1. Installation

1.1 Before You Start

SIGMET's software is supplied on a cdrom containing installation files for HP–UX 11.11 (IRIS only) and RHEL ES 3. Additionally, SIGMET provides an X–Windows based install utility for installing SIGMET software on local or remote systems.

Before you install the IRIS/RDA software, do the following:

1. Check the operating system version.

   Verify that the operating system level is at least the following:

   - HP-UX: V11.11 (IRIS ONLY)
   - Linux: Red Hat Enterprise Linux ES Release 3. See Appendix A for installation details. Other brands of Linux should work fine as long as the kernel version is at least 2.4.21. SIGMET will not be able to answer OS specific support questions on other brands.

SIGMET provides an automated installation procedure and an extensively documented manual installation procedure for installing RHEL ES Release 3. If you are starting with a new computer please go to Appendix A to learn more.

---

2. Use the following sections to perform a new install or upgrade.

1.2 Mount the CD

Login to the system as 'operator' and start the X windows environment if it is not already running. Become the superuser by using the 'su' command and supplying the appropriate root password. Select the root directory.

Insert the cdrom into the cdrom drive. Next you must have the system "mount" the cdrom. This allows for the files on the cdrom to be access in a similar way to files on the hard disk.

On Linux systems:

```
# mount /dev/cdrom
```

The default mount point is /mnt/cdrom. If this does not work, you should probably add the following to your /etc/fstab file:

```
/dev/cdrom  /mnt/cdrom  iso9660 noauto,ro  0 0
```

If your CDROM is an external SCSI CDROM, you may have to create a symbolic link addressing your CDROM before trying to mount it. In this case do:

```
# ln –s /dev/scd0 /dev/cdrom
```

On HP–UX systems:

```
# mount –ocdcase /dev/dsk/device /cdrom.
```

The default mount point is /cdrom. If you know the proper value for ***device*** on your system, the above command is all you need. If you do not know the value of ***device***, you can use the sam utility to determine it. To execute sam, type # **sam**

Once in sam, select **DISKS/FILE SYSTEMS**. Then select either **CD–ROM** or **DISK DEVICES**. Click on the CD–ROM that you will be using highlighting the line. Then click right and release on **ADD DRIVE**. If prompted about a logical volume manage, choose **NO LV Manager**. Set the mount directory to **/cdrom**. Then if MOUNT AT EVERY BOOT is selected, de–select it. Click **OK** once more to actually perform the mount. It is recommended that you now click left on the CD–ROM drive that you using highlighting it. Then click right and release on **View More Information**. This brings up an information window that will tell you the actual name of the *device* that was mounted. Write down and keep this information so in the future you will be able to use the simply mount command to mount the disk instead of the more involved sam procedure.

You must make sure the root user has gunzip in the path. Check this by typing "gunzip". If it is not there, SIGMET recommends that you add symbolic links as follows:

```
# ln –s /usr/contrib/bin/gunzip /usr/bin/gunzip
# ln –s /usr/contrib/bin/gzip /usr/bin/gzip
```

Skip to the next section:

If this is a new installation, go to Section 1.3.2; if this is an upgrade, go to Section 1.7.7. For a general description of the **install** utility and its options, Section 1.8.

1.3 New Software Installation

1.3.1 Preparing for a New Software Installation

Follow the instructions in this section if you are installing a new IRIS/RDA system. If you are **upgrading from a previous IRIS/RDA version, go to Section 1.7**. If you used the **sigconfig** script described in Appendix A, go to **Section 1.5.2**.

If this is a new installation on a new computer or a new hard disk, please read Appendix A before proceeding. **SIGMET strongly recommends the use of its automatic installation procedure** as described in Appendix A.

1. Check that there is enough disk space.

   The IRIS/RDA software uses about 100 megabytes on the disk. The data stored on disk by IRIS typically uses at least 100 megabytes. Thus, you need at least 200 megabytes of free space on the disk before the installation is begun. The software and data can be on separate disks, if desired. If there is not enough space, unnecessary files should be deleted before proceeding. You can see the amount of free disk space using the **`df`** command.

2. Check for conflicting user names.

   The IRIS software (including the installation procedure itself) requires the creation of two new login names: **`operator`** and **`observer`**. If either of these names are already in use, they will have to be changed. You may, however, use any convenient (new or old) group name for these two new login names. In this discussion, it is assumed that the group name is "users". Use the **redhat–config–users** or **linuxconf** utility on Linux or the **`sam`** utility on HP–UX to make the changes.

Important: Do not proceed without making these login name modifications. The install utility will not run if the `operator` name is undefined. If this is a new system (OS installation) please read through Appendix A of this manual and run sigconfig from the CD.

---

3. Choose an anchor point for the IRIS directory tree.

Note: SIGMET recommends using `/usr/sigmet` as the default. You may choose another anchor point, but the remaining discussion assumes the `/usr/sigmet` anchor point.

To create this directory, login as root and then type:

```
# cd /usr
```

```
# mkdir sigmet
# chown operator:users sigmet (use actual group name)
```

4. Create Data Directories

SIGMET software requires a number of directories to hold the data that it generates. These directories may be positioned anywhere within the file system. They have no connection with each other or with the **/usr/sigmet** installation point. The directories are listed below along with their purpose.

ASCOPE	Ascope data files
INGEST	Acquired radar data in polar form
PRODUCT	Normal product files from the product generator
PRODUCT_RAW	Raw product files from the product generator
LOG	Error, status, and history messages
TAPE_INV	Tape inventories for quick retrieval
TEMP	Temporary storage used for network output

Next, create the directories at the operating system prompt. Be sure the owner and group are set to match operator's default. For example:

```
# mkdir /usr/iris_data
# cd    /usr/iris_data
# mkdir ascope
# mkdir ingest
# mkdir log
# mkdir product
# mkdir product_raw
# mkdir tape_inv
# mkdir temp
# chmod 775 *
# chown operator:user *    (use default group for operator)
```

5. Make a backup of your system.

6. Go to Section 1.2.

1.3.2 Performing a New Software Installation

Login to the system as 'operator' and start the X windows environment. Become the superuser by using the 'su' command and supplying the appropriate password (Note – if you are doing a network based 'install to' option, it is not necessary to become the superuser – see section 1.9.2).

Important: A new installation completely overwrites any existing SIGMET files in the ${IRIS_ROOT}/bin, and ${IRIS_ROOT}/config trees. Make sure you do not have any irreplaceable data in these areas before proceeding.

HP–UX SYSTEMS:

```
# cd /cdrom/sigmet/iris/hp-ux
```

LINUX SYSTEMS (IRIS):

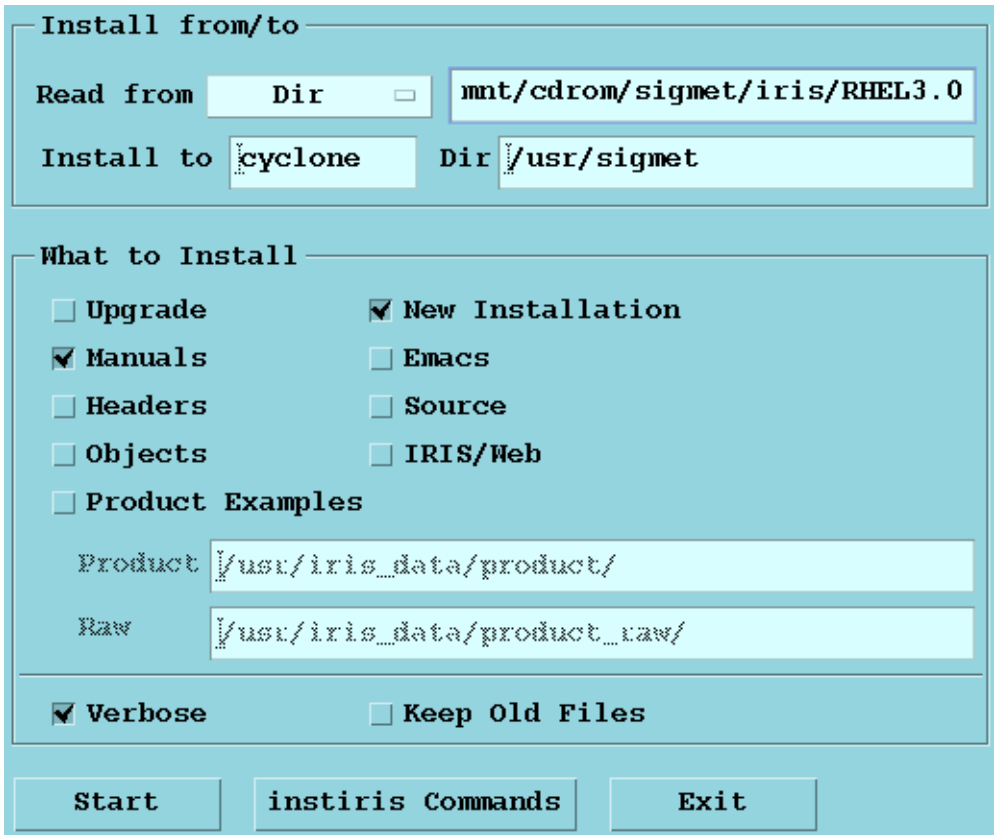
```
# cd /mnt/cdrom/sigmet/iris/RHEL3.0
```

LINUX SYSTEMS (RDA):

```
# cd /mnt/cdrom/sigmet/rda/RHEL3.0
```

ALL SYSTEMS:

```
# ./install
```



In the **`What to Install`** section of the install utility window, you must select **`New Installation`** by clicking the button next to the text. This is shown above.

Note: If you are performing a RDA and IRIS install, first install the IRIS software. Then install the RDA software with the Keep Old Files button pressed.

You may optionally also select other optional SIGMET support software packages to install by pressing their buttons in the **`What to Install`** section. When ready to do the install, just press **`Start`**. For a detailed description of the full capabilities of the install utility, see section 1.8.

The installation utility takes a few minutes to complete and prints progress messages as it runs. Don't forget to unmount the **CDROM** after the installation is complete. Example commands are below:

```
# cd /
```

HP–UX SYSTEMS:

```
# umount /cdrom
```

LINUX SYSTEMS:

```
# eject /mnt/cdrom
```

Continue with the following steps to complete the installation process and required configuration.

1.3.3 Copy SIGMET environment files into /home/operator

SIGMET requires a number of special "environment" files to be included in the /home/operator directory so they can be executed whenever the operator user logs in. Copy the following files as described below:

```
# cd /home/operator
# cp /usr/sigmet/config_template/LINUX/desktop/* .
# mv mwmrc .mwmrc ; mv xinitrc .xinitrc ; mv profile
.profile ; mv Xdefault .Xdefaults ; mv kshrc .kshrc
# chown operator:users *
# chmod 775 *
```

Logout of the computer and login as operator, you should have entered a blue screen with a white x–term window. Continue with the next step to complete the post installation configuration.

1.3.4 Run setup utility

Use the setup utility to configure your software (this utility is described in detail in the *Utilities Manual*).

```
$ setup
```

Successful operation of **setup** does indicate that the utilities can modify the configuration files. All of the configurations in the **setup** utility should be checked for your system.

If your system is an **analysis or display system**, be sure to set that in the General setups, and in the RVP section specify that there is no processor, and the the RCP should specify that there is no antenna.

If your system is a radar system (i.e. connected to an antenna/signal processor) the values in the RVP and RCP sections of setup are partifularly important. These need considerable fine tuning before the RCP can be controlled accurately. For example,

the maximum antenna speeds in both azimuth and elevation should be set at least six degrees per second slower than the maximum the antenna can go. To check how fast it can go, widen the limits, then run the **antenna** utility to see where it maxes out. The **antenna** utility is also discussed in the *Utilities Manual.*

Finally, in the license section of the utility, enter a desired site name. Make sure that this is unique and somehow relates to your radar site. Also, if you plan on sharing data with others, it is important that the site name is unique.

1.3.5 Get your SIGMET license (IRIS or TS Archive systems ONLY)

Run **`show_machine_code`**, which displays licensing information for your system — Machine code, Check code, Site ID, and OS Version number. For new installations, **you need to get a new license number from SIGMET**.

Contact SIGMET (support@sigmet.com)with the information supplied by **`show_machine_code`**, and we will provide you with the license numbers needed for IRIS to operate. If the machine code is the same as on your current license, no upgrade is required.

When the new license has arrived, run the **setup** utility and click the **`License`** button. Type the numbers at the **`Features License`** and **`Products License`** prompts.

For display systems, please use our free IRIS/Display license service on our web site (http://www.sigmet.com).

1.3.6 Run sigmet_env

Run the **sigmet_env** utility program. It checks for obvious mistakes like directories which do not exist. The **sigmet_env** defaults to IRIS only systems, to check the directories on an RDA system, run **sigmet_env –rda**.

1.4 Initial Configuration

1.4.1 Setting up for rcp and other "r" commands

Normally IRIS uses the **rcp** shell command internally to copy product files between computers over the network. For security reasons, this feature is by default blocked, and requires configuration to enable it.

There are two mechanisms to configure the remote commands: The **~/.rhosts** file and the **/etc/hosts.equiv** file. These files are placed on the receiving computer to enable access from the outside world. The .rhosts file is placed in each user's login directory, and the hosts.equiv file is placed in the /etc directory and applies system wide. SIGMET recommends using the hosts.equiv file and checking to make sure there is no .rhosts file for the applicable users. Note that root requires a private .rhosts file to do rcp. This file is basically a list of hostnames and user names. Though wildcarding is allowed, we recommend adding every host and user who needs it.

This can be little tricky to configure, so always test manually. The development cycle is very simple. Test with an rcp command like:

```
$ rcp file host:/directory
```

If it fails, then edit the hosts.equiv file and try again.

Figure 1–1: Sample `/etc/hosts.equiv` File

```
# Format required Linux:
radar.company.com     operator
analysis.company.com  operator

# Format required on HP:
radar.company.com     +
analysis.company.com  +
```

1.4.2 Setting up for scp and other ssh commands

First verify that the ssh daemon is running on both systems. This can be done with the command:

```
$ ps –aef | grep sshd
```

To turn on sshd run the following command as root:

```
# chkconfig ––add sshd
# chkconfig sshd ––levels 345 on
```

This program is configured with the file /etc/ssh/sshd_config. The default file should work OK for IRIS scp needs. The security authorization is different for each user, so be sure to run all the applicable commands while logged in as operator. On the source machine generate your private key with the following command:

```
$ ssh–keygen –t dsa
```

It will prompt you for a password, enter blank for no password. This will create the directory $HOME/.ssh. It will place in it the following files: **id_dsa** and **id_dsa.pub**. You then need to copy the contents of the **id_dsa.pub** file to all the target machines. Append it to the **$HOME/.ssh/authorized_keys** file. If there is no such file, then just rename it to match. If your target is a RedHat 7.1 or earlier machine, the file must be named **authorized_keys2**. Now test by trying a simple command like:

```
$ ssh target date
```

The first time this is run, it will ask you to confirm the target host. Thereafter it will not. It should not ask for a password. If it does, there is a problem. Problems can be easily seen by examining the /var/log/secure log file.

1.4.3 Serial Line Setup

Disable Logins on the Serial Lines:

A problem that frequently arises when assigning the computer's serial lines to IRIS is that the system may already be running a **`getty`** process to log users onto that line. This process must be removed before IRIS can use the serial line. Typically, there are command statements in the **`/etc/inittab`** file that startup **`getty`**. These should be commented out (using a '#' at the beginning of the line) or removed.

Set world read/write permission:

Each serial device must be set so that users can read and write to it. For a device called **`/dev/tty`** the following commands are appropriate:

```
# ls –l /dev/tty
# chmod 666 /dev/tty
```

1.4.4 Create mount point for MO disks

If your system has a MO disk drive, you need to create the mount point. SIGMET recommends /modisk. You will also need to specify the device file in **setup**.

```
# mkdir /modisk
```

1.4.5 Installation Security Issues

During the course of an installation, the **install** utility needs to perform certain root level privileged operations. Such operations include the setting of the ownership and mode of certain IRIS executables. Because normally the **install** utility is run by the root user, these operations are allowed without question.

In the case of doing a network based installation using the "INSTALL TO" option, this installation method is not performed by root, but is instead performed by the operator. To grant the user operator the permission required to do these privileged operations, the iris **install** utility uses the Unix **sudo** facility.

By use of the sudo utility, the operator effectively is allowed to execute the install utility as the superuser. There are some minor security implications of this. Under most circumstances it is fine for the operator to perform these operations. If you wish to allow the operator to perform these actions (normal scenario), then no special action is required – just run install as documented in the following sections.

If you wish to prevent the operator from doing these operations, you should create an empty sudo configuration file using the below commands:

```
# rm –f /etc/sudoers
# touch /etc/sudoers
```

By disabling the root privileges for the **install** utility, when doing a network based "INSTALL TO" installation, after the install is finished, you will need to login to the remote system as root to complete the privileged part of the installation. The **install** utility will prompt you with the exact command to execute in this case.

1.4.6 Startup profile File

SIGMET software requires some configuration of the operating environment before it can run. Most of this is done by defining environment variables in a startup file, then arranging for that file to be executed on each login. We document here only for the Korn shell which is the recommended shell.

1.4.6.1 How to Create and Modify the profile File

If this is a new installation, you must create your working **`profile`** file. You can copy this file from /usr/sigmet/config_template/LINUX/desktop/. This machine–specific file resides in the **`/usr/sigmet/config`** directory. Edit it and make any custom changes for your machine. Finally, in the **`.profile`** scripts of all SIGMET users (located in their default login directories), add a line to source the IRIS configuration file on initial startup. Note that if a set –u command exists in the .profile, then the line below must come before the set –u command:

```
. /usr/sigmet/config/profile
```

When using a workstation running the Common Desktop Environment (CDE), a file called .dtprofile exists in your home directory. If this file exists, you must edit it and at the end insert the following line:

```
DTSOURCEPROFILE=true
```

SIGMET upgrades preserve everything in the **`config`** directory (except the **`DEFAULT`** menu files), but the **`install`** directory is recreated each time. This is the reason for making custom changes within the configuration directory.

If this is an upgrade installation, you already have your old profile file in the configuration directory. You should compare it with the template file in the **`/usr/sigmet/config_template/init`** directory. Check for altered names

or changes to the file structure and edit the old profile file, making whatever changes are needed to bring it in line with the new template. Do not be surprised if there is nothing that needs to be changed.

Whenever you change **/usr/sigmet/config/profile**, you must rerun it:

```
$ cd /usr/sigmet/config
$ . ./profile
```

Then check that your changes are correct by typing:

```
$ env | grep IRIS
```

Changes to the **profile** apply only to applications started in the terminal window where the **profile** was run. To make the changes apply to all applications, you must restart the system.

1.4.6.2 Operator list Defined in the Startup File

These variables specify lists of users who can operate IRIS fully, and who can observe its operation but not make any changes. Note that every IRIS user must share group access to files owner by **operator**.

```
IRIS_OPERATORS='operator joe alan claudia'
IRIS_OBSERVERS='observer'
```

1.4.7 Do platform dependent configuration

On Linux systems:

Goto section 1.5.

On HP–UX systems:

Goto section 1.6.

1.5 Configuring Linux for the first time

For radar systems, the RVP7 must be on and connected at boot time. SCSI devices are automatically assigned device names at boot time. The name chosen for the RVP7 is typically "/dev/sga", but will differ depending on which other SCSI devices you have. To see details on all the SCSI devices type:

```
$ dmesg | grep scsi
```

Once you have determined you device file, you will need to change the protection to allow world access. First check with the **ls –l** command to find what file your device file is linked to. Then set the protection with a command like:

```
# chmod 666 /dev/sg0
```

1.5.1 Automatic startup of X and Default SIGMET Environment

SIGMET recommends that you configure your system to automatically start X when you log in. To do this, you must configure the following files in your home directory:

```
.kshrc
.mwmrc
.profile
.Xdefaults
.xinitrc
```

Example files for all these are shipped in the ${IRIS_ROOT}/config_template/LINUX/desktop directory. Copy the files to your home directories, and rename to add a "." prefix. You can add startup programs to you login screen by adding them to the .xinitrc file just before mwm, for example:

```
xsetroot –solid DarkSlateBlue&
xset s off
xset –dpms
xclock –geometry 70x70–0+0 –bw 0 &
xterm –geometry 80x30+352–342 –font 8x13bold &
mwm
```

1.5.2 Configuring NTP

To configure your machine to time sync with another machine, edit the /etc/ntp.conf file to contain a single line similar to the following:

```
server 198.102.75.10
```

Substitute in the correct IP address of the machine to sync to. To make a computer the time server, use the special address as follows:

```
server 127.127.1.1
```

Ntp will set the time only after 10 minutes of consistent time difference measurements. This means that after booting, the time may change in about 10 minutes. This can cause problems with automatic startup of IRIS. To fix this

problem, create a file **/etc/ntp/step–tickers** and put in just the server IP address, without the word "server". Ntp will then set the date at boot time, if possible. Do not put in the step–tickers file on the time server.

To configure for automatic startup of ntpd, type the following:

```
# chkconfig ––add ntpd
# chkconfig ntpd on
```

This will take effect after the next reboot. To start without rebooting, type:

```
# /etc/rc.d/init.d/ntpd start
```

It will take 10 minutes before it will sync the times. If the times are more than 10 minutes apart, ntp will assume there is an error and never change the time. To check on the status of ntp, type:

```
# ntpq –p
```

The first character before the server host will be "*" when it thinks it is type synced.

Another convenient check to compare the time of your workstation with that of another (such as the ntp server is):

```
# date ; rdate –p nodename
```

Note the semicolon between the two commands allows both the local "date" command to be run simultaneously with the remote date (rdate) command on the other workstation. This allows the times to be easily compared.

You can also manually set the time from another computer with the following command. This will not work if the ntpd is running on your machine.

```
# ntpdate host
```

1.5.3 Configuring xinetd

The default configuration for **xinetd** in Linux allows receipt of only a limited number of remote shell commands per minute. This limit can easily be exceeded with a burst of network transfers between SIGMET systems. This will cause a network send request to become "aborted", and the network link to fail thereafter. SIGMET recommends raising this number to at least 100 on all systems. This is easily done by editing the /etc/xinetd.d/rsh file and adding a line that reads as follows:

```
per_source = 100
```

Changes take effect when you reboot, or send the hup signal to inetd with:

```
kill –s hup /var/run/xinetd.pid
```

These instructions apply to RedHat Linux 7.0 and later.

1.5.4 Configuring the root prompt

All of our SIGMET computers are normally configured to give a shell prompt of "$" for normal users, and "#" for root. This helps to avoid confusion. Unfortunately, starting with RedHat 7.X, this is not the default behavior for root on Linux. To fix this, go to root's home directory of /root. Edit the file .bashrc. Before the aliases, insert the line:

```
PS1="# "
```

1.6 Configuring HP–UX

1.6.1 Create the SCSI Pass-Through Driver for the RVP

The RVP7 communicates with the host computer via a pass–through driver — a UNIX device driver that permits direct transmission of SCSI commands. This driver must be installed and made available when IRIS is first installed.

A SCSI bus ID must be assigned to the RVP7. The ID must be in the range 0–7, and must not conflict with any other device on the bus. Also, ID's 6 or 7 are often reserved for the SCSI initiator device — the "computer end" of the cable. In the following instructions, ***N*** denotes the SCSI ID you have chosen. Be sure to set the jumpers on the RVP7 to match.

Use the **sam** utility to add the **`sctl`** (series 700) or **spt** (series 800) device driver to the system kernel as follows:

---

i **Note: The RVP7 must be connected and powered up and the machine rebooted before these commands are issued.**

---

1) Invoke sam by typing "sam" in a Unix window

2) Choose "Kernel Configuration"

3) Choose "Drivers"

4) Scroll down to "sctl" for series 700 machines or "spt" for series 800 and click left to highlight. If "spt" does not appear in the list and this in a series 800 machine, it will be necessary to first use sam to install the spt fileset from the core–os cdrom.

5) If current state of the driver is "in" then exit from sam and continue below in creating the special device file, else click right and select "Add driver to kernel" followed by "Create a new kernel". System will then need to be rebooted.

Next, create the special device file for the signal processor. The command below is used to do this. Note that this command requires three yet undefined parameters (***MAJOR***, ***K*** and ***N***).

```
# mkdir /dev/scsi
# mknod /dev/scsi/cKtNd0 c MAJOR 0x0KN000 (generic)
# mknod /dev/scsi/c1t4d0 c 203 0x014000
# chmod 666 /dev/scsi/c1t4d0
# ioscan –M DRIVER –H HARDWARE PATH (generic)
# ioscan –M sctl –H 8/16/5.4.0
```

MAJOR is the major number of the device driver. This can be determined by running the "lsdev" command and searching for either sctl or spt.

K is the controller number. This is the logical scsi controller/bus that the signal processor is attached to. This can be determined by running the "ioscan –f" command. The output listing from this command will show the SIGMET RVP7 and

its hardware path. You must look above this line in the listing and find the “ext_bus” listing in the class column that has the same hardware path root as the RVP7. The I (Instance) column of the ext_bus row is the controller (***K***) that must be specified to the mknod command.

Parameter ***N*** is the scsi id of the signal processor. This can also be seen most easily by running the “ioscan –f” command. For example, of the hardware path of the SIGMET RVP7 is 12.4, then the scsi id (***N***) is 4.

The parameter ***DRIVER*** is **spt** for 800 series systems and **sctl** for 700 series systems. The ***HARDWARE PATH*** is the exact hardware path output from the ioscan –f command for the SIGMET RVP7.

1.6.2 Configuring for Automatic Startup

Copy the file **`${IRIS_ROOT}/config_template/init/iris_init_hpux`** to **`/sbin/init.d/iris_init`**. Edit this file if your IRIS root is not /usr/sigmet. Set it’s ownership and protection to match the other files there. Next look in the directory **`/sbin/rc3.d`**. You wish to create a link named **`S800iris`**. First look at the files in that directory, each “S” file represents a process which is automatically started up at boot time. They are started up in numerical order. If there already is a file starting with S990, then try S991, etc. We want this to run very near the end but before the dtlogin. Once you have picked you name, create a link with the command:

```
# ln –s /sbin/init.d/iris_init /sbin/rc3.d/S800iris
```

Similarly, if you want to automatically shut it down, create another link. Be sure to again check to see if the sequence slot is already used. Note that we wish to reverse the order on shutdown, so the filename now starts with K200. Only IRIS versions starting with 6.07 support automatic shutdown.

```
# ln –s /sbin/init.d/iris_init /sbin/rc2.d/K200iris
```

Under HPUX, the **`–auto`** flag causes all IRIS subprocesses to be started in such a way that they ignore the first SIGHUP (hangup) signal that they receive. This is necessary because HPUX sends SIGHUP to every child process that was spawned during the bootup procedure.

1.6.3 Create link for libXmu.sl

IRIS requires this library to run. Create it with the following commands:

```
# cd /usr/lib
# ln –s /usr/lib/X11R4/libXmu.sl libXmu.sl
```

For developers only: IRIS also requires the header files for libXmu to compile. Create it with the following commands:

```
# cd /usr/include/X11
# ln –s /usr/contrib/X11R6/include/X11/Xmu Xmu
```

1.6.4 Check patches

To check what patches are installed on your 10.20 system look in the file /var/adm/sw/patch/PATCH.log.

To fix a memory leak in the mmap call, you need patch PHKL_10259 (PHKL_10260 on 800 systems), or its successor PHKL_18617.

If you system has either PHSS_12824 or PHSS_14040, which are X–windows roll-up patches, you will need to supersede them with PHSS_16120.

If you are having a problem with the library “dld.sl”, try installing the C++ runtime library patch PHSS_14262.

Patches may be found on the extension software CD, on our release in directory ${IRIS_ROOT}/config_template/HPUX, or from the hp ftp site:

`ftp://us-ffs.external.hp.com/hp-ux_patches`

To install patches, place them in a temporary directory, run sh on them, then look at the created .text file for instructions.

1.6.5 Check kernel parameters

You may have to raise some configurable kernel parameters using sam. We recommend a minimum of the following values:

```
maxdsize  256 MB
maxuprc   150
nproc     300
semmni    256
semmns    256
```

1.6.6 Enable windows on the local host

HP–UX seems to ignore the /etc/X0.hosts file for opening windows from it’s own host. To fix this, create a file /etc/dt/config/Xsession.d/090xhost containing the line “xhost +host” where “node” is you local hostname. Set protection to 555, owner and group to bin.

1.6.7 Disabling the Distributed Printer Services

A problem reported to SIGMET by various customers is that the operating system processes associated with the Distributed Printer Services (DPS) sometimes get into a state when they use all of the CPU time on the HP–UX workstation. Mainly the process is called “basicdsd”. If this particular problem is encountered, you can safely disable the DPS by editing the file /etc/rc.config/d.pd and setting the control variable PD_CLIENT equal to 0. This will have no effect on your ability to print.

1.7 Upgrade Installation

1.7.1 Preparing for an Upgrade Installation

1.7.2 When should I upgrade?

If your system is operational and you do not require the new features of a release, then often the best thing is to NOT upgrade. Check the release notes available at www.sigmet.com in the customer support section to see what changes have been made since your current release was installed. Be sure to check the release notes for all intervening releases.

To check the release that you have currently installed, you can type the command:

```
$ show_machine_code –version
IRIS Version 8.00  (indicates version 8.00)
```

1.7.3 What should I do before I upgrade?

1. Save customized files.

As part of the upgrade, many of the files in the SIGMET tree are erased and replaced. If you have placed anything important there, be sure to save it first. The upgrade procedure preserves everything in **${IRIS_CONFIG}**.

2. Print the current setup.

Run the **setup** utility on the old system and generate an ASCII listing file with the File/List command. The format of these files can change between software versions. Therefore, some of the information may have to be entered again.

3. Before you can upgrade, you must maek certain that all SIGMET applications are stopped. You can do this by exiting any IRIS/RDA utilities that you are running and executing the following commands.

```
For IRIS:
$ qiris
$ qant

For RDA:
$ su
$ service rvp8 stop (or service rcp8 stop)
$ service dspexport stop
```

Then type **ps_iris** to verify that all SIGMET processes are stopped. If tehre are remaining processes, you can stop them as root with the "**kill** <process ID number>" command. The process ID number is the first column of numbers from the ps_iris output.

1.7.4 Where to get software upgrades?

The RVP8 and RCP8 (collectively the RDA) and IRIS are active products. New features and bug repairs are provided in the form of software upgrades. Software upgrades from SIGMET can be obtained from two sources:

- **FTP (Internet) Upgrades**- These are available from ftp.sigmet.com. For example, to obtain the release RDA 8.00 you would go to:

  `ftp.sigmet.com/outgoing/releases/rda-8.00`

  Section 1.7.6 shows a typical ftp session. These public releases are FREE of charge but do not include support services unless you are under warranty or have purchased a support contract from SIGMET. Contact support@sigmet.com if you need to arrange a support contract.

- **CDROM Upgrades**- these are provided as part of a support contract or upon request.

1.7.5 How should I upgrade

There are two basic upgrade techniques:

- **Upgrade using "install" utility**- This is the preferred technique since it leaves all configuration files intact. This is described in **Section 1.7.7**.

- **CD–ROM Operating System Upgrade & SIGMET software Full Re-Install**- Backup your configuration files and network files and then do an install from scratch as described in **Appendix A**. This is the preferred technique **only** when it required that you upgrade the operating system which might be necessary in the event of hardware or new hardware.

Once you have decided on either a network or CDROM upgrade, then proceed with the upgrade installation as described in the sections below.

1.7.6 Getting the network upgrade files

There are two ways to get the network upgrade files. Both techniques use ftp to get the files from ftp.sigmet.com. The ftp client service is enabled on your computer allowing you to run an ftp session and "get" files from another networked computer:

- **Use your IRIS workstation/RVP8/RCP8/RCW to get the files-** a one-step procedure that requires that you have internet access.
- **Use another machine on the network–** a two-step procedure to first ftp the files from SIGMET to another computer, and then "get" them from this machine by running ftp on the IRIS workstation/RVP8/RCP8/RCW. Note that an alternative is to copy the files to a CDROM.

No matter what technique you use, you will need some basic familiarity with ftp. Here we will assume that our computer with the name **sigcomp** has direct internet access (the one-step procedure). The two-step procedure is analogous.

The one-step approach: Direct download

- On your IRIS Workstation/RVP8/RCP8/RCW create a directory called /tmp/iris–X.XX. If you will be downloading RDA software also, use a separate directory named /tmp/rda–X.XX. Here the X.XX is the version number of the RDA software that you want to install. This naming convention makes it easy to identify the version of the install files. You should make a directory for each version and type (IRIS/RDA) of software that you download. As **operator** type (assuming version 8.00 for all examples here):

  ```
  $ cd /
  $ mkdir /tmp/rda–8.00
  ```

- Position yourself in the /emp directory by typing:

  ```
  $ cd /tmp/rda–8.00
  ```

Note that on a windows machine, all of the commands below can be typed in the MS–DOS prompt window (remember to use the "\" backslash for DOS).

- Start ftp and follow the sample session below (your responses are indicated by **bold**)

  ```
  $ ftp ftp.sigmet.com
  Connected to ftp.sigmet.com
  220 Welcome to SIGMET
  Name: anonymous
  331 Guest login ok, send your complete e–mail address as password.
  Password: <Use your email address>
  230 Guest login ok, access restrictions apply.
  ftp> cd outgoing/releases/
  ```

```
ftp> dir
```

- You will see a listing of the directories on our web site. You are looking for an "rda" release, then:

```
ftp> cd rda
ftp> dir
```

- You will see a directory listing of available releases. You are looking for an 8.00 release, then:

```
ftp> cd 8.00
```

- You will then need to enter the appropriate directory for your OS version:

```
ftp> cd RHEL3.0
ftp> dir
```

- Now you will see a list of files with file sizes. The **required** files for an installation are (use only these files if you are using a low speed link):

```
app.gz
install
install.gz
install.rf
instiris
```

- Make a list of the files that you want to download including at least the five files in the list above, and then prepare to download by selecting BINARY file transfer: The other files on the FTP site (not shown here nor required for an installation) are default files for configuration and/or development.

```
ftp> bin
200 Type set to I.
```

Important: If you do not specify BINARY transfer, the download will not work.

- Now "get" the files, for example:

```
ftp> get app.z
200 PORT command successful.
150 Opening BINARY mode data connection for app.gz (4897560 bytes)
226 Transfer complete.
```

You will get a confirmation that BINARY transfer is being used and the size of the file in bytes is displayed. Depending on the size of the file and the speed of your connection, the download could take many minutes. When the file transfer is completed, you will get a confirmation message.

You can also use the multiple get command "**mget**" to get all of the files:

```
ftp> mget *
```

You will be prompted for each file download so you can still pick-and-choose by typing "**y**" or "**n**" to select (yes or no).

- After you have downloaded all of the files, then end your session by typing:

```
ftp> quit
221 Goodbye
$ (back to UNIX prompt)
```

For the one-step approach, you have all of the files that you need in the directory /tmp/rda–8.00 on the RVP8.

Completing the two-step approach using another computer

The two step approach is to use another computer to get the upgrade files and then get these files on the RVP8. The first step of getting the files from SIGMET is done analogus to the one-step approach described above. The second step is to ftp the files from the other computer to the RVP8. This is also analogous to the procedure described above. An alternative is to put the files on a CDROM, mount the CDROM on the RVP8 and then copy the files to the RVP8.

The end result of all these various procedures is that the upgrade files are on the RVP8 in a directory called /tmp/rda–X.XX. N

Set the modes on the files

Become root using the **su** command and password. Go to the RVP8 directory where the files were downloaded and change the mode on two of the files that require execute privilege, i.e.,

```
# cd /tmp/rda–8.00
# chmod +x install
# chmod +x instiris
```

You are now ready to move on to the next section.

1.7.7 Performing an Upgrade Installation

Follow these instructions if you are upgrading from a previous SIGMET system.

Login to the system as ’operator’ and start the X windows environment if it is not already running. Become the superuser by using the ’su’ command and supplying the appropriate password (Note network installations do not use root–see section 1.9.2).

⚠ **Warning: An upgrade installation *overwrites* any existing SIGMET files in the `${IRIS_ROOT}/bin` tree. Backup any important files before proceeding.**

HP–UX SYSTEMS:

```
# cd /cdrom/sigmet/iris/hp–ux
```

LINUX SYSTEMS (IRIS):

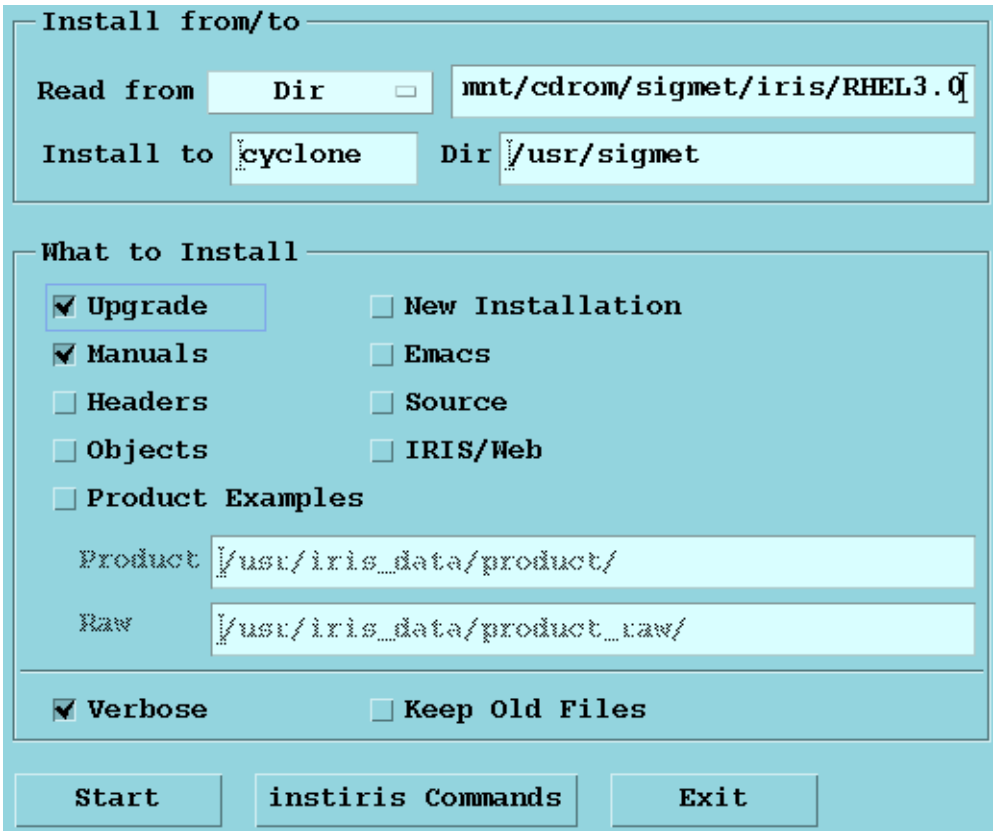
```
# cd /mnt/cdrom/sigmet/iris/RHEL3.0
```

LINUX SYSTEMS (RDA):

```
# cd /mnt/cdrom/sigmet/rda/RHEL3.0
```

ALL SYSTEMS:

```
# ./install
```



In the **`What to Install`** section of the **install** utility window, you must select **`Upgrade`** by pressing its button in. This is shown below:

You may optionally also select other optional SIGMET support software packages to install by pressing their buttons in the **`What to Install`** section. When ready to do the install, just press **`Start`**.

Note: If you are performing a RDA and IRIS install, first install the IRIS software. Then install the RDA software with the Keep Old Files button pressed.

The installation script takes a few minutes to complete and prints progress messages as it runs. Don't forget to unmount the **CDROM** after the installation is complete.

```
# cd /
```

HP–UX SYSTEMS:

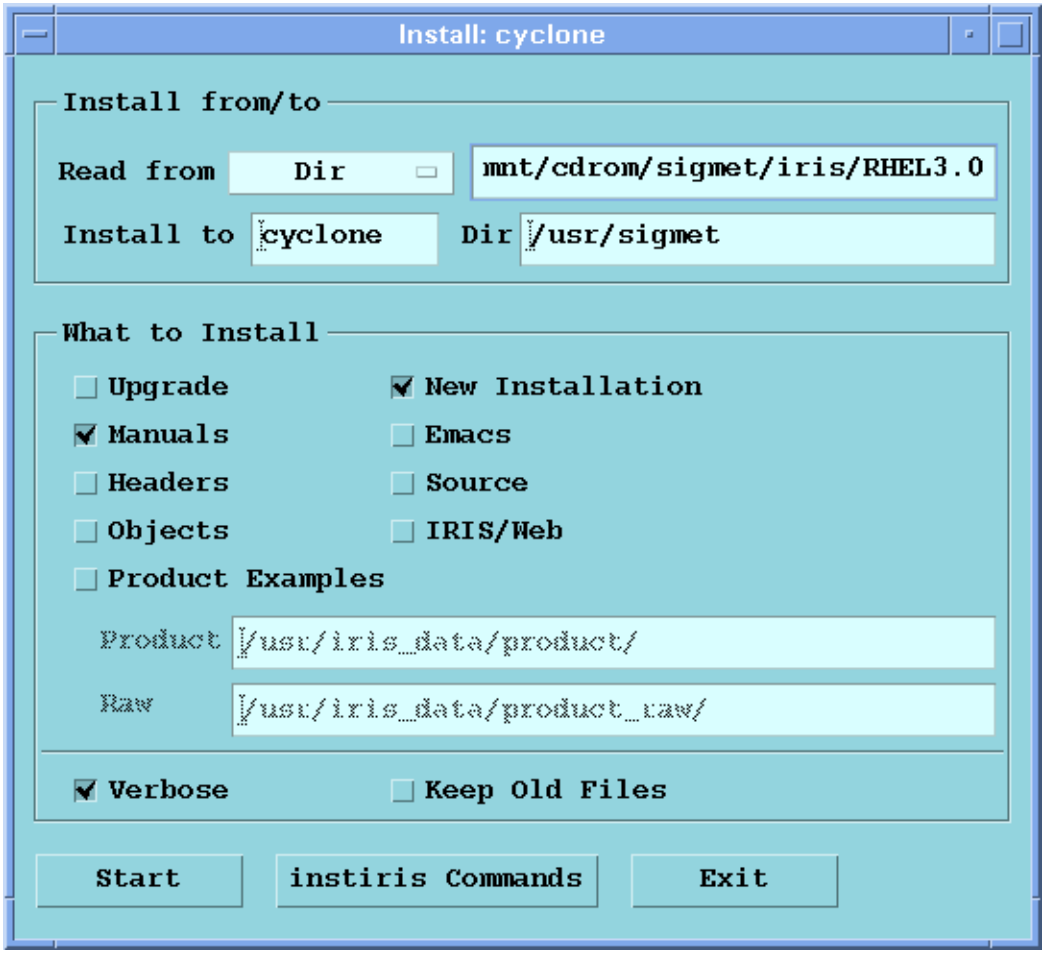
```
# umount /cdrom
```

LINUX SYSTEMS:

```
# eject /mnt/cdrom
```

1.8 Install Utility Window Options

This section describes in detail all the options available in the SIGMET **`install`** utility. It is located on the distribution cdrom, as well as in the **`${IRIS_ROOT}/install`** directory after installation.



The first section is labeled **`Install from/to`**. The default information in this section is shown above. This default information assumes that the installation will be from the directory from which the installation was run, and will be installing to the default directory (/usr/sigmet) in the local computer. In section 1.8 below, other options for **`Install from/to`** will be discussed.

The second section is labeled **`What to Install`**. In this section, you choose which of many different SIGMET files to install. Depending on which button(s) are pressed, there are three basic different installation that can be performed:

1. A New installation of the SIGMET software.
2. An Upgrade installation on a system where an older version of SIGMET software is already installed.

3. Do not install the SIGMET application software, but install one or more different optional support software packages such as **`Manuals`**, **`Emacs`** Editor, IRIS **`Header`** Files, a subset of the IRIS **`Source`** Code, complete IRIS **`Object`** Code, or IRIS **`Product Examples`**. The meaning of each of the optional support software packages is described in section 1.8 below.

It should also be noted that when doing either a New or Upgrade installation, you can also install at the same time one or more of the optional support software packages discussed in item 3 above.

1.8.1 "Read From" Option

The **install** utility allows for the files used in the installation to come from one of three different locations. The default location is the **`DIR`** this means that the installation program will automatically populate the Read From option with the path where the install command was run (this implicitly works for CDROMs). However, by using the **`Read From`** button, you can choose to access the files also from either the files already installed on your system (**`Local`** option).

When selecting the **`DIR`** option, you can type a directory in manually.

When selecting the **`Local`** option, this implies that the files will be taken from the local installation of IRIS or RDA already in this computer and copied to some other destination. In this case in the box to the right of the **`Read From`** button, you must fill in the location of the root directory of the IRIS software on your local system. By default this is the translation of the **`IRIS_ROOT`** environment variable, but you can override this if you wish.

When selection the **`Dir`** option, this implies that the files will be taken from some temporary directory on the local computer and copied to some other destination. In this case in the box to the right of the **`Read From`** button which is automatically populated with the directory where ./install was executed from.

1.8.2 Install To Option (Including Network Installs)

The **`Install To`** fields must be entered in the **install** window. These fields choose where the IRIS software will be installed. By default, these fields will always be filled in in such a way that the IRIS software will be installed to the local system in the default **`/usr/sigmet`** directory.

However, with the **`Install To`** fields, you can choose to install the SIGMET software to any other computer on your network, or to any directory on your computer.

To install to a remote computer **`(Network Install)`** one important step is that the **install** utility must be run from the as operator rather than from root. Thus the command would be:

```
$ cd /usr/sigmet/install
$ ./install
```

In the **install** utility window, enter the nodename of the remote computer that you are installing to in the **`Install To`** field. And if you wish to install to a directory that is different from **`/usr/sigmet,`** enter this directory into the **`Dir`** box immediately to the right of the **`Install To`** box.

These network installations are very convenient, as if you update a single system on your network from a CDROM by choosing all of the default options in the **`Install from/to`** section of the **install** utility, you can then install from the Local directory of this first machine (Local option) to every other machine on your network by putting their hostnames, one at at time, into the Install To field. This is a big time saving operation.

After the SIGMET software has been installed on the remote machine, you must complete the procedure by setting the ownerships, modes, and application defaults on the remote machine itself. This can be done by way of a remote login (as root), as in:

```
# rlogin Remote_Machine –l root
# cd /usr/sigmet/install
# ./instiris –setown –v
```

1.8.3 Manuals Option

To install SIGMET's Online Manuals, prior to starting the install, press the **`Manuals`** button in in the **`What to Install`** section of the **install** utility.

This installs both the online manuals, and the viewer program. It uses approximately 38 Mbytes of disk space. Generally the online manuals are more up to date than the printed versions.

1.8.4 EMACS Text Editor Option

If you would like to install a free copy of GNU EMACS, prior to starting the install, press the **`Emacs`** button in in the **`What to Install`** section of the Install utility.

For EMACS installation you must manually create the directory **/usr/local/emacs** if it does not already exist. EMACS is hard coded to search for libraries rooted at **/usr/local/emacs**. This directory cannot be moved. However, the executable binary **/usr/local/bin/emacs** may be moved freely to any convenient location in your **PATH**.

---

i **Note: EMACS is licensed for unlimited distribution by the Free Software Foundation, Cambridge, Massachusetts, USA. You may continue to redistribute it freely but may neither charge for its use, nor profit from its distribution. Users manuals are helpful and may be ordered from the manufacturer.**

1.8.5 Headers, Source, and Objects

If you would like to install IRIS **Header** files, a subset of the IRIS **Source** Code, or the complete IRIS **Object** Code and Libraries, prior to starting the install, press the **Header**, **Source**, and/or **Objects** button(s) in in the **What to Install** section of the Install Utility.

These options may be used separately or together, and result in the building of sub–directories from your IRIS root:

/usr/sigmet/libs	Common libraries
/usr/sigmet/utils	IRIS utilities
/usr/sigmet/iris	IRIS Client/Server
/usr/sigmet/include	Header files

The Headers option brings in the following ASCII files:

- Header files that define all structures and constants used by IRIS. These are essential for programmers who wish to use IRIS data directly.
- Makefiles to rebuild each section of IRIS.

The Objects option brings in the following machine generated files:

- Compiled object files for all of IRIS. These are necessary in order to use incremental field patches that you may receive from time to time.
- Debugable executable versions of all IRIS modules.

Lastly the Source option, install some program source examples (located in /usr/sigmet/utils/examples) and any additional source code that you have purchased as part of a special agreement with SIGMET.

1.8.6 Product Example file option

When using a SIGMET release cdrom, it is possible to install example **Product and Raw Product** files. To do so, press the **Product Examples** button prior to clicking **Start**. Also, you must fill in the Product and the Raw fields next to the

Product Example button. These fields are automatically filled in with default installation directories for IRIS Products and IRIS Raw Products. If you use different directories for these products, edit these fields prior to starting the install.

This will place a number of IRIS type product files in the above mentioned directories. These files can either be viewed directory, or may be re–ingested to make other products.

1.8.7 Verbose Option

Normally, **install** does its work silently, and prints only minimal progress messages as it runs. To see more details, press the **`Verbose`** button prior to clicking **`Start`**.

1.8.8 Keep Old Files Option

This option should only be used when upgrading or installing a computer system that has both RDA and IRIS software installed. When using this option, make sure that to install IRIS first and then install the RDA software with this button pressed.

1.9 Troubleshooting

This section contains suggestions for fixing common problems.

1.9.1 File Ownership and Protection

Sometimes there are problems after an installation with access to some of the SIGMET files. Typically, this is evidenced by an error message saying that the user does not have privilege to do an operation. This can happen when starting a program or when calibration files are accessed. If you should have this problem, run the **instiris** script, as shown below:

```
# instiris –setown
```

This procedure goes through the **/usr/sigmet** directory tree, changing the owner of all files to **operator** and setting the protection, as follows:

- Directories — **rwxrwxr–x**
- All files, except executable files — **rw–rw–r––**
- Executable files — **rwsrwsr–x**

Always use **instiris –setown** to fix the protection of your SIGMEt files. Do not try to change the protection of these files yourself.

1.9.2 Authorizing remote X–Windows on Your Node

To allow IRIS systems running on other nodes to send output to your screen, enter the command:

```
$ xhost +
```

This command must be done each time you log onto the system, and it cannot be put into a **.profile** file. If you want to authorize windows on your node without first issuing the **xhost +** command, you can create a file called **/etc/X0.hosts** and list the names of each of the nodes in the IRIS system. For example, if the system contains three nodes named **node1**, **node2**, and **node3**, **/etc/X0.hosts** would contain three entries, as follows:

```
node1
node2
node3
```

1.10 Basics of Login, Logout and Shutdown

1.10.1 Power up procedure

When you power the computer on, the system will go through an automatic startup of the operating system at the end of which the SIGMET software starts and performs power–up self tests. This is described in detail in of both the ***RVP8 & RCP8 User's Manuals*.**

If you are not doing any diagnostic or software maintenance work on the system, there is no need to log-in after power-up; simply turn the unit on and your application software will take over.

1.10.2 Local and remote login

There are two ways to login to a SIGMET system:

- **Local login:** The local keyboard, mouse and monitor can be used or via a KVM.
- **Remote login:** If **telnet** is enabled you can use this for remote access. Check with your network administrator.

For the remainder of this discussion it is assumed that local login is used.

1.10.3 Default operator and root login passwords

There are two default users defined in the standard software installation:

- "**root**" with password "**xxxxxxxx**" (8 lower case x's). This is for Operating System maintenance functions.
- "**operator**"with password "**xxxxxx**" (6 lower case x's). This is for SIGMET application software maintenance functions. These are all described in detail in the *Utilities Manual*.

Your system administrator can change either of these passwords by using standard Linux password support.

1.10.4 Login procedure

Local login as operator after power–up

- Connect keyboard, mouse and monitor and then cycle power on the system to force a reboot. This causes Linux to recognize these devices on power-up.
- At the power-up "**login**" prompt type "**operator**" and press **Enter**. When prompted, provide the appropriate password (factory default is "xxxxxx", 6 lower case x's).

Logging-in as operator you will be taken to an X-Window screen.

- Right-click the mouse and select "**New Window**" to get a terminal window.

The top of the terminal window shows, for example:

```
operator on rvp8–1 : /home/operator
```

i.e., your user name, the node name of the system and the current directory path.

If you would like to have a terminal with a bigger font, you can type "sigterm".

Switching from operator to root login using "su"

The easiest way to switch to a root login for system administration work is to type the super user command "**su**" and then give the root password. The prompt will change from $ to # indicating that you are root.

Exiting "su" root login to return to operator

In an X–terminal where you have become the "super user (su)", simply type "exit" to return to operator. The prompt will change from # to $.

Local root login after power-up

To login as root after a power–up or after exiting X–Windows, simply type "**root**" and press **Enter** at the login prompt, then give the appropriate password (factory default is "xxxxxxxx", 8 lower case x's).

You will be in a full screen terminal. This is not as convenient as X–Windows since only one terminal can be displayed on the screen.

If you need a second full screen terminal type "Alt-F2". You can return to your original terminal by typing "Alt-F1". The other function keys can provide additional terminals.

1.10.5 Logout procedure

Logout from X-Windows

- Method 1: Right-click the mouse and select "**Exit**".
- Method 2: Simultaneously press "**Ctrl-Alt-Backspace**" on the keyboard.

You will be logged out and the screen will show the initial login prompt. on the full screen terminal.

Root Logout from full screen terminal

If you logged–in as root from the power-up full screen terminal,

1.10.6 Poweroff shutdown procedure

If you need to swap PCI cards in the chassis, you must first do a poweroff shutdown as described here.

As either operator or root type "**poweroff**". The system will go through a shutdown sequence. When it is done "**Power down**" will be displayed. At this point you can press the power switch located on the right lower front of the chassis.

1.11 RVP8 & RCP8 (RDA) Software Installation

1.11.1 Install the upgraded kernal module

After you upgrade you may get an error message saying that there is a kernel module mismatch. This is easily fixed by rebooting. As an alternative you can restart the kernel module with the following commands:

```
# rdasys stop
# rdasys start
```

1.11.2 Installation steps to "flash" FPGA's in SIGMET devices

In this section you will be installing FPGA software into each of the SIGMET components. You will need to make an inventory of what is in your system and then issue an "**rdaflash**" command to each one as described below:

- Login as "**operator**" with password "**xxxxxx**" (6 x's)
- You will enter X–Windows. Right click the mouse and start a terminal window.
- For each SIGMET component (*italic*) type the appropriate command (**bold**):

SIGMET Component	Unit ID	If RVP8, type:
Standard RVP8/Rx Card	*–0*	**rdaflash –program rvp8rx–0**
Standard I/O–62 Card	*–0*	**rdaflash –program io62–0**
Standard Connector Panel	*–0*	**rdaflash –program io62cp–0**
Optional RVP8/Tx Card	*–0*	**rdaflash –program rvp8tx–0**
Optional 2nd RVP8/Rx	*–1*	**rdaflash –program rvp8rx–1**
		If RCP8, type:
Standard I/O–62 Card	*–0*	**rdaflash –program io62–0**
Standard Connector Panel	*–0*	**rdaflash –program io62cp–0**

- Do a system shutdown by typing "**poweroff**"
- When "**Power down**" is displayed, turn power off with power switch on lower right of front panel.

This completes the FPGA software installation.

1.11.3 Reboot power-up check and RDA diagnostics

After you have completed the installation steps above, you should reboot the unit. You can observe the progress of the reboot on the monitor. In addition, the front panel LED display will show the time of the reboot and display diagnostic messages. Typically about 1 minute is required for the system to complete reboot.

After the reboot is completed, login as operator, right-click to start a terminal window, and then:

- Stop the rvp8 (or rcp8) process by typing:

  $ killall rvp8 *(or* **rcp8***)*

- Run the following diagnostics and observe the results:

```
(for RVP8 and RCP8 systems)
$ rdadiags io62-0          tests I/O-62
$ rdadiags io62cp-0        tests connector panel.
                           Requires test cable

(for RVP8 systems only)
$ rdadiags rvp8rx-0
```

  Run these also for any optional RVP8 cards such as:

```
$ rdadiags rvp8tx-0        tests RVP8/Tx
$ rdadiags rvp8rx-1        tests 2nd RVP8/Rx
```

- Restart the RVP8 or RCP8 process by typing (for the RVP8 example):

  $ rvp8 & or rcp8 &

- Verify that the restart messages show no faults.

1.12 RDA Software Configuration

After the receiving your unit from the factory, or after software re–installation, there are several configuration steps required to customize your system for your particular environment and application. The configuration tools available for this are summarized in the table below.

Configuration Tool	RDA Device	Description of Configuration Features
setup/RVP utility **`setup_dsp.conf`**	RVP8	Configures the local environment required to run RVP8 the support utilities such as **ascope** and **dspx**. Examples include radar equation parameters that are required for calibration, pulse width definitions and PRF request limits.
setup/RCP utility **`setup_ant.conf`**	RCP8	Configures the local environment required to run the RCP8 support utilities that such as **antenna** or **bitex**. Examples include, max allowed AZ/EL velocity request, MIN and MAX elevation angles that can be requested and LAT/LON of radar for sun tracking.
RVP8 TTY setups **`rvp8.conf`**	RVP8	Defines the details of the sampling and processing algorithms as well as the operational configuration of the system. Examples include, IF filter design and selection, PRF limits, relative trigger timing, dual polarization features.
RCP8 TTY setups **`rcp8.conf`**	RCP8	Used to configure which status and control bits are available and define the antenna servo control parameters. Examples include, physical or virtual tachometer selection, shutdown safety criteria and internal antenna simulator on/off.
`softplane.conf`	RVP8	File that is edited which defines the various I/O signals on the I/O-62 connector panel, pin-by-pin. For example, whether a line is an input or output, electrical spec such as RS422 or TTL, what local variable name is associated with each line.
	RCP8	

Important: Both the setup utility and the TTY setups must be configured to customize your system. This is part of the installation procedure.

All of the configuration results are stored as ASCII text ".conf" files, typically in a directory called **/usr/sigmet/** (factory default). The file names are indicated in the table above. Each file has a factory default configuration file that is stored in the template directory (the default is **/usr/sigmet/config_template/init/**). An advantage of this approach is that for a radar network with identical hardware, configuration maintenance can be performed by copying pre-tested files over the network.

The following serve as references and are not repeated here:

setup utility	**IRIS Utilities Manual**
RVP8 TTY setups	**RVP8 User's Manual**
RCP8 TTY setups	**RCP8 User's Manual**

An overview of these setups for both the RVP8 and RCP8 is provided in the next two sections, followed by a description of the softplane.conf file and its configuration.

1.12.1 RCP8 Setup Configuration Summary

The table below summarizes the setups required for the RCP8 and its associated host computer (if any). The three cases are summarized in **"System Network Architecture" section of Chapter 1** of the ***RCP8 User's Manual.***

	Case 1: **Serial Interface**	**Case 2:** **Combined RCP8/RCW**	**Case 3:** **Socket Interface**
RCP8 TTY Setups: Site Host section			
Serial port	/dev/ttyS0 (e.g.)	.../fifo_hostio–x See note	.../fifo_hostio–x See note
Baud rate for host computer I/O	9600	*–Ignored–*	*–Ignored–*
RCP8 Setup: RCP Section			
System has an antenna	Yes	Yes	Yes
Interface Type	Native	Serial	Serial
Main channel device name	Not displayed for interface type Native	.../fifo_hostio–y See note	.../fifo_hostio–y See note
running at		*–Ignored–*	*–Ignored–*
with parity		*–Ignored–*	*–Ignored–*
Antenna angle insertion source	Native RCP8	Normal serial	Normal serial
AntExport Running on RCP8	***No***	***No***	***Yes***
Host RCW Setup Utility: RCP (e.g., running IRIS/Radar)			
System has an antenna	Yes	NA	
Interface Type	Serial		
Main channel device name	/dev/ttyS0/ (e.g.)		
running at	9600		
with parity	No		
Antenna angle insertion source	Normal Serial		

Note: The recommended full path for the fifo interface is **/usr/sigmet/config/fifo_hostio–x**. The other one (it does not matter which) is at **/usr/sigmet/config/fifo_hostio–y**.

For Case 3: AntExport Receiving Workstation	
System has an antenna	Yes
Interface Type	Socket
AntExport hostname/IP-Address	*<for source node running AntExport>*
AntExport Port Number	30745

1.12.2 RVP8 Setup Configuration Summary

The table below summarizes the setups required for the RVP8 and its connection to the associated host radar control workstation (RCW). Please refer to the "System Network Architecture" section of **Chapter 1**, ***RVP8 User's Manual.***

Table 1–1: RVP8 Local Setups: TTY Setups and Setup Utility

RVP8 TTY Setups		
"Mc" Live angle input	**Antenna angles from RVP8 Az/El TTL inputs**	2: Tags
	Antenna angles from RVP8 Az/El S/D inputs	3: S/D
	Antenna angles from other source (e.g., serial input)	1: None
"Mt N" Range Mask spacing	These are all forced by the corresponding settings in the **RVP8 Setup Utility: RVP Section** (See Below)	
"Mt N" Maximum number of Pulses/Sec		
"Mt N" Maximum instantaneous PRF		
RVP8 Setup Utility: RVP Section		
System has a signal processor	Yes	
Interface Type	Native	
Range Mask spacing	Default is 125 m	Configuring these fields will also set the corresponding fields in the RVP8 TTY Setups (See above)
Minimum PRF	As required	
Maximum PRF	As required for max duty cycle	
RVP8 Setup Utility: RCP Section		
Response to Mc Live angle input (above)	***TTL binary angles or S/D***	***"None"***
System has an antenna	Yes	Yes
Interface Type	Not applicable when Antenna angle insertion source is set to RVP8	Serial
Main channel device name		Default: /dev/ttyS0
running at		Default: 9600
with parity		None
Antenna angle insertion source	Native RVP8	Normal serial
DspExport Running on RVP8	***Yes***	***Yes***

Table 1–2: Setup Utility on RCW (Radar Control Workstation) running IRIS

IRIS Host RCW Setup Utility: RVP Section

System has a signal processor	Yes
Interface Type	Socket
DspExport hostname/IP–Address	*<hostname or IP address of RVP8>*
DspExport Port Number	Default: 30740

IRIS Host RCW Setup Utility: Ingest Section

Response to Mc Live angle input (above)	***TTL binary angles or S/D***	***"None"***
Manner of angle acquisition	Binary TAGS	Serial Stream

1.12.3 Configuring the softplane.conf file

What is the softplane.conf file ?

The softplane.conf file is used to define pin–by–pin assignment of I/O functions to various connectors on the I/O-62 connector panel. It is a plain text ASCII file that is self-documented. Since the RVP8 and RCP8 have virtually no jumpers, or wirewrap, all I/O configuration on the I/O-62 connector panel is done by software approach according to this file.

Where is softplane.conf ?

The file resides in the IRIS_CONFIG directory. Typically this is **/usr/sigmet/config** (this is the default directory that is factory installed). The factory configurations are also available in the **/usr/sigmet/template/init** directory so that you can always return to the factory defaults if you need to.

When do I need to change softplane.conf ?

The softplane.conf file that is shipped with your system is configured for a standard connector panel with I/O as described in the *RCP8 User's Manual*. As long as you use the standard I/O pin assignments, then you do not need to change softplane.conf.

If you need to redefine some of the I/O pins on the connector panel, or add additional SIGMET cards such as a second I/O-62 then you must change softplane.conf.

Editing softplane.conf

You will need to use a text editor to modify the softplane.conf file. There are two editors included in the system:

- **vi** The generic UNIX editor that is available on every UNIX system. It is really arcane to use, but many people know how to use it out of necessity or they are simply used to it now.
- **emacs**: This is more user friendly with keyboard commands and mouse support when you are in X–Windows so it a little easier to learn than **vi**.

If you are not familiar with either of these, then you will need to either find someone who is or learn how to use these tools.

To start an editing session you would do the following as **operator**:

```
$ cd /usr/sigmet/config
$ vi softplane.conf    or     emacs softplane.conf
```

softplane.conf file: RVP8 example

An example from the beginning and some excerpts from the softplane.conf file are shown below (note that the command "**cat**" causes the file to be listed on a terminal):

```
$ cat /usr/sigmet/config/softplane.conf
# *********************************
# *                               *
# *  Softplane Configuration File  *
# *                               *
# *********************************

# The following general purpose control and status signals
# can be routed to/from any available hardware pin.  The '~'
# prefix character may be used for signal inversion.
#
#         Control Outputs            Status Inputs
#        -----------------          ---------------
#            cPedAZ[15:0]             sPedAZ[15:0]
#            cPedEL[15:0]             sPedEL[15:0]
#          cEarthAZ[15:0]                sServoPwr
#          cEarthEL[15:0]                   sLocal
#               cServoPwr                 sStandby
#           cCabinetRelay                 sLowerEL
#            cTransmitPwr                 sUpperEL
#            cPWidth[3:0]              sTransmitPwr
#              cTrigBlank             sTransmitLocal
#              cRadiateOn               sPWidth[3:0]
#             cRadiateOff                 sTrigBlank
#                  cReset                   sRadiate
#          cIrisMode[2:0]               sAirflowFlt
#              cAux[63:0]                sWavegpFlt
#                    true             sInterlockFlt
#                   false            sMagCurrentFlt
#                                            sReset
#                                    sIrisMode[2:0]
#                                       sAux[319:0]

splConfig.sVersion = "7.32"

# ------------------- IO62 Slot #0 -------------------
#
splConfig.Io62[0].lInUse = 1

# The remote backpanel type should be one of the following:
#     Direct : Direct I/O with IO62 connector itself
#     IO62CP : Standard IO62-CP connector panel
#     RVP88D : RVP8 portion of WSR88D panel
#     RCP88D : RCP8 portion of WSR88D panel
#
splConfig.Io62[1].sExtPanel = "IO62CP"

# TTL/CMOS on J1
#
splConfig.Io62[0].Opt.Cp.J1.pin01 = "sPedAZ[0]"
splConfig.Io62[0].Opt.Cp.J1.pin02 = "sPedAZ[1]"
splConfig.Io62[0].Opt.Cp.J1.pin03 = "sPedAZ[2]"
...
```

```
# Relays and relay drivers on J6
#
splConfig.Io62[0].Opt.Cp.J6_IntRelay1 = ””
splConfig.Io62[0].Opt.Cp.J6_IntRelay2 = ””
splConfig.Io62[0].Opt.Cp.J6_IntRelay3 = ””

splConfig.Io62[0].Opt.Cp.J6_ExtRelay1 = ””
splConfig.Io62[0].Opt.Cp.J6_ExtRelay2 = ””
splConfig.Io62[0].Opt.Cp.J6_ExtRelay3 = ””
splConfig.Io62[0].Opt.Cp.J6_ExtRelay4 = ””

# BNC testpoint monitors
#
splConfig.Io62[0].Opt.Cp.J13_BNC = ””
splConfig.Io62[0].Opt.Cp.J16_BNC = ””

# BNC trigger drivers direct from IO62 PCI card.
# Special signals ’trigger[8:1]’ may also be used here.
#
splConfig.Io62[0].Opt.Cp.J14_BNC = ”trigger[1]”
splConfig.Io62[0].Opt.Cp.J15_BNC = ”trigger[2]”
splConfig.Io62[0].Opt.Cp.J17_BNC = ”trigger[3]”
splConfig.Io62[0].Opt.Cp.J18_BNC = ”trigger[4]”

# RS232 TTY transmitters from IO62
#
splConfig.Io62[0].Opt.Cp.TTY0_Tx = ””
splConfig.Io62[0].Opt.Cp.TTY1_Tx = ””

# ------------------- IO62 Slot #1 -------------------
#
splConfig.Io62[1].lInUse = 0

# ------------------- IO62 Slot #2 -------------------
#
splConfig.Io62[2].lInUse = 0
....
#     <End of softplane definitions>
```

softplane.conf file: RCP8 example

An example from the beginning and some excerpts from the softplane.conf file are shown below (note that the command "**cat**" causes the file to be listed on a terminal):

```
# **********************************
# *                                *
# *  Softplane Configuration File  *
# *                                *
# **********************************
# The following general purpose control and status signals
# can be routed to/from any available hardware pin.  The '~'
# prefix character may be used for signal inversion.
#
#         Control Outputs            Status Inputs
#        -----------------          ---------------
#            cPedAZ[15:0]             sPedAZ[15:0]
#            cPedEL[15:0]             sPedEL[15:0]
#          cEarthAZ[15:0]                sServoPwr
#          cEarthEL[15:0]                   sLocal
#               cServoPwr                 sStandby
#           cCabinetRelay                 sLowerEL
#            cTransmitPwr                 sUpperEL
#            cPWidth[3:0]              sTransmitPwr
#              cTrigBlank             sTransmitLocal
#              cRadiateOn               sPWidth[3:0]
#             cRadiateOff                 sTrigBlank
#                  cReset                   sRadiate
#          cIrisMode[2:0]               sAirflowFlt
#              cAux[80:0]                sWavegpFlt
#                    true             sInterlockFlt
#                   false            sMagCurrentFlt
#                                             sReset
#                                    sIrisMode[2:0]
#
sAux[319:0]splConfig.sVersion = "8.00"
# ------------------- IO62 Slot #0 -------------------
#
splConfig.Io62[0].lInUse = 1# The remote backpanel type should be
one of the following:
#     Direct : Direct I/O with IO62 connector itself
#     IO62CP : Standard IO62-CP connector panel
#     RCP88D : RCP8 portion of WSR88D panel
#     RVP88D : RVP8 portion of WSR88D panel
#
splConfig.Io62[0].sExtPanel = "IO62CP"# TTL/CMOS on J1
#
splConfig.Io62[0].Opt.Cp.J1.pin01 = "sPedAZ[0]"
splConfig.Io62[0].Opt.Cp.J1.pin02 = "sPedAZ[1]"
splConfig.Io62[0].Opt.Cp.J1.pin03 = "sPedAZ[2]"
splConfig.Io62[0].Opt.Cp.J1.pin04 = "sPedAZ[3]"
splConfig.Io62[0].Opt.Cp.J1.pin05 = "sPedAZ[4]"
splConfig.Io62[0].Opt.Cp.J1.pin06 = "sPedAZ[5]"
splConfig.Io62[0].Opt.Cp.J1.pin07 = "sPedAZ[6]"
splConfig.Io62[0].Opt.Cp.J1.pin08 = "sPedAZ[7]"
splConfig.Io62[0].Opt.Cp.J1.pin09 = "sPedAZ[8]"
splConfig.Io62[0].Opt.Cp.J1.pin10 = "sPedAZ[9]"
splConfig.Io62[0].Opt.Cp.J1.pin11 = "sPedAZ[10]"
```

```
splConfig.Io62[0].Opt.Cp.J1.pin12 = ”sPedAZ[11]”
splConfig.Io62[0].Opt.Cp.J1.pin13 = ”sPedAZ[12]”
splConfig.Io62[0].Opt.Cp.J1.pin14 = ”sPedAZ[13]”
splConfig.Io62[0].Opt.Cp.J1.pin15 = ”sPedAZ[14]”
splConfig.Io62[0].Opt.Cp.J1.pin16 = ”sPedAZ[15]”
splConfig.Io62[0].Opt.Cp.J1.pin17 = ””
splConfig.Io62[0].Opt.Cp.J1.pin18 = ””
splConfig.Io62[0].Opt.Cp.J1.pin19 = ””
splConfig.Io62[0].Opt.Cp.J1.pin20 = ””# TTL/CMOS on J2
#
splConfig.Io62[0].Opt.Cp.J2.pin01 = ”cEarthAZ[0]”
splConfig.Io62[0].Opt.Cp.J2.pin02 = ”cEarthAZ[1]”
splConfig.Io62[0].Opt.Cp.J2.pin03 = ”cEarthAZ[2]”

. . .

splConfig.Io62[0].Opt.Cp.J2.pin15 = ”cEarthAZ[14]”
splConfig.Io62[0].Opt.Cp.J2.pin16 = ”cEarthAZ[15]”
splConfig.Io62[0].Opt.Cp.J2.pin17 = ””
splConfig.Io62[0].Opt.Cp.J2.pin18 = ””
splConfig.Io62[0].Opt.Cp.J2.pin19 = ””
splConfig.Io62[0].Opt.Cp.J2.pin20 = ””# TTL/CMOS on J4
#
splConfig.Io62[0].Opt.Cp.J4.pin01 = ”sPedEL[0]”
splConfig.Io62[0].Opt.Cp.J4.pin02 = ”sPedEL[1]”
splConfig.Io62[0].Opt.Cp.J4.pin03 = ”sPedEL[2]”

. . .

splConfig.Io62[0].Opt.Cp.J4.pin15 = ”sPedEL[14]”
splConfig.Io62[0].Opt.Cp.J4.pin16 = ”sPedEL[15]”
splConfig.Io62[0].Opt.Cp.J4.pin17 = ””
splConfig.Io62[0].Opt.Cp.J4.pin18 = ””
splConfig.Io62[0].Opt.Cp.J4.pin19 = ””
splConfig.Io62[0].Opt.Cp.J4.pin20 = ””# TTL/CMOS on J5
#
splConfig.Io62[0].Opt.Cp.J5.pin01 = ”cEarthEL[0]”
splConfig.Io62[0].Opt.Cp.J5.pin02 = ”cEarthEL[1]”
splConfig.Io62[0].Opt.Cp.J5.pin03 = ”cEarthEL[2]”

. . .

splConfig.Io62[0].Opt.Cp.J5.pin15 = ”cEarthEL[14]”
splConfig.Io62[0].Opt.Cp.J5.pin16 = ”cEarthEL[15]”
splConfig.Io62[0].Opt.Cp.J5.pin17 = ””
splConfig.Io62[0].Opt.Cp.J5.pin18 = ””
splConfig.Io62[0].Opt.Cp.J5.pin19 = ””
splConfig.Io62[0].Opt.Cp.J5.pin20 = ””# TTL/CMOS on J7

#
splConfig.Io62[0].Opt.Cp.J7.pin01 = ”sAux[0]”
splConfig.Io62[0].Opt.Cp.J7.pin02 = ”sAux[1]”
splConfig.Io62[0].Opt.Cp.J7.pin03 = ”sAux[2]”
splConfig.Io62[0].Opt.Cp.J7.pin04 = ”sAux[3]”
```

```
splConfig.Io62[0].Opt.Cp.J7.pin05 = ”sAux[4]”
splConfig.Io62[0].Opt.Cp.J7.pin06 = ”sAux[5]”
splConfig.Io62[0].Opt.Cp.J7.pin07 = ”sAux[6]”
splConfig.Io62[0].Opt.Cp.J7.pin08 = ”sAux[7]”
splConfig.Io62[0].Opt.Cp.J7.pin09 = ”sAux[8]”
splConfig.Io62[0].Opt.Cp.J7.pin10 = ”sAux[9]”
splConfig.Io62[0].Opt.Cp.J7.pin11 = ”sAux[10]”
splConfig.Io62[0].Opt.Cp.J7.pin12 = ”sAux[11]”
splConfig.Io62[0].Opt.Cp.J7.pin13 = ”sAux[12]”
splConfig.Io62[0].Opt.Cp.J7.pin14 = ”sAux[13]”
splConfig.Io62[0].Opt.Cp.J7.pin15 = ”sAux[14]”
splConfig.Io62[0].Opt.Cp.J7.pin16 = ”sAux[15]”
splConfig.Io62[0].Opt.Cp.J7.pin17 = ”sAux[16]”
splConfig.Io62[0].Opt.Cp.J7.pin18 = ”sAux[17]”
splConfig.Io62[0].Opt.Cp.J7.pin19 = ”sAux[18]”
splConfig.Io62[0].Opt.Cp.J7.pin20 = ”sAux[19]”
#
# Eight IO62 line pairs on J3
#
splConfig.Io62[0].Opt.Cp.J3_01_14.lRS422 = 0
splConfig.Io62[0].Opt.Cp.J3_01_14.iTerm = 0
splConfig.Io62[0].Opt.Cp.J3_01_14.pinPos = ”cPWidth[0]”
splConfig.Io62[0].Opt.Cp.J3_01_14.pinNeg = ”cPWidth[1]”splCon-
fig.Io62[0].Opt.Cp.J3_02_15.lRS422 = 0
splConfig.Io62[0].Opt.Cp.J3_02_15.iTerm = 0
splConfig.Io62[0].Opt.Cp.J3_02_15.pinPos = ”cRadiateOn”
splConfig.Io62[0].Opt.Cp.J3_02_15.pinNeg = ”cRadiateOff”splCon-
fig.Io62[0].Opt.Cp.J3_03_16.lRS422 = 0
splConfig.Io62[0].Opt.Cp.J3_03_16.iTerm = 0
splConfig.Io62[0].Opt.Cp.J3_03_16.pinPos = ”cServoPwr”
splConfig.Io62[0].Opt.Cp.J3_03_16.pinNeg = ”cTransmitPwr”splCon-
fig.Io62[0].Opt.Cp.J3_04_17.lRS422 = 0
splConfig.Io62[0].Opt.Cp.J3_04_17.iTerm = 0
splConfig.Io62[0].Opt.Cp.J3_04_17.pinPos = ”cReset”
splConfig.Io62[0].Opt.Cp.J3_04_17.pinNeg = ””splCon-
fig.Io62[0].Opt.Cp.J3_05_18.lRS422 = 0
splConfig.Io62[0].Opt.Cp.J3_05_18.iTerm = 1
splConfig.Io62[0].Opt.Cp.J3_05_18.pinPos = ”sPWidth[0]”
splConfig.Io62[0].Opt.Cp.J3_05_18.pinNeg = ”sPWidth[1]”splCon-
fig.Io62[0].Opt.Cp.J3_06_19.lRS422 = 0
splConfig.Io62[0].Opt.Cp.J3_06_19.iTerm = 1
splConfig.Io62[0].Opt.Cp.J3_06_19.pinPos = ”sRadiate”
splConfig.Io62[0].Opt.Cp.J3_06_19.pinNeg = ””splCon-
fig.Io62[0].Opt.Cp.J3_07_20.lRS422 = 0
splConfig.Io62[0].Opt.Cp.J3_07_20.iTerm = 1
splConfig.Io62[0].Opt.Cp.J3_07_20.pinPos = ”sServoPwr”
splConfig.Io62[0].Opt.Cp.J3_07_20.pinNeg = ”sTransmitPwr”splCon-
fig.Io62[0].Opt.Cp.J3_08_21.lRS422 = 0
splConfig.Io62[0].Opt.Cp.J3_08_21.iTerm = 1
splConfig.Io62[0].Opt.Cp.J3_08_21.pinPos = ”sReset”
splConfig.Io62[0].Opt.Cp.J3_08_21.pinNeg = ””
# Two RS-422 Tx/Rx chips on J3
#
splConfig.Io62[0].Opt.Cp.J3_09_22 = ””
splConfig.Io62[0].Opt.Cp.J3_10_23 = ””
# Seven IO62 line pairs on J9
#
splConfig.Io62[0].Opt.Cp.J9_01_14.lRS422 = 0
```

```
splConfig.Io62[0].Opt.Cp.J9_01_14.iTerm = 1
splConfig.Io62[0].Opt.Cp.J9_01_14.pinPos = ”sWavegpFlt”
splConfig.Io62[0].Opt.Cp.J9_01_14.pinNeg = ”sAirflowFlt”splCon-
fig.Io62[0].Opt.Cp.J9_02_15.lRS422 = 0
splConfig.Io62[0].Opt.Cp.J9_02_15.iTerm = 1
splConfig.Io62[0].Opt.Cp.J9_02_15.pinPos = ”sInterlockFlt”
splConfig.Io62[0].Opt.Cp.J9_02_15.pinNeg = ”sMagCurrentFlt”splCon-
fig.Io62[0].Opt.Cp.J9_03_16.lRS422 = 0
splConfig.Io62[0].Opt.Cp.J9_03_16.iTerm = 1
splConfig.Io62[0].Opt.Cp.J9_03_16.pinPos = ”sLocal”
splConfig.Io62[0].Opt.Cp.J9_03_16.pinNeg = ”sStandby”splCon-
fig.Io62[0].Opt.Cp.J9_04_17.lRS422 = 0
splConfig.Io62[0].Opt.Cp.J9_04_17.iTerm = 1
splConfig.Io62[0].Opt.Cp.J9_04_17.pinPos = ”sLowerEL”
splConfig.Io62[0].Opt.Cp.J9_04_17.pinNeg = ”sUpperEL”splCon-
fig.Io62[0].Opt.Cp.J9_05_18.lRS422 = 0
splConfig.Io62[0].Opt.Cp.J9_05_18.iTerm = 0
splConfig.Io62[0].Opt.Cp.J9_05_18.pinPos = ””
splConfig.Io62[0].Opt.Cp.J9_05_18.pinNeg = ””splCon-
fig.Io62[0].Opt.Cp.J9_06_19.lRS422 = 0
splConfig.Io62[0].Opt.Cp.J9_06_19.iTerm = 0
splConfig.Io62[0].Opt.Cp.J9_06_19.pinPos = ””
splConfig.Io62[0].Opt.Cp.J9_06_19.pinNeg = ””splCon-
fig.Io62[0].Opt.Cp.J9_07_20.lRS422 = 0
splConfig.Io62[0].Opt.Cp.J9_07_20.iTerm = 0
splConfig.Io62[0].Opt.Cp.J9_07_20.pinPos = ””
splConfig.Io62[0].Opt.Cp.J9_07_20.pinNeg = ””
# Relays and relay drivers on J6
#
splConfig.Io62[0].Opt.Cp.J6_IntRelay1 = ”cPWidth[0]”
splConfig.Io62[0].Opt.Cp.J6_IntRelay2 = ”cPWidth[1]”
splConfig.Io62[0].Opt.Cp.J6_IntRelay3 = ””splCon-
fig.Io62[0].Opt.Cp.J6_ExtRelay1 = ””
splConfig.Io62[0].Opt.Cp.J6_ExtRelay2 = ””
splConfig.Io62[0].Opt.Cp.J6_ExtRelay3 = ””
splConfig.Io62[0].Opt.Cp.J6_ExtRelay4 = ””
# BNC testpoint monitors
#
splConfig.Io62[0].Opt.Cp.J13_BNC = ””
splConfig.Io62[0].Opt.Cp.J16_BNC = ””
# BNC trigger drivers direct from IO62 PCI card.
# Special signals ’trigger[8:1]’ may also be used here.
#
splConfig.Io62[0].Opt.Cp.J14_BNC = ””
splConfig.Io62[0].Opt.Cp.J15_BNC = ””
splConfig.Io62[0].Opt.Cp.J17_BNC = ””
splConfig.Io62[0].Opt.Cp.J18_BNC = ””
# RS232 TTY transmitters/Receivers from IO62
#
splConfig.Io62[0].Opt.Cp.TTY0_Tx = ””
splConfig.Io62[0].Opt.Cp.TTY1_Tx = ””
splConfig.Io62[0].Opt.Cp.TTY0_Rx = ””
splConfig.Io62[0].Opt.Cp.TTY1_Rx = ””
# ------------------- IO62 Slot #1 -------------------
#
splConfig.Io62[1].lInUse = 0
# ------------------- IO62 Slot #2 -------------------
#
```

```
splConfig.Io62[2].lInUse = 0
# ------------------- IO62 Slot #3 -------------------
#
splConfig.Io62[3].lInUse = 0
```

softplane. conf organization and syntax

The softplane.conf file is used to define every I/O pin on every connector, on the PCI cards themselves and on the connector panel. There are two primary definitions that are made for each pin:

- **Physical interface-** the electrical properties (RS422 output, analog input, TTL output, etc.).
- **Logical interface-** The internal variable name that is associated with each pin.

With this in mind, we can describe the syntax of the file.

- "**#**" at the beginning of a line indicates a comment. These are used for internal documentation and if you make changes you should comment them, for example

```
# TTL I/O on J7
#
# Modification by REP on 2 Apr 03
# Added new interlock input on connector panel J7 pin07
...
```

- The top part of the file provides a list of internal variables names that are used to define the logical interface to the softplane. These are divided into status inputs (also called indicators) and control outputs (also called requests). For example, sPedAZ0 corresponds to the LSB of a digital azimuth angle relative to the antenna pedestal. The tables on the next page provide a summary of the available status and control variable names.

Important: This table is subject to change

- Each definition line in the file starts with the keyword text:

```
# splConfig...
```

- The first un-commented line of the file indicates the version of the IRIS support software that was last used to machine–generate the file. This is an information only field for traceability purposes and is thus not edited. In the example we have this shown as:

```
# splConfig.sVersion = ”7.32”
```

In any event, on the TTL connectors (J1, J2, J4, J5, J7) each of these connectors must be exclusively used for INPUT (s vars) or OUTPUT (c vars). You can not mix these on an individual connector.

Table 1–3: Summary of softplane.conf Status and Control Bits

Control Output	Meaning/Interpretation
cPedAZ[15:0]	16 bits of antenna azimuth angle relative to the pedestal (fixed base system)
cPedEL[15:0]	16 bits of antenna elevation angle relative to the pedestal (fixed base system)
cEarthAZ[15:0]	16 bits of antenna azimuth angle relative to the earth (moving platform)
cEarthEL[15:0]	16 bits of antenna elevation angle relative to the earth (moving platform)
cServoPwr	To control servo power on
cCabinetRelay	To control a relay signal
cTransmitPwr	Request transmit power on
cPWidth[3:0]	Request one of four pulse widths
cTrigBlank	Trigger blanking signal
cRadiateOn	Request radiate on
cRadiateOff	Request for radiate off
cReset	Request a reset of external equipment
cIrisMode[2:0]	Request the application software (e.g., IRIS) to switch to 1 of 8 operating modes.
cAux[63:0]	Arbitrarily assigned output requests
true	Internal logic variable
false	Internal logic variable

Status Input	Meaning/Interpretation
sPedAZ[15:0]	16 bits of antenna azimuth angle relative to the pedestal (fixed base system)
sPedEL[15:0]	16 bits of antenna elevation angle relative to the pedestal (fixed base system)
sServoPwr	Servo power on indicator
sLocal	Antenna local mode indicator, usually tied to an external local/remote switch.
sStandby	Radar ready to radiate indicator
sLowerEL	Lower limit switch indicator
sUpperEL	Upper limit switch indicator
sTransmitPwr	Transmitter cabinet power on indicator
sTransmitLocal	Transmitter local mode indicator, usually tied to an external local/remote switch.
sPWidth[3:0]	Indicator of the current pulse width
sTrigBlank	Indicator that trigger blanking is requested, usually from an external source.
sRadiate	Radiate on indicator
sAirflowFlt	Cooling airflow fault indicator
sWavegpFlt	Wave guide pressure fault indicator
sInterlockFlt	Master interlock fault indicator
sMagCurrentFlt	Transmitter overload fault indicator
sReset	Request for reset coming from external source
sIrisMode[2:0]	Information on which operating mode is active in the application software
sAux[319:0]	Arbitrary status indicators

- Next, each piece of hardware is identified as being either in use or not in use.

```
splConfig.Io62[0].InUse = 1  if in use
splConfig.Io62[0].InUse = 0  if unused or not installed
```

  Currently, the I/O-62 is the only I/O device supported by the softplane.

- The method of connecting to the I/O-62 is specified next, for example**:**

```
splConfig.Io62[0].sExtPanel = ”DIRECT”
```

  Currently, the options are:

Type of Connection	softplane Descriptor
Direct connect to I/O-62 via 62 pin connector	**DIRECT**
I/O-62 Connector Panel (used for RVP8 and RCP8)	**IO62CP**
WSR88D connector panel, RVP8 portion	**RVP88D**
WSR88D connector panel, RCP8 portion	**RCP88D**

- The assignments for each connector and each pin are then made. For convenience, these are usually grouped together by connector. For example let’s say that , Pin 1 of connector J1 on the I/O-62 connector panel is assigned to be the LSB of the input azimuth angle, i.e.,

```
# TTL/CMOS on J1
#
splConfig.Io62[0].Opt.Cp.J1.pin01 = ”sPedAZ[0]”
…
```

- The notation ”” indicates that no assignment is made.

```
# BNC testpoint monitors
#
splConfig.Io62[0].Opt.Cp.J13_BNC = ””
```

  In the example above the “pin name” is J13_BNC.

- Putting a ~ in front of a logic variable inverts the variable.

```
splConfig.Io62[0].Opt.Cp.J1.pin03 = ”~sPedAZ[2]”
```

Check in the /usr/sigmet/config_template/init directory for other examples of softplane configurations.

1.13 Testing, Backup, and Calibration

After software installation and before calibration, it is possible to check performance and detect errors in the installation. If DSP calibrations are off, radar data may not be available.

1.13.1 Ascope Test (RVP8 Installations ONLY)

Run the Ascope utility by typing **`ascope`**. This serves as an overall test of the signal processor. See the *Utilities Manual* for details. If the displayed PRF does not match what is requested, the processor type may be set incorrectly. Be careful with high PRFs because the pulse width control may not be working yet.

1.13.2 Antenna Test (RCP Installations ONLY)

Run the Antenna utility by typing **`antenna`**. It should be able to control the position and speed of the antenna as documented in the *Utilities Manual*.

1.13.3 IRIS Test (IRIS systems ONLY)

As a final test, run IRIS and schedule a simple task that moves the antenna, runs the signal processor, and generates a PPI product. Be sure to check for messages in the message menu. Normal startup should produce no messages.

1.13.4 Print Special Files

Because every system is customized on installation, it is desirable to save the work that was done at installation time, in case it needs to be repeated. Our recommendation is to print the Setup listing file generated by the **`all`** command and the Antsetup listing file.

1.13.5 Make a Full Backup

You should backup your system on a regular basis. This is the only way to restore your disk if data are lost. For HP systems follow the system backup procedures recommended by these manufacturers. In addition, the sigbru utility can be used for auxiliary backup. Linux customers can use the backup procedure recommended by Red Hat or can use the sigbru utility for complete backup and restore. Sigbru is described in Appendix D.

1.13.6 DSP Calibration (RVP Installations ONLY)

Below is a list of the calibrations for the signal processor in the suggested order. Next to each is the name of the utility program that can help perform the calibration. How to perform each calibration is described in the *Utilities Manual*.

1. Adjustment of I, Q and Log pots. (Ascope)
2. Calibration of the Log channel. (Zauto)
3. Adjustment of Gain Control pots. (Gaintest)
4. Calibration of the Gain Control signal. (Gaincal)
5. Selecting the STC waveform. (Stcwave)
6. Computing the AGC response table. (Agcal)