IRAF NEWSLETTER

July 1992 Number 12

Central Computer Services National Optical Astronomy Observatories* P. O. Box 26732 Tucson, AZ 85726

Table of Contents

System News 1

Retiring IRAF Support for Certain Platforms 2

1992 Software Conference Update 2

Multiobject SPP 3

World Coordinate System Support in IRAF V2.10 5

Improved CL Scanning and Printing Facilities 7

EPS Output Files in IRAF Version 2.10 8

Major Changes to WFITS in IRAF V2.10 8

A New Version of IMCOMBINE 9

Coordinate Overlay Task WCSLAB Installed in TV Package 11

New Task HFIX Announced 12

Changes to the ICFIT Routine 12

New DIGIPHOT Packages in IRAF V2.10 13

APPHOT Package Update 13

DAOPHOT Package Update 14

PHOTCAL Package Added to DIGIPHOT 15

New OBSERVATORY Database 16

CCDRED in V2.10 16

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

Radial Velocity Package Update 17

The Version 2.10 NOAO Spectroscopy Packages 17

What's New in SPLOT 19

Changes to the ONEDSTDS Directory in IRAF Version 2.10 21

Interfacing an HP LaserJet Printer to IRAF 21

New MEM0 Add-on Package 23

IRAF Tools for Accessing the ADC CDROM 24

IRAF Documentation 26

Add-on Software Available for IRAF Versions 2.8/2.9/2.10 27

Erratum 28

IRAF Version 2.10 Revisions Summary

IRAF SUPPORT SERVICES

Internet:	iraf@noao.edu
SPAN:	noao::iraf or 5355::iraf
IRAF HOTline:	(602) 323-4160
FAX:	(602) 325-9360
Network Archive:	ftp iraf.noao.edu
(anonymous ftp)	ftp 140.252.1.1

The IRAF NEWSLETTER is published roughly three times a year (February, June, and October) by the Central Computer Services, National Optical Astronomy Observatories, P. O. Box 26732, Tucson, AZ 85726. Editors: Jeannette Barnes, Doug Tody

System News

The IRAF version 2.10 distribution began in late April with the release of V2.10 Sun/IRAF. The Sun/IRAF distribution was updated to IRAF V2.10.0 in early July, adding the completed V2.10 revisions summary and updated installation and site manager's guides, and correcting bugs found during the first two months of the release. Those sites that installed IRAF V2.10 prior to July 8 can upgrade their systems by installing the patch0 update available in the IRAF network archive. Sites installing V2.10.0 from scratch will automatically get the latest version. As was the case with V2.9, an additional series of patch upgrades (V2.10.1, V2.10.2, etc.) will likely follow throughout the next year containing bug fixes and minor enhancements for the core IRAF system and NOAO packages.

Initial releases of IRAF for the IBM RS/6000 and Macintosh systems running A/UX, and upgrades to IRAF V2.10 for DEC Ultrix, VAX/VMS, SGI, HP, and so on will follow throughout the coming months. See the accompanying article in this newsletter about the current status of supported IRAF hosts.

Detailed release notes for IRAF V2.10, the *IRAF Version 2.10 Revisions Summary*, are attached to this newsletter and are included in the distribution itself (type "news"). Most of the articles in this newsletter also discuss many of the new features available in the V2.10 release. A brief summary of what is new follows.

The magtape subsystem has been completely rewritten, generalizing the device model and making it table driven via a "tapecap" file. New device entries can be added by the user at a remote site by merely modifying the tapecap file, without need to modify the actual device driver in IRAF. Generic device entries are provided for most devices, e.g., DAT, Exabyte, 1/2 inch reel tape, 1/4 inch cartridge tape, floppy disk, and so on.

A new IRAF network driver has been added which eliminates the repetitive password prompts encountered with the old system. In most cases network connections will now be transparent, and the user need not know what network node a resource resides on.

World coordinate system support has been added to the most commonly used core system tasks, allowing for example, RA and DEC or spectral units to appear on the axes in plots produced by general IRAF utility tasks. The old catch-all NOAO PROTO package has been replaced by the new core system packages PROTO and OBSOLETE, and by the corresponding new NOAO packages NPROTO and NOBSOLETE. All core system tasks present in the old NOAO PROTO package have been moved into the core system where they belong, making the core system more self contained. The main core system packages are now loaded automatically at login time.

Much work was done on the QPOE, MWCS, and other interfaces as part of the new image structures project. Some modest changes were made to IMTOOL allowing user defined lookup tables and a wider selection of color lookup tables. Many other system enhancements and bug fixes were made as noted in the detailed release notes.

Two new X11 system utilities have been written; an X11 (Xt) based IRAF help browser and a magtape status utility used to monitor tape jobs. These will be released, along with the new XGTERM graphics terminal emulator, in a small package of X window system support utilities due out later this year as an add-on for V2.10.

A completely new DIGIPHOT package was installed in the NOAO layered package. This new package includes a revised version of the aperture photometry package APPHOT, the new IRAF DAOPHOT package, a new photometry calibration package PHOTCAL, and a new photometry tools package PTOOLS. The IRAF version of DAOPHOT was a collaborative effort with Dennis Crabtree and Peter Stetson at the DAO, and PHOTCAL was a collaborative effort with Pedro Gigoux at CTIO. The new RV package, used to compute radial velocities using Fourier cross correlation techniques, has been installed. New versions of the APEXTRACT and ONEDSPEC packages are available which use the new world coordinate system facilities and which eliminate the need to linearize spectra. Much reorganization and work has been done on

the IMRED (instrument reduction) packages including the addition of automated reduction scripts. A new version of IMCOMBINE is available providing support for combining misregistered images, and enhancements to the pixel combine and rejection algorithms.

The main IRAF system development priorities for the near future are the remaining V2.10 updates for supported platforms, the establishment of an IRAF news network (will be tied into USENET), continued work on X11 support and the general areas of graphics and imaging, window system support and user interfaces, and development of the ICE software for integrating IRAF with data acquisition. The issue of window user interfaces in particular has received and will continue to receive a lot of attention, although to date most of our work has been R&D. The X11 support package due out later this year is only the start, most of our R&D in this area is addressed at projects planned for FY93 or later.

Doug Tody

Retiring IRAF Support for Certain Platforms

We are considering retiring IRAF support for several older platforms which are now obsolete or which are little used.

IRAF for the Sun 386i will no longer be supported after the release of Sun/IRAF Version 2.10.0. Sun no longer provides upgrades to this machine and we are also finding it increasingly difficult to maintain our local Sun386i that is used for IRAF support. After the release of IRAF Version 2.10.0 our Sun386i will be upgraded to an IPX.

We still have a functional Apollo/Domain system but there are very few sites still using this system, it is old and obsolete, and it not the easiest system to support. We do support the new HP platforms, which replace the Apollo (HP bought out Apollo one or two years back).

Other systems which are on the IRAF endangered species list are VAX/Ultrix, MIPS (SGI has bought out MIPS and these systems are expected to merge in the future), Alliant, and Convex. Continued support for these systems will depend upon demand and the availability of manpower. All are relatively easy to support but usage appears to be diminishing.

Doug Tody

1992 Software Conference Update

The *Second Annual Conference on Astronomical Data Analysis Software and Systems* (ADASS) will be held in Boston on November 2-4, 1992 (some additional meetings not part of the main conference are planned for November 5-6). The Conference is sponsored by NOAO, SAO, and STScI, and is being hosted by SAO. The ADASS Conference provides a forum for scientists and programmers concerned with algorithms, software, and software systems employed in the reduction and analysis of astronomical data.

The first two and one half days of the conference will be devoted to contributed and invited papers given in general session, with some users group sessions and special interest group meetings being held during the evenings and on the afternoon of the third day. Some additional "tag-along" workshops are planned for the days following the main conference. Contributed posters and demos will be on display throughout the conference. We expect the Proceedings of ADASS ’92, like those of the first conference, to be published as part of the Astronomical

Society of the Pacific Conference Series. Since the dates for this meeting encompass Election Day, instructions for casting absentee ballots are included with the registration materials, which were mailed in early May.

Contributions to the program are invited in the areas of algorithms, software, systems, and related topics, with emphasis on practical solutions to the problems of treating real data. The special topics for this year's conference are next generation systems and languages, user interfaces and data visualization, data acquisition, and databases, catalogues, and archives. As always, papers in other areas may be accepted as well.

Invited speakers for this year's program include Miguel Albrecht (ESO), Geoff Croes (NRAO), Alan Farris (STScI), Margaret Geller (Harvard-SAO), Jim Gettys (DEC), Andre Heck (Strasbourg), James Himer (Calgary), Mark Johnston (STScI), Bob Kibrick (UCO/Lick), Doug Tody (NOAO), Michael Van Steenberg (NSSDC), and Don Wells (NRAO).

The early registration deadline for the Conference is July 15; registrations after that date are accepted but at a higher registration fee. The abstract deadline is August 3, and the abstracts must be accompanied by the abstract fee. Participants interested in displaying exhibits or computer software should contact the address below.

If you are interested in registering for this conference, please contact Tricia Buckley at SAO (MS-83, 60 Garden St., Cambridge, MA 02138) or send e-mail to softconf@cfa.harvard.edu (Internet) or 6699::softconf (SPAN). Registration and abstract forms are available on-line via anonymous ftp to sao-ftp.harvard.edu; get the files pub/softconf/register and pub/softconf/abstract. Detailed information on the conference arrangements and program is available upon request.

Jeannette Barnes
George Jacoby
Doug Tody

Multiobject SPP

Don't worry, this article isn't about object oriented programming or even about new SPP libraries. Rather it is about this astronomer's experience writing an applications package in SPP. In the fall of 1990, I began the process of moving the coordinate processing and target selection code that supports the MX Spectrometer from our Data General MV/10000 onto a Sun workstation. Since some of this code was already in its second translation from one FORTRAN to another, I was less than enthusiastic about the conversion process. Since we already had a package of IRAF scripts for reducing MX Spectrometer data, it made sense that the coordinate processing code should also run under the IRAF environment. I originally planned to use the IMFORT interface or the STSDAS F77VOS interface, but a little encouragement from Skip Schaller convinced me that I should take the plunge and convert the code directly to SPP. This would produce a package that is more portable in the future and can be easily exported for my collaborators to run at their home institutions.

The first thing I learned about programming in SPP is that there are three important manuals to read:

A Reference Manual for the IRAF Subset Preprocessor Language, D. Tody, 1983, *doc$spp.hlp*.

Programmer's Crib Sheet for the IRAF Program Interface, D. Tody, 1983, *doc$crib.hlp*.

IRAF Standards and Conventions, E. Downey, et al, 1983, *doc$std.ms*.

These manuals are showing their age in places, but almost everything you need to know is in there. The bad part was that they all contain important things you need to know about the language, so sometimes you need to look in all three manuals in order to find the answer. The

other thing that was confusing at first was learning the proper structure for an IRAF external package. The best solution was to copy the structure of previously existing packages. Translating the old FORTRAN code was very straightforward since SPP has a lot of structure in common with FORTRAN. Primarily I/O and memory allocation statements needed significant changes. I had almost no experience with C, so things like pointers were tricky at first. Then I learned that the parts of SPP that weren't like FORTRAN were probably from C and that reading a C manual often provided an explanation.

MXPACKAGE now has two operational subpackages. MXPACKAGE.MXREDUCE contains CL scripts for use during data reduction. These are primarily drivers for other NEWIMRED tasks (the scripts have not yet been upgraded to IRAF V2.10). MXPACKAGE.MXCOORD contains SPP programs for coordinate processing and target selection algorithms for assigning objects to fiber probes. These programs convert coordinates from the catalog epoch where they were measured into apparent coordinates on the telescope focal plane for the date/time of observation. Precession, nutation, aberration and refraction are all considered. Then one of several algorithms can be selected to assign fibers to the target objects. The package currently contains about 12000 lines of code.

project --- project celestial coordinates on the focal plane.
select --- select targets for the fiber probes to make target files.
crossref --- make a cross reference listing of objects and targets.
xcheck --- check a target file for possible collisions.
mxplot --- plot MX probes and target positions.
autoselect --- select targets from a list of celestial coordinates.
makeid --- generate a slit/target apid file for reduction.
telcat --- generate a telescope catalog from coordinate files.
tcopy --- copy a target file by reading and writing.
xcopy --- copy a xy file by reading and writing.

A third subpackage, MXPACKAGE.MXOBSERVE, is now under development. This package will provide a communication interface to the MX Spectrometer for downloading programs and coordinates and calibration files. A fourth subpackage, MXPACKAGE.MXPROTO, contains development tasks. These are presently a series of post-processors for the output of the RV0.FXCOR cross-correlator.

Would I do this again? Definitely yes! I have almost entirely replaced FORTRAN with SPP as my everyday language of choice. My third order telescope design program has also been converted to SPP. This is not so much because of SPP as a language, but rather because of the consistent user interface that I can use when switching from data reduction, to CCD spectroscopy, to processing MX coordinates, to cooking mirrors in the oven.

Would I recommend SPP to others? Again yes. The only significant obstacle was the lack of a good Reference Manual. Zoltan Levay from Space Telescope has recently rectified that problem by compiling a new and very complete manual entitled *SPP Programmer's Reference.*

What's next? I am currently in the process of generalizing the MX coordinate code so that it is applicable to any one of a number of multifiber spectrometers starting with HYDRA, ARGUS, AUTOFIB, AMOS, etc. Instrument specific parameters will be stored in separate parameter files. This code is also proving useful for simulating new instrument geometries with larger numbers of fibers.

Thanks to Skip Schaller of Steward Observatory for his SPP advice, and to Micheal Keane of UC Santa Cruz for the use of his astrometry code, and to Simon White of Cambridge for the use of his simulated annealing algorithm for target assignments. For further information on the MXPACKAGE, contact jhill@as.arizona.edu.

John Hill
Steward Observatory

World Coordinate System Support in IRAF V2.10

IRAF V2.10 marks the general introduction of consistent world coordinate system (WCS) support in both the IRAF core system and NOAO applications packages. Although the IRAF WCS interface itself was added to IRAF in Version 2.9, applications support was both limited and application dependent prior to IRAF V2.10 (see the article *The New World Coordinate System (MWCS) Interface* in IRAF Newsletter Number 9, February/June 1990).

IRAF supports three builtin coordinate systems: 1) "logical" coordinates which are pixel coordinates relative to the current image section, 2) "physical" coordinates which are (in the case of raster images) pixel coordinates relative to the original raw data image, and 3) "world" coordinates which may be any general world coordinate system. For example, the coordinates of the lower lefthand pixel in an image section of the new test image described by "dev$wpix[23:33,50:60]" are (1.0, 1.0) in the "logical" coordinate system, (23.0, 50.0) in the "physical" coordinate system, and (13:28:04.6, 47:24:35.9) in the "world" coordinate system. The "physical" and "world" coordinate systems are invariant to linear transformations of the parent image, e.g. an image shift, meaning that the pixel coordinates in the "physical" system and the "world" coordinates of an object do not change.

IRAF currently supports the following nonlinear "world" coordinate systems at the system level: 1) the TAN, SIN, ARC, and GLS sky projections, 2) the MULTISPEC coordinate system used by the spectral reduction and analysis packages, and 3) "sampled" coordinate systems where the transformation is defined by an array of points. In addition any linear coordinate system can be defined.

Automatic WCS updating of the image header is now enabled by default (it was disabled by default in IRAF V2.9). This affects all the IMAGES package geometric operator tasks including: IMCOPY, SHIFTLINES, IMSHIFT, MAGNIFY, BLKAVG, BLKREP, IMTRANSPOSE, ROTATE, IMLINTRAN, REGISTER, GEOTRAN, IMSLICE, and IMSTACK. This means that tasks like the IMSHIFT task will now shift the image pixels and update the world coordinate system. The following example shows the WCS header keywords of the new test image "dev$wpix" before and after being shifted +2.0 pixels along each axis.

Original dev$wpix WCS image header keywords:

```
CRPIX1  =                257.75
CRPIX2  =                258.93
CRVAL1  =      201.94541667302
CRVAL2  =              47.45444
CDELT1  =        -2.1277777E-4
CDELT2  =         2.1277777E-4
CTYPE1  = 'RA---TAN'
CTYPE2  = 'DEC--TAN'
```

Shift command:

```
cl> imshift dev$wpix outpix 2.0 2.0
```

Outpix WCS image header keywords:

```
CRPIX1  =                259.75
CRPIX2  =                260.93
CRVAL1  =      201.94541667302
CRVAL2  =              47.45444
CDELT1  =        -2.1277777E-4
CDELT2  =         2.1277777E-4
CTYPE1  = 'RA---TAN'
CTYPE2  = 'DEC--TAN'
```

```
WCSDIM  =                    2
CD1_1   =  -2.1277777000000E-4
CD2_2   =  2.12777770000000E-4
LTV1    =                   2.
LTV2    =                   2.
LTM1_1  =                   1.
LTM2_2  =                   1.
WAT0_001= 'system=image
WAT1_001= 'wtype=tan axtype=ra
WAT2_001= 'wtype=tan axtype=dec
```

The LISTPIXELS task is now able to list pixels in "logical", "physical", or "world" coordinates, using formats supplied by the user or read from the image header. For example, the pixel coordinates of an image with a TAN sky projection WCS can be listed in hh:mm:ss.s and dd:mm:ss format, and the pixel coordinates of a spectrum can be listed in Angstroms to any desired precision. Similarly the RIMCURSOR task is now able to accept a user or image header format specification for the pixel coordinates. The first two examples below show how to use the LISTPIXELS and RIMCURSOR tasks to list out the pixels of the test image "dev$wpix" in hh:mm:ss.s and dd:mm:ss format. The third example shows how to list a spectrum in world coordinates to the desired precision and pipe the results to a file.

```
cl> listpixels dev$wpix[255:259,255:259] wcs=world \
>>> formats="%H %h"

cl> display dev$wpix 1
cl> rimcursor wcs=world wxformat=%H wyformat=%h

cl> listpix spec0005.ms[*,3] wcs=world \
>>> format="%10.3f %-15.7g" > sfile
```

WCS support has been added to the PLOT package tasks IMPLOT, GRAPH, PROW(S), and PCOL(S) and to the IMEXAMINE task. These tasks all permit the user to set the world coordinate system that the coordinates will be plotted in, to change the WCS interactively (IMPLOT and IMEXAMINE), to specify the format that the coordinates will be plotted in, and to change those formats interactively (IMPLOT and IMEXAMINE). The following command plots aperture 3 of a MULTISPEC format image in wavelength units.

```
cl> implot <spectrum> 3 wcs=world
```

Users should be aware that the PCOL(S) and PROW(s) tasks still expect the input data specification to be in "logical" coordinates, even if the output is in "world" coordinates. In the following example row 20 in the test image "dev$wpix" is plotted in "world" coordinates.

```
cl> prow dev$wpix[*,20:100] 1 wcs=world
```

Two new tasks, WCSRESET and WCSEDIT, have been added to the PROTO package. WCSRESET is used to reset a named WCS to the identity transformation. It is also quite useful for removing the effects of previous linear transformations from the "physical" coordinate system. WCSEDIT is used to enter a new WCS or edit an existing one. The first example below removes all coordinate system information from a copy of the the test image "dev$wpix". The second example resets the "physical" coordinate system to the "logical" coordinate system, while leaving the "world" coordinate system unaltered.

```
cl> imcopy dev$wpix newpix
cl> wcsreset newpix world

cl> imcopy dev$wpix newpix
cl> wcsreset newpix physical
```

A new WCS can be entered with the WCSEDIT command as follows. Note that the test image "dev$pix" does not have a "world" coordinate system.

```
cl> imcopy dev$pix newpix
cl> wcsedit newpix wcs=image interactive+
```

WCSEDIT can also be used to edit a single attribute of an existing WCS. For example, default formats can be entered into the image WCS with WCSEDIT as follows.

```
cl> imcopy dev$wpix newpix
cl> wcsedit newpix format %H 1
cl> wcsedit newpix format %h 2
```

LISTPIXELS, RIMCURSOR and the plotting tasks will now make use of these format specifications to print out the pixel coordinates.

Brief descriptions of the IRAF WCS system can be found in the help for the WCSEDIT and WCSRESET tasks, and for spectra in the help for the ONEDSPEC package, **phelp onedspec.package**. Detailed documentation for the IRAF world coordinate system interface MWCS can be found in the file *iraf$sys/mwcs/MWCS.hlp*. This file can be formatted and printed with the command **help iraf$sys/mwcs/MWCS.hlp fi+ | lprint**. Details of the (proposed) FITS world coordinate system representation, on which the format of the IRAF image headers is based, can be found in the document *World Coordinate Systems Representations Within the FITS Format* by Hanisch and Wells, available from our anonymous ftp archive.

Lindsey Davis
Frank Valdes

Improved CL Scanning and Printing Facilities

IRAF V2.10 includes the long-awaited ability to scan from a pipe (or whatever the STDIN is connected to). Also added are new formatted I/O routines SCANF, FSCANF, and PRINTF. These are like the Unix/STDIO functions but use IRAF formatted I/O which supports such features as %h and %*.*. This means, for example, that the way to scan an image header to extract the exposure time, RA, and DEC keywords using HSELECT, to then compute the mean value for the image using IMSTATISTICS, and finally to nicely print these values, has been greatly simplified as shown in the following CL script.

```
# EXAMPLE - Example of scan and printf usage.
# Note that the format can be a variable as well.

procedure example (image)

string    image                 {prompt="Image name"}
```

```
string   format = "%s: %h %h %m %5.1f\n" {prompt="Format"}

begin
        string      im
        real        exptime, ra, dec, mean

        im = image
        hselect (im, "itime,ra,dec", yes) | scan (exptime, ra, dec)
        imstatistics (im, fields="mean", format=no) | scan (mean)
        printf (format, im, exptime/3600., ra, dec, mean/exptime)
end
```

Note that for scanning multiple lines of data, for example using HSELECT on a list of images, it is still necessary to use FSCAN to read from a file as is described in *An Introductory User's Guide to IRAF Scripts*. More information and limitations on these new features may be found in the *IRAF Version 2.10 Revisions Summary* document and in the system notes file *iraf$local/notes.v210*.

Suzanne Jacoby

EPS Output Files in IRAF Version 2.10

In IRAF Version 2.10 it is now possible to output plot files in Encapsulated PostScript format. The EPS files are output to the current directory with file names such as *sgixxxxx.eps* where "xxxxx" is a five digit number that changes for each plot. The EPS files can then be directed to a PostScript printer or included in documents.

EPS output is enabled by specifying one of the EPS graphics output "devices". Three such logical devices are available: *eps* produces a plot placed at the top of a page, *epsl* produces a landscaped plot, and *epsh* produces a plot placed in the middle of a page. These options can be invoked with *:.snap*, by specifying the graphics output device via a hidden parameter when a task is run, or by setting the environment variable *stdplot* before running the task.

Chris Biemesderfer

Major Changes to WFITS in IRAF V2.10

WFITS has been modified to write IEEE floating point FITS files by default (FITS bitpix = -32 or -64), instead of scaled 32 bit integer FITS files, if the input image pixel type is real or double precision. If the user overrides the default "bitpix", and chooses to scale the floating point data into FITS integers ("bitpix = 16" or "bitpix = 32"), the scaling parameters "bscale" and "bzero" are printed on the terminal. Since "bscale" is the ratio of the dynamic range in the data to the dynamic range in the FITS integers, it provides an accurate estimate of the maximum loss of precision to be expected when the FITS file is reread. Avoiding this loss of precision was the primary motivation for changing the WFITS default output format. Images with integer pixel types will still be written in FITS integer format by default (bitpix = 16 for short integer pixels and bitpix = 32 for long integer pixels). Users who are planning to read their data into local programs with a local FITS reader should check to see whether their local reader supports IEEE FITS format, and select "bitpix" accordingly if it does not.

WFITS has been modified to set the FITS tape blocking factor to the value of the "fb" parameter in the *dev$tapecap* file. For variable blocksize devices such as 9-track tapes, Exabytes, and most DAT drives, the user can override this built in default blocking factor by setting the WFITS "blocking_fac" parameter to any number between 1 and 10. For fixed blocksize devices, such as cartridge tapes, the blocksize and the FITS blocking factor are read from the *dev$tapecap* "bs" and "fb" parameters respectively. In this case the user is no longer required to know or set the tape blocksize and FITS blocking factor and cannot override the *dev$tapecap* entries. When WFITS is run the user will see either "blkfac = 1-10" if the output device is a variable blocksize device, or "blkfac = fixed" if the output device is disk or a fixed blocksize device. It is the responsibility of the local site manager to set the *dev$tapecap* "bs" and "fb" parameters to values appropriate for the devices in use at the local site. Users who are planning to read their data with a local FITS reader should check to see if their local reader can read tapes with a blocking factor greater than 1, and either set the WFITS "blocking_fac" parameter or request their site manager to reconfigure the *dev$tapecap* entries appropriately, if it cannot.

Lindsey Davis

A New Version of IMCOMBINE

A major new version of IMCOMBINE and the closely related CCDRED.COMBINE and ONEDSPEC.SCOMBINE has been released with V2.10. An external add-on version of IMCOMBINE for V2.9 IRAF is also available in the IRAF network archive. This article summarizes the capabilities of the IMCOMBINE task and also a proposed change in the weighting of the weighted average. CCDRED.COMBINE is essentially the same with the extra feature of recognizing and grouping images by their CCD type. The SCOMBINE task for spectra shares some of the algorithms but not the masking and offsetting. Instead, it allows spectra to be combined in wavelength even when the dispersions are not identical.

The new version of IMCOMBINE provides the same capabilities as the previous version but adds many new capabilities. The new capabilities may be categorized as follows.

- Combine images with different sizes
- Combine images with integer offsets
- Exclude bad pixels using mask images
- Separate threshold step to exclude pixels
- Separate rejection step to exclude pixels
- Additional 1D rejection growing
- Separate combining method with choice of mean or median
- New rejection algorithms
    - CCDCLIP - Clipping using CCD noise parameters
    - CRCLIP - Cosmic ray clipping using CCD noise parameters
    - PCLIP - Clipping using percentiles
- Choice of mean or median in rejection algorithms
- Separate scaling, zero level, and weighting options with choice of mean, median, mode, or exposure statistics
- Projection option to combine across highest dimension

The ability to "mosaic" images without manually inserting each one in a bigger frame of the same size and origin was one of the main requirements for the new task. For large grids with large detectors it was very wasteful of disk space. This capability can be used to construct a

single image from a set of offset exposures or to make a mosaic for display purposes (possibly including gaps between the images which can be filled with the "blank" value).

The new task is still limited in the number of images which can be combined by the number of files which IRAF can handle simultaneously (which is a larger number than most operating systems allow). For ".imh" images this is a little over 100 images. If one wants to combine more, other than by breaking things up into groups, one may stack them into a higher dimensional image (with IMSTACK) and use the projection option. This stacked format and projection combining may also be desirable for data consisting of a large number of small images, for example an IR imager, to minimize the number of files.

Pixel mask "images" are a relatively new feature of IRAF, particularly outside of the XRAY and STSDAS packages. A separate newsletter article will discuss them and only a quick summary is given here. A pixel list image (of which a mask is a special case) is a type of file format which compactly describes integer valued images, especially if there are only a small number of values which define regions. These files have a standard extension of ".pl" (pixel list) and may be used anywhere an image may be used; hence it is considered one of the image types supported by IRAF. A mask would be an image defining good pixels by one value, most efficiently as zero, and bad pixels by a second value, say one. Because masks are new and not used in many applications yet, there are few tools available for making them. However, if you have any recipe that can make an image of 0's and 1's (such as with MKPATTERN or IMREPLACE) you can explicitly use the ".pl" extension to make them. To convert from a standard array type image to a pixel list image you can simply use:

```
cl> imcopy badpix mask.pl
```

The new IMCOMBINE is one of the tasks which can use a pixel mask image. A mask is assigned to an image by setting a header keyword. Then when images are combined pixels flagged as bad in the pixel mask will be excluded from the rejection and combining algorithms.

One of the important organizational changes between the old and new version of IMCOMBINE is to separate out three steps which were formally combined together. These are threshold rejection, algorithmic rejection, and the statistic used to combine the final set of pixels. In particular, previously one could use threshold rejection but then nothing else. Threshold rejection is an alternative way of masking bad pixels. If the bad pixels are set to some value outside the data range, say with IMREPLACE, then the threshold parameters allow excluding them from the rest of the combining operations. The combining statistic has been separated out to allow choosing an average or a median independent of any rejection algorithm. (Speaking of medians, in the new version a median of an even number of points is now the average of the middle two values rather than the lower one).

In a similar vein, there were requests to allow both scaling and zero level adjustments independently. This is now possible. Also, there was interest in having additional choices in scaling, zero level, and weighting statistics beyond exposure time and mode. The new version includes mean and median as additional statistics. The region in which the statistics are calculated also had to be generalized to allow defining it either relative to the input images or the output images because of the new offsetting capability.

In addition to some of the previous rejection algorithms there are several new ones. The most effective for images having noise statistics characterized by a readout noise and gain is to apply a sigma clipping based on these parameters. A new and interesting algorithm which is relatively insensitive to objects appearing in a few of the images at each point, the case when making sky flats from disregistered observations, is a percentile clipping algorithm. In this algorithm the "sigma" for clipping is determined by the difference between the median and a specified percentile. The help page lists and describes the rejection algorithms.

The weighted average in both the old and new versions uses sigma weighting. In particular, the weights are the square root of the exposure time or image statistic (mean, mode, or median). This has been the case for many years and those who critiqued the task did not comment on this choice of weights. The choice of weights in a weighted average requires adopting some

criterion for what is desired in the final average. It was recently pointed out that the standard criterion given in basic statistics texts is to minimize the variance in the final average. This is akin to a least squares criterion. In this case the weights should be the variances and not the sigmas. Because this is the simplest and most common recommendation for weights in a weighted average it is proposed to change the weights in IMCOMBINE, CCDRED.COMBINE, and ONEDSPEC.SCOMBINE to variance weighting for the next release (possibly V2.10.1); the weights would be the exposure time, mean, median, or mode rather than the square root. Comments to iraf@noao.edu or via mail are welcome.

Note that the current weighting, which is analogous to minimizing the absolute difference rather than the variance, is very similar or identical if all the weights are similar or equal. So for the most common cases of no weighting, using a median, or combining images with nearly identical signal-to-noise, there will be little or no difference. It is only in the case of a weighted average of images with significantly different signal-to-noise, for example images which have exposure times differing by factors of 2 or more, which will differ somewhat in the final variance statistics. In all cases, there is no difference in the mean signal levels but only in the noise statistics.

Frank Valdes

Coordinate Overlay Task WCSLAB Installed in TV Package

The WCSLAB task, written by members of the STScI SDAS programming group and available in the STSDAS package, has been installed in the IRAF TV package in IRAF V2.10. WCSLAB reads the world coordinate information stored in an image header or supplied by the user in the WCSPARS parameter set to draw a labelled coordinate grid. At present, WCSLAB can handle the TAN, SIN, and ARC sky projection coordinate systems and any linear coordinate system. WCSLAB does not support the IRAF MULTISPEC format which is best handled using the SPECPLOT task.

The output coordinate grid may be overlayed on the image display, the default, or on a previous plot, for example a contour plot. WCSLAB attempts to pick reasonable default window and viewport parameters but the user can adjust these if the defaults chosen by WCSLAB are unsatisfactory. The following two examples, which users can try for themselves, illustrate typical usage of WCSLAB. Note the name of the new test image "dev$wpix". Dev$wpix is the same as "dev$pix" but has a defined sky projection world coordinate system.

Overlay a coordinate grid for wpix on the 512 square image display.

```
cl> display dev$wpix 1 xmag=0.8 ymag=0.8
cl> wcslab dev$wpix 1
```

Overlay a coordinate grid for wpix on a contour plot.

```
cl> contour dev$wpix per-
cl> wcslab dev$wpix dev=stdgraph append+
```

WCSLAB attempts to determine the optimum range and interval of labels appropriate for a given coordinate system. However, if the WCSLAB defaults are unsatisfactory, there is a large menu of parameters, located in the WLPARS parameter set, which the user can modify to alter the appearance of the plot.

Jonathan Eisenhamer
Lindsey Davis

New Task HFIX Announced

A new task for modifying image headers is available in the IRAF V2.10 PROTO package. HFIX takes the supplied command and applies it to the user area of the image header. The most common (and default) command is a text editor. However the command can be any IRAF command including CL escapes to host commands. This task can be quite powerful, and indeed HFIX may be the only way to recover from certain pathological situations involving image headers. The help page gives several examples. The following examples illustrate some more sophisticated cases which came up recently. In these examples below the temporary text file containing the header to be edited is referred to as $fname.

[1] We have seen images where user area keywords can not be accessed individually and output from IMHEADER appears skewed across the page. Such an image could have been created with a program outside the IRAF IMFORT or IMIO interfaces or with an improper FITS format. In one notable case a FITS file included NULL control codes in the header cards. It may be possible to save such images using HFIX to call REBLOCK. Of course, the exact REBLOCK command line will depend on the details of the botched header, but, for example:

```
cl> hfix <image> command="reblock $fname temp inrec=79 \
>>> outrec=80 padchar=\n; del $fname; rename temp $fname"
```

[2] Another application of HFIX, again to recover from an improper FITS file, is for incorrectly formatted keyword=value card images; for example the = sign in column 10 instead of column 9. HFIX is used to invoke the Unix stream editor, to replace each occurrence of

```
keyword  = value        with
keyword = value
```

```
cl> hfix <image> command="!sed 's/ =/=/' $fname > temp; \
>>> mv temp $fname"
```

This replaces the first occurrence on each line of " =" with "=", effectively deleting a space.

Suzanne Jacoby
Frank Valdes

Changes to the ICFIT Routine

A couple of often requested changes have been made to the interactive curve fitting routine (ICFIT) used in a number of tasks. These are to allow adding points in order to constrain a fit and to remove the limitation on the length of the sample region strings. A new key, "a", allows adding a point with a specified weight. By making the weight large it is possible to force the fitted function to be closer to what is desired. We recognize that this is not as effective as user defined spline knots but it was a change which could be made more easily at this time.

Sample regions are used to restrict the fit to certain parts of the data range. Default sample regions are often specified in task parameters and new sample regions may be specified interactively either with the cursor or an explicit colon command. In all cases the regions are described at some point by a character string. The problem has been that after a certain number of regions the maximum length of the character string is reached. The regions specified in a task parameter are limited by the maximum length of a string of such a parameter which is 160

characters. There is no easy way to modify this length. Instead the syntax of a sample region string now includes references to a file containing a list of regions. This is done using the same syntax you already know for a list of filenames by preceding the file with the "@" character. For the interactive cursor and colon commands the internal string limit is now 1024 characters, much longer than before. Anything longer can use the "@file" syntax in a colon command.

Another useful change is the new key "z" to delete individual sample regions. This is the opposite of the "s" key used to define regions with the cursor.

Frank Valdes

New DIGIPHOT Packages in IRAF V2.10

Three new packages, DAOPHOT, PHOTCAL, and PTOOLS have been added to the digital stellar photometry package DIGIPHOT.

IRAF DAOPHOT is the IRAF implementation of the DAO crowded-field photometry package DAOPHOT-I, written by Peter Stetson of the DAO. The initial port of DAOPHOT-I to IRAF was done in collaboration with Dennis Crabtree also of the DAO. A version of IRAF DAOPHOT-I has been available to users as part of the IRAF add-on package TESTPHOT since January 1990.

PHOTCAL is a photometric calibration package containing a set of tasks for computing the transformations from the instrumental system to the standard photometric system, and applying the computed transformations to the data. PHOTCAL is a new package not available prior to IRAF V2.10. PHOTCAL was written in collaboration with Pedro Gigoux at CTIO.

PTOOLS is a photometric tools package containing a set of tasks for performing operations on the photometric catalogs produced by APPHOT or DAOPHOT. Tasks exist to examine the photometry catalogs interactively, edit the photometry catalogs either interactively or in batch mode, and do simple dump, sort and select operations on the output files. A version of PTOOLS has been available to users as part of the IRAF add-on package TESTPHOT since January 1990.

Most but not all DAOPHOT tasks can be configured to write out or read in data in ST tables format. Users who wish to use the ST tables input/output option in DAOPHOT must acquire and install the TABLES package, available from either NOAO or ST. The TABLES package is not required for users who select the default text output format. All PHOTCAL or PTOOLS tasks which require DAOPHOT output as input can read either text or ST tables files.

Lindsey Davis

APPHOT Package Update

The following major changes have been made to the APPHOT package in IRAF V2.10.

The new PTOOLS package task PEXAMINE for interactively examining and/or editing the photometry catalogs produced by the APPHOT package tasks has been added to the APPHOT package. PEXAMINE is designed to aid the user in assessing the accuracy of the photometry, in diagnosing problems with particular catalog objects, in searching for relationships between the computed quantities, and in editing the catalog based on these observed relationships.

The PTOOLS package task TXDUMP has been added to the APPHOT package and replaces the APSELECT task. Users with scripts referencing APSELECT should simply replace references to APSELECT with references to TXDUMP. The functionality is the same.

Two new parameters "obstime" and "otime" have been added to the DATAPARS task. These parameters can be used to set the time of observation, either by extracting it from the image header using the "obstime" parameter, or setting a specific value using the "otime" parameter. The time of observation is written to the photometry catalog. Users may have to unlearn the DATAPARS task to pick up the new feature.

All computed magnitudes will be set to INDEF for apertures which contain bad data, i.e. data which is less than the DATAPARS parameter "datamin" or greater than the DATAPARS parameter "datamax". The PIER flag and PERROR string will still be set as before. This change was made to help PHOTCAL users easily identify saturated stars and stars which are outside the linearity regime of the detector.

All the APPHOT tasks have been modified to verify the "datamin" and "datamax" parameters in verify mode ("verify = yes" in non-interactive mode, the "v" keystroke command in interactive mode). This change can help avoid unpleasant surprises for users who like to set their "datamin" value to nsigma * sky and who then change images to one with a significantly different sky level.

Lindsey Davis

DAOPHOT Package Update

The DAOPHOT package is now installed in the DIGIPHOT package in IRAF V2.10. Users familiar with the TESTPHOT version of DAOPHOT should review the following summary of the major differences between the DIGIPHOT and TESTPHOT versions of DAOPHOT. All DAOPHOT users are encouraged to retire the TESTPHOT version of DAOPHOT when they acquire IRAF V2.10.

The functionality provided by the TESTPHOT TTOOLSX package has been replaced by the TABLES package. TABLES is an external add-on package developed at STScI which contains all the tasks previously available in TTOOLSX plus many more. Users who wish to use the ST tables output option in DAOPHOT must acquire and install the TABLES package from either NOAO or STScI. Users who select the default text output format do not require the TABLES package. DAOPHOT will warn the user at load time if TABLES is not available.

The new PTOOLS package task PEXAMINE for interactively examining and/or editing the photometry catalogs produced by the DAOPHOT package tasks has been added to the DAOPHOT package. PEXAMINE is designed to aid the user in assessing the accuracy of the photometry, in diagnosing problems with particular catalog objects, in searching for relationships between the computed quantities, and in editing the catalog based on these observed relationships.

The TESTPHOT PTOOLS tasks APPEND, CONVERT, DUMP, RENUMBER, SELECT, and SORT which are loaded by the DAOPHOT package have had their names changed to PAPPEND, PCONVERT, PDUMP, PRENUMBER, PSELECT, and PSORT. This change avoids collisions with general existing and possible future CL facilities like SORT. Users will need to update their scripts with the new names.

The "text" parameter, used to specify the format of the DAOPHOT output files, has been moved from the DAOPARS parameter set to the DAOPHOT package parameter set.

Two new parameters "obstime" and "otime" have been added to the DATAPARS task. These parameters can be used to set the time of observation, either by extracting it from the image header using the "obstime" parameter, or setting a specific value using the "otime" parameter. The time of observation is written to the photometry catalog.

All the DAOPHOT tasks, except the interactive PSF fitting task PSF, now accept lists of input and output images and photometry files.

The PSF task now stores up to 66 stars by default (instead of the original 22) in the output PSF image header or the number permitted by the current value of "min_lenuserarea", whichever is greater.

The DAOPARS parameters "psfrad", "fitrad", and "matchrad" are now defined in units of the DATAPARS "scale" parameter. If "scale = 1.0" all units are in pixels as before. PSF images created with DAOPHOT in the TESTPHOT package are still readable with the new version.

All the DAOPHOT tasks have been modified to verify the "datamin" and "datamax" parameters in verify mode ("verify = yes" in non-interactive mode, the "v" keystroke command in interactive mode if applicable). This change can help avoid unpleasant surprises for users who like to set their "datamin" value to nsigma * sky and who then change images to one with a significantly different sky level.

Lindsey Davis

PHOTCAL Package Added to DIGIPHOT

The photometric calibration package PHOTCAL, written in collaboration with CTIO, has been added to the DIGIPHOT package in IRAF V2.10. PHOTCAL is a set of tasks for determining the transformation from the instrumental photometric system to the standard photometric system, and applying the transformation to the data. PHOTCAL contains tasks for: 1) creating and/or editing standard star catalogs and observations catalogs, 2) creating, checking and/or editing the configuration file which specifies the format of the standard star and observations catalogs and the form of the transformation equations, 3) solving the transformation equations, and 4) applying the transformations to the observations.

Since PHOTCAL requires its input to be in the form of simple text files, it can be used to process data reduced with non-IRAF programs as well as the output of APPHOT and DAOPHOT. Various preprocessor tasks exist in PHOTCAL for combining data on the same object taken through different filters.

PHOTCAL maintains a directory of standard star catalogs. Currently the only supported catalog is the Landolt UBVRI catalog. Others will be added in the future. User suggestions and/or contributions to the catalog library are encouraged.

A brief introduction to PHOTCAL is available on-line by typing the command **`phelp pcintro`**. More detailed information on using the package, especially in the context of calibrating APPHOT and DAOPHOT reductions, can be found in *A User's Guide to Stellar CCD Photometry with IRAF* by Phil Massey and Lindsey Davis available from the IRAF network archive.

Lindsey Davis
Pedro Gigoux

New OBSERVATORY Database

Some astronomical tasks require information about the location of the observation such as the latitude. In V2.10 this information is determined from an observatory database using the name or abbreviated identification of the observatory. The simplest and safest way is if the observatory identification is recorded in the image header under the keyword OBSERVAT. If this is done then independent of the setting of any parameters the correct information will be retrieved from the database. For example the NOAO observatories, CTIO and KPNO, provide this keyword and have entries in the database so that reducing data alternately (or even simultaneously) from these observatories will be done correctly without having to set any hidden parameters. Generally the tasks which use the observatory information will record the information used in some type of log output for your confirmation.

If either the data does not have an observatory identification in the image header or the observatory information is not in the database then things become a little more complicated. In the latter case we would appreciate receiving observatory information (name, identification label, latitude, longitude, altitude, and time zone) for inclusion in the distributed database; email iraf@noao.edu. There are options to allow a site to define a default observatory and/or alternate database in the site configuration files, for individual users to define these with simple variables in their login files, or to simply enter values in a parameter file (the latter is similar to the previous mechanism). Also the individual tasks provide a parameter to override the default observatory. Rather than describe these alternatives, which can sound confusing without some reflection, please read the help topic for "observatory".

Frank Valdes

CCDRED in V2.10

The CCDRED package has a small number of changes though functionally it is basically the same. One significant change is that COMBINE and the related CCD type-specific combining script tasks are based on the new IMCOMBINE task. This new task is discussed in a separate article. The main effect is that there are more parameters and capabilities for combining images, but the defaults provide essentially identical functionality.

Two common "gotchas" affecting CCDRED, which have been discussed in past newsletters, have been fixed in V2.10. The first is that loading a package such as CCDRED in the *login.cl* or *loginuser.cl* scripts now works properly. Previously doing this froze the package parameters so that they could not be changed during the IRAF session. This was a significant problem with CCDRED since the package parameters control the instrument setup. The second problem was the termination of CCDPROC with a misleading error about accessing a temporary image. This was caused by the improper deletion of images using the Unix "rm" command which would leave hidden files that are hard links to the originally image header. Attempting to process a new image with the same name caused the error. While it is still improper to delete images using Unix commands the condition of having leftover hidden link images is now silently handled.

The original design of CCDPROC had in mind the possible desirability of producing short (16 bit) output images to save disk space. With every larger CCD format this remains a useful feature. However, the parameter specifying the output pixel type, "pixeltype", also defined the default calculation type. Generally it is not desirable to use short arithmetic for intermediate products so this was a problem. The pixel type parameter has now been extended to take two words to specify the output pixel type and the calculation pixel type. If the second word is missing then a real calculation type is selected. Thus, setting the pixel type to "short" or "short

real" now does the internal calculations in real floating point arithmetic but truncates the final processed image to a 16 bit integer to save a factor of two in disk space.

For efficiency flat field division is not checked for zero (or negative) divisors. While it is rare to find a zero value in a processed flat field it is not impossible. A new option is to specify a minimum output value, with the "minreplace" parameter, to be produced when processing a flat field. The default value of 1 will insure that when this flat field is applied to object images an arithmetic abort will not occur. It is important to realize the distinction between checking the flat field divisor on each pixel when applied to each object image, which is not done, with replacing low pixel values when processing a flat field image, which is done. The latter is only done once and only to one type of image being processed. This technique will not prevent arithmetic aborts if the flat field is modified outside of CCDRED with a task which allows zero or negative values.

The last change to note is that there is a new task which aids in setting up CCD header translation files. This task, CCDINSTRUMENT, works by example with existing image headers making it somewhat simpler to get the translation format correct.

Frank Valdes

Radial Velocity Package Update

The external RV0 package has been included in V2.10 IRAF as the new NOAO.RV package (not to be confused with the old prototype RV external package). This new package is largely what people have been using for a while now, however there have been a few changes to accommodate some new system enhancements in IRAF. For example, the OBSERVATORY pset has been removed and the "observatory" parameter is now a string variable indicating the observatory name (e.g. "kpno", "ctio", etc).

The FXCOR task was modified to understand the new MWCS (coordinate system) image header information that is output by the spectroscopy tasks, although older header formats are still supported. There have been numerous bug fixes since the last release of the external package and users are encouraged to refer to the Revisions file, **`phelp rv.revisions opt=sys`**, for a list of changes. Since this is a newly installed package there may still be bugs, and indeed several have shown up since the V2.10 release in April; these have been fixed in the IRAF V2.10.0 (patch0) release that is now available in the IRAF network archive. Please contact the author (fitz@noao.edu, 5355::fitz) with any questions or problems you may have.

Mike Fitzpatrick

The Version 2.10 NOAO Spectroscopy Packages

The IRAF NOAO spectroscopy software has undergone many major revisions in Version 2.10. The purpose of this article is not to describe the revisions but to call your attention to a number of documents which are available covering the changes and to highlight a few of the general areas which are novel or of special importance.

The NOAO spectroscopy packages are ONEDSPEC, APEXTRACT, LONGSLIT, MULTISPEC, and a number of new and old packages in IMRED. Of these packages the LONGSLIT package is the only one which is basically unchanged. The MULTISPEC package has been

eliminated as obsolete and unsupportable with the intention to someday implement some of its unique capabilities in future tasks or packages.

The fundamental spectroscopy packages are ONEDSPEC and APEXTRACT. These have been greatly revised. There are revision notes and summaries accompanying this newsletter as well as specific documents available in the network archive. The documents are *iraf/docs/onedv210.ps.Z* and *iraf/docs/apexv210.ps.Z*. You may also get a summary of the task revisions in the ONEDSPEC package directly on-line using the REVISIONS section of the help pages. To do this type:

```
cl> phelp onedspec.* section=revisions
```

In addition to the task help pages there are new on-line help topics which discuss general issues and algorithm details. The list of help topics is found by typing "help <package>" where package is one of the packages. I urge you to read the general package discussion for APEXTRACT and ONEDSPEC. This is best done with the commands:

```
cl> phelp onedspec.package
cl> phelp apextract.package
```

Besides the basic ONEDSPEC and APEXTRACT packages there are a number of new generic and instrument specific packages in the IMRED package designed for the reduction and calibration of spectroscopic data. The organization and philosophy of these various packages are discussed in an article to appear in the Proceedings for the first Astronomical Data Analysis Software and Systems Conference. Within the the spectroscopic reduction packages are a number of consolidated and streamlined tasks for specific types of data. For example there are specialized tasks for extracting and calibrating multifiber data. Each of these specialized tasks has a user's guide which may be obtained from the network archive in the directory *iraf/docs*.

There are many changes in the ONEDSPEC package and even the major highlights make a substantial list. Here I mention the highlights of the highlights which are generalized dispersion coordinates, the choice of sinc interpolation, use of long slit spectra in most tasks without need to "extract", greater flexibility with determining and applying dispersion functions (IDENTIFY/REIDENTIFY/DISPCOR), and a number of new tasks. The coordinate system changes are mentioned in other articles as one of the main focuses of the new release. Coordinates are now handled by a system wide interface allowing different applications to obtain coordinates, such as to allow the IMAGES and PLOT tasks to use wavelengths from spectra, and to keep track of changes to the image sampling due to, for instance, image sections.

The coordinate system interface allows different applications to define specialized coordinate functions which can be used by other applications without knowing the details of the keywords or function. One coordinate type common to many applications are independent linear coordinates. This is what is used in long slit calibrated spectra and linearized one dimensional spectra. This type of system is straightforward and is largely familiar. In addition the spectral applications define a new coordinate system type called "multispec". This type allows storing multiple spectra with independent dispersion functions in a single image and the dispersion functions can be linear, nonlinear functions, and look-up table type coordinates. The details of this are given in the on-line help "onedspec.package".

A significant point about the "multispec" coordinate type is the nonlinear functions. Basically, the dispersion functions determined by tasks such as IDENTIFY and ECIDENTIFY are stored in the image headers and used directly without requiring the spectra to be resampled. It is now possible, if desired, to skip linearizing spectra and to flux calibrate and measure equivalent widths without resampling.

A related capability is that if spectra need to be combined arithmetically the operations are done in wavelength space. Spectra will be automatically resampled to match dispersions (not necessarily linear) before the operation. This is in contrast to combining spectra by matching pixel coordinates as was the case previously.

An attempt has been made to allow use of prior spectral image formats as input; any new spectra created will use the new keywords. In particular this means the old "onedspec" and "multispec/echelle" formats. However, things are somewhat sensitive so if you attempt to modify anything, say with HEDIT, the spectra may not be recognized correctly. One example of this is that in the previous releases W0 and WPC had precedence over CRVAL1 and CDELT1/CD1_1; the reverse is now the case. So if you modify W0 and WPC intending to make a simple change to the wavelength scale this will not work.

Now let's turn to the APEXTRACT package. The new version retains some of the flavor of the previous version and adds new tasks and algorithms. There are also several new help topics which discuss the general aspects of the algorithms and software. The highlights of the highlights in this package are a consolidated task called APALL and new and improved cleaning and variance weighting (optimal extraction) algorithms. The new task collects all the parameters which are used in a single parameter set. It was a common complaint that the organization of parameters by function causes concern and confusion about what parameters are actually used. For most people the new APALL will be the preferred task though the original division of functions into separate tasks is still available.

The new cleaning and weighting extraction algorithms are more robust than the previous ones. They do rely on having a good knowledge of the readout noise and gain (this is a more conventional form than the old v0 and v1 parameters). They can make a useful improvement in the extracted spectra. For reference, there is an paper in the Tucson software conference proceedings giving some quantitative measures of the performance of the new algorithms.

Frank Valdes

What's New in SPLOT

Since SPLOT is probably the most heavily used task in the one dimensional spectroscopy packages it merits it's own article. There have been many changes made in the V2.10 version of SPLOT. Many of these are behind the scenes support for general changes made to the spectroscopy software as a whole. An example of this is use of world coordinate systems defined in the image headers. Any of the various coordinate systems will be accepted. This includes linear coordinates, the new non-linear spectral dispersion functions, the multiaperture spectral format, and even RA/DEC coordinates. As part of this capability image sections may now be used without the problem of confusing the dispersion coordinates. In particular, plotting a part of a very long spectrum will provide the correct coordinates.

SPLOT is a very function rich task, probably too rich. This means that all of the available keys are used. In order to introduce new capabilities there had to be a reorganization of the key definitions. Therefore, many of the keys you are familiar with may now be different. You should either carefully read the help page, or more likely, type "?" when you first use the new task.

Judging from early feedback the most common confusion concerning the new task is how to overplot a spectrum. Since there are now a number of ways to select a second spectrum, particularly with multiaperture or long slit spectra, the "o" does not query for the spectrum to be overplotted. Instead it sets a flag to cause the next spectrum to be overplotted rather than erasing and drawing a new graph. Thus, following the "o" key you would use one of the many keys to select what to overplot. This behavior is similar to that of IMPLOT.

Another source of possible confusion is the behavior of the "r" redraw key. The old redraw key actually did two things; it autoscaled and redrew the entire spectrum. This made it incompatible with the "a" key (and the related "," and "." keys) which nicely allows selecting a region of the

spectrum to be expanded and autoscaled. In the new version the redraw key redraws the current spectrum in the current graph limits. To autoscale and redraw the entire spectrum the new "c" key is used.

As mentioned earlier, there are now several ways to select a new spectrum to be plotted. The "g" key to specify a new image name is unchanged. However, if the current image is a multiaperture image or a long slit image, and a different aperture, band (from the 3D APEXTRACT "extras" format), line, or column is desired, the keys "#", "%", "(", and ")" may be used. The latter two are particularly efficient for examining the many fiber spectra produced by such instruments as Argus and Hydra.

One of the requests we frequently received was to allow deblending of more than four components. There is now no limit to the number of components, though the speed of deblending slows down dramatically if the number becomes too large. In addition, the deblending now includes simultaneous fitting for the background as well as a few new features described in the help. Selection of what is to be fit is now more obvious rather than using a set of special acronyms.

At the opposite extreme of fitting more Gaussian components, people sometimes used the deblending option to fit a Gaussian profile to a single line. Rather than go through all of the steps and keystrokes required in the general deblending operation there is a new key, "k", which given two continuum endpoints fits a single Gaussian, prints and records the result, and immediately returns to the main SPLOT command menu.

The old "k" key as well as "h" and "v", which provided specialized equivalent width measurements, have been combined into a new "h" key. A second key then selects which of the algorithms to use. A new feature is that the profile parameters generated by any of these keys may be used with the "-" key to subtract this profile. Of special interest in this regard are the one-sided algorithms. One can use one side of a blended-profile for estimating a Gaussian profile and then subtract this estimate from under the blended side.

The "t" key now allows a spectrum to be fit interactively with the interactive curve fitting (ICFIT) commands. By judicious use of sample regions and interactive rejection one can fit the continuum or replace the rejected points. The result of this fitting can either be ignored to use ICFIT just for study, be subtracted from the original continuum, be used to create a smooth continuum spectrum, or be used to replace the rejected points by an estimated continuum. This is essentially identical to what can be done with the CONTINUUM and SFIT tasks.

With the enhanced support of world coordinate systems comes arithmetic operations performed at matching wavelengths rather than matching pixel coordinates. In the "f" function mode any arithmetic operations performed with a second spectrum will first resample the second spectrum to the same dispersion as the current spectrum (even non-linear dispersions) before performing the operation.

In the past there have been occasional attempts to use SPLOT on spectra with units other than Angstroms. In particular, either using a velocity scale or units such as inverse centimeters. This required fudging the coordinate system keywords and other monkey business. It is clear that to support multiwavelength investigations it is desirable to allow the user to determine the units and not the application. While this change has not yet been fully made in the spectroscopy package as a whole and the units in the image have to be defined in terms of Angstroms, the plotting tasks SPLOT and SPECPLOT now allow converting between various units. SPLOT can plot spectra in units ranging from Hertz to Mev. In addition the units may be modified by taking the inverse, such as for inverse centimeters, and the log, such as for log(Hz). Finally, a velocity relative to some origin may be plotted. The units are set and changed with the "units" task parameter, the units colon command, and the new "v" key for velocity. By the way, the various measurement commands will be performed using the selected units so that, for example, equivalent widths may be reported in microns if the units are microns.

It is now easier to switch back and forth between pixel coordinates and whatever dispersion units are selected. The "$" key is a toggle to switch back and forth. Previously after switching

to pixel coordinates it was difficult to return to the original wavelength coordinates.

One new capability of interest to those with long slit spectra, or any two dimensional image for that matter, is that SPLOT can be used directly to display or measure features in the sum of a number of columns or lines. In a sense this makes SPLOT a kind of IMPLOT with a large set of specialized options pertaining to spectra. Two common uses are for examining raw long slit spectra immediately after acquiring them at the telescope (before any wavelength and flux calibration) and for examining fully calibrated (by the LONGSLIT package for instance) long slit spectra without having to extract regions to a one dimensional spectral format. The number of lines or columns to sum is set with the package "nsum" parameter. The initial direction of the plotted vector, the so-called dispersion axis, is determined either from the DISPAXIS parameter in the image header or by the "dispaxis" package parameter. After the initial graph the direction of plotting and the number of lines/columns to sum can be interactively changed with new colon commands. Note in particular that it is perfectly acceptable to set the dispersion axis to be that which is actually the cross-dispersion dimension in order to examine and fit profiles across the dispersion.

In addition to the various changes to the single cursor keystroke commands there are now a greater variety of colon commands. These include toggling logging, entering comments in the log file of measurements, changing any of the options initially set with the options parameter, and setting the coordinate units and changing the dispersion axis and summing used with two dimensional images, as described previously.

Frank Valdes

Changes to the ONEDSTDS Directory in IRAF Version 2.10

The *onedstds* directory in *iraf$noao/lib* has contained linelists and flux calibration data used by the ONEDSPEC, TWODSPEC, and IMRED packages. The contents of this directory have now been divided; the linelists have been moved to *iraf$noao/lib/linelists* but the flux calibration data remain in *iraf$noao/lib/onedstds*. For convenience, when the NOAO package is loaded the logical directories *onedstds* and *linelists* are defined corresponding to the previously noted directories. There are README files in each directory.

Several new linelists have been added to the *linelists* directory including several lists of night sky lines and infrared emission line lists for argon, krypton, and xenon. The thorium lists have been updated as well. As part of the patch0 release, a linelist specific to the KPNO/Hydra instrument for 6 built-in CuAr hollow cathode lamps was added.

CTIO has added several new sets of flux calibration standards to the *onedstds* directory, and the *spec50cal* flux standards now include data out to 1 micron for some stars (this list was combined with the old *spec50redcal* data).

Please see the README files for more detailed information.

Frank Valdes

Interfacing an HP LaserJet Printer to IRAF

Site support has received many requests recently for information on interfacing the HP LaserJet III series of printers to IRAF. This article addresses the more general topic of how to interface any new printer to IRAF and uses the HP LaserJet as an example. Complete descriptions of the graphics interface and interfacing graphics devices can be found in the GIO help document (see

sys$gio/doc/gio.hlp) and in *The IRAF Simple Graphics Interface (SGI)* document (available as *iraf$doc/sgi.ms* or from the network archive as *iraf/docs/sgi.ps.Z*). It should be noted that this article **will not** cover installing new printers on VMS hosts, or how to get hardcopy output from the SunView Gterm program (contact site support for questions on these subjects, which may be covered in a future article).

Making a new printer known to IRAF is a two-step process; editing the *dev$termcap* file and editing the *dev$graphcap* file. The termcap file is used by the LPRINT command for text file printing. The graphcap file is used for graphics plots by the CL, either for plotting in an interactive session or for making hardcopies. We will begin with a discussion of the changes necessary to the termcap file. The format of this file is the same as the standard Unix termcap file (see termcap(5)), and declares not only the graphics terminals supported by IRAF, but also the printers used by the IRAF LPRINT command.

The key to reading the termcap (or any similarly formatted) file is the "tc" field, which acts as a pointer to another entry in the file for the same device. An example entry for the HP LaserJet IIIsi might look like:

```
hplp|lw|lw1|      :tc=uhplj:

uhplj|Generic entry to HP IIIsi LaserJet:\
    :co#80:li#66:os:pt:ta^I:\
    :DD=hplj,/tmp/asfXXXXXX,!{ lpr -Plw1 $F; rm $F; }:
```

In this example the printer is known to IRAF by any of the aliases "hplp", "lw", or "lw1". The "tc" field of the first entry tells the reader to continue to the "uhplj" entry for more information. The DD field (device-dispose control string) consists of three comma separated fields, the third of which contains the command used to dispose of the output at the host level. Here we see that the printer is defined to the host as "lw1" which **must** be a known device name in */etc/printcap*. Most problems with adding new printers are caused by not having the correct DD string, which should be the same command used at the host level to print the file (the "$F" is a placeholder for a temporary file used internally).

Once the printer is known to the termcap file, the next step is to edit the *dev$graphcap* file to make the printer known as a graphics hardcopy device for the ":.snap" cursor mode command and other plots that go through the CL. Graphcap entries are similar in form but usually more complex than the corresponding termcap entry. The DD string is more complex since it typically calls an SGI metacode translator to produce the PostScript (or other graphics language) required by the printer.

A graphcap entry for the same HP IIIsi printer would look something like:

```
hplp|lw|lw1|      :tc=uhplj:

uhplj|UNIX generic interface to HP LaserJet IIIsi:\
    :BF:WS:XO#0:YO#0:LO#2:LS#2:\
    :DD=uhplj,tmp$sgk,!{ sgidispatch sgi2uhplj $F \
    -l$(XO) -b$(YO) -w$(PX) -h$(PY) $F | lpr -Plw1; rm $F; }&:\
    :tc=sgi_hplaserjet3:

sgi_hplaserjet3|HP LaserJet laser printer:\
    :kf=bin$x_sgikern.e:tn=sgikern:cw#.0125:ch#.0294:\
    :ar#0.755:xs#.2692:ys#.2032:xr#2400:yr#3180:\
    :XO#75:YO#75:XW#2400:YW#3180:PX#2400:PY#3180:LO#1:LS#0:\
    :BI:BF:MF#8:RO:NB#8:
```

Here the device is known by the same three aliases, but you'll see there is much more information about the printer itself. In the V2.10 release of IRAF the last two entries are already in the distributed graphcap file, and only minor editing of the "uhplj" entry's DD string should be

necessary.

In the example given, the "sgi2uhplj" translator is used to produce HP Printer Command Language output from IRAF SGI metacode. However, many sites purchase cartridges for these printers making them PostScript devices. In this case it may be possible to use one of the existing "uapl" entries in the graphcap file to output PostScript to the printer. These entries use the "sgi2uapl" translator to produce PostScript from the SGI metacode. Another consideration is that certain cartridges (e.g. the Pacific Page PostScript cartridge) may require a filter at the host level to switch the printer between native HP graphics and PostScript mode (this filter is available by contacting site support).

Lastly, even with a properly configured graphcap entry there may be reasons why output does not appear at the printer: not enough printer memory, a maximum file size defined in the */etc/printcap* file (see the "mx" description in printcap(5)), or the graphics buffer not being flushed (try typing "gflush" or ":.gflush" from cursor mode to flush the buffer). A useful debugging tool is to spool the output of the SGI translator to a file and then try to print that file from the host level. (For example, try modifying the "uhplj" entry so that instead of piping the translator output to lpr you redirect it to a file in your home directory, then simply print the file by hand using the same lpr command.) Having the file available at the host level will allow you to experiment with different lpr commands, printcap entries, or even SGI translators until you find one that works. Note however that if you modify the graphcap entry you **must** type GFLUSH in the CL to reinitialize the graphics system and pick up the change.

Readers are encouraged to contact the IRAF group with any questions about graphcap or termcap entries.

Mike Fitzpatrick
Suzanne Jacoby

New MEM0 Add-on Package

A task for image restoration using the Maximum Entropy Method (MEM), called IRME0, and a number of related tasks/procedures have been developed in IRAF by Dr. Nailong Wu from Beijing Astronomical Observatory, Chinese Academy of Sciences, while a visiting scientist at NOAO.

These tasks/procedures are available in the new MEM0 add-on package. A detailed description of these programs is presented in the technical report *MEM Task IRME0 in IRAF*. Interested IRAF users can retrieve the package from the IRAF network archive in the *iraf/contrib* directory. The technical report is also available over the network in the *iraf/docs* directory. Only a brief description of the task IRME0 is given in this article. Specific questions about the programs should be directed to Dr. Wu (nailong@stsci.edu).

IRME0 is a task for 2-D image restoration using MEM. In searching for the maximum value of entropy, the approximate Newton method for optimization is used. To run the task, several input image files are required: (1) the degraded image to be restored, (2) the point spread function, and optionally (3) the prior estimated image. The basic input parameters required for deconvolution are: the rms value of the noise, the total power (flux) of the image, and the number of pixels in the psf. These parameters may be entered by the user or calculated automatically.

Output messages help the user to monitor the iterations. The total CPU time is estimated to be "somewhat more than the CPU time for one FFT x 2 x the number of iterations". At NOAO, for example, the CPU time for one FFT is about 7 seconds for a 256x256 image on Draco (VAX 8600), so 30 iterations require roughly 600 seconds or 10 minutes. On Gemini (Sun 4/370), the required CPU time is approximately 180 seconds or 3 minutes for the same image and number of iterations. Space for seven images (including working space) must be held in core memory.

Therefore, to process a 256x256 image requires about 0.25 x 7 = 1.75 Mbytes of memory. The following table presents the CPU time in seconds for a 2-D FFT on a variety of host computers at NOAO.

Computer	64x64	128x128	256x256	512x512	1024x1024
SPARCstation 2	0.03/0.05	0.22/0.25	1.0	4.7	20
SPARCserver 670MP	0.03/0.05	0.20/0.25	1.0	4.7	21
IBM RS/6000	0.21	0.25	1.1	4.7	20
Sun 4/370	0.08	0.4	1.9	8.5	37
Silicon Graphics	0.25/0.07	0.9/0.3	3.5/1.3	14/6	55/26
DECstation 3100	0.28/0.07	1.0/0.35	3.9/1.6	15/7	62/31
VAX 4000-200	0.15	0.8	3.3	15	75
VAX 8600	0.3	1.5	6.5	29	124
Sun 3/260	0.9	4.1	18	80	350
Sun 3/60	1.21/1.27	5.6/5.9	25/27	110/120	
VAX 750	1.5	7.0	30	145	

(Note: CPU time at first call / at second (third ...) call in IRFFTES)

Satisfactory results were obtained in the preliminary tests. It should be pointed out that this first MEM task in IRAF is not perfect, but usable. More tests are needed. The development of a better MEM task on the basis of IRME0 is under consideration. Users are encouraged to run the task, find bugs, and even revise the source code. Advice: learn something about MEM and SPP before rolling up your sleeves to do the latter!

Nailong Wu
Visiting Astronomer, NOAO

IRAF Tools for Accessing the ADC CDROM

An IRAF external package for accessing many of the 114 catalogs in Volume I of the Astronomical Data Center (ADC) CD-ROM collection has been developed. The tasks are specific to the text file format CD-ROM. There are two tasks, one for selecting and printing data from tabular catalogs and one for extracting spectra to 1D IRAF spectral images.

The CATALOG task provides a convenient user interface to a large number of tabular catalogs. The catalog to be accessed is specified by a simple catalog name. One goal of this task is that users need not know the CD-ROM directory structure or the full file names. A list of catalogs, organized by type of data, may be paged by specifying "?" for the catalog name.

The purpose of this task is to allow selecting and printing a subset of the data in the designated catalog. The output is a readable text table of the field values for the selected entries. This table may be printed to the terminal, piped to another task, redirected to a file, or explicitly directed to a file.

The catalogs are tables having a number of entries with each entry having the same set of columns or fields. Thus, one may select entries based on some function of the fields and then select which fields to be output. Since one doesn't initially know what fields are contained in a particular catalog a list of the available fields may be paged by entering "?".

To select entries a boolean selection expression is specified. The expression consists of various operators applied to the catalog fields. Hence the field names must also be known here and a "?" value for the expression parameter will provide a list of fields. The expression syntax is

similar to that used in HEDIT and HSELECT. A typical expression consists of a set of equality or inequality tests on various fields combined together by logical ands and ors. The expression may be taken from a file by specifying @<file> where file is the filename containing the expression.

A powerful feature is the ability to define special macro fields which are expressions and then refer to these expressions by name in both the fields and expression parameters. Two such macros, "ra" and "dec", are automatically defined if the catalog contains fields specifying the right ascension hours (RAH), minutes (RAM), and seconds (RAS) and the declination degrees (DecSign and DecD), minutes (DecM), seconds (DecS) separately. The macros combine these fields into a single numeric field which may be used for selection and printing in sexagesimal notation.

An example of using the catalog task to examine the available catalogs, select a catalog, list the fields of the catalog, and extract a subset of entries is shown below.

```
ad> catalog
Catalog name (? for list): ?
<List of catalogs>
Catalog name (? for list): fk4
Subcatalog name (? for list): ?
Choose one of the following subcatalogs in fk4:
        data50
        data75
        descript
        polar55
        polar60
        polar65
        polar70
        suppl
Subcatalog name (? for list) (?): polar70
Fields to print (? for list): ?
<List of fields>
Fields to print (? for list): FK4,ra,dec,Mag,SpType
Selection expression (? for list): dec<-85:30 && Mag<7 && evsptype (SpType, "A")
1663  10:32:53.15 -85:56:08.30  6.74 A0
1665  13:35:43.58 -85:38:01.92  5.65 A2
 920  15:11:36.63 -88:01:29.54  6.52 A2
 921  16:49:23.73 -86:19:03.43  6.13 A0
1669  22:31:50.54 -88:58:29.39  6.54 A5
```

The SPECTRA task provides a convenient user interface to the catalogs of spectra contained in the CD-ROM collection. The catalog to be accessed is specified by a simple catalog name. One goal of this task is that users need not know the CD-ROM directory structure or the full file names. A list of catalogs is paged by specifying "?" for the catalog name.

The purpose of this task is to allow extracting a subset of the spectra in the designated catalog into one dimensional IRAF spectral images. Each catalog or library of spectra consists of a number of similar spectra. To designate a spectrum or spectra to be extracted one specifies a list of identification numbers which are just the order index in the catalog. To get a directory of the spectra with the identification numbers and titles one enters "?" for the spectra parameter. The selected spectra are output as one dimensional IRAF spectral images. The image parameter specifies a root image name to which the spectrum ID number is appended. The image header contains sufficient information to allow plotting and manipulating the spectra with the IRAF spectroscopy tasks.

Available ADC CD-ROM Catalogs of Spectra	
iuelda	IUE Low-Dispersion Spectra Reference Atlas. I. Normal Stars
iueostar	IUE Atlas of O-Type Stellar Spectra from 1200 to 1900 A
spatlasb	Stellar Spectrophotometric Atlas 3160-5740 A
spatlasr	Stellar Spectrophotometric Atlas 5760-10620 A
splib	A Library of Stellar Spectra
spstd	Spectrophotometric Standards
uvbs	Ultraviolet Bright Star Spectrophotometric Catalogue
uvbssupp	Supplement to the UV Bright Star Spectrophotometric Cat

This software is available as the file *iraf.old/adccdrom.tar.Z* in the IRAF anonymous ftp account or via other media upon request. The tasks were developed under V2.10 but it is expected that the CATALOG task is compatible with V2.8 and V2.9 and the SPECTRA task is compatible with V2.9.

Frank Valdes

IRAF Documentation

Most of the IRAF documentation is now available in the IRAF network archive (in the directory *iraf/docs*) as compressed PostScript or compressed ASCII text files. This includes past issues of the *IRAF Newsletter*. The *iraf/docs/README* file contains a complete listing of what is available in the archive.

A number of new or updated IRAF documents or manuals have become available, mostly dealing with the release of IRAF version 2.10. In the following, the file name in the network archive is given at the end in parentheses.

- A simple Table of Contents for all of the IRAF Newsletters is available only in the network archive. (TOC_news.txt)
- Two Table of Contents files are available that list the contents of IRAF User Handbooks Volumes 1A and 2B, that can be purchased from NOAO. (TOC210_vol1a.txt, TOC210_vol2b.txt)
- *APEXTRACT Package Revisions Summary: IRAF Version 2.10*, by Francisco Valdes, May 1992. (apexv210.ps.Z)
- *Sun/IRAF Installation Guide*, by Doug Tody, Revised July 1992. (suniraf.ps.Z)
- *Sun/IRAF Site Manager's Guide*, by Doug Tody, Revised July 1992. (sunsmg.ps.Z)
- *Sun/IRAF Version 2.10 Directory Structure*, May 1992. (d210sun.ps.Z)
- *A User's Guide to CCD Reductions with IRAF*, by Philip Massey, June 1992. (ccduser2.ps.Z)
- *A User's Guide to Stellar CCD Photometry with IRAF*, by Philip Massey and Lindsey Davis, April 1992. (daophot2.ps.Z)
- *Guide to the Coude Three Fiber Reduction Task DO3FIBER*, by Francisco Valdes, April 1992. (do3fiber.ps.Z)
- *Guide to the ARGUS Reduction Task DOARGUS*, by Francisco Valdes, April 1992. (doargus2.ps.Z)
- *Guide to the Slit Spectra Reduction Task DOECSLIT*, by Francisco Valdes, April 1992. (doecslit.ps.Z)

- *Guide to the Kitt Peak Coude Fiber Reduction Task DOFIBERS*, by Francisco Valdes, April 1992. (dofibers2.ps.Z)
- *Guide to the Fiber Optic Echelle Reduction Task DOFOE*, by Francisco Valdes, April 1992. (dofoe.ps.Z)
- *Guide to the HYDRA Reduction Task DOHYDRA*, by Francisco Valdes, April 1992. (dohydra.ps.Z)
- *Guide to the Kitt Peak Coude Slit Reduction Task DOSLIT*, by Francisco Valdes, April 1992. (doslit2.ps.Z)
- Glossaries of the IRAF and NOAO Packages, IRAF Version 2.10. (glos210a.ps.Z, glos210b.ps.Z)
- *MEM Task IRME0 in IRAF*, by Nailong Wu, May 1992. (mem0.ps.Z)
- *ONEDSPEC/IMRED Package Revisions Summary: IRAF Version 2.10*, by Francisco Valdes, May 1992. (onedv210.ps.Z)
- *A User's Guide to Reducing Slit Spectra with IRAF*, by Phil Massey, Frank Valdes, and Jeannette Barnes, April 1992. (spect.ps.Z)
- *Preliminary Test Procedure for IRAF, IRAF Version 2.10*, by Jeannette Barnes, revised May 1992. (testproc2.ps.Z)
- *IRAF Version 2.10 Revisions Summary*, July 1992. (v210revs.ps.Z)
- A directory called *focas* has been added containing three FOCAS documents.

Please contact us if you would like to receive printed copies of any of the IRAF documents (iraf@noao.edu).

Jeannette Barnes

Add-on Software Available for IRAF Versions 2.8/2.9/2.10

The following software packages are available as add-ons to IRAF Versions 2.8, 2.9, and in some cases 2.10. Note that several of the add-ons have been included in the V2.10 release and so are not considered add-ons to V2.10. All packages are available via file transfer from the IRAF network archive on iraf.noao.edu (Internet node 140.252.1.1) in the directory *iraf.old* and have *readme* files containing instructions for transfer and installation. Unless specified otherwise please contact the IRAF hotline for further information (iraf@noao.edu).

- ADC CD-ROM utility package - two IRAF tasks that extract data from Volume I of the ADC CD-ROM text version. See the accompanying article in this newsletter.
- IMCOMBINE task (V2.9 only) - the new IMCOMBINE task, installed in V2.10, is available as an add-on for IRAF Version 2.9 (see the accompanying article in this newsletter).
- ARTDATA (V2.8 only) - the new artificial data package. This is available as an add-on to IRAF Version 2.8 only as it is included with IRAF Versions 2.9 and 2.10. See the article in IRAF Newsletter Number 10 (October 1990).
- IUEECHELLE package - a prototype package to support a particular format of IUE Echelle spectra. See a discussion of this software in IRAF Newsletter Number 7 (June 1989). For further information contact Frank Valdes (fvaldes@noao.edu, 5355::fvaldes).
- NEWIMRED (V2.8 and V2.9 only) - the first complete release of the new spectroscopic reduction packages for a variety of KPNO/CTIO instruments. A revised version of this package has been included with IRAF Version 2.10. See articles in IRAF Newsletters Number 8 (October 1989), Number 9 (February/June 1990), and Number 10 (October 1990). Contact Frank Valdes for further information (fvaldes@noao.edu, 5355::fvaldes).

- DAOPHOT/IRAF (V2.8 and V2.9 only) - the IRAF DAOPHOT digital stellar photometry package. DAOPHOT/IRAF is included in IRAF Version 2.10, but the original TEST-PHOT version is still available as an add-on for older releases of IRAF. See a discussion of the add-on software in IRAF Newsletter Number 8 (October 1989).
- RV0 (V2.8 and V2.9 only) - the Radial Velocity analysis package. A beta test version of this software was released in April 1991; anyone using the original prototype version of RV should upgrade to this new, much more thoroughly tested version. RV0 is included in IRAF Version 2.10 as the package RV. Contact Mike Fitzpatrick for more information (fitz@noao.edu, 5355::fitz).
- Volume rendering software - this software has been discussed in previous issues of the IRAF Newsletter (Number 5 October 1988 and Number 6 February 1989).
- IRAF demos (V2.8 and V2.9 only) - a set of IRAF demos for tasks in the IRAF and NOAO packages as well as an IMTOOL tutorial. A revised package was made available in September 1990. Contact Jeannette Barnes for further information (jbarnes@noao.edu, 5355::jbarnes).
- Kernel server kits - may be installed to remotely access tape drives or the workstation display via IRAF networking, as an alternative to installing or NFS mounting the full IRAF system (UNIX/IRAF hosts only). Note that if IRAF is already installed on another node in your local network which is architecturally compatible and accessible via NFS, it may be simpler to NFS mount and install IRAF than to install the kernel server kit.
- SAOimage - an X Window System based display server for IRAF developed by Mike VanHilst at the Center for Astrophysics for workstations running X11 (see article in IRAF Newsletter Number 8 October 1989).
- UISDISP display software for VMS Workstations - this software was discussed in IRAF Newsletter Number 7 (June 1989). This version is included with VMS/IRAF version 2.9. For further information please contact Nigel Sharp (sharp@noao.edu, 5355::sharp).
- Gould DeAnza IP8400/8500 display software (VMS only).

Several add-on packages are also available in the *iraf/contrib* directory. The software in this directory is contributed by persons outside the IRAF group and the author or authors are solely responsible for the software. Users with IRAF-related software they would like to share are encouraged to contribute their own software to this directory (the directory is world writable).

- MEM0 package - a new MEM package using the Newton-Raphson optimization method, developed at NOAO by Nailong Wu, a visiting astronomer from Beijing Astronomical Observatory. Questions about the algorithms can be sent directly to nailong@stsci.edu. See the accompanying article in this newsletter.
- STSDAS TABLES V1.2.1 - the latest release of the TABLES package distributed by the STSDAS group at Space Telescope Science Institute.

The IRAF Group

Erratum

In the last issue of the *IRAF Newsletter* (April 1991 Number 11), in the "System News" article, we incorrectly referred to the Silicon Graphics Personal IRIS workstation on loan to us from Silicon Graphics Computer Systems as a "Silicon Graphics Personal IRIS 2". There is no Silicon Graphics computer with that exact name.

The Editors