DEC Alpha Digital UNIX/IRAF Installation Guide

Doug Tody

Mike Fitzpatrick

IRAF Group
National Optical Astronomy Observatory†
Revised July 2002

ABSTRACT

This document describes how to install or update IRAF on a DEC Alpha running Digital UNIX V4.x (aka Compaq Tru64) system. Both standalone and network installations are described. Only those issues which one must understand to install DUNX/IRAF are discussed here; a companion document, the *UNIX IRAF V2.12 Site Manager's Guide*, deals with other issues such as interfacing new devices, configuring the IRAF networking system, adding layered software, and so on.

July 16, 2002

†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 DUNX/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
       - 2.1.2.1 Create the 'iraf' User Account 3
       - 2.1.2.2 Create the IRAF Directory Tree 3
   - 2.2. Install the files 4
     - 2.2.1. Installing from a network distribution 4
     - 2.2.2. Installing from CD-ROM 5
     - 2.2.3. Configuring the BIN directories 5
   - 2.3. Merge local revisions back into the new system 6
   - 2.4. Run the INSTALL Script 6
     - 2.4.1. System Settings 7
     - 2.4.2. System Verification 8
     - 2.4.3. System Installation 9
     - 2.4.4. Post-Install Configuration 9
       - 2.4.4.1 IRAF Networking Setup 10
       - 2.4.4.4 Strip System Sources 10
     - 2.4.5. Post-Install Verification 11

3. **System Checkout** 11
   - 3.1. Frequently Asked Questions 12

Appendix A. A Complete Example 13

Appendix B. A Complete Example Using Split Distributions 14

DEC Alpha Digital UNIX/IRAF Installation Guide

Doug Tody

Mike Fitzpatrick

IRAF Group
National Optical Astronomy Observatory†
Revised July 2002

1. Introduction

Before installing DUNX/IRAF, one must 1) obtain an appropriate DUNX/IRAF distribution, e.g., from the IRAF network archive on `iraf.noao.edu` or one of it's mirrors (or by ordering a 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. V2.12 DUNX/IRAF supports Digital UNIX V4 and V5 systems using the Digital V4 or V5 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.

The amount of disk space required to install IRAF depends upon the system configuration, primarily the number of layered packages installed on top of the core system in the case of DUNX/IRAF. The main system, including both the core system and NOAO package sources, requires about 112 Mb, less if stripped after installation. The core system binaries require about 35 Mb and the NOAO package binaries requires about 24 Mb.

2. Installing DUNX/IRAF

Although the details of how DUNX/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 the 'iraf' user; superuser permission is required to initially create the 'iraf' user account and in the final stages of the installation to run the *install* script.

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.

---

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

```
# Prepare the root IRAF directory.
if new installation
    create 'iraf' user 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/imtoolrc
dev/tapecap.*
dev/termcap
hlib/extern.pkg
hlib/login.cl
hlib/zzsetenv.def
local/.login
local/.cshrc
```

Note: If updating an existing V2.11 installation remember to save *all* the tapecap files in the iraf$dev directory. Some of these files may have been created as symbolic links to the more generalized `tapecap.solaris` and/or `tapecap.sunos` files; to preserve the links it's preferrable to use `tar` over `cp` or `mv` when creating the old system backup files.

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

---

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

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...

2.1.2.1. Create the 'iraf' User Account

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. For that reason it's **critical** that 1) the iraf account be created to use a C-shell as it's login shell, 2) the account be created *before* the IRAF core system files are unpacked so that the preconfigured IRAF login files overwrite any system skeleton files that are created when the 'iraf' account is first setup, and 3) the iraf login directory be defined as $iraf/local (and not just $iraf as one might expect). Missing any one of these points could cause problems in properly setting up the IRAF environment needed for installation and maintainence.

Assuming the default IRAF directory structure (see below) the `/etc/passwd` file entry for the IRAF account would be something like the following:

```
iraf:*:312:12:IRAF system login:/iraf/iraf/local:/bin/csh
```

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.

2.1.2.2. Create the IRAF Directory Tree

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 (so they are not lost entirely during 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, or when NFS mount points may prevent a common path from being established. Try to keep the path to the physical IRAF root directory short to simplify the filename expansion when IRAF is run.

A typical IRAF installation consists of the main IRAF release, a number of BIN directories (the IRAF and NOAO package binaries), and additional directories for layered software such as STSDAS, PROS, display tools like X11IRAF, DS9, 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.alpha         # DUNX/IRAF binaries iraf core system
/iraf/irafbin/noao.bin.alpha    # DUNX/IRAF binaries NOAO package
/iraf/x11iraf                   # X11IRAF (i.e. XGterm/XImtool) directory
/iraf/extern                    # external package directory
/iraf/extern/stsdas             # external layered package
/iraf/extern/mscred             # 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. Digital UNIX binaries for the core IRAF system would be in /iraf/irafbin/bin.alpha, with a link $iraf/bin.alpha pointing to this directory (more on this later). Keeping external packages and add-on software out of the main IRAF tree simplifies later updates to either the core system or the add-on, mixing them in the IRAF tree is discouraged.

2.2. Install the files

2.2.1. Installing from a network distribution

DUNX/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*`/DUNX`, where "v*nnn*" is the IRAF version number, e.g., subdirectory `iraf/v212/DUNX` for V2.12 DUNX/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.dunx.gen`. The distributions files are stored in the archive as a large compressed tar archive. For sites with slow network connections it may be easier to download the distribution as a split-set of files and reconstruct them locally later. In this case the distribution "file" is stored in the archive *"splits"* directory as a subdirectory wherein the large distribution file has been split into a number of smaller pieces, e.g.,

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

Assuming that a single-file installation was chosen one would only need to download the `as.dunx.gen.Z` file from the archive. The IRAF source tree would then be restored with the command (remember to login as the 'iraf' user first!):

```
% whoami
iraf
% cd $iraf
% cat /path/as.dunx.gen.Z | zcat | tar -xpf -
```

On the other hand, if a split-file distribution was downloaded, we assume that the directory `as.dunx.gen` as shown above has been recreated somewhere on the local machine onto which IRAF is to be installed. We can then restore the main IRAF source tree as follows (remember to login as the 'iraf' user first!):

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

After the above finishes the root IRAF directory should appears as follows (this is for V2.12).

```
HS.DUNX.GEN  bin.alpha    doc          math         pkg          unix
IS.PORT.GEN  bin.generic  lib          mkpkg        sys
bin          dev          local        noao         tags
```

The file `bin.alpha` is a symbolic link to the IRAF BIN directory (for binary executables), which probably does 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 §2.2.3.

2.2.2. 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 above.

2.2.3. 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, commonly referred to as the IB (IRAF binary) and NB (NOAO binary) distributions respectively.

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 V2.12 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.alpha -> ../irafbin/bin.alpha
```

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.alpha or some other architecture. The bin.alpha link can be modified as desired but bin and bin.generic should be left alone.

Assume that the bin.alpha directory has been created somewhere (e.g. in the /iraf/irafbin directory) and that the ib.dunx.alp distribution files for the core IRAF system binaries have been downloaded from the network archive or will be taken from the CD-ROM. We can then restore the binaries with the following commands (after creating the e.g. /iraf/irafbin/bin.alpha directory):

```
% cd $iraf/bin.alpha
% cat /path/ib.dunx.alp/ib.* | uncompress | tar -xpf -
```

Similarly, to restore the NOAO package binaries (after creating the e.g. /iraf/irafbin/noao.bin.alpha directory):

```
% cd $iraf/noao/bin.alpha
% cat /path/nb.dunx.alp/nb.* | uncompress | tar -xpf -
```

Note that you must first create and cd to the proper BIN directory for *each* distribution before unpacking the files. Verify that this command worked as expected and you did indeed end up in the proper directory before attempting to unpack the files. Another common error you should avoid is to unpack both the IB and NB distributions in a single directory, this will lead to a "cannot open connected subprocess" error inside IRAF when trying to access core iraf and/or NOAO package tasks.

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 and which should have been backed-up prior to the upgrade. When propagating revisions made to these files, be sure to not just simply 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. 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.

2.4. Run the INSTALL Script

Once all of the IRAF files have been restored to disk the DUNX/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, and 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.

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

---

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

†† These commands assume you are using a C-shell (or equivalent such as 'tcsh'), if not then you can start one with the "csh" or "tcsh" commands before beginning. Commands such as 'source' and 'setenv' may not be understood by other shells and the IRAF environment will not be properly defined.

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.

In "no-execute" mode (i.e. with the "-n" flag) the output will look identical to the standard execution however no files are actually changed in the system.

2.4.1. System Settings

The installation begins by first querying the user for a number of system settings such as the iraf root directory, an image pixel storage directory, and where to put the commands in the system. The meaning of these fields is described below but the exchange with the install script will be something like the following:

```
% ./install -n
                    IRAF V2.12 System Installation
                    ==============================

     Welcome to the IRAF  installation script.  This  script  will
first prompt you for several needed path  names.  The system  will
be verified for proper structure before the actual install begins,
all error must must be  corrected  before you  will be  allowed to
continue. Recommendations for fixing problems will be  made but no
corrective action will be taken directly. Once properly installed,
you will be allowed to do some minimal configuration.

For each prompt: hit <CR> to accept the default value, 'q' to quit
or 'help' or '?' to print an explanation of the prompt.

=======================================================================
======================== Query for System Settings ====================
=======================================================================

New iraf root directory (/iraf/iraf):
Default root image storage directory (/iraf/imdirs):
Local unix commands directory (/usr/local/bin):
Directory for local shared libraries (/usr/local/lib):
```

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 pixel data for users using the default *OIF* IRAF image format, FITS image users do not make use of this directory but it should still be created. The value entered should be the path to a publicly-writeable directory of a designated data or scratch disk on your system; this directory and any user subdirectories will be created as needed by the install script. The "*local unix commands directory*" is where the UNIX callable IRAF startup commands (e.g. '*cl*', '*mkiraf*, etc) 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 or /opt/local/bin are the most common values.
Lastly, the "*directory for local shared libraries*" is where the IRAF shared library is to be

† 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 normally deleted whenever the system reboots.

installed. This will typically be /usr/local/lib however it may be a system directory such as /lib or /usr/lib as well.

2.4.2. System Verification

Once all the needed information has been entered, the script then echoes the system settings it's been given (or determined automatically about the machine), and performs a number of integrity checks to be sure the path definitions are appropriate, the IRAF directory tree structure is correct, and the BIN directories are populated correctly, etc. Should any of these tests fail, instructions for correcting the error or determining more about the problem will be printed, however the script will take no corrective action itself. All tests must be passed before the installation will be allowed to continue past this stage. For example, the above install session might continue as follows:

```
=============================================================
====================== Verifying System Settings ======================
=============================================================

Hostname     = lyra                 OS version   = OSF1 V4.0
Architecture = alpha                HSI arch     = alpha
New iraf root = /iraf/iraf          Old iraf root = /iraf/iraf
New imdir    = /iraf/imdirs         Old imdir    = /iraf/imdirs
Local bin dir = /usr/local/bin
Local lib dir = /usr/local/lib

Checking definition of iraf root directory ...          [  OK  ]
Checking iraf root and imdir directory ...              [  OK  ]
Checking iraf directory write permissions ...           [  OK  ]
Checking for iraf user account ...                      [  OK  ]
    Checking iraf user login directory ...              [  OK  ]
    Checking iraf user account shell ...                [  OK  ]
Checking file ownerships ...                            [  OK  ]
Checking file permissions ...                           [  OK  ]
Checking proper iraf tree structure in /iraf ...
    Checking for 'iraf' subdir ...                      [  OK  ]
    Checking for 'irafbin' subdir ...
        Checking for fallback tree structure ...        [  OK  ]
    Checking for 'irafbin/bin.alpha' subdir ...         [  OK  ]
    Checking for 'irafbin/noao.bin.alpha' subdir ...    [  OK  ]
Checking Core system binary directory ...               [  OK  ]
Checking NOAO package binary directory ...              [  OK  ]
Checking that local bin directory exists ...            [  OK  ]
Checking for existing commands directory...             [ WARN]

   ***  IRAF commands were found in the directory:
   ***
   ***          /opt/local/bin
   ***
   ***      These commands may conflict with the commands now
   ***  being installed in: '/usr/local/bin'
   ***
```

Do you want to delete commands in the old directory? (yes): no

Proceed with installation? (yes):

In this example the final test fails when commands from a previous installation are found which may conflict with the definitions for the current install. Here, we replied '*no*' to simplify the output, however it's recommended that you reply '*yes*' to such a question so as to minimize confusion amongst users as to which iraf command is really being used, to clean out invalid command links, etc.

2.4.3. System Installation

At this point we can reply '*yes*' to the prompt to proceed with the actual installation. Unlike earlier versions of the install script, this new version simplifies the output to print the steps being taken and a singular report of success or failure. Error messages from any stage will output any specifics about what failed along with a recommended fix. Again no corrective action will be taken by the install script directly. For example, the output produced from a successful installation will look something like:

```
=================================================================
========================= Begin Installation =========================
=================================================================

                          Editing Paths

Editing iraf user .login/.cshrc paths ...                [ OK ]
Editing iraf/imdir paths into system files ...           [ OK ]

                     Checking File Permissions

Checking iraf file permissions ...                       [ OK ]
Creating root imdir at /iraf/imdirs ...                  [ OK ]
Reset /tmp sticky bit setting ...                        [ OK ]
Setting tape device permissions ...                      [ OK ]
Checking alloc.e permissions ...                         [ OK ]

                        Creating File Links

Checking for /iraf symlink ...                           [ OK ]
Creating <iraf.h> symlink ...                            [ OK ]
Creating iraf command links in local bin dir ...         [ OK ]
Marking system update time hlib$utime ...                [ OK ]

                    Creating Graphics Device Files

Creating /dev/imt1 fifo pipes for image display ...      [ OK ]
Creating /dev/imt fifo pipes link ...                    [ OK ]
Creating /usr/local/lib/imtoolrc link ...                [ OK ]
Checking if termcap file contains an XGterm entry ...    [ OK ]
Checking graphcap file for XGterm/imtool entries ...     [ OK ]
```

Not all errors found at this stage are considered fatal, however the installer should examine each message and take whatever action is suggested or required. The install script may be re-run later with the same inputs to fix any missed steps.

2.4.4. System Post-Install Configuration

A detailed explanation of the various post-install configuration setup can be found in the **Unix IRAF Site Manager's Guide** however the install script is able to do some of the more common setup as part of the installation on each node. This includes things which may be needed for each iraf client node such as creating a local tapecap file link or entry for the machine in the local IRAF network, or things which may only be needed on the server such as stripping the system of sources.

Once the installation is complete you will be prompted with whether to continue to the post-install configuration stage. If you say *no* at this stage the system is fully functional however some features may still need to be configured by hand. Replying *yes* at this stage will allow you to continue where you may choose to configure one or all of the available features.

2.4.4.1. IRAF Networking Setup

IRAF networking allows you to access remote devices (image displays, tape drives, etc), images, or files from within an IRAF session. All nodes in the *"local IRAF network"*, including the local node, must be defined in the **dev$hosts** file for this to be enabled. A *dev$hosts* entry consists simply of the node name and the local path to an *irafks.e* kernel server binary for that system. The install script is able to determine this information automatically and can add the local machine to the network when the install script is run (e.g on each of the client machines getting their IRAF from a central server).

The networking setup looks something like the following:

IRAF Networking Config

```
    IRAF Networking can be used to access a remote image, tape
device, display server, or other network service. It's config-
uration is not a requirement for normal IRAF operations and it
can be updated at any time by editing the IRAF dev$hosts file
with new entries.

    In this stage we will add an entry for the current platform
to the hosts file.  In a local network installation this script
should be run on each system to add a networking entry as well
as to install other system files needed by IRAF.

Configure IRAF Networking on this machine?  (yes):                    (1)

Recommended dev$hosts file entry used for this machine:

    lyra        : lyra!/iraf/iraf/bin.alpha/irafks.e

Proceed with this entry?  (yes):                                      (2)
Do you want to sort the hosts file by node name?  (yes):              (3)

Checking that iraf networking is properly enabled...              [ OK ]
```

In prompt **(1)** you can reply *no* to skip the setup, hitting the <*CR*> or *yes* will continue with the rest of the output. The script next tries to determine the appropriate entry for this machine and asks in **(2)** whether you would like to accept this entry. Here we accept the default, but replying *no* at this stage will allow you to manually edit the *dev$hosts* file and insert your entry, or you will be asked if you want to skip the setup entirely. If an entry for the machine already exists in the file you will be shown the recommended entry and the existing one and asked if you want to overwrite the entry in the file. Once the entry has been added, you are asked in **(3)** if you would like to sort the hosts file alphabetically by node name (the default is to add the local node at the end of the file). For networks with many nodes this simplifies management of the hosts file since it's easier to locate a particular machine and spot duplicates. Lastly, a check is performed to see whether IRAF networking has been properly configured for this machine. If this last test fails you will be asked whether you want to try again with a different entry or move on.

2.4.4.2. Strip System Sources

The final stage in the post-install configuration is another space-saving step which was often overlooked when left as an exercise in the Site Manager's Guide but which can be important on some systems. The AS distribution file contains the source files for each task in IRAF as well as the various runtime files needed. It is safe to delete the program sources provided:

1) You do not plan to do any development of the IRAF code itself.
2) You do not wish to compile IRAF patches locally and will only be installing prebuilt binaries (external packages and local tasks may still be compiled on a stripped system).

The default is to leave the sources in the system since ar times it will be necessary for site support to supply a code change to fix a particular bug, or recommend a piece of code for inspection to clarify a problem. Workarounds exist for these cases and if disk usage is a concern on a particular system stripping the sources will not hurt IRAF in any way.

The exchange with the install script will look something like:

Strip IRAF System Sources

```
    Source code for all IRAF tasks and interfaces is included with
this installation, but is strictly only required if you plan to
develop this code.  The sources may be deleted from the system with-
out affecting the runtime environment (including help pages, compil-
ation of external packages or local task development) allowing you
to reclaim 50-60Mb of disk space for the system.  Stripping sources
is recommended for systems very short on space, leaving it on the sys-
tem will allow IRAF site support to send code fixes and compilation
instructions as needed to fix problems which have no other workaround.
```

Do you wish to strip the system of sources? (no): yes

```
Stripping core system sources ...                       [  OK  ]
Stripping NOAO package sources ...                      [  OK  ]
```

Note that all external packages (STSDAS, TABLES, MSCRED, etc) can similarly be stripped of sources using something like "mkpkg -p <*pkg*> strip".

2.4.5. Post-Install Verification

The post-install verification stage of the install script currently checks two things: the first is that there is a suitable image display server (e.g. XImtool, DS9, SAOtng/SAOimg) available on the machine, and the second is that there is a suitable graphics terminal (e.g. XGterm or Xterm) on the system. It's important to note that aside from an Xterm window which is standard on most systems, display servers and XGterm must be installed separately from the rest of IRAF.

XImtool and XGterm are distributed as part of the X11IRAF package from the IRAF archive (iraf.noao.edu or it's mirrors) in the /iraf/x11iraf directory. There are separate README and installation instructions for these tools which should be consulted. The choice of display server is sometimes a matter of personal preference, alternate servers are available from the IRAF archive "/contrib" directory.

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, i.e., 'alpha' for Digital UNIX.

```
% 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 workstation that will be used as a server for multiple client systems in a local network. The IRAF directories will be located at /u3/iraf using a symbolic link /iraf to point to this location. 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.dunx.gen/as.* | uncompress | tar -xpf -
%
% cd /iraf                                 # create BIN directories
% mkdir irafbin
% mkdir irafbin/bin.alpha
% mkdir irafbin/noao.bin.alpha
%
% cd $iraf/bin.alpha                       # unpack core bin.alpha
% cat /scr0/ib.dunx.alp/ib.* | uncompress | tar -xpf -
%
% cd $iraf/noao/bin.alpha                  # unpack NOAO bin.alpha
% cat /scr0/nb.dunx.alp/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
```

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 *Unix IRAF V2.12 Site Manager's Guide* for information on how to configure IRAF for the local site.

Appendix B. A Complete Example Using Split Distributions

Assume we are installing IRAF for the first time on a workstation that will be used as a server for multiple client systems in a local network. The IRAF directories will be located at /u3/iraf using a symbolic link /iraf to point to this location. 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.dunx.gen/as.* | uncompress | tar -xpf -
%
% cd /iraf                                   # create BIN directories
% mkdir irafbin
% mkdir irafbin/bin.alpha
% mkdir irafbin/noao.bin.alpha
%
% cd $iraf/bin.alpha                         # unpack core bin.alpha
% cat /scr0/ib.dunx.alp/ib.* | uncompress | tar -xpf -
%
% cd $iraf/noao/bin.alpha                    # unpack NOAO bin.alpha
% cat /scr0/nb.dunx.alp/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
```

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 *Unix IRAF V2.12 Site Manager's Guide* for information on how to configure IRAF for the local site.