an RJE workstation also includes printer and punch devices identified by numbers. The different configuration options available for each of these devices reflect the different host usage of each device type. SNAplus2 RJE punch devices can also handle output intended for exchange devices. See the section “Exchange Devices” later in this chapter for more information.

SNAplus2 RJE printers and punches should be configured so that there is an equivalent SNAplus2 device for each host device. Host data sent to remote printer 1 (for example), is then routed to SNAplus2 printer 1 (designated PRT1). Host data sent to remote punch 3 is routed to SNAplus2 punch 3 (designated PUN3). Host data sent to exchange device 2 is routed to SNAplus2 punch 2 (since SNAplus2 punches can also handle exchange device output).

The following sections provide more information on the output options associated with each type of output device.

Printers

Printers produce formatted output. They are typically used where the output is required to be printed. The data sent to a printer device includes control characters that provide the necessary formatting information. This formatting information comes from the host. In addition, you can use the following SNAplus2 options to control the format of printer output.

The following output options for printers can be defined by using **`snaprjecust`** (the workstation customization program) or by editing the minor options in the style file (see Chapter 6, “SNAplus2 RJE Workstation Configuration”).

- Forms Control Buffer (FCB)
- Form Feed Passthrough
- Maximum Printer Line Length
- Code Conversion (data translation from EBCDIC to ASCII)

Punches

Punch devices produce unformatted “raw data” output, such as simple text files, as opposed to the formatted output produced by printers. For example, if you use RJE to retrieve a file that has been sent to the host for storage, the host will generally send the data to a punch device. The data can consist of variable length or fixed records (depending on the host configuration).

The following parameters for punches can be defined by using **`snaprjecust`** (the workstation customization program) or by editing the minor options in the style file (see Chapter 6, “SNAplus2 RJE Workstation Configuration”).

- Code Conversion (data translation from EBCDIC to ASCII)
- NL option
- Binary Output
- EBCDIC New-line Option

Exchange Devices

When the host sends output data as exchange data, it specifies a particular exchange device to which the output is routed. SNAplus2 supports exchange devices by routing any output for exchange device *n* to punch device *n*. For example, if the host specifies that an output device is exchange device number 3, it sends the output for this exchange device to punch number 3. However, the job logging information will indicate that the output is exchange output (see “RJE Job Log File” later in this chapter).

The options for controlling exchange device output are the same as for punch output. See the *HP-UX SNAplus2 Administration Guide* for more information.

Sending PDIR Information

When the host sends output data to a printer or punch device, it may supply information on how to process the data. This information is in the form of a PDIR (Peripheral Data Information Record), which precedes the data to which it applies. The host will normally send a PDIR at the start of the output data for a job (this depends on the host configuration). It may also send further PDIRs interspersed with the data.

SNAplus2 RJE allows you the option of retaining PDIR information for use in processing the output or discarding it. This option, **`pdir_passthrough`**, is defined in the RJE style file for each output device. You can use the **`snaprjecust`** customization program to turn this option on or off (see Chapter 6 for information on how to use this program). The use of PDIR information depends on the method of routing the output.

A PDIR consists of the following seven parameters:

DATE The date of creation of the host data set (host file) being sent as output data. This is in the format MM/DD/YY (month, day, year).

TIME The time of creation of the host data set (host file) being sent as output data. This is in the format HH.MM.SS (hours, minutes, seconds).

FORMS The name (up to eight EBCDIC characters) of the printer forms to be used. Blank indicates “use the standard forms”.

FCBNAME The name (up to eight EBCDIC characters) of the FCB to be used to format the output. (The host system may restrict this to four characters). If this name matches the name of an FCB configured for SNAplus2 RJE, then this FCB is used to process the output; otherwise, it is ignored. Blank indicates “use the default FCB”.

TRAIN The name (up to eight EBCDIC characters) of the printer train to be used. Blank indicates “use the standard train”.

COPIES The number (shown in EBCDIC characters) of additional copies of the printed output to be produced. Zero indicates “print only one copy”; 1 indicates two copies in total.

VOLIO An indication of the volume of output produced (shown in EBCDIC characters). Note that this only provides information on the relative sizes of different jobs.

JOBNAME,
STEPNAME,
PROCSTEP,
DDNAME,
SPINNO These five parameters make up the DSNAME parameter used by the host to identify the job. Many host systems use only JOBNAME and do not include the remaining parameters.

REQID PDIR request identifier. This takes one of the following values:

0 - Standard PDIR

1 - Job separator PDIR

2 - System message PDIR

The REQID parameter is not passed to output programs, but is available in the temporary PDIR file accessible by output programs.

When PDIR information is retained for use in processing output, it is stored in files with a fixed format. An example of a PDIR file is shown below.

```
     OPEN  WKSTPRT1
     PDIR  11/14/91 18.28.46 STD    STD6    QN    0 30 NEWJOB1  None  None  None  None  0
1054 PDIR  11/14/91 18.28.46 STD    STD6    QN    0 50 NEWJOB1  None  None  None  None  1
2835 CLOSE
```

The first and last lines of the file are OPEN and CLOSE. The RJE device name follows OPEN. The RJE device name is a combination of the workstation name and the device identifier (PRT*n* for printer number *n*). CLOSE indicates the end of the data.

Other lines of the PDIR file are followed by a list of the PDIR parameters as described above. If the host leaves any of the PDIR parameters blank, SNAplus2 RJE inserts the string “None”.

The first column of the PDIR file indicates the byte offset into the output data at which the information was received. OPEN and the first PDIR will be at offset 0, and CLOSE will be at the offset representing the end of the file.

Routing Output

Output for a printer or punch can be routed in the following ways:

Directory The host sends output directed to this device to a new file in a directory specified in the configuration. The file is named RJE_abcdef, where abcdef represents a string of six alphabetic digits. The host creates file names in alphabetic order, but not necessarily in consecutive order.

If you specify the **`pdir_passthrough`** option, PDIR information is written to a file that has the same name as the output file but with extension .i. The format of the file is as described in the previous section.

Program Output directed to this output device is used as the standard input to an executable program or shell script, whose name and path are specified in the configuration.

Note that the program does not start until it has successfully received all the data for a job. If an error occurs during receipt of the data, the program will not start.

If you specify the **`pdir_passthrough`** option, the parameters from the first received PDIR pass to the program as parameters. The program can then use the values of these parameters to determine how to process the data. See “Using Output Programs” below for more information.

The path name of the temporary output file being piped to the program is also passed as a parameter to the program. The PDIR file for this output can be derived by adding the **`.i`** extension. If there are no PDIR parameters, the path name will be the only parameter. If there are PDIR parameters, it will be the last parameter.

File

Output is sent to a file whose name and path are specified in the configuration of the output device. If a subsequent job specifies the same output device, its output will be sent to the same file. The configuration specifies whether it will overwrite the existing contents of the file or be appended to it.

If the data is replacing the contents of an existing file, and an error occurs while the data is being received, the file will be deleted unless the **`keep_failed_files`** minor option is set. If **`keep_failed_files`** is set, that data will be saved in a temporary file.

If the data is being appended to the end of an existing file, SNAplus2 RJE writes to a temporary file until all the data has been received or until an error occurs. The temporary file is named **`TMP_abcdef`**, where abcdef represents a string of six alphabetic digits. The file is created in the **`/var/opt/sna/`*`WKST`*`/output`** directory. If all the data is received successfully, SNAplus2 RJE then appends it to the existing file and removes the temporary file. If an error occurs, the incomplete data is left in the temporary file, and the existing file is not changed. An error log message will indicate the name of the file.

If you specify the **`pdir_passthrough`** option, PDIR information is written to a file that has the same name as the output file but with extension **`.i`**. This PDIR file is replaced or appended in the same way as the output file. The format of the PDIR file is as described in the previous section except when output is being appended and an error occurs. Then the temporary PDIR will contain the line CLOSE ERROR instead of CLOSE.

usr card If you want to include a usr card when you have your output routed, you need to configure usr card routing for each output device. You can route the output by sending it to either the **`/opt/sna/bin/rjeusr`** program or the **`/opt/sna/bin/rjeusrpad`** program.

The **`rjeusr`** program routes output according to a usr card specification. If no usr card is present in a job, **`rjeusr`** leaves the output in **`/var/opt/sna/bin/WKST/output`** and renames the output file PRT*nnnn* (for printer files), PUN*nnnn* (for punch files), or EXC*nnnn* (for exchange device files). For more information see “Usr Card Parameters” later in this chapter.

The **`rjeusrpad`** program has an extra feature. It pads punch records with blanks. (See the following section on punch padding.)

The above mentioned minor options (**`pdir_passthrough`** and **`keep_failed_files`**) can be defined by editing the RJE style file.

Punch Padding

The punch_padding option offers several variations on how punch output is padded.

punch_padding Option The preferred method for punch padding is to specify the **`punch_padding`** option in the **`define_minor_options`** record in the RJE style file. Specifying this option allows you to pad all punch output without configuring an output program to do the padding**.**

Note that when **`punch_padding`** causes RJE to add blanks, these blanks are either ASCII or EBCDIC, depending on whether translation is done.

This method of padding punch records does not depend on the addition of new-line characters. If you do not desire new-line character insertion, you must use the **`snaprjecust`** program to configure **`NL between records`** off, or set **`add_punch_nl`** to NO in the style file.

NOTE: If output is sent to **`rjeusrpad`**, new-line characters will be removed. Use **`rjeusr`** with the **`punch_padding`** option to route output with insertion of new lines.

rjeusrpad Program Another method for punch padding is to use the **`rjeusrpad`** program. The **`rjeusrpad`** program pads punch output that is logged as being received in TEXT mode (ASCII or EBCDIC). Check the audit log to determine if your data was received as text or transparent. While actual output data may be both text and transparent, if any record is a transparent record (sent with a transparency character), it is logged as TRANSP. In the case of data that is logged as TRANSP, punch records will not be padded. (For other types of punch padding, see the next section.)

In order for the **`rjeusrpad`** program to work correctly, you must configure NL (new-line character) insertion for the device. The **`rjeusrpad`** program uses the new-line characters to determine where padding is necessary. It pads records to the maximum record length configured for the device on the host. New-line characters are stripped from the record after the padding is done. If **`rjeusrpad`** is not able to successfully pad the output file, the file is deleted.

Using Output Programs

You can use program output to process data received from the host without PDIRs or with PDIRs. Each method is described in the following sections.

Program Output Without PDIRs

Once the output from the host has been received successfully, the output program is invoked with the path name of the temporary output file as a parameter. The data from the host is piped as standard input to the program. For example, the data could then be piped to the HP-UX print spooler **`lp`**, or to a shell script that calls **`lp`** with suitable parameters.

Program Output With PDIRs

Once the output from the host has been received successfully, the output program starts. The parameters from the first PDIR received with the data pass as parameters to the output program along with the pathname of the temporary output file. The data from the host is piped as standard input to the program. In other words, the host data is piped to the following PDIR parameters:

program_name, date, time, forms, FCBname, train, copies, volIO, jobname, stepname, procstep, ddname, spinno, reqid

The program can then use the PDIR parameters to determine how to process the data. Additionally, the temporary PDIR file for the output can be accessed by adding the **`.i`** extension to the parameter specifying the temporary output file name.

For example, the following shell script calls the print spooler **`lp`** with different options according to the form name and number of copies specified by the ***forms*** and ***copies*** parameters. (Note that ***copies*** is the number of additional copies, not the total. **`copies=0`** represents 1 copy.)

```
#
copies=expr $6 + 1
cat | lp -f$3 -n $copies
```

If this script, named **myprog**, is called with the PDIR parameters in the example PDIR file shown earlier in this chapter, the host output would be piped to:

```
myjob  11/14/91 18.28.46 STD    STD6    QN    0 30 NEWJOB1  None  None  None  None  1
```

myprog would then calculate the number of copies as 1 and pipe the output to:

```
lp -fSTD -n1
```

A more complicated example of an output program is listed in Appendix A of this manual.

General Guidelines for Output Programs

The program runs with the user and group ID of the primary user and group of the workstation as specified in the workstation configuration unless the permissions on the file allow it to run as owner (for example, the executable has the setuid bit on).

The environment in which the program runs is inherited from the user who started the workstation. However, it is preferable to set any required environment variables explicitly from within the program, rather than relying on the inherited environment. Similarly, if the program needs to access files, it should specify the full path of the file to open (rather than relying on the environment).

If the program uses STDOUT and STDERR, reroute the output (for example, to a file) instead of sending it to the screen (which is the default for HP-UX STDOUT and STDERR).

Routing Received Files with a usr Card

A usr card is an instruction to SNAplus2 RJE that can be included in a job file. It is used to do the following:

- route a single job’s files returning from the SNA host computer.
- notify the user when the file has been returned.

Syntax

`usr=(`*`logon,place`*`[,`*`notify`*`])`

The usr card enables the user to determine where the returning file will be stored. For example, if you queue a file for transmission to the host, and you expect two files to be returned from the SNA host computer, each of those two returning files must contain a usr card if you want to choose the destinations for each of the two files.

To enable this option, configure a printer or punch to send output to the program **`/opt/sna/bin/rjeusr`** or **`/opt/sna/bin/rjeusrpad`**.

If you use a usr card, you should take some precautions regarding the following:

- the values used for the **`logon`** field on the usr card
- using the **`-f`** option with the **`snaprjesend`** command on a file that contains usr card(s).

When **`rjeusr`** or **`rjeusrpad`** handles a file, it changes the group of the file to the primary group of the userid in the logon field of the usr card. Note that files without a usr card will be written with the user and group ID specified in the configuration for the workstation.

Also, if a usr card is used to change the userid of an output file to a userid that is different from that configured for the workstation, **`rjeusr`** or **`rjeusrpad`** will not be able to overwrite this file with other output.

If you use the **`-f`** option, the entire file, including any usr cards, is folded to upper case characters. If you have included a usr card, for example, **`usr=(bc,/users/bc,77`**), it will be folded to **`USR=(BC,/USERS/BC,77`**).

When a file is returned from the SNA host, **`rjeusr`** or **`rjeusrpad`** scans the file for usr cards. If **`rjeusr`** or **`rjeusrpad`** finds a usr card that is in all upper case characters, it folds the entire usr card line back to lower case. Therefore, if you use the **`-f`** option with the **`snaprjesend`** command, and you have usr card(s) in your file, you should use control lines to control the **`-f`** option within the file. For example, if you want to specify directories and/or files in the *place* field with one or more uppercase characters, use control lines. See Example 2 under “Examples of Using Control Lines” in chapter 4.

Usr Card Parameters

The following paragraphs describe the usr card parameters:

logon Your logon entry in **`/etc/passwd`**. The *logon* can be as many as 8 characters.

place *Place* is a path to a directory or a file. The place field can contain as many as 60 characters. Path names in *place* are relative to your logon directory as specified in **`/etc/passwd`** if they do not begin with a slash.

If *place* is a file, the file may or may not have execute permission. If *place* is a file without execute permission, the received file is copied into *place*, overwriting any existing data. Note that in this case, if several returning files have usr cards that specify the same place field, each of the files will overwrite the previous file. The file user ID (UID) and group ID (GID) are changed to the UID and GID of the logon. The file permissions are set to 640.

If *place* has execute permission, *place* is executed and the received file is routed to the executable file. The *place* file will execute with the user and group ID of the primary user and group of the workstation unless the permissions on the file allow it to run as the owner (for example, the executable has setuid bit on). If the setuid bit is set, the effective user is the owner, but the real user is still the primary user for the workstation. (See the man page for setuid(2) for more information.)

When the executable *place* file finishes execution, it returns an exit value to **`rjeusr`** or **`rjeusrpad`**. If this exit value is non-zero, **`rjeusr`** or **`rjeusrpad`** will log an error message to the error log file. You should use, for example, the exit system call if *place* is a C program, or the exit command if *place* is a shell script to return an explicit exit value of zero to **`rjeusr`** or **`rjeusrpad`**.

The **`rjeusr`** and **`rjeusrpad`** programs pass the following arguments to the executable:

$1 = *workstation*
$2 = *base directory*
$3 = *filename* (of the received file)
$4 = *logon* on usr card
$5 = *directory* given for logon in **/etc/passwd**

These arguments append to the value in *place* in the sequence shown. In addition, if PDIR parameters are available to **`rjeusr`** or **`rjeusrpad`** (see “Sending PDIR Information” earlier in this chapter), these parameters will be appended to the arguments. The arguments are separated from *place* and from each other by a single space. Make sure your executable (program or shell script) is written to handle these arguments.

If *place* is a directory, the received file is copied into the directory with a unique system-generated name:

PRT*nnnn* for printer files
PUN*nnnn* for punch files
EXC*nnnn* for exchange device files

where *nnnn* is the next available number in the sequence from 0000 to 9999.

The directory specified in *place* must be writable by the primary user and group for the workstation. See “Precautions When Receiving Files” in this chapter.

NOTE: *Place* cannot be a program in the root directory.

[,notify] An optional field where you can indicate how to be notified when files are received. This field must contain two characters. The table on the following page lists values for the field **`,notify`**, and the kind of notification procedures the values indicate.

For example, to receive a notification message to your terminal or mail, enter one of the following values: 11, 21, 41, or 71.

The default notify value is 77. If you do not specify a notify value in the usr card, the default value of 77 will be used.

Table 1 **Notification Values**

Notification Method When File is Received			
Values	**Mail**	**Terminal**	**None**
11, 21, 41, 71	*	*	
12, 22, 42, 72	*		
14, 24, 44, 74		*	
17, 27, 47, 77 (default is 77)			*

Note that if you select notification to terminal, there are three circumstances requiring some caution:

- If you are logged on to multiple terminals (or you are in multiple windows), the messages go to the lowest tty value for your specified user_id. This means that the notification message may come back to a terminal other than the one from which you issued the **`snaprjesend`** command.
- If you are running windows on your terminal and if the terminal (not an individual window) is the lowest shell, the message goes to the terminal rather than into a window.
- Messages will overwrite a graphical window display. To recover, you must repaint the screen.

You must be logged onto the HP-UX system to receive notification messages to your screen. If you are not logged on, SNAplus2 RJE will discard screen messages.

Including the usr Card in Your File

To use the usr card successfully, apply the following rules:

- You must include a usr card in the first 22,000 bytes of data of each received file from the SNA host computer in order to determine the destination for that file.
- The usr card must be on a single line.
- The usr card may not contain any tabs or blanks.
- The characters “usr” in the usr card are not case sensitive and will be found if typed in all capital letters. If the entire usr card is in uppercase, upon receipt by RJE, **`rjeusr`** or **`rjeusrpad`** folds the entire card to lowercase.
- The contents of the usr card are case sensitive. The *logon* and place can be in either uppercase or lowercase, but you are responsible for making sure that the case remains the same during the queuing process. Refer to the section called “The **`snaprjesend`** Command” for more information about the queuing process.
- A usr card can be located anywhere on a line, with any characters preceding it and following it. For example, it is convenient to include a usr card in a JCL comment as follows:

  ```
  //* a comment usr=(bc,/users/bc,41) more...
  ```

- If more than one usr card is included in a file coming back from the SNA host, only the first one will be used for routing and notification.
- The path name must be the full path name or be relative to the logon directory for *logon*. Do not use metacharacters such as “~”.
- If the output is not ASCII, the product will not be able to scan for a usr card. Therefore, the output will not be routed. This can happen if the **`-b`** option is used with the **`snaprjesend`** command or if the output is not translated to ASCII when it is received from the SNA host.

NOTE: If your *logon*, as listed in the usr card, does not match your logon entry in **/etc/passwd**, you will not receive notification that the file has been received from the SNA host computer, regardless of the notification code you may have specified. A warning will also be logged in the audit log file in this case.

Example of usr Cards

The following example is an illustration of a file called **testfile** which a user is sending to an SNA host computer. The JCL commands in **testfile** specify that the host will return two files to the HP 9000 computer: a print file, and a punch file. To determine the destination for each of these files, the user includes two usr cards as in the following example:

```
//TESTJOB JOB ('ACCTING*INFO'),'PGMR NAME',CLASS=F
//* usr=(jts,/users/sna/testdir/PR1,11)
//STEP1 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSUT2 DD SYSOUT=B
//SYSIN DD DUMMY
//SYSUT1 DD *
usr=(jts,/users/jts/IBMstuff/PU1,11)
ABCDEFGHIJKLMNOPQRST
UVWXYZabcdefghijklmn
opqrstuvwxyz01234567
89!@#$$Ÿ&*()-_=+[{]}
;:’",<.>/? |\~
/*
```

The first usr card indicates that the print file will be routed to the directory **/users/sna/testdir/PR1**. The second usr card specifies that the punch file will be routed to the directory **/users/jts/IBMstuff/PU1**.

To queue the file shown in the figure above and to display the results of queuing the file, the user, logged on as jts, enters the following command:

```
snaprjesend -l RJE5 testfile
```

The result of the queuing process is a file that looks like the previous figure. This result appears on screen.

Note that in the example, the usr cards specify a notification code that causes the user to be notified by both a terminal message and by mail when the files are received.

If the user in the example checks for mail messages after the file has been successfully received back from the host and routed, the messages will be in the following form:

`RJE workstation` ***`WKST`*** **`has routed output sent to`** ***`output_dev`*****`.`**

`Output was routed to` ***`filename`*****`.`**

Routing Received Files Without a usr Card

If a printer or punch is configured to send output to **`rjeusr`** or **`rjeusrpad`** and no usr card is present, the file is placed in the output directory of the workstation's subdirectories (also referred to as the base directory). For example, for workstation RJE1, the output would be placed in the following directory:

`/var/opt/sna/rje/RJE1/output`

A unique system-generated name is assigned to each received file. If the file is a print file, the name is in the following format:

`/var/opt/sna/rje/`***`WKST`*****`/output/PRT`*****`nnnn`***

where *nnnn* is a four-digit sequence number starting at 0000. The sequence starts at 0000 each time a new instance of SNAplus2 RJE is started.

If it is a punch file, the name is in the following format:

`/var/opt/sna/rje/`***`WKST`*****`/output/PUN`*****`nnnn`***

If it is an exchange device file, the name is in the following format:

`/var/opt/sna/rje/`***`WKST`*****`/output/EXC`*****`nnnn`***

The owner and group for the file are the primary user and group configured for the workstation. The permissions are 640.

Taking Precautions When Receiving Files

The following two sections inform you of some precautions you should take when using SNAplus2 RJE to receive files.

Routing by Default

If you do not include usr card(s) in your files, the returned file is kept in the workstation's base directory (**`/var/opt/sna/rje/`*`WKST`*`/output`**). This file will be a copy of an original output file and will be renamed **`PRN`***`nnnn`*, **`PUN`***`nnnn`*, or **`EXC`***`nnnn`*. If **`rjeusr`** or **`rjeusrpad`** cannot access the workstation's base directory, it logs an error message and deletes the output.

Routing by usr Card

If you include usr card(s) in your files, **`rjeusr`** or **`rjeusrpad`** writes to the file or directory you have specified in the place field. If **`rjeusr`** or **`rjeusrpad`** cannot write to place, it logs an error message and notifies you of the error by mail (see “Receiving Error Notification” in this chapter).

In this case, **`rjeusr`** or **`rjeusrpad`** routes output to the base directory and assigns a unique system generated name as described in this chapter under “Routing Received Files Without a usr Card”.

Receiving Error Notification

Regardless of the value you have set for the notify field on the usr card, if there is an error in the routing process, **`rjeusr`** or **`rjeusrpad`** will notify you through mail of the error. Here is an example of the notification you receive.

Example

```
The RJE workstation RJE1 has attempted to route output sent to
PRT1. An error has been detected: SNA9555

SNAplus2 RJE:9555:  RJE1  PRT1  rjeusr/rjeusrpad - received
non-zero exit status after executing /users/user1/test.  Output
will remain in /var/opt/sna/rje/RJE1/output/PRT1.
```

In this example, **`rjeusr`** or **`rjeusrpad`** routed the data to the base directory, **`/var/opt/sna/rje/RJE1/output`** and kept the data in the file with the system generated name **`PRT1`**.

Running the RJE Workstation

The RJE workstation is the program responsible for taking jobs from the spool and sending them to the host for processing. It also receives output returned by the host and routes it as required.

Depending on your SNAplus2 RJE configuration, you may have more than one RJE workstation. You must configure each RJE workstation to run on a particular SNAplus2 computer (which may be either a server or a client). You can start a workstation only on its configured computer, and you must issue any other RJE commands related to the workstation (stopping the workstation, spool commands, or the console program) on the same computer.

A group of users will be assigned to each workstation. Your system administrator will tell you which workstation(s) you are to use.

The workstation does not have to be running when you use the spool commands. You can use the spool commands at any time to add, list, or delete jobs on the spool, and you can run the workstation at any time to process any jobs that are waiting on the spool. The only restriction is that you must start the SNAplus2 software in order for the workstation to run.

While the workstation is running, it takes jobs from the spool in the order in which you submit them, and sends them to the host. When the workstation sends a job to the host, it deletes the job from the spool. If the workstation encounters an error while sending the job (for example, a session failure), it will retry up to 99 times to send the job, depending on the -a option value specified with **snaprjesend**. The default is 10. After the specified number of failed attempts to send the same job, the workstation leaves the job on the spool and continues with the next job.

Workstation Directory Structure

SNAplus2 creates several files and directories for each RJE workstation. It creates these on the computer for which the workstation is configured as subdirectories of **/var/opt/sna/rje/*WKST***, where WKST (in uppercase) is the name of the RJE workstation. See the *HP-UX SNAplus2 Installation Guide* for more details if necessary.

Apart from the subdirectories **`output`** and **`.pgmout`** (temporary files for output to programs), you cannot access these files and directories directly. They are for internal use by RJE programs.

If the required files and directories do not already exist from a previous run of the workstation, SNAplus2 RJE creates them when you start the workstation or submit a job. Since you may require these files and directories later, the workstation program does not remove them when it stops running. As a result, you can spool jobs for the workstation while it is not running.

Clearing a Workstation's Directories

You can remove all the files and directories associated with a workstation, using the **`snaprjeclr`** command. For example,

```
snaprjeclr RJE1
```

The files and directories will be rebuilt if a new job is submitted for the workstation or if the workstation is restarted. You must use the command on the computer where the RJE workstation is configured, and the SNAplus2 software must be started on that computer.

This command is for *SNA user or root only*. Use this command only if a workstation is no longer being used and has been removed from the configuration or if a serious error has occurred. Clearing the workstation's directories will remove the following:

- any spooled jobs
- stored status information for sent jobs
- file on the default output directory
- stored history of console activity

See Chapter 4 for more information on this command.

If you need to remove individual files from directories other than the output directory (for example, to clean up after a serious error condition), you must log on as root, since the files are protected. See your system administrator if you need to remove files and do not have the necessary privilege.

Specifying New Temporary Directory Paths

Currently, if output is sent to a user program, all output is initially sent to a temporary file in **`/var/opt/sna/rje/`*`WKST`*`/.pgmout`**. Also, if the **`rjeusr`** or **`rjeusrpad`** programs are used and they leave output in the base directory, this output is placed in **`/var/opt/sna/rje/`*`WKST`*`/output`**.

You can specify a new directory path for these two directories by defining the **`temp_directory`** option in the **`define_minor_options`** record in the style file (see Chapter 6 of this manual). The RJE workstation will create the directories **`.pgmout`** and **`output`** for this path. If they exist, the permissions will be modified to be the permissions of the user and group configured for the workstation. The user and group configured for the workstation must be able to access the specified path, or an error will be reported.

Do not remove these directories while RJE is running. If you try to remove them, RJE will loop while trying to access the directories.

Setting logon_retry

Normally, when RJE sends a logon to an IBM host and it receives an error sense code, such as 08570002 (application (JES2) not available), it will not retry the logon. This avoids an indefinite retry of the logon when there may be a host or configuration error.

To force the workstation to retry sending the logon string periodically until it succeeds, set the **`logon_retry`** option to YES in the **`define-minor-options`** record in the style file.When you set the **`logon_retry`** option to YES, and RJE has previously had a session with the host, a JES (Job Entry Subsystem) termination will cause RJE to retry the logon (about every 30 seconds). This will result in RJE automatically establishing a session after JES is restarted.

If this is the first time RJE has attempted to establish a session with the host and it receives an error sense code, it will not retry the logon. This feature applies only if RJE has previously logged on to the host and the host terminated the session and if the workstation is defined to send its logon string at startup. You can use the **`snaprjecust`** customization program to set **`Session startup by Workstation Logon`**, or specify the **`send_logon`** parameter in the **`define_workstation_options`** record).

To disable this feature, set **`logon_retry`** to NO.

Starting the Workstation

Before starting an RJE workstation, be sure that the SNAplus2 daemon is running. See your system administrator if necessary, or refer to the *HP-UX SNAplus2 Administration Guide*.

Also, be sure that your login ID is in the group configured for this workstation.

To start the RJE workstation, use the following command:

```
snaprjestart WKST
```

NOTE: The workstation name (*WKST*) can be only four characters long and is always in uppercase. If you enter it in lowercase, SNAplus2 RJE will convert it to uppercase. For more information on this command, see chapter 4 of this manual.

Normally, once the workstation starts, it will continue to run until you stop it with the **`snaprjestop`** command (see the section “Stopping the Workstation” later in this chapter). Alternatively, you can specify a time-out parameter with the **`-sn`** option when starting the workstation. In this case, the workstation stops when both of the following conditions are true:

- All the spooled jobs have been sent to the host and the specified time (in minutes) has elapsed with no RJE activity such as, no jobs submitted, no console commands entered, and no output or console messages received.
- The workstation has lost contact with the local node, and the specified time (in minutes) has elapsed without reestablishing contact. This generally occurs because the local node has been stopped.

If you need to stop the workstation quickly without waiting for RJE activity to end, use the **`snaprjestop`** command.

Displaying Workstation Status

To display a workstation's status information, ensure that your login ID is in the group configured for the workstation. See your system administrator if necessary.

Use the **`snaprjestat`** command to display information about the workstation's LUs, devices, or both, depending on the options you specify. For more information on this command, see Chapter 4 of this manual.

To display status information on sent jobs, use the **`snaprjelst`** command to check the status (**waiting**, **sending in progress**, or **failed**) of jobs that have been spooled for a particular RJE workstation. Normally, when SNAplus2 sends a job successfully to the host, it removes the stored status information for the job, so that the **`snaprjelst`** command no longer returns it.

You can set up the workstation to retain status information on sent jobs so that you can view the information and check that the jobs have been sent. Use the **`list_sent_jobs`** option in the RJE workstation style file (see Chapter 6 of this manual for more information on this option).

If you use this option, SNAplus2 will retain the stored status information until you purge it with the **`snaprjepur`** command. You should use this command regularly to avoid storing too much information, since retaining status information on large numbers of files will use up disk space and may affect the performance of RJE programs.

For more information on the **`snaprjelst`** and **`snaprjepur`** commands, see Chapter 4 of this manual.

Stopping the Workstation

To stop the RJE workstation, your login ID must be in the group configured for the workstation. Use the following command to stop the workstation. For more information on this command, see Chapter 4 of this manual.

```
snaprjestop WKST
```

Following are three options for stopping a workstation:

Wait for inactivity (-s*n*)

With this option, the workstation continues sending any spooled jobs. It stops only when the spool is empty and no jobs have been submitted or output received from the host for a specified time.

Default: The default is a one-minute inactivity time-out (**`-s1`**). If no option is specified, the workstation stops after one minute of no RJE activity.

Because you may submit jobs at any time regardless of whether the workstation is running, you may add further jobs to the spool after you issue the stop command. The workstation continues to send these jobs, and does not stop until the spool is empty. It may, therefore, take some time for the stop command to take effect. If necessary, you can override this activity by issuing the stop command again with one of the other options.

Finish Job (-j)

With this option, the workstation finishes sending the current job, if any, and then stops. Any other spooled jobs remain on the spool and may be sent during a later run.

Immediate (-i)

With this option, the workstation stops immediately, even if it is currently sending a job to the host or receiving data. Since it disrupts both sending and receiving RJE data, you should only use this option in error situations.

Getting Information About RJE Jobs

SNAplus2 logs detailed information about actions that occur on jobs and their data files that are received from the host. It sends information to the **RJE job log file** (if the **job_logs** option is defined) and to the RJE **audit log file** (if audit logging is enabled). Another way of getting RJE job information is by using the **snaprjelst** command.The following sections explain these three methods.

Audit Log File

The default name for the audit log file is **/var/opt/sna/sna.aud**. Since the audit log file is a text file, you may view it with a text editor or write your own program to filter and format the information you desire.

Note that SNAplus2 backs up the audit log file and resets it when it reaches a size of one megabyte. The backup file is named bak.aud. The following example shows some typical entries:

```
------------------ 11:29:14 PDT  09 Apr 1996 ------------------
RJE     Message 32770 - 35, Subcode: 0 - 1
Log category: AUDIT     Cause Type: Audit
System: hpntcbf
Process ID: 4101 (snaprjesend)

RJE:RJEF  snaprjesend - Job file submitted to job spooler
User id  = bill
Userfile = demo2B
Filetype = MIXED
*
```

Cause: RJE has processed a file submitted by a user and has submitted the processed file to the job spooling program **rjesub**. See the *HP-UX SNAplus2 RJE User's Guide* for more information.

Action: No action is required.

```
------------------ 11:29:17 PDT  09 Apr 1996 ------------------
RJE     Message 16389 - 90, Subcode: 215 - 1
Log category: AUDIT     Cause Type: Audit
System: hpntcbf
Process ID: 4107 (rjesub)

RJE RJEF: Job spooled.
Logon    = bill
Options  = -
Userfile = demo2B
Spool ID = SPL_rttnyq
```

Cause: An RJE user has successfully added a job to the spool. This message shows details of the spooled job.

```
------------------ 11:29:18 PDT  09 Apr 1996 ------------------
RJE     Message 16389 - 101, Subcode: 102 - 1
Log category: AUDIT     Cause Type: Audit
System: hpntcbf
Process ID: 3967 (snaprjestart)

RJE RJEF: Job sent to host
Logon       = bill
Retry count = 0
Userfile    = demo2B
Spool ID    = SPL_rttnyq
```

Cause: The workstation shown has successfully sent a job to the host. This message shows details of the job.

```
------------------ 11:29:23 PDT  09 Apr 1996 ------------------
RJE     Message 16389 - 63, Subcode: 68 - 1
Log category: AUDIT     Cause Type: Audit
System: hpntcbf
Process ID = 3978 (snaprjestart)

RJE RJEF: Received host output on PRT1 device
Routed to file /home/bill//RJEFPRT1
Size of output = 5971 bytes
File type      = PRINTER
Mode           = TEXT
```

Cause: The workstation shown has successfully received data from the host. The message shows the type of device to which the data is being sent (printer, punch, or exchange), and either PRT (printer) or PUN (punch) followed by the printer or punch number.

If output is being appended to a file, note that the output size shown is the size of the complete file after the new data has been appended, not the volume of data added.

```
------------------ 11:29:23 PDT  09 Apr 1996 ------------------
RJE     Message 16389 - 63, Subcode: 55 - 1
Log category: AUDIT     Cause Type: Audit
System: hpntcbf
Process ID: 3967 (snaprjestart)

RJE RJEF: Received host output on PUN2 device
Routed to program /opt/sna/bin//rjeusrpad
Size of output = 111 bytes
Filetype       = PUNCH
Mode           = TEXT
```

Cause: The workstation shown has successfully received data from the host. The message shows the type of device to which the data is being sent (printer, punch, or exchange), and either PRT (printer) or PUN (punch) followed by the printer or punch number.

If output is being appended to a file, note that the output size shown is the size of the complete file after the new data has been appended, not the volume of data added.

```
------------------ 11:29:23 PDT  09 Apr 1996 ------------------
RJE     Message 32770 - 83, Subcode: 0 - 1
Log category: AUDIT     Cause Type: Audit
System: hpntcbf
Process ID: 4115 (rjeusrpad)

RJE: RJEF  rjeusr/rjeusrpad has routed output for PUN2.
Output contains no usr card
Filename = /var/opt/sna/rje/RJEF/output/PUN0004
```

Cause: An output device was configured to do usr card routing by using the **`rjeusr`** or **`rjeusrpad`** program. A usr card was not found in the output, so the output file was not routed. The output is left in the file specified with the message.

Action: If you want the output routed, use a usr card in the data. Consult the *HP-UX SNAplus2 RJE User’s Guide* for information about the usr card. If a usr card is used, the device must be configured to convert EBCDIC data to ASCII in order for the usr card to be readable.

```
------------------ 12:58:31 PDT  09 Apr 1996 ------------------
RJE     Message 32770 - 66, Subcode: 0 - 1
Log category: AUDIT     Cause Type: Audit
System: hpntcbf
Process ID: 4148 (rjeusrpad)

RJE: RJEF  rjeusr/rjeusrpad has routed output for PUN2
usr logon = bill
Filename  = /home/bill/file1
```

Cause: **`rjeusr`** or **`rjeusrpad`** has successfully routed an output file to the indicated file name.

Action: No action is required.

```
------------------ 13:00:20 PDT  09 Apr 1996 ------------------
RJE     Message 32770 - 67, Subcode: 0 - 1
Log category: AUDIT     Cause Type: Audit
System: hpntcbf
Process ID: 4176 (rjeusrpad)

RJE: RJEF  rjeusr/rjeusrpad has routed output for PUN2.
usr logon = bill
Program   = /home/bill/program
```

Cause: **`rjeusr`** or **`rjeusrpad`** has successfully routed an output file to the indicated program name.

Action: No action is required.

RJE Job Log File

Entries are written to the RJE job log file only if the **`job_logs`** option is set to YES in the workstation’s style file. (For more information on setting this option, see Chapter 6 of this manual.) If this option is set, SNAplus2 writes an entry to the file each time a job is spooled to the workstation, sent to the host or cancelled, and each time output is received from the host.

The file is a text file that can be viewed using a standard ASCII text editor. Each entry in the file consists of two lines: the first gives the date and time at which the entry was written, and the second gives details of the event that caused the entry. The example below shows some typical entries.

```
Tue Apr 09 11:29:18 PDT 1996
RJEF QUED bill - * SPL_rttnyq demo2B

Tue Apr 09 11:29:18 PDT 1996
RJEF SENT bill 0 * SPL_rttnyq demo2B

Tue Apr 09 11:29:23 PDT 1996
RJEF CANL bill * * SPL_rttnyq demo2B

Tue Apr 09 11:29:23 PDT 1996
RJEF RECV * PRT1 TEXT 5971 /home/bill//RJEFPRT1

Tue Apr 09 11:29:23 PDT 1996
RJEF RECV * PUN2 TEXT 111 /opt/sna/bin//rjeusrpad

Tue Apr 09 11:29:23 PDT 1996
RJEF ROUT * PUN2 NO-USR * /var/opt/sna/rje/RJEF/output/PUN0004

Tue Apr 09 12:58:32 PDT 1996
RJEF ROUT bill PUN2 USR * /home/bill/file1

Tue Apr 09 13:00:22 PDT 1996
RJEF ROUT bill PUN2 USR * /home/bill/program
```

The first two fields of the second line of each entry give the name of the workstation to which the job was submitted (or the workstation that processed the host output), and the type of event that caused the log.

The event type is one of the following:

QUED The job was successfully submitted to the workstation and is queued waiting to be sent to the host.

SENT The workstation has sent the job to the host.

CANL The user cancelled the job while it was waiting on the spool.

RECV The workstation has received output from the host.

ROUT The job was routed by **`rjeusr`** or **`rjeusrpad`**.

The remaining fields in the entry depend on the event type. The fields shown for each event type are listed below. In some cases, the character “*” appears as a field in an entry. This indicates that a field stored internally by SNAplus2 is not used for that event type.

QUED: Job Submitted to the Workstation

The remaining fields are as follows:

- The user name of the user who submitted the job to the workstation
- The combination of command-line options used on the job submission command (see Chapter 4 for more information on these options).
- “*” (unused field)
- The spool ID of the submitted job
- The job file name that was specified on the job submission command

SENT: Job Sent to the Host

The remaining fields are as follows:

- The user name of the user who submitted the job to the workstation
- The number of retries before the job was successfully sent (a value of zero indicates that the job was sent on the first attempt. “1” indicates one failure before a successful send)
- “*” (unused field)
- The spool ID of the submitted job
- The job file name that was specified on the job submission command

CANL: Job Cancelled

The remaining fields are as follows:

- The user name of the user who cancelled the job (this is normally the user who submitted the job to the workstation, except that a user logged on as **root** can cancel any job)
- “* *” (two unused fields)
- The spool ID of the submitted job
- The job file name that was specified on the job submission command.

RECV: Output Received from the Host

The remaining fields are as follows:

- “*” (unused field)
- The name of the output device specified by the host: PRT*n*, PUN*n*, or EXC*n* (printer, punch or exchange device)
- The data type specified by the host: TEXT (text data) or TRANSP (binary data)
- The size of the output file (in kilobytes)
- The directory and file name of the output file (or, for program output, the name of the output program to which the output was sent)

ROUT: Output Routed by rjeusr/rjeusrpad

The remaining fields are as follows:

- The name of the user on the usr card. If no usr card is used, this field will be “*” (unused field).
- The name of the output device to which the output was routed: PRT*n*, PUN*n*, or EXC*n*.
- If output was routed with a usr card, “USR” appears in this field. If no usr card was used, “NO-USR” appears in this field.
- “*” (unused field)
- The directory and name of the output file to which the output was routed

The RJE job log file will grow without bounds. You should check the file at intervals and delete it, or delete older entries to prevent the file from growing too large and using too much disk space.

Other Sources of Information

Another way getting information about RJE jobs is with the **`snaprjelst`** command. If you use this command with the **`-s`** option, you can list all of the jobs that have been sent to the host. This includes only the jobs that have been successfully sent to the host.

If you use this command with the **`-b`** option, you can list both jobs waiting and jobs sent, including jobs that were not successfully sent to the host.

For more information on the **`snaprjelst`** command, see Chapter 4.

Workstation Mail Messages

For most messages that RJE prints to STDERR, there is a corresponding error message printed to either the audit log or error log.

The following are mail messages that you may receive from the SNAplus2 RJE workstation:

SNAplus2 RJE:9592

MESSAGE `Message from SNAplus2 RJE workstation `***`WKST`***`. Notification ID `***`#number`***`. Error receiving to file or program `***`name`***`. Output file has been deleted. Incomplete data file may be written for next output to `***`output device`***`. Check error log file for SNAplus2 error messages.`

CAUSE RJE has encountered an error while receiving an output file. The number of the error message is displayed.

ACTION Check the audit log file and error log file for error messages related to the SNAplus2 error number. Check the output files for the specified device. (The name of the output file can be determined from the audit log file if job logging is enabled). In the case of output being appended to a file, a temporary file may exist in **`/var/opt/sna/rje/`*`WKST`*`/output`**. The name of this file will be logged to the audit log (see “Routing Output” earlier in this chapter). If the output files contain incomplete data, resubmit the job.

SNAplus2 RJE:9593

MESSAGE

`Message from SNAplus2 RJE workstation` *`WKST`*`. Error receiving to file or program` *`name`*`. Output file has been deleted. Incomplete data file may be written for next output to` *`output device`*`. Check error log file for SNAplus2 error messages. The workstation has reached the maximum number of times it will mail messages about this error. Restart the workstation if you wish to continue receiving this message.`

CAUSE

RJE has encountered an error while receiving an output file. There is a counter that determines how many mail messages will be sent for this error. It has reached the maximum number of times it will send out this message.

ACTION

Check the audit log file and error log file for error messages related to the SNAplus2 error number. Check the output files for the specified device. (The name of the output file can be determined from the audit log file if job logging is enabled). In the case of output being appended to a file, a temporary file may exist in **`/var/opt/sna/rje/`*`WKST`*`/output`**. The name of this file will be logged to the audit log (see “Routing Output” earlier in this chapter). If the output files contain incomplete data, resubmit the job. Restart the workstation if you wish to reset the counter for mailing messages about this error.

4

SNAplus2 RJE Commands

This chapter explains each of the commands used to control SNAplus2 RJE. These commands allow you to start and stop the workstation, submit, list, or cancel jobs, check the workstation status and issue host commands.

snaprjestart

The **`snaprjestart`** command starts the RJE workstation which performs the following tasks:

- Transfers jobs from the spool to the host for processing
- Receives output from the host and routes it to the appropriate file or program.
- Provides access to host console information

Start the SNAplus2 RJE workstation on the computer for which it is configured. The SNAplus2 software must also be started on that computer. See your system administrator if necessary, or refer to the *HP-UX SNAplus2 Administration Guide* for information on RJE workstation configuration and starting the SNAplus2 software.

Syntax

`snaprjestart [-sn] WKST`

Options

-sn

The **`-sn`** option, where *n* is a decimal number (minimum 1), indicates that the workstation should stop after *n* minutes with no RJE activity. In other words, the workstation stops when there have been none of the following for *n* minutes:

- Jobs on the spool waiting to be submitted

- RJE console commands

- Output returned from the host

- Console messages from the host

If this option is not used, the workstation continues to run indefinitely, whether or not any work arrives. To stop it, you must use the **`snaprjestop`** command.

Parameters

WKST	***WKST*** is the name of the RJE workstation to start. This must be the name of an RJE workstation in the SNAplus2 configuration file, and your user ID must be in the group configured for this workstation. See your system administrator if necessary.

Example

```
snaprjestart -s2 RJE1
```

When you use the command as in the example above, the program responds with the following message:

```
RJE Workstation WKST has started
```

Other messages may be sent to standard error, indicating (for instance) that an invalid option or an invalid workstation name was supplied, that the SNAplus2 software was not started, or that the workstation was already running.

Check the error log file for other information when error messages display.

snaprjestop

The **`snaprjestop`** command stops the RJE workstation.

You must use this command on the computer on which the workstation is running (the computer for which it is configured). See your system administrator if necessary, or refer to the *HP-UX SNAplus2 Administration Guide* for information on RJE workstation configuration.

Syntax

```
snaprjestop [-sn | -j | -i] WKST
```

Options

-sn The **`-sn`** option is the “inactivity” option. It stops the workstation after *n* minutes of inactivity. The workstation continues to process any jobs waiting on the spool. It stops when there have been none of the following for *n* minutes (minimum 1):

- Jobs on the spool waiting to be submitted

- RJE console commands

- Output returned from the host

- Console messages from the host

-j The **`-j`** option is the “finish job” option. If the workstation is currently sending a job to the host, and/or receiving data from the host, it finishes these transactions. It then stops without processing further jobs. Any unprocessed jobs remain on the spool and may be sent during a later run of the workstation.

-i The **`-i`** option is the “immediate” option. When you use this option, the workstation stops immediately. Any job currently being sent to the host, and any output currently being received from the host is abandoned. Use this option only in error conditions since it will disrupt RJE processing. If a job was abandoned during sending, you will need to cancel and resubmit it in order to send it to the host when the workstation is restarted. (Some hosts may try to resend output that was abandoned during receiving.)

Only one of these options may be specified. If none is specified, the default is **`-s1`** (stop as soon as one minute has elapsed without RJE activity).

When you use the **`-s`*`n`*** option, the workstation may take some time to stop. If necessary, you can use **`snaprjestop -i`** or **`snaprjestop -j`** to override a pending **`snaprjestop -s`*`n`*** if you need to stop the workstation quicker.

Parameters

WKST ***WKST*** indicates the name of the RJE workstation to stop. This must be the name of a running RJE Workstation.

Example

```
snaprjestop -s2 RJE1
```

When you use the command as in the example above, the program responds with the following message:

`RJE Workstation` ***`WKST`*****`: Stop command issued, level`** *level*

level indicates the stop option used - “Immediate”, “Finish Job”, or “Inactivity”.

Other messages may be sent to standard error, indicating (for instance) that an invalid option or an invalid workstation name was supplied, that a parameter in the workstation style file was invalid, or that the workstation was not running.

Check the error log file for other information when error messages display.

snaprjesend

The **`snaprjesend`** command submits a job file to the RJE workstation for transfer to the host. It produces a spool file for the named workstation and returns the file ID for this spool file. Normally, the job file will be searched for the control lines and it will be converted from ASCII to EBCDIC. You can override these options if required.

You must submit the job on the computer for which the RJE workstation is configured, and the SNAplus2 software must be started on that computer. See your system administrator if necessary, or refer to the *HP-UX SNAplus2 Administration Guide* for information on RJE workstation configuration or for information on starting the SNAplus2 software.

The RJE workstation does not have to be running when this command is issued. The file is simply added to the spool. When the workstation is run at some later time, it will send the job to the host for processing.

You can also start the workstation at the same time you submit the job (see “Options” below.) However, this does not guarantee immediate sending of this job, since other jobs already spooled for this workstation will be processed first.

Syntax]

```
snaprjesend [options] WKST filename
```

Options

-b Indicates that SNAplus2 should treat the job file as a binary file. Job files will normally be submitted in ASCII, and SNAplus2 translates them to EBCDIC. This option overrides the conversion, and sends the file as submitted. This option remains enabled until the next end-of-file (EOF) is encountered. No scanning for control lines occurs with this option. If you specify **`-b`** in the command line, you must specify a file name also. Only the **`-q`** option can be used in conjunction with this option.

-c Lists control lines to standard error as the file is interpreted for transmission.

-e Specifies the use of the RJE exchange device as the source of the file being sent. Data is sent as a basic exchange data set with a default record length of 128 bytes.

-f Folds lower-case characters to upper-case characters.

-h Lists the input lines to stdout without expanding tabs. The **`-h`** option cannot be used with the **`-l`** option. (Binary data will not be listed.)

-i Does not scan the input for control lines. This option remains enabled until the next end-of-file (EOF) is encountered.

-l Lists the input lines to stdout. Note that this listing shows the tabs expanded. The tab size is eight characters. (Binary data will not be listed.)

-q Does not submit the file for queuing, but processes the file normally. This is useful for checking your file before you queue it for transmission. Use the **`-q`** and the **`-l`** options together, or the **`-q`** and the **`-h`** options together to display the results of the queuing process to stdout.

-t Prints the character > on standard error when input sources are opened.

-y Suppresses error diagnostics and submits the file anyway.

-a*n* Specifies the number of retry attempts for this job. If an error occurs during the sending of this job, the workstation will retry sending the job as many as *n* times. The maximum value for *n* is 99. *n* must be specified with this option. If **`-a`*`n`*** is not specified, the number of retry attempts defaults to 10.

-z

Submits file for transmission without interpreting the file. Any other options specified in this command that require reading the file will be ignored. You must include a file name when you specify this option. This is the fastest way to queue a file, because the input is not interpreted line by line. Only the **`-q`** option may be used with this option. Any file that you queue with this option will be translated ASCII to EBCDIC.

-m*n*

the **`-m`*`n`*** option indicates that the job file should be sent in records of n bytes (in the range 80 - 248). If this option is not used, the default record length is 80 bytes for punch data (without the **`-e`** option), or 128 bytes for exchange data (with the **`-e`** option). See “Sending Exchange Data” in chapter 3.

-r [-s*n*]

The **`-r`** option runs the RJE workstation if it is not already running after submitting the job.

The **`-s`*`n`*** option is only valid if the **`-r`** option is also specified. The combination of **`-r`** and **`-s`*`n`*** is equivalent to the command **`snaprjestart -s`*`n WKST`***, which specifies that the workstation should start, complete all outstanding jobs, and then time out after *n* minutes with no RJE activity. See the section on the **`snaprjestart`** command for more information.

Parameters

WKST The name of the RJE workstation in the SNAplus2 configuration file that will process the job. This parameter must be specified whether or not the workstation is currently running.

file The name of the file to be sent. If file is not specified, input is taken from STDIN. When a file is interpreted by snaprjesend, each control line must be no greater than 512 characters and any tabs in the line will be expanded to eight blanks. The maximum fully qualified file name length is 1024 characters. However, for job spooling and audit logging, the job file name cannot exceed 200 characters. If the file name exceeds 200 characters, the job will be sent, but the job file name will not be logged in the audit log file or show in the job spooler listing.

For job files that are sent as mixed data (-b and -z options are not used) no one section of data (the job file or the individual included files) can be larger than 2 gigabytes.

Examples

To send a file named **rocky** to the workstation RJE1 without any options, you enter the following:

```
snaprjesend RJE1 rocky
```

To view the results of the queuing process for the file **rocky** without actually queuing the file, you enter the following:

```
snaprjesend -q -l RJE1 rocky
```

or

```
snaprjesend -ql RJE1 rocky
```

Using Control Lines

In either form, this command displays the results of the queuing process on the screen and does not queue the file.

Control lines can be used to modify the source data being sent to the SNA host. Control lines can be included in the source file or entered from the terminal depending on the command options being used.

A control line begins with a tilde (~). You can insert control lines into your file to open another input source within a job, or to set or clear command line options. A line in the source data that is recognized as a control line will cause snaprjesend to perform a function and the line will not be included in the queued data. The function of the control line is specified by use of the following arguments:

- the period (.)
- the hyphen (-)
- the plus sign (+)
- the colon (:)
- the exclamation mark (!)
- the tilde (~)

Control Line Syntax

When you use control lines, follow these syntactical rules:

- The tilde (~) must be the first character in the line.
- There cannot be a space between the tilde and its argument.
- The character after the tilde must be the control argument.
- Whenever control lines are scanned, the maximum control line size is 512 characters including the ~ character and the control arguments.

Control Line Commands

The control line commands are listed on the following pages.

Syntax | **Function**

~.

Close the current source. If the file is being scanned for control lines (the **`-b`** and **`-i`** options are not set), enter **`~`**. to close the current source.

Example usage: If you have entered input from the terminal, for example, and you want to close the current source, use the **`~`**. command.

Note that if you are entering data from the terminal and the input is not being scanned for control lines (the **`-b`** option or the **`-i`** option is set), close the current source by entering [CTRL]-D once at the beginning of a line or three times if you are not at the beginning of a line.

~-option(s)

Set an option or options. The snaprjesend options that can be set include the following: If you specify more than one option, there must be no spaces between the options.

Example usage: To set the options **`-f`** and **`-c`**, for example, you would enter the command:

```
~-fc
```

In this example, lower case to upper case folding of characters will occur and control lines will be listed on stderr until the options are cleared.

~+option(s)

Clear an option or options. The snaprjesend options that can be cleared include **`-c, -f, -h, -l`**, and **`-t`**. If you specify more than one option, there must be no spaces between the options.

Example usage: To clear the options **`-f`** and **`-c`**, for example, you would enter the command:

```
~+cf
```

Lower case to upper case folding of characters will not occur and control lines will not be listed on stdout. The options **`-f`** and **`-c`** would be cleared regardless of their previous settings.

~- [option]:[prompt]

Open STDIN as new source and optionally prompt user.

You can specify either the **`-b`** or the **`-i`** as an option to take input from this source. The -b and the **`-i`** options cannot be specified at the same time.

Note that there must be a space between the **`~-`** and the option. The **`-`** must be specified with the option.

If you specify the **`-b`** option or the **`-i`** option, interpretation of control lines is halted until an EOF for the new source is read. At that time, the options will no longer be in effect. This allows you to interpret control lines for some sources and not for others. For example, this is useful if you want to send binary data to an SNA host computer and you need to include JCL before the data.

Refer to the **`~.`** option for information about entering data from STDIN.

If STDIN is the terminal, note the following: all input taken from the terminal in binary mode is entered as character data, and carriage returns are included in the file whenever you press Return.

Example usage:

To open STDIN as source and specify the **-b** option and a prompt that says Data being read from **bin_file**, for example, you would enter:

```
~- -b:Data being read from bin_file
```

~+ [option]:[prompt] Open terminal as source and optionally prompt user.

You can specify either the **-b** or the **-i** option to take input from this source. The **-b** and the **-i** options cannot be specified at the same time.

Note that there must be a space between the **~-** and the option. The **-** must be specified with the option.

If you specify the **-b** option or the **-i** option, interpretation of control lines is halted until an EOF for the new source is read. At that time, the options will no longer be in effect. This allows you to interpret control lines for some sources and not for others. For example, this is useful if you want to send binary data to an SNA host computer and you need to include JCL before the data.

Refer to the **~.** option for information about entering data from STDIN.

Note the following: All input taken from the terminal in binary mode is entered as character data, and carriage returns are included in the file whenever you press Return.

Example usage:

To open terminal as source and specify the **-b** option and a prompt that says Enter data now, for example, you would enter:

```
~+ -b:Enter data now
```

~! [option]command

Execute an HP-UX command

Execute the specified HP-UX command through the shell. Any output from the command will be included in the file to be transmitted. Valid **snaprjesend** options are either **-b** or **-i**. Note that **-b** and **-i** cannot be specified at the same time. These options will not be in effect after the shell command has terminated.

If you do not specify the **-b** option, snaprjesend expects that all data in a job is text data. If you include data within the job that contains non-text characters (such as nulls), a problem might occur when you submit the job. You might receive the following error message:

```
SNAplus2 RJE:9513 - error copying
data during interpretation
```

To avoid this problem, include the data as binary data. If you want to convert this data from ASCII to EBCDIC, you can do it while including the data with the **iconv** command. For example:

```
~! -b iconv -f roman8 -t american_e filename
```

(See the man page iconv (1) or the *HP-UX Reference* Vol1: Section 1).

Note that if you use **~! -b**, a record separator will be placed after the last byte of data. This will not necessarily occur if you do not specify **-b**. If the **-b** option is not used, end all included text files with a new-line character in order to separate them from your JCL.

If the shell command returns a non-zero status, the **`snaprjesend`** command will terminate, and the job will not be submitted. This error can be overridden with the **`-y`** option.

~:message Print the contents of message on stdout.

~~comment Ignore this line.

NOTE: You must type in commands with the exact spacing shown in this list.

Control Line Examples

The following examples show how you might use control lines with the snaprjesend command.

Example 1

To send a binary file to an SNA host computer with one **`snaprjesend`** command that needs to include JCL statements as well as the file name, you would do the following:

- Create a new file that consists of a series of control lines to include the different input source files within the job.
- Set options within the control lines so that the JCL statements are translated from ASCII to EBCDIC, but the binary data file is not translated.

If you have the following files,

File Name	Contents
JCLFILE	JCL commands that must be translated from ASCII to EBCDIC
datafile	a file of binary data that must not be translated from ASCII to EBCDIC
MOREJCL	JCL commands that must be translated from ASCII to EBCDIC

you can create the following master file called **`masterfile`**:

File Contents	What the Control Line Does
~! cat JCLFILE	reads the file **`JCLFILE`** and sends the file in text mode.
~! -b cat datafile	reads the file **`datafile`**, and turns on the **`-b`** option to suppress character translation and send the file in transparent mode. The **`-b`** option remains in effect until the EOF is encountered in **`datafile`**.
~! cat MOREJCL	reads the file **`MOREJCL`** and sends that file in text mode.

To send the job created by **`masterfile`** to the workstation called **`RJE1`**, you would then enter the **`snaprjesend`** command:

```
snaprjesend RJE1 masterfile
```

Example 2

To send the file, **`masterfile`**, to the SNA host converting all lowercase characters (except those in a usr card) to uppercase, enter this command:

```
snaprjesend -f RJE1 masterfile
```

where **`masterfile`** contains the lines:

```
       .
       .
       .
    //* JCL cards here
    ~~ Comment line. The usr card will remain in lowercase.
    ~+f
    //* usr=(sam,/users/sam/results,77)
    ~-f
    ~~ Comment line. Folding to uppercase continues here.
   //* DATA HERE
       .
       .
       .
```

Example 3

Use the **`~- [option]:prompt`** control line for opening STDIN as a new source. You can use this control line to redirect input from a file. Note that the prompt in this case is only a message to indicate that the file is being read since the file is being redirected to STDIN. Use the **`snaprjesend`** command:

`snaprjesend RJE1 masterfile < datafile`

where **`masterfile`** contains the lines:

```
 .
 .
 .
//* JCL cards here
~~ Comment line. Read the file from STDIN and do not
~~ translate to EBCDIC. Also, display a message to the user
~~ the file is being read.
~- -b:Data being read from STDIN
//* ADDITIONAL DATA HERE OR END OF FILE
 .
 .
 .
```

Example 4

To interactively enter data from the terminal during the **`snaprjesend`** command, use the **`~+ [option]:prompt`** control line.

Enter the command:

`snaprjesend RJE1 masterfile`

where **`masterfile`** contains:

```
       .
       .
       .
    //* JCL cards here
    ~~ Prompt the user to begin to enter data. Turn off
    ~~ control lines so that if the data contains a tilde
    ~~ character, it will not be interpreted incorrectly.
    ~+ -i:Begin entering data from the terminal now
       .
       .
       .
```

Example 5

To view the input as the file is being scanned, use the **`~-`**options control line with either the **`-h`** or **`-l`** option. The -h option will not expand tabs to 8 blanks and the -l option will expand tabs to 8 blanks.

For example:

`snaprjesend RJE1 masterjob`

where **`masterjob`** contains:

```
       .
       .
       .
    //* JCL cards here
   ~~ Note that since the control line for listing input is not
    ~~ at the top of the file, none of the JCL lines above will
    ~~ be listed to the screen.  The following lines will be
    ~~ listed to the screen with tabs expanded to 8 blanks.
    ~~ lines are listed:
    ~-l
   //* END OF FILE
   /*
```

The number of retries specified with the **`-a`** option for **`snaprjesend`**. If this option is not used, the default is ten retries.

snaprjelst

The **`snaprjelst`** command lists all the job files on the spool for an RJE workstation. The default of this command is to list only jobs that have been submitted either by you or by other users in your group and have not yet been sent to the host. However, by using the **`-s`** or **`-b`** option with the **`snaprjelst`** command, you can list jobs sent or both jobs waiting and sent.

You must use this command on the computer for which the RJE workstation is configured, and the SNAplus2 software must be started on that computer. However, the RJE workstation need not be running. See your system administrator if necessary, or refer to the *HP-UX SNAplus2 Administration Guide* for information on RJE workstation configuration or for information on starting the SNAplus2 software.

Syntax

```
snaprjelst [ -s | -b ] [-l] [-a] WKST
```

Options

-s

The **`-s`** option is for "sent jobs only". By default, the **`snaprjelst`** command lists only jobs that have been submitted but not yet sent to the host. This includes jobs that are currently being sent, or jobs for which send attempts have failed. The **`-s`** option lists only jobs that have been successfully sent to the host (see also **`-b`** below).

-b

The **`-b`** option is for "both waiting jobs and sent jobs". This option lists all submitted jobs, regardless of whether they have been successfully sent to the host (see also -s above).

-l

The **`-l`** option indicates "long listing". By default, this command lists only the spool ID, login name, and job file name for each job. The **`-l`** option includes more detailed information. See "Using the Command" below for more information.

-a The `-a` option indicates “all users”. By default, this command lists only the jobs that were submitted using your user ID. The `-a` option lists all jobs on the spool, submitted either by you or by other users of the workstation.

Parameters

WKST The name of the RJE workstation for which you want to list the submitted jobs. This must be the name of an RJE Workstation in the SNAplus2 configuration file, whether or not the named workstation is currently running.

Example

To list all the job files on the spool submitted with your user ID for workstation RJE1, type the following command:

```
snaprjelst RJE1
```

To list all the jobs on the spool submitted by you or other users for workstation RJE1 and to obtain the long listing, type the following command:

```
snaprjelst -l -a RJE1
```

The program responds with the following short listing if you do not specify the -l option:

```
spool-id, login name, jobfilename
```

If you specify the -l option, the program responds with the following long listing:

```
spool-id, login name, date, time, status, maxretry,
retrycount, options, filesize, jobfilename
```

Following are the descriptions of these parameters:

spool-id The spool ID returned when the job was submitted. This is the ID required by the **`snaprjecan`** command to cancel the job. For a spooled job, this ID begins with the characters SPL.

For a job that has been sent to the host (listed using the **`-s`** or **`-b`** option), this ID begins with the characters STS, and is the ID required by the **`snaprjepur`** command to purge the stored status information for this job.

login name The login name of the user who submitted the job.

date and time The date and time when the job was submitted.

status Status can take the following values:

WAITING - the job has not yet been sent to the host

SENDING - the job is currently being sent

FAILED - the attempt to send the job failed (either the RJE workstation terminated during sending and has since been restarted, or it has retried the maximum number of times without success - see retrycount below).

SENT - the job has successfully been sent to the host.

If you use the **`-s`** option, only jobs in the SENT state will be listed. If you do not use either **`-s`** or **`-b`**, only jobs in the WAITING, SENDING, or FAILED state will be listed.

maxretry The number of times the RJE workstation will try to send the job. Normally, this number is 10; however, it will retry the maximum number of times specified when the job was submitted. If it fails after the maximum number of retries, the Job’s status changes to FAILED and the job is abandoned.

retrycount The number of times the RJE workstation has tried to send the job.

options Shows whether the following options were specified with the **`snaprjesend`** command: **`-z, -b, -e, -m`**.

filesize The size of the job file in kilobytes.

jobfilename The name of the job file submitted.

Example of short listing

```
SPL_fgdjlh jim        jim/myjob.file
```

Example of long listing

```
SPL_fgdjlh jim     11/11/91 12:47:56 WAITING  10   0 -   5 jim/myjob.file
```

Other messages may be sent to standard error, indicating the following:

- An invalid option was used.
- An invalid workstation name was supplied.
- No jobs were on the spool.

snaprjecan

The **`snaprjecan`** command cancels a job on the spool. The user who submitted the job must be the same one who cancels it, using the same user ID.

You must use this command on the computer for which the RJE workstation is configured, and the SNAplus2 software must be started on that computer. However, the RJE workstation need not be running. See your system administrator if necessary or refer to the *HP-UX SNAplus2 Administration Guide* for information on RJE workstation configuration or for information on starting the SNAplus2 software.

Syntax

```
snaprjecan [-f] [-r] WKST spool id [spool id . . ]
```

```
snaprjecan [-f] [-r] WKST ALL
```

Options

-f

The **`-f`** option indicates “force”. Normally, you can cancel a job only if it is waiting for transmission to the host. The snaprjecan command will not cancel a job that is in the process of being sent or that has been abandoned after a failed attempt to send it. The **`-f`** option forces the job to be cancelled whatever its status.

If you cancel a job with the option after SNAplus2 RJE has started sending it to the host, sending will continue. However, if this send attempt fails (for example, if a connection failure occurs), no retries will be made. (Normally, SNAplus2 RJE would attempt to retry up to the maximum retry count if such an error occurred.)

-r

The **`-r`** option indicates “root”. Normally, you can cancel a job only if you submitted it; that is, you must have submitted the job with the same user ID that you used to log on. The snaprjecan command will not cancel a job that was submitted with a different user ID. The **`-r`** option, which is valid only if you are logged on as root, forces the job to be cancelled regardless of the user ID with which it was submitted.

Parameters

WKST ***WKST*** indicates the name of the RJE workstation to which the job was submitted. This must be the name of an RJE workstation in the SNAplus2 configuration file, whether or not the named workstation is currently running.

spool-id *spool-id* indicates the spool ID of the job to be cancelled. You can cancel more than one job by specifying more than one spool ID.

The spool ID for a job is returned by the snaprjesend command when the job is submitted. You can also use the snaprjelst command to list the spooled jobs for a workstation with their spool IDs.

To cancel all jobs submitted with your user ID (or all jobs regardless of user ID if you are using the **`-r`** option), replace the list of spool IDs with the word **ALL** or **all**.

Example

To cancel a job on the spool for workstation RJE1, type the following command:

```
snaprjecan RJE1 SPL_fjghld
```

When you use the command as in the example above, the program responds with the following message:

```
Job SPL_fjghld cancelled
```

Other messages may be sent to standard error, indicating (for instance) that a parameter in the workstation style file was invalid, or that an invalid option or an invalid workstation name was supplied. In particular, the **`-r`** option is treated as an invalid option if you are not logged on as root.

snaprjepur

The **`snaprjepur`** command purges status information for a job that has been sent to the host. The job must be purged by the user who submitted it (using the same user ID), except that a user logged on as **root** can purge any job.

SNAplus2 stores information on sent jobs only if the **`list_sent_jobs`** option is set in the workstation style file (see Chapter 6 of this manual for more information). If you use this option, you should use the **`snaprjepur`** command regularly to ensure that SNAplus2 does not retain too much information (which may affect disk occupancy or the performance of RJE programs).

You must use this command on the computer for which the RJE workstation is configured (unless your system administrator has set up the configuration to allow spool commands to be issued from other computers), and the SNAplus2 software must be started on that computer. However, the RJE workstation need not be running. See your system administrator if necessary, or refer to the *HP-UX SNAplus2 Administration Guide* for information on RJE workstation configuration or on starting the SNAplus2 software.

Syntax

```
snaprjepur [ -r ] WKST spool-id [spool-id.. ]
```

```
snaprjepur [ -r ] WKST ALL
```

Options

-r Indicates **Root**. Normally you can purge information for a job only if you submitted it (that is, if you are logged on using the same user ID with which the job was submitted); the **`snaprjepur`** command will not purge information for a job that was submitted using a different user ID. The **`-r`** option, which is valid only if you are logged on as root, forces the information to be purged regardless of the user ID with which the job was submitted.

Parameters

WKST The name of the RJE workstation to which the job was submitted. This must be the name of an RJE Workstation in the SNAplus2 configuration file, whether or not the named workstation is currently running. The workstation name is not case sensitive.

spool-id The spool ID of the job for which information is to be purged. You can purge information for more than one job by specifying more than one spool ID.

To check the spool ID for a sent job, use the **`snaprjelst`** command (with the **`-s`** option) to list the sent jobs for the workstation with their spool IDs. The spool ID is similar to the spool ID returned to the **`snaprjesend`** command when the job was submitted, except that the initial characters **SPL** are replaced by **STS** when the job is successfully sent.

To purge information on all jobs submitted using your user ID (or all jobs regardless of user ID if you are using the **`-r`** option), replace the list of spool IDs by the word **ALL** or **all**.

Example

To purge information on a job sent to the host, type the following command:

```
snaprjepur RJE1 STS_fjghld
```

When you use the command as in the above example, the program responds with the following message:

```
Job STS_fjghld cleared
```

Other messages may be sent to standard error, indicating (for instance) that a parameter in the workstation style file was invalid, or that an invalid option or an invalid workstation name was supplied. In particular, the **`-r`** option is treated as an invalid option if you are not logged on as root.

snaprjestat

The **`snaprjestat`** command is used to check the status of RJE workstation devices and LUs. Before you can view the status, you must have run the workstation at least once so that workstation directories are present. Additional information about jobs submitted and data files received can be obtained by looking at the audit log file. (See the section “Getting Information About RJE Jobs”, earlier in this manual).

Syntax

```
snaprjestat [-d] [-l] WKST
```

Options

By default, **`snaprjestat`** displays both workstation device status and LU status. The user can specify a list of workstation names, or this list can be read from STDIN.

-d When the **`-d`** option is used, **`snaprjestat`** reports on workstation device status only.

-l When the **`-l`** option is used, **`snaprjestat`** reports on workstation LU status only.

Parameters

WKST The name of the workstation whose status information will be displayed. This workstation must be an RJE workstation in the current configuration file.

You may specify more than one RJE workstation here, provided that you leave a space after each. Alternatively, you may enter the **`snaprjestat`** command and press Return. Then type the name of an RJE workstation followed by Return to view its status. You may continue to view the status of further workstations by typing their names followed by Return. When you have finished, press CTRL-D to exit the command. You may also redirect STDIN to a file containing workstation names.

As an RJE user, you may check on the status of workstation devices and LUs. For example, you may wish to see if workstation LUs are active and connected to a host. The **`snaprjestat`** command displays the following information for workstation devices:

WKST The name of the RJE workstation.

device The name of the device owned by the workstation. This name can be a printer (PRT*n*), punch (PUN*n*), console (CONS), or reader (RDR1) device (*n* is a one digit decimal number.)

status Each workstation device will have a status of one of the following:

Active Currently receiving or transmitting data.

Inactive Currently no activity for this device.

Suspended The host has suspended this device.

LUname The name of the LU being used by this device. If a device is inactive, the LU name is displayed as *.

action If a device is inactive, action is displayed as *. For Active or Suspended devices, action may be one of the following:

Printers and Punches:

Output to *type name* *type* indicates whether the output is being sent to a program, a directory, or a file. *name* is the name of the program, directory or file to which the output is being sent.

Reader:

JOB: *spool-id* The device is currently sending a job with the spool ID of *spool-id*.

Console:

Outbound The console is currently receiving data from the host.

Inbound The console is currently sending a command to the host.

For each workstation LU, `snaprjestat` displays the following information:

WKST The workstation name.

LUname The name assigned to the LU when it was configured.

status For an LU, status can be one of the following:

Disconnected The node or communications link to the host is not active.

Pending The node and host communications link are active, but a session has not been established with the host.

Receiving The LU is being used by a device to receive a file or console message from the host.

Sending The LU is being used by a device to send a file or console command to the host.

Suspended The LU is currently suspended by the host while receiving a file.

Idle A session exists for the LU, but there is currently no activity.

device The device that is currently using this LU. For disconnected, pending or idle LUs, device is indicated as *.

The **`snaprjestat`** command will return the following exit codes:

-1 An error has occurred with the command.

0 The workstation is active and connected.

1 The workstation is active, but no LU is currently active (all LUs are either disconnected or pending).

2 The workstation is inactive

The **`snaprjestat`** command always returns a value of 0 if more than one workstation is specified and there is no command error.

Example

`snaprjestat RJE1`

If you enter the command above, your screen displays the status of configured devices for workstation RJE1 and their current actions. It also shows the status of configured LUs for RJE1. The following figure is an example of the screen display. The RJE1 workstation has five printers, one punch device and three LUs assigned to it, and the printer and punch devices are designated by **`PRT`** and **`PUN`** plus the printer or punch number.

```
WKST  Device   Status      LU Name   Action
--------------------------------------------------------------
RJE1  CONS     Active      FOURTH    Outbound
RJE1  PUN1     Suspended   FOURTH    Output to file /var/opt/sna/rje/RJE1 \
 /output/punch.out
RJE1  RDR1     Active      NUMBER10  Job: SPL_HKUWSX
RJE1  PRT5     Inactive    *         *
RJE1  PRT4     Inactive    *         *
RJE1  PRT2     Inactive    *         *
RJE1  PRT1     Active      LU3       Output to program /usr/mb/misc/mikeprog

WKST  LU Name  Status      Device
--------------------------------------------------------------
RJE1  FOURTH   Receiving   CONS
RJE1  LU3      Receiving   PRT1
RJE1  NUMBER10 Sending     RDR1
```

If the command detects an error, it writes a message to standard error. This message may indicate (for instance) that an invalid option or an invalid workstation name was supplied, that a parameter in the workstation style file was invalid, that the SNAplus2 software was not started, or that the workstation was already running.

snaprjecmd

The **`snaprjecmd`** command sends host commands through the RJE console program (Chapter 5) to a workstation from the command line.

You must specify a workstation name with this command and optionally specify a host command string. If no host command string is specified, then input is taken from standard input. STDIN may be redirected to a file containing host commands.

NOTE: The purpose of this command is primarily for issuing host commands programmatically. For most purposes, it is easier to use the snaprjecon program to access a workstation's console. (See chapter 5).

Syntax

`snaprjecmd` ***`WKST hostcommand`***

Parameters

WKST — ***WKST*** is the name of the RJE workstation to which the command is being sent. This must be the name of a running RJE workstation, and your user ID must be in the group configured for this workstation. See your system administrator if necessary.

hostcommand — The command you are sending to the host Job Entry Subsystem (JES). The syntax of these commands is defined by the host. (See your host JES documentation for more information.)

If you do not enter a command on the command line, SNAplus2 RJE will read commands in from standard input. You can redirect STDIN to a file of commands. Note that when you enter a host command on the command line, any characters such as $ must be preceded by `\` to prevent their interpretation by the HP-UX shell. This is not necessary for commands entered from standard input. For the commands from standard input, each line should contain one host command.

The output from the command may be an error message:

```
RJE WORKSTATION WKST IS NOT RUNNING
```

Or the output could be confirmation that commands are sent to the host:

```
COMMAND SENT TO THE HOST: hostcommand
```

Error messages for the command will be displayed in STDERR. Host information will be sent to the workstation console log file. You can use the console program to view this information.

Note that if an interactive console is running for this workstation, **snaprjecmd** host commands and responses to those commands will be displayed on the console.

Other messages may be sent to standard error, indicating (for instance) that a parameter in the workstation style file was invalid, or that an invalid workstation name was supplied.

snaprjeclr

The **`snaprjeclr`** command removes the files and directories associated with an RJE workstation. The files and directories will be rebuilt if a new job is submitted for the workstation, or if the workstation is restarted.

The command must be issued by the system administrator. Note that this command will remove all of the following:

- Jobs spooled for this workstation
- Status information stored for sent jobs
- Output files on the default output directory
- Console information

The command should only be used when the workstation is no longer required, or to clean up after a serious error condition. Before using the command, you must ensure that the workstation and the console program are not running.

If the workstation's main directory contains an RJE style file, this file is copied into the main RJE directory **`/var/opt/sna/rje`** before the workstation's files and directories are deleted. If you are removing the workstation from the configuration, you should delete this file. If you are only clearing the directories and intend to continue using the workstation, you should recreate the directory **`/var/opt/sna/rje/`*`WKST`***, where *WKST* is the name of the workstation, and move the style file into this directory.

You must use the command on the computer for which the RJE workstation is configured, and the SNAplus2 software must be started on that computer. Refer to the *HP-UX SNAplus2 Administration Guide* for information on RJE workstation configuration or on starting the SNAplus2 software.

Syntax

```
snaprjeclr WKST
```

Parameters

WKST *WKST* is the name of the RJE workstation for which you want to clear the directories. It is not case sensitive.

Example

```
snaprjeclr RJE1
```

If the command completes successfully, no messages are returned.

A message is sent to standard error if SNAplus2 RJE detects an error. This may indicate (for instance) that an invalid workstation name was supplied, that a parameter in the workstation style file was invalid, or that the workstation was running and so the directories could not be cleared.

5

SNAplus2 RJE Console Program

This chapter explains how to use the SNAplus2 RJE console program to view messages produced by the host and to issue commands to the host.

RJE Console Program

The console program provides access to the host's Job Entry Subsystem console. You can use it to view messages produced by the host while processing jobs, to issue host commands, and to view the host's responses to console commands. You can either view this information as it becomes available, or browse through previous commands and messages. The commands you can issue are determined by the operator privilege for your workstation, as specified in the host configuration.

To run the RJE console program, use the **`snaprjecon`** command. You may run the console program at any time, provided the SNAplus2 software is started. However, if the RJE workstation is not running, or is running but not connected to the host, you can use it only for viewing previous information. You will not be able to issue host commands.

You can issue host commands programmatically by using the **`snaprjecmd`** command. For more information, see the description of this command in chapter 4.

Starting the Console Program

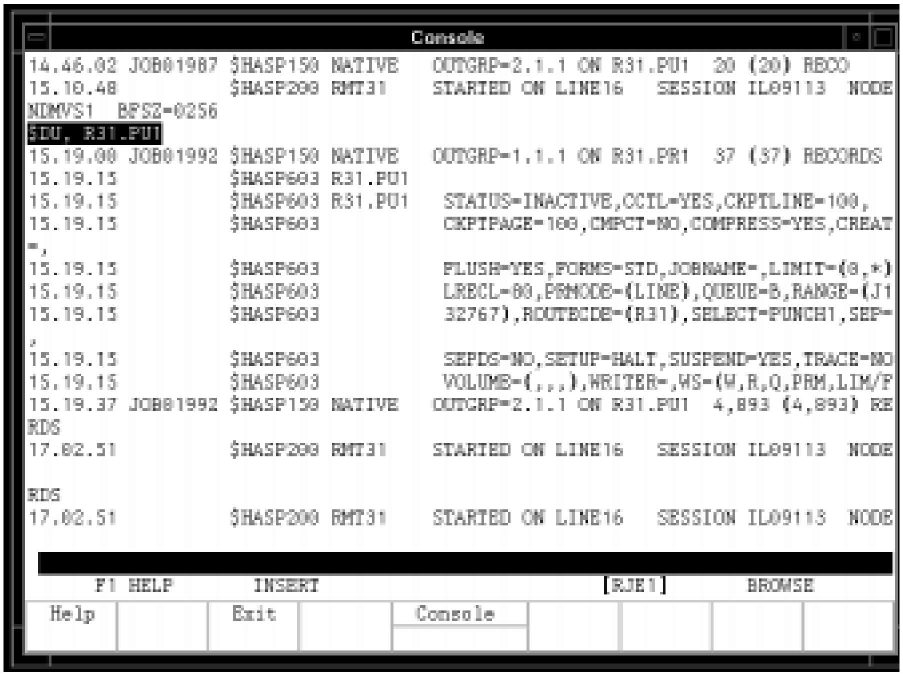
You must start the console program on the computer for which the RJE workstation is configured, and the SNAplus2 software must be started on that computer. Your user ID must be in the group configured for this workstation. See your system administrator if necessary, or refer to the *HP-UX SNAplus2 Administration Guide* for information on RJE workstation configuration or for information on starting the SNAplus2 software.

Use the following command to start the RJE console program.

```
snaprjecon WKST
```

WKST is the name of the RJE workstation for which you want to view console information.

When you use the command as in the above example, the following console screen displays:



```
14.46.02 JOB01987 $HASP150 NATIVE   OUTGRP=2.1.1 ON R31.PU1   20 (20) RECO
15.10.48          $HASP200 RMT31    STARTED ON LINE16   SESSION IL09113   NODE
NDMVS1  BFSZ=0256
$DU, R31.PU1
15.19.00 JOB01992 $HASP150 NATIVE   OUTGRP=1.1.1 ON R31.PR1   37 (37) RECORDS
15.19.15          $HASP603 R31.PU1
15.19.15          $HASP603 R31.PU1   STATUS=INACTIVE,CCTL=YES,CKPTLINE=100,
15.19.15          $HASP603           CKPTPAGE=100,CMPCT=NO,COMPRESS=YES,CREAT
=,
15.19.15          $HASP603           FLUSH=YES,FORMS=STD,JOBNAME=,LIMIT=(0,*)
15.19.15          $HASP603           LRECL=80,PRMODE=(LINE),QUEUE=B,RANGE=(J1
15.19.15          $HASP603           32767),ROUTECDE=(R31),SELECT=PUNCH1,SEP=
,
15.19.15          $HASP603           SEPDS=NO,SETUP=HALT,SUSPEND=YES,TRACE=NO
15.19.15          $HASP603           VOLUME=(,,,),WRITER=,WS=(W,R,Q,PRM,LIM/F
15.19.37 JOB01992 $HASP150 NATIVE   OUTGRP=2.1.1 ON R31.PU1  4,893 (4,893) RE
RDS
17.02.51          $HASP200 RMT31    STARTED ON LINE16   SESSION IL09113   NODE

RDS
17.02.51          $HASP200 RMT31    STARTED ON LINE16   SESSION IL09113   NODE


     F1 HELP       INSERT                          [RJE1]        BROWSE
```

Figure 2 **SNAplus2 RJE Console Screen**

The bottom line of the screen is the status line. It includes the following:

- reminders of the keystrokes for on-line help
- whether command editing is in INSERT or REPLACE mode
- the workstation name and its current status
- whether the console display is in BROWSE or FOLLOW mode.

The meaning of each of the status line messages is explained in later sections of this chapter.

The line above the status line, which appears in reverse video, is the command input area where you can type host commands. The remainder of the screen displays the console messages sent by the host, and also the text of any commands you have entered. Messages and commands display in the order in which the console program receives them. Commands you have entered display in reverse video.

Understanding Status Line Information

The following indicators appear on the status line of the console program.

HELP [f1] The f1 key displays help information on how to use the console program.

INSERT Command editing is in insert mode. This means that any text you type into the command input area is inserted at the current cursor position.

REPLACE Command editing is in replace mode. This means that any text you type into the command input area overwrites the character at the current cursor position.

To switch between insert and replace mode, use the Insert char key.

WKST The name of the RJE workstation for which console information is being displayed. The status of the workstation is indicated as follows:

[WKST] If the workstation name is shown in brackets, this indicates that the workstation is inactive.

WKST- If the workstation name is followed by a **-** sign, the workstation is active, but the connection to the host is not established. This may be because the local node or link service has not been started, because a link error has occurred, because the host has rejected a logon attempt from the workstation, or because the LUs are not active on the host. Check with your system administrator if necessary.

WKST+ If the workstation name is followed by a **+** sign, this indicates that the workstation is active and that the connection to the host has been established. Host commands may be entered.

FOLLOW The console program is in follow mode. This means that you are viewing new console messages as they are received.

BROWSE The console program is in browse mode. This means that you are viewing earlier console messages.

Using the Console Program

The RJE console program allows you to do the following:

- Enter host JES commands, and view the host's responses.
- View earlier console information (commands that you have entered and output from the host).
- Perform a temporary exit to the HP-UX command prompt, leaving the console program running.

These options are discussed in the following sections.

Entering Host Commands

To enter a host command, simply type the text of the command and press Return. It appears on the reverse-video command input area at the bottom of the console screen. After you press Return to send the command, it is echoed on the console screen and removed from the command input area.The syntax of these commands is defined by the host. See your host JES documentation for more information on the commands you can use.

If you make a mistake while typing the command, use the following keys to correct it.

- Backspace
- Left arrow to move to the left
- Right arrow to move to the right
- CTRL-T to move to the end of the line.
- CTRL-W to move to the start of the line
- ESC to clear the command input area without sending the command so that you can start again
- Insert char to toggle between insert mode and replace mode. In insert mode, a character you type is inserted at the current cursor position. In replace mode, a character you type overwrites the character at the current cursor position. The status line displays the indicator **`INSERT`** or **`REPLACE`** to remind you which mode you are in.

The console program retains the text of the ten previous commands. This means that you can repeat a previous command without having to type it in again. The keystrokes CTRL-P (previous command) and CTRL-O (next command) can be used to browse through the available commands. As you move forwards or backwards through them, each command in turn displays on the command input area. Press Return to issue the displayed command, or use the editing keys described above to change the text of the command before you issue it.

Viewing Console Information

When you first start the console program, the indicator **FOLLOW** appears at the right-hand end of the status line. This indicates that the console is in follow mode and that the most recent console information is displaying. As more messages are received from the host, or as you type further commands, the screen display scrolls so that the most recent information always displays.

You can use the console program to view earlier console activity, by using the following keys:

- Up arrow to scroll the display back one line.
- Down arrow to scroll the display forward one line
- Prev or CTRL-B to scroll the display back one screen
- Next or CTRL-F to scroll the display forward one screen
- Home to return to the earliest available information
- Shift-Home or CTRL-E to return to the latest available information

While you are browsing through previous console information, the status line displays the indicator **BROWSE** instead of **FOLLOW**. The display is now in browse mode instead of follow mode. When you move forward to the end of the available information (either by using Shift-Home or by scrolling forward until you reach the end), the console automatically returns to follow mode, and the indicator changes from **BROWSE** back to **FOLLOW**.

While in browse mode, host messages continue to be received and stored. However, you are limited to browsing back through 20,000 bytes of previous output. Also, if you stay in browse mode while more than 20,000 bytes of information is received from the host, some of the data will be lost. When you move forward past the lost data, the message **`DATA LOST`** displays to indicate this.

You can specify the maximum amount of console data that is saved by specifying the **`console_log_size`** option in the **`define_minor_options`** record in the RJE style file.

The SNAplus2 RJE console uses two files that have stored console output from the host. Output is toggled to these two files. When the length of one file exceeds the maximum length, the other file initializes. Output is then written to that file. By default, the maximum size of the files is approximately 10,000 bytes each. (This limits the total amount of data that can be viewed by the RJE console to 20,000 bytes).

This maximum length can be changed by setting the **`console_log_size`** option to a value of up to 1000000000. This sets each console file to one billion bytes. Therefore, the minimum allowable value is 10000 (ten thousand bytes), and the maximum allowable value is 1000000000 (one billion bytes).

If the value you specify is not in this range, an error message will be logged, and the console file length will default to 10,000 bytes.

You can enter host commands whether the console is in browse mode or in follow mode. However, commands and the host responses to them are always added to the end of the available console information. This means that in browse mode, they will not be visible on the screen until you move toward the end of the file. If you need to see the screen display of a command and the host's response as soon as you issue it, return to follow mode by moving to the end of the available console information.

While you are using the console program, your screen may become corrupted if other processes write to it, or if characters are lost due to line transmission problems. To redraw the screen, you can use the keystroke CTRL-R. This does not affect any command data you have entered.

Exiting Temporarily to the HP-UX Command Prompt

To move temporarily from the console program to an HP-UX command prompt, leaving the console program running, use the ***Tempexit*** keystroke, CTRL-Z. The prompt **RJECON** (RJE console) displays, reminding you that you are in a temporary exit shell from the console program.

You can then issue any other commands as required. For example, if you start the console program and find that the RJE workstation is not running, you can start the workstation from a temporary exit shell. You can also issue other HP-UX commands not related to SNAplus2 RJE

To return to the console program from a temporary exit, use the HP-UX end-of-file (EOF) key sequence for your terminal (typically CTRL-D).

Ending the Console Program

To end the console program, use the keystroke CTRL-X. Alternatively, you can use the f3 key if it is available on your terminal's keyboard. Either of these keystrokes returns you to the HPUX command prompt.

Note that console information is still collected even after you end the console program, as long as the RJE workstation is running. This information can be viewed later if you restart the console program.

6

SNAplus2 RJE Workstation Configuration

SNAplus2 RJE Workstation Configuration

This chapter gives configuration information for the RJE Workstation and how it communicates with the host and the output devices available to it. It also explains how to use the customization program and how to edit the minor options in the style file.

To configure the RJE workstation, you will use the following three tools:

- **`snaprjecust`** customization program
- a text editor for the RJE style file
- **`xsnapadmin`** administration program

Overview of the Workstation Style File

The workstation style file contains information on how the workstation interacts with the host Job Entry Subsystem, how it interacts with the local HP-UX computer to process input and output data, and the printer and punch devices available for processing output data. It may be maintained by the system administrator or by the users of the workstation.

The RJE workstation style file is an ASCII text file with information stored in readable text format. Because the file is a text file, you can check or modify the workstation definition using a standard text editor. SNAplus2 also provides a menu-driven customization program, **`snaprjecust`**, that allows you to set most of the parameters in the style file. You are recommended to use this program if you need to modify the file. However, you will need to use a text editor in order to set the less common style file parameters in the **`define_minor_options`** record.

The style file is named *WKST*.**sty**, where *WKST* is the name of the workstation, and is stored in the directory **/etc/opt/sna**.

RJE Style File Customization Program

The customization program provides a character-based menu interface that allows you to view, modify, and save the contents of a workstation style file. It provides defaults for the style file parameters wherever possible, and also performs cross-checking to ensure that the parameters you specify are valid. Help screens provide context-sensitive information on how to set the parameters in each dialog, and on the menu options provided by the program.

The program can be used to define the style file for a configured RJE workstation, whether or not the workstation is running. You can use it to set up the style file for the first time, or to modify an existing style file. You can also use the program offline to set up a new style file (or modify an existing one) that is *not* associated with a configured workstation. This allows you to set up multiple style files with different customization (and copy the appropriate one to the correct filename before you start the workstation), or to set up style files for use with workstations on different computers.

The following sections explain how to start the program and provide an overview of how to use the program to customize and save the style file. For more detailed information on using a particular menu item or dialog box in the program, move to the menu item or dialog box, and press F1 to access the help screen for that item or dialog box.

If you need more information on using the menu interface, press F1 to access a help screen, and then tab to the **`< UI Info >`** button on the help screen and press Enter. This displays help information on using the menu interface.

Starting the Program - Style File for a Configured Workstation

To customize the style file for a configured workstation, start the program with the following command:

```
snaprjecust WKST
```

Replace *WKST* with the name of the RJE workstation.

If a style file already exists for this workstation, the program opens the file and displays the current customization. Otherwise, it displays a message asking whether you want to create the file. Choose **`< OK >`**to create the file, or **`< Cancel >`** to exit the program without creating it. If you create a new file, all parameters are initially set to default values, which you can then customize as required.

Starting the Program - Setting Up a Style File Offline

To create or modify a style file that is not associated with a configured workstation, start the program with the following command:

```
snaprjecust -s style_file_name
```

Replace **`style_file_name`** with the name of the style file that you want to create or modify. The filename must consist of 1-4 characters followed by the extension .sty, but you do not need to include the extension on the command line (SNAplus2 will add it automatically). If you do not specify a path for the file, SNAplus2 assumes that it is in the current directory.

If the specified style file already exists, the program displays the current customization. Otherwise, it displays a message asking whether you want to create the file. Choose **`< OK >`** to create the file or **`< Cancel >`** to exit the program without creating it. If you create a new file, all parameters are initially set to default values, which you can then customize as required.

Customizing the Workstation

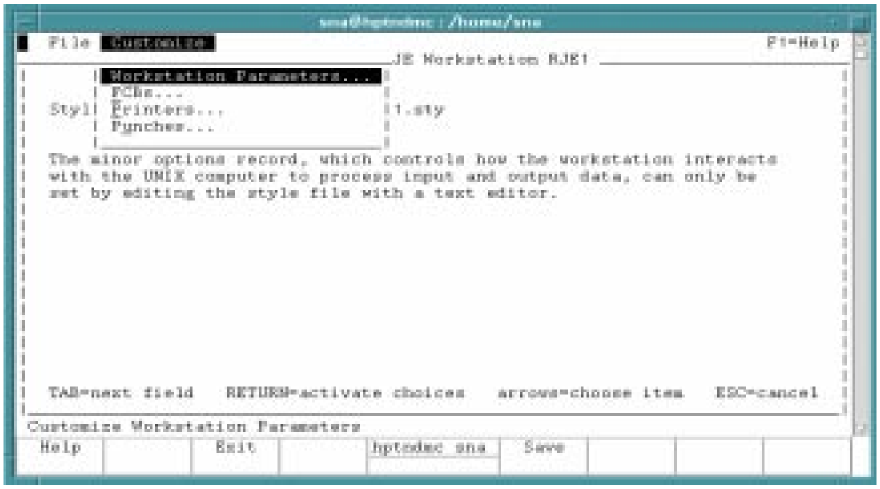
To customize the workstation, select the **`Customize`** menu. Then choose one of the following menu options:

Workstation Parameters Choose this option to customize how the workstation interacts with the host (for example, to set up a logon string to be sent to the host when establishing the communications session).

FCBs Choose this option to customize Forms Control Buffers (FCBs); an FCB specifies the page layout used by an RJE printer device.

Printers Choose this option to customize printer devices that the workstation can use to process output returned from the host.

Punches Choose this option to customize punch devices that the workstation can use to process output returned from the host.



```
sna@hptndmc : /home/sna
  File  Customize                                                F1=Help
                                     _JE Workstation RJE1 ________________
|        | Workstation Parameters... |
|        |  FCBs...                  |
|   Styl |  Printers...              |1.sty
|        |  Punches...               |
|        |___________________________|
|  The minor options record, which controls how the workstation interacts
|  with the UNIX computer to process input and output data, can only be
|  set by editing the style file with a text editor.

|  TAB=next field   RETURN=activate choices   arrows=choose item   ESC=cancel
 Customize Workstation Parameters
  Help          Exit          hptodmc sna   Save
```

Figure 3 **Customization Menu**

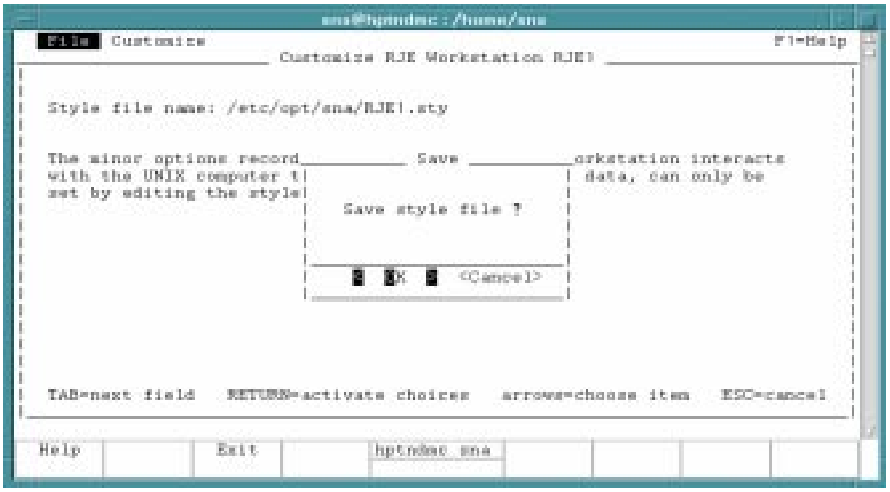
For help on each customization screen, press the F1 key.

Saving the Style File

After making changes to the workstation's customization, you need to save these changes to the style file. To do this, select the **`File`** menu, and then choose Save.

The program prompts you to confirm saving the changes. Choose **`< OK >`** to save the changes, or **`< Cancel >`** if you do not want to save them. If you choose **`< Cancel >`**, any changes you have made are still displayed in the program (so that you can make further changes if necessary before saving the file), but are not written to the file.

If you save changes to the style file while the workstation is running, the program displays a message to remind you to stop and restart the workstation to ensure that it picks up the new customization. Choose **`< OK >`** to acknowledge this message.



```
                          sna@hptndmc : /home/sna
  File  Customize                                              F1=Help
  ____________________ Customize RJE Workstation RJE1 ____________________

   Style file name: /etc/opt/sna/RJE1.sty

   The minor options record__________ Save __________orkstation interacts
   with the UNIX computer t|                        | data, can only be
   set by editing the style|                        |
                           |    Save style file ?   |
                           |                        |
                           |________________________|
                           |   < OK >   <Cancel>    |
                           |________________________|

   TAB=next field   RETURN=activate choices   arrows=choose item   ESC=cancel
  ________________________________________________________________________
  Help          Exit          hptndmc sna
```

Figure 4 **Save Style File Screen**

Exiting the Program

To exit from the program when you have finished customizing the workstation, select the `File` menu and choose `Exit`.

If you have not made any changes to the customization, or have saved any changes you have made, the program ends.

If you have made changes and have not saved them, the program prompts you to save the changes before exiting. You have the following options:

- Choose `< Yes >` to save the changes and then exit the program.
- Choose `< No >` to exit the program without saving the changes (so that the workstation's customization reverts to the last saved version).
- Choose `< Cancel >` to abandon the exit. The program continues to run, and your changes are still displayed in the program. You can make further changes if necessary, and then save them before you exit the program.

Messages displayed by the program

When you start the program, or when you choose `< OK >` to accept the information in a dialog box, SNAplus2 may display a message relating to the action you have taken. Three different message types may be displayed, as follows:

Error Error messages indicate a condition that requires action. This may be an error detected by SNAplus2; for example, a failure to open or write to the specified style file. It may also be an error that you made in a dialog box field, such as typing invalid data in an edit box and choosing `< OK >`. You must return to the edit box and correct the entry, or choose `< Cancel >` and abandon the dialog box.

Warning Warning messages indicate a condition that may require action. For example, if you do not specify an output path for a printer device, a warning message informs you that the default path will be used. You can either continue with the default path, or return to the dialog box and type in a suitable path.

Information Information messages provide confirmation that an action has been accepted (such as saving the style file), or provide information related to the previous action that may be useful to you. No action is required.

The title of a message box indicates whether it is an Error, Warning, or Information message. For all types of message boxes, choose the **`< Exit message >`** push button to acknowledge the message and continue with the program.

For Error and Warning message boxes, there is also a **`< More Info >`** push button. If you choose this push button, the message box is replaced by a larger box providing more information. This box includes the message text that was shown in the original message box, and also more detailed information on the condition that caused the message and the action you should take. Again, choose the **`< Exit Message >`** push button to acknowledge the message and continue with the program.

Another information message appears on the main screen of the program. This message informs you that in order to define the minor options, which control how the workstation interacts with the HP-UX computer to process input and output data, you must edit the style file with a text editor. This message remains on the screen and does not require you to take any action.

For detailed information on defining the minor options, see the following section, “Text File Format”.

Text File Format

The RJE workstation style file is an ASCII text file, with information stored in readable text format. This allows you to check or modify the workstation definition using a standard text editor.

The file consists of a series of records, each of which defines a set of options for the workstation or for the resources associated with it. The records that can be included are discussed in more detail in **RJE style file records**, later in this chapter.

Each record is defined in the following format:

```
[record_name]
parameter_name     =     value
parameter_name     =     value
             .
             .
             .
parameter_name     =     value
```

The record_name must be enclosed in square brackets. It is followed by a series of parameter entries, each on a separate line. A back-slash character (\) at the end of a line indicates that the entry continues on the next line.

All the parameters associated with a particular record must be listed after the record_name for that record, and before the record_name for the next record in the file. However, the order of individual parameters within a record is not important (except where this is indicated in the command descriptions later in this chapter). Also, note that SNAplus2 provides defaults for many parameters, so you need not specify every parameter explicitly

Parameter Types

Each parameter in a record is of a defined type — character string, decimal number, or defined constant. For example, in the **define_minor_options** record, **temp_directory** is a character string, **console_log_size** is a decimal number, and **punch_padding** is a defined constant. In the record descriptions later in this chapter, the information given for each individual parameter includes its type.

The following paragraphs define how to specify a parameter of each type, and provide information on valid entries for each type.

Character string

A character string, for example RJE1. The individual parameter description gives details of the maximum and minimum length of each string. When the string must consist of characters from a particular character set (such as alphanumeric, A-string, or HP-UX filename characters), this is noted in the parameter description. Otherwise, you can use any character that displays in your computer's local character set. Character strings are case-sensitive

Decimal Number

A numeric value, for example 128. The individual parameter description gives details of the maximum and minimum values. Specify numeric values in decimals, not in hexadecimals.

Defined Constant

One of a set of defined values, specified as an ASCII character string without quotation marks, for example **`program`**. In some cases, a defined constant parameter may be set to a combination of two or more of the defined values, as indicated in the parameter description. Where this is not indicated, only one value may be used.

Defined constants are used for parameters where there is a fixed set of valid values, such as **`File/Dir/Program.`** The individual parameter description lists the defined values for each parameter. Defined constants are not case-sensitive. You can use either uppercase or lowercase.

Default Parameter Values

Some parameters in style file records, such as the name of the output device you are defining, must always be specified explicitly. For other parameters, SNAplus2 RJE provides default values, so that for a standard setup you do not need to specify every parameter in a record. (However, you may prefer to specify parameters explicitly, to make it easier to check the values that are being used.)

The individual parameter descriptions later in this chapter include information on default values where applicable. If no default value is shown for a parameter, you must specify it explicitly.

Blank Space

Embedded space characters are valid within the **`logon_text`** parameter in the **`define_workstation_options`** record, but not within any other parameter value. For example, the character string **FORM001** is a valid value for a forms control buffer name, but **FORM 01** is not.

All blank spaces before or after descriptors, parameter names, or parameter values (that is, any combination of spaces and tabs) are ignored. Completely blank lines are also ignored. You may prefer to add blank lines between records to make the file more readable, but this is not required.

RJE Style File Records

The style file can contain the following records:

define_workstation_options Specifies how the workstation interacts with the host.

define_minor_options Specifies how the workstation interacts with the HP-UX computer to process input and output data.

define_forms_control Defines a Forms Control Butter, which specifies a page layout used by an RJE printer device.

define_ws_device Defines an RJE output device (printer or punch).

The following sections describe these records.

define_workstation_options

The **`define_workstation_options`** record controls the workstation's interaction with the host Job Entry Subsystem.

This record must always be included in the style file.

Parameter Name	**Type**	**Length**	**Default**
`[define_workstation_options]`			
`description`	`character`	`31`	`(null string)`
`logon-text`	`character`	`256`	`(null string)`
`send_logon`	`constant`		
`inbound_compression`	`constant`		

description An optional text string describing the workstation. This string is for information only. It is stored in the style file and displayed in the RJE style file customization program.

logon_text

The logon string required by the host Job Entry Subsystem. This string must be specified if the `send-logon` parameter below is set to YES. Check with your host Job Entry Subsystem personnel to determine the correct string to use. If `send_logon` is set to NO, this parameter must not be specified.

send_logon

Specifies whether the RJE workstation sends its logon string to the host to initiate RJE startup, or waits for the host to initiate the startup.

Allowed values:

YES The workstation sends a logon message to the host as soon as the SSCP session is established. The `logon_text` parameter must be specified.

NO The workstation waits for a BIND message from the host, and does not send the logon string. The `logon_text` parameter must not be specified.

Check with your host Job Entry Subsystem personnel to determine the correct option to use.

inbound_compression

Specifies whether job files submitted to this workstation are to be compressed for transfer to the host. Compression can improve the overall processing time for a job if you are sending large files containing sequences of repeated characters; that is, more than four of the same character (including the space or null characters) in succession.

Check with your host Job Entry Subsystem personnel whether compression is supported. If compression is supported, you can choose whether to use it. Do not specify compression if the host configuration does not support it. Note that SNAplus2 RJE always accepts compressed files sent from the host.

define_minor_options

The **define_minor_options** record controls how the workstation interacts with the HP-UX computer to process input and output data.

Parameter Name	**Type**	**Length**	**Default**
[define_minor_options]			
description	character	31	(null string)
job_logs	constant		NO
list_sent_jobs	constant		NO
temp_directory	character	256	/var/opt/sna/rje/*WKST*/.pgmout
punch_padding	constant		PAD_NONE
ff_passthrough	constant		NO
printer_nl	decimal		132
console_log_size	decimal		10000
ebcdic_newline	constant		NO
logon_retry	constant		NO
mail_on_events	constant		YES
keep_failed_files	constant		NO
run_in_foreground	constant		NO
punch_nl_IRS	constant		YES
ignore_rec_length	constant		NO
display_sscp_messages	constant		NO

description An optional text string describing the minor options. This string is for information only. It is stored in the RJE style file.

job_logs Specifies whether the workstation should log messages about the progress of jobs it is processing. Note that these messages are not written to the standard SNAplus2 audit or error log file, but to a separate file for each workstation. This file is named **`job.log`**, and is stored in the main workstation directory **`/var/opt/sna/rje/WKST`**. For more details of the format of this file, see Chapter 3 of this manual.

Allowed values:

YES	Log messages about job progress.
NO	Do not log messages.

If you specify YES, SNAplus2 writes to this file each time a job is spooled, cancelled, or sent to the host, and each time output is received from the host. In addition, if output is sent to the **`rjeusr`** or **`rjeusrpad`** programs, it will add entries about output routing. You should check this file at intervals, and either delete the file or delete older entries from it, to ensure that the file does not grow too large and take up too much disk space. If you need to delete individual entries from the file rather than deleting the whole file, this must be done when the SNA software is not running.

list_sent_jobs Specifies whether the workstation should retain information about jobs after it has sent them to the host, so that you can use the **`snaprjelst`** command to list them.

Allowed values:

YES	Retain information about sent jobs.
NO	Do not retain information about sent jobs.

If you set this option to YES, note that SNAplus2 will retain information about the sent jobs until you purge them using the **`snaprjepur`** command. You should use this command regularly to ensure that the volume of information stored does not grow too large (which will affect disk occupancy and the performance of the RJE programs). See Chapter 4 of this manual for more information on RJE commands.

Note also that SNAplus2 does not purge any existing stored information when you change this option from YES to NO. You should use the **`snaprjepur`** command to do this explicitly.

temp_directory Specifies the directory where the workstation stores temporary files that it uses during job processing. This must be the full path of a directory on the computer where the workstation runs. The primary user and group of the workstation must have read and write access to this directory.

punch_padding Specifies whether to pad punch records out to the full record length during output, and (if so) the pad characters to be used.

Allowed values:

PAD_NONE No padding

PAD_TEXT Pad text records with spaces (either ASCII or EBCDIC, depending on the setting of **`translate_data`** in the **`define_ws_device`** record).

PAD_TRANSP Pad transparent data records with null characters.

PAD_TRANSP_SPACE Pad transparent data records with spaces.

To specify that both text records and transparent data records are to be padded, specify the two constants combined using a + character:

PAD_TEXT + PAD_TRANSP or

PAD_TEXT + PAD_TRANSP_SPACE

ff_passthrough Specifies whether the workstation should pass through Form (FF) characters received from the host in printer output, or convert them to line-feed characters. This option is ignored for punch output.

Allowed values:

YES	Pass through FF characters without conversion.
NO	Convert FF characters to the appropriate number of line-feed characters to reach the end of the page (as defined by the forms control buffer used by the printer device).

Select the option that generates the output format expected by the printer devices or output programs you are using.

printer_nl Specifies the maximum line length in characters for printer output. The workstation will insert a new-line character each time the output line reaches this length. The valid range is 132 - 255. This option is ignored for punch output.

console_log_size Specifies the maximum size in bytes of each of the two files used to hold console log data for this workstation, in the range 10,000 - 1,000,000,000. The first file can grow to the size specified here. Information is then written to the second file. When the second file reaches the same size, the first file is cleared, and the information is redirected to the first file. Using two files means that console logging can be active for an extended period of time, with a maximum disk space usage of twice the value specified here.

The following parameters should all be left with their default settings. They are used only in special cases. For most installations, the default values are correct and should not be changed.

ebcdic_newline

If the workstation’s output devices expect data in EBCDIC format and so require data to be passed through without EBCDIC-ASCII translation, set this option to YES to indicate that any new-line characters inserted by the workstation should be EBCDIC and not ASCII new-line characters. Otherwise, set it to NO to insert ASCII new-line characters.

logon_retry

This option causes RJE to retry sending a logon string after a host terminated session is reestablished. In some situations, a link outage may occur due to a host shutdown. When the link is reestablished, RJE will send a logon string. Normally, if the application (such as JES2) is available, the session will be reestablished. If the application is not available at that time, RJE will not resend the logon string.

To have RJE resend the logon string periodically until it succeeds, set the **`logon_retry`** option to YES. This option applies only if the workstation is defined to send its logon string at startup (by setting the **`send_logon`** parameter). Note that when this option is set to YES, messages will be logged to the audit log file each time the workstation attempts to contact the host. If the **`display_sscp_messages`** option below is set, messages may also be written to the RJE console log file. Frequent logon retries (for example, because of an unreliable communications link), may result in a large amount of data being written to these files. To avoid losing data when the files wrap, be sure to allow the maximum specified file sizes for these files.

mail_on_events

Specifies whether the workstation should send mail messages to the primary user of the workstation about the progress of jobs it is processing.

Allowed values:

YES	Send mail messages about job progress.
NO	Do not send messages.

keep_failed_files

In general, the host's configuration should specify a large checkpoint size, to ensure that it will always resend the complete output data and not try to restart part way through. For example, to ensure this in a typical JES2 workstation configuration, set CKPTLINE and CKPTPAGE to 10000 and CKPTMODE to PAGE.

If a communications link failure occurs while the host is sending output to the workstation, the host will retry sending the output when the communications link is restored. SNAplus2 RJE assumes that the host will resend complete data, and so the workstation normally deletes the temporary file containing incomplete output when it detects failure.

If the host's checkpoint size is set so that it restarts part way through the output data, you have the option of setting **`keep_failed_files`** to YES so that SNAplus2 will not delete incomplete files. In this case, although the output data file contains only the last section of the data (from the point at which the host restarted), the user can recover the earlier data from the temporary files stored for the workstation. The name of the temporary output file is logged to the error and audit logs.

run_in_foreground

The RJE workstation normally runs in the background as a daemon process. To force it to run in the foreground, set **`run_in_foreground to YES.`**

punch_nl_IRS

When sending punch output, the host specifies the line length to be used for the output from each job. SNAplus2 normally inserts a new-line character each time an output line reaches this length. To override the specified line length so that a new-line character is only inserted when an IRS (Inter-Record Separator) character is received in the host data, set **`punch_nl_IRS`** to YES.

ignore_rec_length

When sending job data to the host, SNAplus2 normally inserts an IRS (Inter-Record Separator) character each time the line length reaches the specified record length. To override the specified record length so that an IRS is only inserted when the job data contains a new-line character, set **`ignore_rec_length`** to YES.

display_sscp_messages

When the RJE workstation attempts to establish a session with the host for one of its LUs, the host may send data on the SSCP session for that LU. This data may simply be a start-of-day message when the session is established successfully, or may contain diagnostics information indicating why the session could not be established. If the workstation receives SSCP data, SNAplus2 writes the first 80 bytes of this data to the log file. However, because only 80 bytes can be included in the log message, some of the data may be lost.

To specify that SSCP data should be written to the RJE console as well as to the log file, set **`display_sscp_messages`** to YES. In this case, the complete data sent by the host is written to the console log file, and can be viewed using the console program. To specify that SSCP data should not be written to the console, set this option to NO.

This option is intended for use when you are having problems establishing sessions to the host. Displaying the complete SSCP data on the console allows you to check for error messages indicating why the session could not be established. In normal use, you should set this option to NO to avoid filling up the console log file with large amounts of data (particularly if the host sends a full-screen banner message at start of day).

define_forms_control

The define_forms_control record defines a Forms Control Buffer that determines the page layout used by an RJE printer device. Up to ten Forms Control Buffers may be defined in each style file; each one may be used by one or more printer devices.

Parameter Name	Type	Length	Default
[define_forms_control]			
description	character	31	(null string)
form_name	character	8	
form_length	decimal		66
first_line	decimal		1
last_line	decimal		66
channel_2	decimal		0
channel_3	decimal		0
channel_4	decimal		0
channel_5	decimal		0
channel_6	decimal		0
channel_7	decimal		0
channel_8	decimal		0
channel_9	decimal		0
channel_10	decimal		0
channel_11	decimal		0
channel_12	decimal		0

description An optional text string describing the Forms Control Buffer. This string is for information only. It is stored in the style file and displayed in the RJE style fine customization program.

form_name A name identifying this Forms Control Buffer. This name should not match any other **`form_name`** in the style file. If you include two Forms Control Buffers with the same name, only the second definition will be used.

form_length The page length used to format the output data, specified by the number of lines the printer can print on each page. Specify a value in the range 1 - 255.

first_line The line number (within the specified page length) on which the first line of data should be printed. This determines the top margin on the printed page. For example, 1 specifies no top margin (print on the first printable line of the page), and 5 specifies a top margin 4 lines deep.

Specify a value from 1 to the number of lines specified by the **`form_length`** parameter above.

last_line The line number (within the specified page length) on which the last line of data should be printed. This determines the bottom margin on the printed page. For example, on a 66-line page, 66 specifies no bottom margin (print down to the last printable line of the page), and 62 specifies a top margin 4 lines deep.

Specify a value between the values of the **`first_line`** and **`form_length`** parameters above.

channel_2 - channel _12 Vertical channel positions 2 to 12 specify “vertical tabs”— specific positions on the page at which data can be printed. When the data form the host contains the control code “go to vertical channel *n*”, the line number specified here for that vertical channel determines the position on the page at which subsequent data will be printed.

For each of the channels 2 to 12, specify a line number, between the values of the **`first_line`** and **`form_length`** parameters above. Alternatively, if you specify zero for a particular vertical channel position, the output device will simply interpret the “go to vertical channel *n*” instruction as a line feed.

define_ws_device

The **`define_ws_device`** record defines a printer or punch device used for processing RJE output. Up to 9 printer and 9 punch devices may be defined for each workstation. Each device, identified by its device type (printer or punch) and device number (1 - 9), must match a device defined at the host for this workstation.

Some host systems make a distinction between punch devices and exchange devices, and require the use of exchange devices for transferring data records of more than 80 bytes. SNAplus2 does not make this distinction, because it does not impose a restriction on the record length for punches.

If the host requires output to be sent to an exchange device, you can include this in the RJE style file using a punch device. Any output for exchange device *n* will be routed to the punch device *n*. For example, if the host specifies that an output device is exchange device number 3, you need to define a punch with device number 3 for the workstation, and output for this exchange device will be sent to punch number 3. Note that in this case, the host configuration must not also include a punch device number 3. Problems may occur if the host attempts to route outputs from two different jobs to a punch and an exchange device that have the same number.

Parameter Name	**Type**	**Length**	**Default**
`[define_ws_device]`			
`description`	`character`	`31`	`(null string)`
`device_number`	`decimal`		
`device_type`	`constant`		`PUN`
`form_name`	`character`	`8`	`(null string)`
`translate_data`	`constant`		`YES`
`translate_transp_data`	`constant`		`YES`
`concatenate`	`constant`		`YES`
`output_type`	`constant`		`DIR`
`pdir_passthrough`	`constant`		`NO`
`add_punch_nl`	`constant`		`YES`
`output_path`	`character`	`256`	`/var/opt/sna/rje/`*`WKST`*`/output`
`output_file`	`character`	`(see notes)`	

description An optional text string describing the device. This string is for information only. It is stored in the style file and displayed in the RJE style file customization program.

device_number The number that the host uses to identify this device, in the range 1 - 9. This must not match the device number of any other device of the same type (printer or punch, as specified by the **`device_type`** parameter) defined for the workstation.

The host configuration identifies each printer or punch only by its number. The range of numbers for each device type is always consecutive and starts at 1. For example, if two printers and three punches are defined at the host, they will be assigned printer numbers 1 and 2 and punch numbers 1, 2, and 3. You must set up the number for each device so that the range of numbers for each device type is consecutive and starts at 1 to ensure that the SNAplus2 configuration matches the host configuration. If there are gaps in the sequence of device numbers, the RJE programs will fail with an error message indicating that the style file is invalid.

device_type Specifies the type of output device. Use this parameter to define the appropriate devices of each type to match the host configuration.

Allowed values:

PRT	Printer device
PUN	Punch device (note that this value is also used for exchange devices (see the note in the introduction to this section).

form_name For a printer device, this parameter specifies the name of the Forms Control Buffer that defines the output format. It must match the name specified in a **`define_forms_control`** record in the same workstation style file. For a punch device, this parameter is ignored.

translate_data Specified whether data received from the host is translated from EBCDIC to ASCII before being routed to the appropriate file or program.

Allowed values:

YES	Received data translated from EBCDIC to ASCII.
NO	Received data is not translated.

translate_trans_data Specifies whether transparent data received from the host is translated from EBCDIC to ASCII before being routed to the appropriate file or program. The host output may contain control codes indicating that a section of data is to be treated as transparent data. This means that the printer should pass it through as received without checking it for control codes.

Allowed values:

YES	Received transparent data is translated from EBCDIC to ASCII.
NO	Received transparent data is not translated.

If **`translate_data`** above is set to NO, this parameter must also be set to NO.

concatenate Specifies whether data routed to a file is concatenated to the end of any existing data in the file. This option applies if **`output_type`** below is set to FILE. Otherwise, it is ignored.

Allowed values:

YES	Data routed to a file is concatenated to the end of the file.
NO	Data routed to a file replaces the current contents of the file.

output_type Specifies how output received from the host for this printer is handled.

Allowed values:

DIR Output from each job is sent to a separate file in a particular directory. The directory must be specified using the **`output_path`** field (see below). The file will be given the name RJE_*abcdef*, where *abcdef* is an alphabetic string unique to each job. The strings are assigned in sequence so that alphabetical order corresponds to the order in which the outputs were received from the host. The **`output_file`** parameter must not be specified.

FILE Output from all jobs is sent to a single file, which must be specified using the **`output_file`** and **`output_path`** fields (see below).

PROGRAM Output from all jobs is used as the standard input to a program, which must be specified using the **`output_file`** and **`output_path`** fields (see below). If output from more than one job is directed to the same program at the same time, multiple instances of the program will be started.

pdir_passthrough Specifies whether the information in any PDIRs (Peripheral Data Information Records) accompanying host output is routed with the host output.

Allowed values:

YES	PDIR information is routed as described below.
NO	PDIR information is discarded.

The usage of this field depends on the value of the **`output_type`** field, as follows:

- If **`output_type`** is DIR or FILE, the information in any accompanying PDIRs is sent to a second file, with the same name as the output file but with the extension **`.i`**. For file output, this file is appended or overwritten in the same way as the output file.
- If **`output_type`** is PROGRAM, the information in the first PDIR sent to the device is passed to the program as parameters. See Chapter 3 of this manual for information on how the program should use PDIR information.

add_punch_nl Specifies whether to insert a new-line character after each record of the host output. This option applies only to punch devices; do not specify it for printer devices.

Allowed values:

YES	Insert a new-line character after each record, regardless of whether the host data stream includes new-line characters.
NO	Do not insert new-line characters. Select this option if the host includes the required new-line characters in the data stream.

output_path Specifies the directory to which output is routed. This must be the full path of a directory on the computer where the workstation runs.

The usage of this field depends on the value of the **output_type** field, as follows:

- If **output_type** is DIR, the field specifies the directory in which output files are created. The primary user and group of the workstation must have read/write access to this directory.
- If **output_type** is FILE, the field specifies the directory path of the file to which host output sent to this device is written. The primary user and group of the workstation must have read/write access to this directory.
- If **output_type** is PROGRAM, the field specifies the directory path of the program to which host output sent to this printer is routed.

output_file Specifies the file to which output is routed. The maximum length of this parameter is determined by the maximum filename length on the HP-UX system. This file must exist in the directory specified by **output_path** on the computer where the workstation runs.

The usage of this field depends on the value of the **output_type** field, as follows:

- If **output_type** is DIR, this field must not be specified.
- If **output_type** is FILE, the field specifies the file to which host output sent to this device is written. The primary user and group of the workstation must have read/write access to this file. The concatenate field (described above) specifies whether output from subsequent jobs will overwrite the file or be appended to it.
- It **output_type** is PROGRAM, the field specifies the program to which host output sent to this printer is routed. The primary user and group of the workstation must have execute access for this program. The host output will be used as standard input to the program.

Diagnostics

SNAplus2 checks the contents of the workstation style file when you start the workstation, when you use the spool commands, or when you start the console program. If the contents of the style file are invalid, a message is written to standard error and the command fails. Full details of the style file error are written to the error log file.

Check and correct the contents of the style file, and then retry the command. See the error log file for more information if necessary.

Configuring RJE Workstation with xsnapadmin

In addition to using **`snaprjecust`** and the text style file to configure the workstation, you must define some parameters with **`xsnapadmin`**. This information is stored in the SNAplus2 configuration file, which is set up and maintained by the system administrator and contains information that is required for RJE communications. For additional information, see the *HP-UX SNAplus2 Administration Guide*.

Workstation Configuration Information

The following parameters must be defined for the RJE workstation:

workstation name The name of the workstation (maximum of 4 characters).

primary user The user ID of the main user of this workstation. This must be a valid user ID on the server or client on which this workstation will run and a valid ID in the group for this workstation.

group name The group name of the users who use this workstation. This must be a valid group name for this workstation.

system name The name of the HP-UX server or client on which this workstation will run.

LU name Names of the LUs that this workstation uses.

A

Program Output with PDIRs

The following example is listed to show you another way of using output programs to process data from the host with PDIRs. It is being presented only as an example. It is not supported code.

Output Program Example

This example consists of two parts: a script that is configured as the output program, and a C program that must be compiled and installed as **`/opt/sna/bin/catrje.`**

The script uses this program to produce sections of the offset that are indexed by byte offsets from the PDIR file. The script is used to separate the header and trailer from an output file and then to append the header and trailer to each other section of output. Copies of these assembled sections of output then print, according to the number of copies specified in the PDIR.

Example:

```
/*Compile this program and install it as/opt/sna/bin/catrje*

#include <stdio.h>
void main(argc,argv)
int argc;
char *argv[];

{
int ch;
int count = 1;
int offset1,offset2;

   offset1 = atoi(argv[1];
   offset2 = atoi(argv[2];
   ch = getc(stdin);
   while ((ch != EOF) && (count <= offset2))
    {
      if (count >= offset1)
       putc(ch,stdout);
       ch = getc(stdin);
       count++;
    }

}
```

```
#!/bin/ksh
#This script is only an example. It is not supported code and
#Hewlett-Packard does not provide assistance if it does not work
#or needs modification.

#This script requires that the output file contains at least
#three PDIRs. There is a header, a trailer and at least one
#other section of output. Otherwise, it will not work.

#Change the name of the printer= statement to assign the
#desired printing program.
```

```
#The location of the temporary output file is the 14th
#parameter passed to the program. From this name the script
#can get the name of the PDIR file.

output file=$14
pdir_file=$output_file.i
###################################################

#Turn off output from stdout or stderr; comment these lines
#out for debugging purposes
exec 2> /dev/null
exec 1> /dev/null
###########################################################

exec 3< $pdir_file
integer count=2
integer n=0
integer copies
integer offset
integer end
integer trailer
integer start_header
integer start_trailer
integer start_body

#Read in PDIR first 5 values into arrays
while read -u3 offset[n] type[n] date[n] time[n] form[n] \
  fcb[n] train[n]
   do
     ((n = n + 1))
   done

#end = last line of PDIR file
((end = n - 1))

#exit if we have not read at least 5 lines (3 PDIRs plus OPEN and
#CLOSE statements)
if ((end < 5))
  then
   exit 1
fi

#trailer = trailer PDIR
#start header = first byte of header, start_trailer = first byte
#of trailer
((trailer = end -1))
((start_header = ${offset[1]} + 1))
((start_trailer = ${offset[trailer]} + 1))

#cp the header and trailer to temporary files
cat $output_file | catrje $start_header ${offset[2]} > \
  $output_file.header
cat $output-file | catrje $start_trailer ${offset[end]} > \
  $output_file.trailer

#Create output files consisting of header, body, trailer
#Body is created by starting at first byte after the PDIR
#offsets and terminating at byte offset of next PDIR.
```

```
#This code looks at the COPIES value for each body, and prints
#out more than one copy if specified. Other PDIR values can
#also be examined and used.

while ((count < trailer))
  do
    ((num_copies = ${copies[count]} + 1))
    ((start_body = ${offset[count]} + 1))
   cat $output_file | catrje $start_body ${offset[count + 1]} \
      > $output_file.body

    printer=”lp -n$num_copies”
    cat $output_file.header $output_file.body \
      $output_file.trailer | $printer

    ((count = count + 1))
  done

rm -f $output_file.header
rm -f $output_file.body
rm -f $output_file.trailer
```

Index

Numerics

- 128-byte data records, 33

A

- access permissions, 58
- accessing job spool, 32
- add_punch_nl, 45
- ASCII to EBCDIC conversion, 80
- audit log file, 64
- audit log file example, 64, 65, 66

B

- binary data, 16
- binary files, sending, 34, 35
- BROWSE mode, 118
- browse mode, limitations, 119

C

- cancelling jobs from spooler, 32, 98
- CANL, 69, 70
- checking RJE status, 102
- code conversion, 16
- command input area, console program, 114
- commands
  - snap start, 20
  - snapman, 20
  - snaprjecan, 25, 32, 98
  - snaprjeclr, 108
  - snaprjecmd, 106
  - snaprjecon, 23, 113
  - snaprjelst, 25, 29, 71, 94
  - snaprjepur, 100
  - snaprjesend, 24, 80
  - snaprjestart, 21, 61, 76
  - snaprjestat, 22, 102
  - snaprjestop, 25, 63, 78
  - spool, 28
- components of RJE, 28
- compressing job files, 35
- configuration information, 37, 57
- configuring equivalent SNAplus2 device, 37
- configuring workstation with xsnapadmin, 152
- console information, 113, 120
  - loss of, 119
  - viewing, 118
- console log size, 119
- console program, 15, 23, 33, 111, 112
  - correction keys, 117
  - ending, 120
  - scroll keys, 118
  - starting, 113
  - status line, 23, 114, 115
- console_log_size, 119
- control line
  - arguments, 84
  - commands, 85
  - syntax, 84
  - using with snaprjesend, 84
- cron script, 18
- Customize menu, 126
- customizing workstation, 126

D

- daemon, 20
- DATA LOST message, 119
- data records, 35
- DDNAME, 40
- default error log file, 21
- default parameter values, 132
- default record lengths
  - command-line option, 33
  - example, 33
- define_forms_control, 143
- define_forms_control parameters
  - channel_2 - channel_12, 144
  - description, 143
  - first_line, 144
  - form_length, 143, 144
  - form_name, 143
  - last_line, 144
- define_minor_options, 45
- define_minor_options parameters
  - console_log_size, 138
  - description, 135
  - display_sscp_messages, 141
  - ebcdic_newline, 139
  - ff_passthrough, 137
  - ignore_rec_length, 141
  - job_logs, 135
  - keep_failed_files, 140
  - list_sent_jobs, 136
  - logon_retry, 139
  - mail_on_events, 140
  - printer_nl, 138
  - punch_nl_IRS, 141
  - punch_padding, 137
  - run_in_foreground, 141
  - temp_directory, 137
- define_workstation_options, 133
- define_workstation_options parameters
  - description, 133
  - inbound_compression, 134
  - logon_text, 134
  - send_logon, 134
- define_ws_device, 145
- define_ws_device parameters
  - add_punch_nl, 149
  - concatenate, 147
  - description, 146
  - device_number, 146
  - device_type, 146
  - form_name, 146
  - output_file, 150
  - output_path, 150
  - output_type, 148
  - pdir_passthrough, 149
  - translate_data, 147
  - translate_trans_data, 147
- deleting job from spool, 25
- diagnostics, 151
- directory, output routed to, 42

E

- ending console program, 120
- environment variable, setting, 47
- error log file, default, 21
- error messages
  - from snaprjecust, 128
  - from workstation, 72
- error while sending, 30
- example of long listing, 97
- example of short listing, 97
- examples
  - notification of errors, 56
  - using control lines with snaprjesend, 90
- exchange data, 16, 33
- exchange devices, 36, 38
- exiting
  - console program, 120
  - RJE workstation, 23
  - snaprjecust, 128

Index

F
file compression, 16
File menu, 127
file transfer, storage and retrieval, 18
file, output routed to, 43
FOLLOW mode, 118
formatting information, printer output, 37

G
getting started, 19
group ID, 47

H
help key (f1), 115
help screen, 115
host commands, 112, 117
  entering, 117
  issuing, 106
host configuration, 112
host console information, 15
host data sets, 18
host output, 57
  receiving, 30
  routing, 30

I
information messages, 129
input processing, 32
INSERT mode, 117
interactions between RJE components, 28
issuing host commands, 106

J
Job Control Language (JCL), 30
job file, 29
  compression, 35
  contents, 30
  sample, 24, 31
job processing stages, 29
  adding to spooler, 29
  creating job file, 29
  listing spooled jobs, 29
  receiving host output, 30
  routing host output, 30
  running RJE workstation, 29
  sending jobs to host, 30
  viewing console information, 30
job queueing, 15
job spool, accessing, 32
job submission, 32
job submission, retries, 57
job_logs, 64, 68
JOBNAME, 40

L
list_sent_jobs, 25, 100
listing spooled jobs, 32
listing submitted jobs, 94
login ID requirements, 61
logon_retry, 59
logon_text, 132
long listing, example, 97
lost data, console, 119
ls (link station), starting, 20

M
messages
  status line, 23
modifiable data record size, 16
multiple LUs, 16
multiple RJE workstations, 15
MVS/JES2, 14
MVS/JES3, 14

N
new-line characters (NL), 35, 45
node, starting, 20
notification of errors, examples, 56
notification of received files, 52
notification values table, 53
notify values, 52

O
operator privilege for remote workstation, 112
output data, processing, 36
output devices
  exchange devices, 36
  for formatted data, 36, 37
  for unformatted data, 36, 38
  printers, 36, 37
  punches, 36, 38
output file
  appending, 43
  overwriting, 43
output program example, 154
output programs
  general guidelines, 47
  user & group ID, 47
output routing, 42
  to directory, 42
  to file, 43
  to program, 42
  to usr card, 44
output to disk, 15

P
padding punch records, 17, 45
parameter type
  character string, 131
  decimal number, 131
  defined constant, 131
PDIR file
  appended, 43
  overwritten, 43
PDIR file example, 40
PDIR parameters, 39, 46
  COPIES, 39
  date, 39
  DDNAME, 40
  FCBNAME, 39
  FORMS, 39
  JOBNAME, 40
  PROCSTEP, 40
  REQUID, 40
  SPINNO, 40
  STEPNAME, 40
  TIME, 39
  TRAIN, 39
  VOLIO, 40
pdir_passthrough, 39
PDIRs, 16, 39
Peripheral Data Information Record (PDIR), 16, 39
place parameter, 50, 51
port, starting, 20
precautions, when receiving files, 56
printer output, formatted, 37
printers (printer devices), 15, 36
processing input data, 32
processing output data, 36
PROCSTEP, 40
program output, 15
program, output routed to, 42

Index

punch output, 38
punch padding, 45
punch_padding, 45
punches (punch devices), 15, 36

Q
QUED, 69

R
receiving files, precautions
  when routing by default, 56
  when routing by usr card, 56
record lengths, 33
RECV, 69, 70
REPLACE mode, 117
REQID, 40
RJE audit log file, 64
RJE audit log file example, 64
RJE commands, 75
RJE console program, 33
RJE device name, 40
RJE job log file, 64, 68
RJE job log file example, 68
RJE job logging event
  CANL, 69, 70
  QUED, 69
  RECV, 69, 70
  ROUT, 69, 71
  SENT, 69, 70
RJE jobs
  getting information about, 64
RJE workstation, 28, 32
  configuring, 35
  exit, 23
  running, 29
  starting, 21
RJE workstation configuration, 122
RJECON prompt, 120
rjeusr program, 16, 44
rjeusrpad program, 17, 44
ROUT, 69, 71
routing
  by default, 56
  by usr card, 56
routing output
  to directory, 42
  to file, 43
  to program, 42
routing program output
  with PDIRs, 46
  without PDIRs, 46
routing received files
  with usr card, 48
  without usr card, 55
running RJE workstation, 57

S
saving style file, 127
screen redrawing, 119
scrolling of console display, 118
sending exchange data, 33
sending jobs, 32
  binary files, 34, 35
  text files, 34
  to host, 18, 24
SENT, 69, 70
setting environment variables, 47
short listing, example of, 97
snap start command, 20
SNAplus2 daemon, 29, 32
SNAplus2 RJE
  features, 15
  getting started, 19
  structure of, 28
  typical uses, 18
SNAplus2 RJE components
  output, 28
  RJE workstation, 28
snapman command, 20
snaprjecan command
  example, 99
  options, 98
  parameters, 99
  syntax, 98
  using, 25, 32, 98
snaprjeclr command
  example, 109
  parameters, 108
  syntax, 108
  using, 108
snaprjecmd command
  parameters, 106
  syntax, 106
  using, 106
snaprjecon command
  syntax, 113
  using, 23, 113
snaprjecust, 45, 123, 124
  exiting, 128
  messages, 128
snaprjelst, 71
snaprjelst command
  example, 95
  options, 94
  parameters, 95
  syntax, 94
  using, 25, 29, 94
snaprjepur command
  example, 101
  options, 100
  parameters, 101
  syntax, 100
  using, 100
snaprjesend command
  control lines, 84
  example, 83
  options, 80
  parameters, 82, 83
  syntax, 80
  using, 24, 80
snaprjestart command
  options, 76
  parameters, 77
  syntax, 76
  using, 21, 61, 76
snaprjestat command
  example, 105
  options, 102
  parameters, 102
  syntax, 102
  using, 22, 102
snaprjestop command
  example, 79
  options, 63, 78
  parameters, 79
  syntax, 78
  using, 25, 78
SPINNO, 40
spool commands
  cancel, 28
  list, 28
  send, 28
spool control commands, when to use, 57
stages in job processing, 29

Index

starting
  console program, 113
  daemon, 20
  ls (link station), 20
  node, 20
  port, 20
  RJE workstation, 21, 61
status
  of jobs queued at the host, 30
  of the workstation, 22
status line messages, 23
  BROWSE, 23, 114
  FOLLOW, 23, 114
  INSERT, 23, 114
  REPLACE, 23, 114
  RJE-, 23
  RJE+, 23
status line, console program, 114, 115, 118
STEPNAME, 40
stopping RJE workstation, 25, 61, 63, 78
structure of SNAplus2 RJE, 28
style file customization program, 124
style file records
  define_forms_control, 133
  define_minor_options, 133
  define_workstation_options, 132, 133
  define_ws_device, 133
submitting jobs, 80

T
tail command, 21
temp_directory, 59
Tempexit keystroke, 120
temporary exit to HP-UX command prompt, 120
text files, sending, 34, 35
text style file
  parameter types, 130
  parameters, 130
TRANSP, 45

U
unformatted data, punch output, 38
user ID, 47
using RJE console program, 111
usr card
  example, 54
  -f (fold) option, 48
  including in job file, 53
  logon field, 48
  notification caution, 52
  notification values, 52
  precautions, 48
  routing, 16, 24, 44, 48
  rules for use, 53
  syntax, 48
  user & group ID, 48
usr card parameters
  logon, 50
  notify, 52
  place, 50, 51

V
VSE/POWER, 14

W
warning messages, 128
workstation
  configuration, 57
  error messages, 72
  error while sending, 57
  RJE, 57
  status of, 22
  stopping, 61, 63, 78
  when to run, 57
workstation style file
  for a configured workstation, 125
  for an unconfigured workstation, 125
  overview, 123
  records, 130
  starting snaprjecust, 125
  text file, 130

X
xsnapadmin, 152