Sun/IRAF V2.11 Installation Guide

Doug Tody
Mike Fitzpatrick

IRAF Group
National Optical Astronomy Observatories†
August 1997

ABSTRACT

This document describes how to install IRAF V2.11 on a Sun workstation or server running either SunOS 4.1 or Solaris V2.4 (or later), or update an existing installation. Both standalone and networked, multiple architecture configurations are described. Only those issues which one must understand to install IRAF are discussed here; a companion document, *Sun IRAF Site Manager's Guide*, deals with other issues such as interfacing new devices, configuring the IRAF networking system, Sun/IRAF shared libraries, adding layered software, and so on.

August 30, 1997

†Operated by the Association of Universities for Research in Astronomy, Inc. under cooperative agreement with the National Science Foundation.

Contents

1. **Introduction** 1

2. **Installing Sun/IRAF** 1
   - 2.1. Prepare the root IRAF directory 2
     - 2.1.1. If updating an existing IRAF installation 2
     - 2.1.2. If installing IRAF for the first time 3
   - 2.2. Install the files 4
     - 2.2.1. Installing from a network distribution 4
     - 2.2.2. Installing from tape 4
     - 2.2.3. Installing from CD-ROM 5
     - 2.2.4. Configuring the BIN directories 6
   - 2.3. Merge local revisions back into the new system 6
   - 2.4. Run the INSTALL Script 7

3. **System Checkout** 8
   - 3.1. Frequently Asked Questions 9

Appendix A. A Complete Example 10

Sun/IRAF V2.11 Installation Guide

Doug Tody
Mike Fitzpatrick

IRAF Group
National Optical Astronomy Observatories†
August 1997

1. Introduction

Before installing Sun/IRAF, one must 1) obtain an appropriate Sun/IRAF distribution, e.g., from the IRAF network archive on `iraf.noao.edu` or one of it's mirrors (or by ordering a tape or CD-ROM distribution from NOAO), 2) select the server or node on which the system is to be installed and arrange for sufficient disk space to hold the system, and 3) set aside sufficient time to do the installation. If these directions are followed carefully and mistakes are avoided the basic installation should only take a half hour or so. Additional time may be required to customize the system to configure the local tape drives and other devices, set up IRAF networking, and so on. V2.11 Sun/IRAF supports SunOS 4.1 using the SunPRO V1.3 or V1.4 compilers, and Solaris versions 2.4 and later using the SunPRO V2.0, V3.0 and V4.0 compilers. Both C and Fortran compilers are required for compilation (either building external packages, doing IRAF development, or local IMFORT programming), no compilers are required for a runtime user system.

Starting with the V2.11 release the SunOS and Solaris versions of IRAF have been merged into a common system allowing for a single installation on a heterogenous network of Sun workstations. The basic installation is still the same with the exception of the added binaries and a bit more work in merging old configuration files if you are replacing two existing systems for each OS. SunOS and Solaris are supported as separate architectures (*sparc* and *ssun*). Sites may install only one or both architectures. We will discuss the installation and basic configuration of a multi-architecture system, detailed configuration is covered in the accompanying *Sun/IRAF Site Manager's Guide*.

The amount of disk space required to install IRAF depends upon the system configuration, primarily whether this is a multi-architecture installation and the number of layered packages installed on top of the core system. The main system, including both the core system and NOAO package sources, requires about 80 Mb, less if stripped after installation. The core system binaries for SunOS require about 22 Mb and the NOAO package binaries requires about 11 Mb, and 38Mb and 22Mb respectively for Solaris.

2. Installing Sun/IRAF

Although the details of how Sun/IRAF is installed or updated depend upon the type of distribution and the desired local system configuration, the basic procedure is always the same. First one obtains the distribution files, then one follows the procedure outlined below to install the system. Most of these steps should be performed while logged in as 'iraf'; superuser permission is required in the final stages of the installation to run the *install* script.

---

†Operated by the Association of Universities for Research in Astronomy, Inc. under cooperative agreement with the National Science Foundation.

System managers familiar with the installation of typical UNIX programs should beware that IRAF, being a large system in its own right and not a UNIX program, does not always follow to the usual conventions. It is wise to read and adhere to the installation instructions to avoid problems.

```
# Prepare the root IRAF directory.
if new installation
    create iraf account
else if updating an old installation
    save locally modified files; delete old system

# Install the files.
login as iraf
unpack the core system distribution
configure the BIN directories

# Merge local revisions into new system.
if updating an old installation
    merge locally modified files back into new system

run the iraf install script to complete the installation
checkout the new system
```

If problems should arise during the installation help is available via the IRAF HOTLINE (520-318-8160), or by sending email to `iraf@noao.edu`.

2.1. Prepare the root IRAF directory

2.1.1. If updating an existing IRAF installation...

If you are updating an existing IRAF installation then you will be replacing IRAF by the new version, and IRAF should already have an account and root directory on the desired host system. You should save any locally modified files and delete the old system, e.g., login as IRAF and enter something like the following.

```
% cd $iraf†
% tar -cf /scr0/oiraf.tar local dev unix/hlib
% /bin/rm -rf *
```

There are many possible variations on this, e.g., you could use *mv* to move the above directories to someplace outside the main IRAF directory tree. Although it is probably simplest and safest to save entire directories as in the example, in practice only a few files are likely to have been modified. These are the following.

```
dev/graphcap
dev/hosts
dev/tapecap*
dev/termcap
hlib/extern.pkg
hlib/login.cl
hlib/zzsetenv.def
local/.login
```

In the case where you will be installing V2.11 Sun/IRAF to replace existing SunOS and Solaris IRAF systems it will be necessary to save these files from *both* systems so they can be merged in the final configuration. If updating an existing V2.11 installation remember to save *all* the tapecap files.

---

† $iraf symbolizes the UNIX pathname of the root IRAF directory. If no "iraf" environment variable is defined just supply the actual pathname.

Once the old system (or systems) has been deleted you are ready to install the new one, as described in §2.2. It is important to delete the old system first to avoid creating junk files or directories when the new system is installed (due to file or directory name changes or deletions). Once the new system has been restored to disk, do *not* merely restore the files saved above, as you will need to carefully merge local changes into the versions of the files supplied with the new IRAF release (more on this later).

2.1.2. If installing IRAF for the first time...

If you are installing IRAF for the first time then the first step is to set up a new account for the user '`iraf`'. This is necessary for IRAF system management, which should always be done from the IRAF account. The IRAF account has custom login files which set up a standard UNIX environment for IRAF system management. Having an IRAF account provides a convenient place (the IRAF system manager's login directory) to keep scratch files created during system configuration, and since it has a standard environment it not only simplifies routine maintanence but provides a baseline for performance when problems are encountered.

The location of the IRAF root directory is arbitrary. Our practice here is to locate the software in a system file storage area separate from the rest of the operating system files (to simplify OS upgrades), and then use a symbolic link such as /iraf to point to the actual root directory. This makes life simpler if IRAF is NFS mounted on several machines and it is later necessary to move the IRAF files. Try to keep the path to the physical IRAF root directory short to avoid filename truncation problems when IRAF is run.

The login directory for the IRAF account should be $iraf/local (e.g., /iraf/iraf/local), rather than the IRAF root directory $iraf as one might expect. This is done to provide a work area for local files separate from the main IRAF directory tree, to simplify updates and make it easier to keep track of what has been locally added and what is standard IRAF. In any case, make sure that when the IRAF account is set up the login directory is set correctly, or the IRAF environment will not be set up properly and later problems are sure to result.

A typical IRAF installation consists of the main IRAF release, a number of BIN directories (the IRAF binaries), and additional directories for layered software such as STSDAS, PROS, and so on. If sufficient disk space is available to keep everything in one area the following directory structure is recommended.

```
/iraf/iraf                      # iraf root directory ($iraf)
/iraf/iraf/local                # iraf login directory (~iraf)
/iraf/irafbin                   # iraf BIN directories
/iraf/irafbin/bin.sparc         # SunOS binaries iraf core system
/iraf/irafbin/noao.bin.sparc    # SunOS binaries NOAO package
/iraf/irafbin/bin.ssun          # Solaris binaries iraf core system
/iraf/irafbin/noao.bin.ssun     # Solaris binaries NOAO package
/iraf/extern                    # external package directory
/iraf/extern/stsdas             # external layered package
/iraf/extern/xray               # external layered package
    (etc.)
```

For the purposes of this example we assume that the IRAF files are stored in /iraf; as we say this might be a link and the actual directory is arbitrary. Given this directory the IRAF root `$iraf` would be "/iraf/iraf/" and the login directory for the IRAF account would be /iraf/iraf/local. The e.g. Solaris binaries for the core IRAF system would be in /iraf/irafbin/bin.ssun, with a link $iraf/bin.ssun pointing to this directory (more on this later).

Given the above directory structure the `/etc/passwd` file entry for the IRAF account would be something like the following.

```
iraf:*:123:12:IRAF system login:/iraf/iraf/local:/bin/csh
```

2.2. Install the files

If you are installing from tape skip forward to §2.2.2. If you are installing from a network distribution (i.e., from disk) continue with the next section.

2.2.1. Installing from a network distribution

Sun/IRAF is available over the network via anonymous ftp from the node `iraf.noao.edu` (140.252.1.1), in the subdirectory `iraf/v`*nnn*`/SSOL`, where "v*nnn*" is the IRAF version number, e.g., subdirectory `iraf/v211/SSOL` for V2.11 Sun/IRAF.

If IRAF is being installed from a network distribution or CD-ROM all the architecture independent IRAF files for both the core IRAF system and the NOAO packages will be in the distribution file `as.ssol.gen`. This "file" is stored in the archive as a directory wherein the large distribution file has been split into a number of smaller pieces, e.g.,

```
% ls as.ssol.gen
CHECKSUMS           as.ssol.gen.Z.12    as.ssol.gen.Z.26
FILES.Z             as.ssol.gen.Z.13    as.ssol.gen.Z.27
as.ssol.gen.Z.00    as.ssol.gen.Z.14    as.ssol.gen.Z.28
as.ssol.gen.Z.01    as.ssol.gen.Z.15    as.ssol.gen.Z.29
as.ssol.gen.Z.02    as.ssol.gen.Z.16    as.ssol.gen.Z.30
as.ssol.gen.Z.03    as.ssol.gen.Z.17    as.ssol.gen.Z.31
       (etc.)
```

Assume that the directory `as.ssol.gen` as shown above has been recreated somewhere on the machine on which IRAF is to be installed. We can restore the main IRAF source tree as follows.

```
% whoami
iraf
% cd $iraf
% cat /path/as.ssol.gen/as.* | uncompress | tar -xpf -
```

After the above finishes the root IRAF directory should appears as follows (this is for V2.11).

```
HS.SSOL.GEN     bin.ffpa        dev             math            sys
IS.PORT.GEN     bin.generic     doc             mkpkg           tags
bin             bin.sparc       lib             noao            unix
bin.f68881      bin.ssun        local           pkg
```

The files `bin.ssun` and `bin.sparc` are links to the IRAF BIN directories (for binary executables), which probably do not exist yet. The `bin` file is a symbolic link to the empty `bin.generic` directory indicating that the system is configured "generically" (i.e. not for development on a particular architecture), this link should *NOT* be changed by hand as it is needed by the system. Binaries for the various architectures will be found automatically by the system at runtime. Configuring the BIN directories is discussed in section §2.2.4.

2.2.2. Installing from tape

If you have not already done so, log into the IRAF account so that the files when restored will belong to IRAF. Mount the distribution tape, which may be a cartridge tape, a 6250 bpi 9 track tape, a DAT tape, an Exabyte, or whatever.

IRAF distribution tapes consist of multiple files separated by tape marks, with a TOC (table of contents) file as the first file on the tape. To find out what is on the tape, rewind it and read out the TOC file as follows (the actual device name will likely be different than that shown in the examples).

```
% mt -f /dev/rmt/0hn rewind;  cat /dev/rmt/0hn
```

This should cause a TOC file to be listed similar to the following, except for the file names which will vary depending upon what type of distribution you have. The example below is for a distribution of Sun/IRAF.

```
0   Table of Contents

1   AS.SSOL.GEN   27.9Mb   IRAF, NOAO packages and Solaris sources
2   IB.SSOL.SUN   19.6Mb   Solaris binaries for IRAF core system
3   NB.SSOL.SUN   15.2Mb   Solaris binaries for NOAO packages
4   IB.SSOL.SOS   14.3Mb   SunOS binaries for IRAF core system
5   NB.SSOL.SOS   12.5Mb   SunOS binaries for NOAO packages
```

Here, the first column is the file number on the tape, the TOC file being file zero (the first distribution file is number one), the second column is the name of the tape file, the third column is the file size in megabytes (this tells you how much space will be needed to unpack the file on disk), and the last column is a description of the file contents.

There are three types of tape files in the example shown: the `AS` file, which is all the IRAF sources (the core IRAF system, NOAO packages, and the Sun/IRAF host system interface), the `IB` files, or IRAF core system binaries, and the `NB` files, or NOAO package binaries. The NOAO package sources are included in the `AS` file since most people requesting IRAF are expected to want the astronomical reduction software, although IRAF can be configured without this if desired. All of the file objects are UNIX *tar* format files, with the exception of the TOC file which is a simple text file. The distribution files may be compressed if this was necessary to fit all the files on a tape.

The following commands would suffice to restore the main IRAF system to disk, given the distribution tape described by the TOC file in our example above. Once again, the tape device file and block size shown in the example will very likely have to be changed to whatever is needed for the tape device being used.

```
% whoami
iraf
% cd $iraf
% mt -f /dev/rmt/0hn rewind;  mt -f /dev/rmt/0hn fsf 1
% tar -xpbf 126 /dev/rmt/0hn
```

After the above tar file read operation, the tape is left positioned to *just before the EOF of the file just read*, since *tar* stops reading the file data before reading the physical EOF. Hence, an *mt fsf* will be required to position to the next file on the tape. Any combination of *fsf* (forward skip file) or *bsf* (backward skip file) operations may be used to position to a file on a 9 track tape. On a cartridge tape, it is best to plan things so that only forward file skips are used, using a rewind and forward skip if it is necessary to position to an earlier file on the tape.

Once the main system, containing only sources, is installed it is possible to create one or more empty BIN directories for the executables, then compile and link the full system. More commonly one will merely read the precompiled executables off the distribution tape, as we discuss in the next section.

2.2.3. Installing from CD-ROM

At the time of this writing the CD-ROM distributions of IRAF are snapshots of selected areas of the network ftp archive. Installing from CD is very much like a network distribution except that the CD may be mounted as a local filesystem and accessed directly when unpacking the files. There is no real reason to copy the files to a local disk before installation, simply use */mount*`/iraf/v`*nnn*`/SSOL` as the path prefix when accessing the `AS` and BIN distribution files. Details on how to unpack the distribution file are covered in §2.2.1 when discussing installation from a network distribution.

2.2.4. Configuring the BIN directories

In IRAF all the files specific to any particular architecture are contained in a single directory called the BIN, or "binary", directory. To run IRAF you must install not only the AS (all-sources) directory tree, but the BIN directory for each architecture. The IRAF core system and the NOAO packages have separate BIN directories.

The BIN directories for the IRAF core system or a layered package (such as NOAO) are located, logically or physically, in the root directory of the IRAF core system or layered package. Every layered package has its own set of BIN directories. In the distributed V2.11 system you will find the following BIN files (directories or symbolic links) at the IRAF root.

```
link        bin -> bin.generic
directory   bin.generic
link        bin.ssun -> ../irafbin/bin.ssun
link        bin.sparc -> ../irafbin/bin.sparc
```

If the IRAF directory structure is set up as described in §2.1.2, with $iraf located at /iraf/iraf and the BIN directories stored in /iraf/irafbin, then these links will not have to be modified. If a different directory structure is used you will have to modify the links accordingly.

The *bin* link and the *bin.generic* directory are required for the correct operation of the IRAF system software (*mkpkg*) and are maintained automatically by the IRAF software management utilities. *Under no circumstances should "bin" or "bin.generic" be modified or deleted.* It is a very common error to manually delete the bin link and manually set it to bin.ssun or some other architecture. The bin.ssun link can be modified as desired but bin and bin.generic should be left alone.

Assume that the bin.ssun directory has been created somewhere (e.g. in the /iraf/irafbin directory) and that the `ib.ssol.sun` distribution files for the core IRAF system solaris binaries have been downloaded from the network archive or will be taken from the CD-ROM. We can restore the solaris binaries with the following commands.

```
% cd $iraf/bin.ssun
% cat /path/ib.ssol.sun/ib.* | uncompress | tar -xpf -
```

Similarly, to restore the NOAO package solaris binaries:

```
% cd $iraf/noao/bin.ssun
% cat /path/nb.ssol.sun/nb.* | uncompress | tar -xpf -
```

A similar sequence can be used to restore the SunOS binaries for both the core system and the NOAO package.

The procedure for restoring a BIN directory from a tape distribution is similar to that described in §2.2.2 for the core system. For example,

```
% cd $iraf/bin.ssun
% mt -f /dev/rmt/0hn rewind;  mt -f /dev/rmt/0hn fsf 2
% tar -xpbf 126 /dev/rmt/0hn
```

would restore the core system bin.ssun directory from a cartridge tape containing an uncompressed `ib.ssol.sun` as file 2 on the tape.

2.3. Merge local revisions back into the new system

If this is a new IRAF installation this step can be skipped, but you will still need to do the final configuration as directed in the Site Manager's Guide. Otherwise, once the new system has been restored to disk any local revisions made to the previous IRAF installation should be merged back into the new system. See §2.1.1 for a list of the files most likely to be affected. When propagating revisions made to these files, be sure to not replace the entire file with your saved version, as the version of the file in the new release of IRAF will often contain important

additions or changes which must be preserved. It is best to merge your revisions into the version of the file which comes with the new IRAF.† This task will be easier if the revisions have been localized as far as possible, e.g., keep all `graphcap` additions together at the head of the file, so that they may merely be transferred to the new file with the editor. Note that when updating existing SunOS/IRAF and Solaris/IRAF systems to a common V2.11 Sun/IRAF system the `dev$tapecap` files are different for the two architectures, you will need to merge the saved files separately to the `tapecap.sunos` and `tapecap.solaris` files. See the Site Manager's Guide for more information on this. The task of propagating revisions will also be much easier if detailed notes have been kept of all revisions made since the the last release was installed.

Beginning with IRAF version 2.8, one should no longer install locally added software in the core system LOCAL package. This significantly complicates updates and is no longer necessary as, due to the layered software enhancements introduced in V2.8 IRAF, it is now straightforward for each site to maintain their own custom LOCAL package external to the core IRAF system. The core system LOCAL is now only a *template-local* to be copied and used as the starting point for a custom LOCAL layered package. The layered software enhancements, and the procedure for building a custom LOCAL, are discussed further in the *Sun/IRAF Site Manager's Guide.*

2.4. Run the INSTALL Script

Once all of the IRAF files have been restored to disk the Sun/IRAF installation script must be run to complete the system installation. The install script modifies the system as necessary to reflect the new root directory and new default image storage and local bin directories, checks the mode and ownership of a number of files, installs a small set of IRAF commands in UNIX, and so on.

To make a trial run of the install script, enter the following commands:

```
% setenv iraf /path/iraf/
% cd $iraf/unix/hlib
% source irafuser.csh
% ./install -n
```

and answer the questions (don't forget the trailing '/' in the "setenv iraf"). The "-n" argument tells install to go through the motions without actually doing anything, so that one can see what will be done before committing to it.

Installing IRAF requires a few changes to be made to system directories outside the IRAF directory tree. Two fifo device entries are made in /dev. A symbolic link "iraf.h" is created in /usr/include. A number of links (cl, mkiraf, etc.) are made in /usr/local/bin or some similar directory which most users can be expected to have in their search path. The tape allocation task alloc.e is made suid root (there are no known security loopholes, although we cannot make any guarantees). A symbolic link imtoolrc is created in /usr/local/lib (but only if this directory already exists; since this file is needed for image display it is a good idea to create /usr/local/lib before running the install script if it doesn't already exist). Lastly the iraf path is edited in to several iraf system files so it is vital this path be correct for your system.

Following one or more trial "no execute" ("-n") runs to see what the install script will do, the install script should be run without the "-n" to complete the installation. This must be done by the superuser as root permission is required to carry out the necessary additions to UNIX. It is essential for the superuser account to also have the proper iraf environment defined before the install script is run, both the $iraf root path and the variables defined by sourcing the irafuser.csh file.

---

†The UNIX utility *diff* is useful for comparing files to see what has changed.

It is necessary to run the install script separately on each node from which IRAF will be used. If a single version of IRAF is installed on a server and NFS mounted on one or more clients, the install script must be run first on the server and then on *each* client (when installing on a client there will be warnings about insufficient permission to make changes to files on the NFS mounted partitions, which can be ignored). To install IRAF on a diskless client it may be necessary to run the install script *on the server* to do the install for the client, since the client's /usr/include and /dev directories may only be writable by root on the server. On some systems /usr is mounted read-only, and must be unmounted and remounted read-write before doing the installation to allow an entry to be made in /usr/include. Once the installation is complete the default mount access mode may be restored.

The exchange with the install script will be along the lines of the following (this example is for a sparc server):

```
% ./install -n
new iraf root directory (/iraf/iraf):
default root image storage directory (/d0/iraf):
local unix commands directory (/usr/local/bin):
install iraf for machine type ssol
old iraf root = /usr/iraf, old imdir = /d0/iraf
installing iraf at /iraf/iraf, imdir=/d0/iraf, lbindir=/usr/local/bin
proceed with installation? (yes):
```

The "iraf root directory" is the value of $iraf (minus the trailing '/'in this case). The "root image storage directory" is the default place to put image data for users; the program may prompt with /tmp if it cannot find any likely looking data storage areas on your system, but /tmp is not a good place to put image data as the contents are deleted whenever the system reboots. The value entered should be the path to a public iraf subdirectory of a designated data or scratch disk on your system. Lastly, the "local unix command directory" is where the UNIX callable IRAF startup commands will be defined. This should be a UNIX directory which is in the default path of anyone who might want to use IRAF; /usr/local/bin is the most common value.

After answering with "yes" or hitting return in response to the "proceed with installation" query, the script will issue a series of messages as it checks the system and performs the installation, possibly answering additional questions in the process. In "no-execute" mode (i.e. with the "-n" flag) the output will contain a diff of the files that will be edited.

3. System Checkout

The basic IRAF system should be usable once the files have been restored to disk, the binaries have been configured or generated, and the install script has been run. To verify that the basic system comes up and runs successfully, login as iraf and startup the CL (IRAF command language) from the iraf account. You should be able to login as IRAF and type "cl" to start IRAF, using the login files which come with the distributed system.

```
% login iraf
% cl
```

To more thoroughly test the installation it is a good idea to test IRAF from a user account. To do this you login to a user account and run the *mkiraf* task to set up the IRAF login files. This will create or initialize the user's `uparm` (user parameter) directory, and create a new `login.cl` file. It may also be desirable to edit the user's `.login` file to modify the way the environment variable `IRAFARCH` is defined. This variable, required for software development but optional for merely using IRAF, must be set to the name of the desired machine architecture, e.g., 'ssun' for Solaris and 'sparc' for SunOS.

```
% mkiraf
Initialize uparm? (y|n): y
Terminal types: xgterm,xterm,gterm,vt640,vt100,etc."
Enter terminal type: xgtermc
A new LOGIN.CL file has been created in the current directory.
You may wish to review and edit this file to change the defaults.
```

The *cl* command should now start up the CL, which will clear the screen and print out a startup message. The standard test procedure included in Volume 1A of the *IRAF User Handbook* should be run to verify the installation, this document is also available as

```
ftp://iraf.noao.edu/iraf/docs/testproc2.ps.Z
```

3.1. Frequently Asked Questions

Many common installation problems are answered in our Frequently Asked Questions file available from either

```
ftp://iraf.noao.edu/iraf/FAQ/FAQ
http://iraf.noao.edu/iraf/faq
```

or from the various iraf archive mirror sites. The WWW version of the FAQ is searchable by keyword.

Appendix A. A Complete Example

Assume we are installing IRAF for the first time on a sparcstation that will be used as a server for both Solaris and SunOS systems in a local network. The IRAF directories will be located at /u3/iraf using a symbolic link /iraf to point to this location. We will configure both sets of binaries, locating the BIN directories in the directory /iraf/irafbin. The local user commands will be placed in /usr/local/bin. We will be installing from a network distribution with the distribution files located in /scr0.

The first step is for the superuser to create an account for the fictitious user '`iraf`', with home directory /iraf/iraf/local and shell /bin/csh. The /u3/iraf directory is created owned by IRAF, and pointed to by the link /iraf. We then login as IRAF (a warning message will be printed since there is no login directory) and proceed as follows.

```
% whoami
iraf
%
% setenv iraf /iraf/iraf/                    # set root directory
% mkdir /iraf/iraf
%
% cd $iraf                                   # unpack main IRAF distribution
% cat /scr0/as.ssol.gen/as.* | uncompress | tar -xpf -
%
% cd /iraf                                   # create BIN directories
% mkdir irafbin
% mkdir irafbin/bin.ssun
% mkdir irafbin/noao.bin.ssun
% mkdir irafbin/bin.sparc
% mkdir irafbin/noao.bin.sparc
%
% cd $iraf/bin.ssun                          # unpack core bin.ssun
% cat /scr0/ib.ssol.sun/ib.* | uncompress | tar -xpf -
%
% cd $iraf/noao/bin.ssun                     # unpack NOAO bin.ssun
% cat /scr0/nb.ssol.sun/nb.* | uncompress | tar -xpf -
%
% cd $iraf/bin.sparc                         # unpack core bin.sparc
% cat /scr0/ib.ssol.spc/ib.* | uncompress | tar -xpf -
%
% cd $iraf/noao/bin.sparc                    # unpack NOAO bin.sparc
% cat /scr0/nb.ssol.spc/nb.* | uncompress | tar -xpf -
%
% cd $iraf/unix/hlib                         # run the INSTALL script
% source irafuser.csh
% ./install -n
% su
# ./install
# exit
%
% cd
% source .login                             # read new .login
% rehash                                    # pick up new iraf commands
% cl                                        # verify that the CL runs
```

This will fully install IRAF on a server, on a standalone system system where only one set of binaries are needed you can skip either the bin.ssun or bin.sparc binary installation. If this version of IRAF will be accessed via NFS by client nodes then the IRAF install script must be run on each client node as well. Installing IRAF does not allow one to access local tape drives, printers, and so on. Refer to the *SunOS/Solaris IRAF V2.11 Site Manager's Guide* for information on how to configure IRAF for the local site.