IRAF NEWSLETTER

October 1990 Number 10

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

Table of Contents

System News 1

Release of IRAF Version 2.9.1 3

Changes to the ARTDATA Package: Version 1.1 4

CCDRED News 5

IMRED Package Revisions Summary: IRAF Version 2.10 5

ONEDSPEC Package Revisions Summary: IRAF Version 2.10 6

APEXTRACT Package Revisions Summary: IRAF Version 2.10 7

Radial Velocity Analysis Package Being Revised 8

Photometry Package, BASPHOTC, Available from Hawaii 9

Problems with Long Image Headers 10

Getting Help in IRAF 11

Parameter Sets or Psets 11

Saving IRAF Plots in PostScript Files 13

Using PostScript Fonts with IRAF Plots 14

FOCAS News #4 16

Astronomical Calculator and Calendar Programs 17

IRAF Documentation 17

Add-on Software Available for IRAF Versions 2.8/2.9/2.9.1 18

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

IRAF HOTLINE SERVICES

telephone:	(602) 323-4160
FAX:	(602) 325-9360
Internet:	iraf@noao.edu
SPAN:	noao::iraf or 5355::iraf
BITnet:	iraf@noao.edu (through a gateway)
UUCP/Usenet:	{arizona,decvax,ncar}!noao!iraf uunet!noao.edu!iraf

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

An update to version 2.9 IRAF, known as version 2.9.1, was prepared during the summer and released in early September. A major objective of the version 2.9.1 update was to support the beta release of PROS, the X-ray data analysis software for the ROSAT mission. In addition, IRAF V2.9.1 includes limited support for the Exabyte tape drive under various SunOS drivers, limited support for the new Sun/IRAF version 1.3 Fortran compiler, an updated ARTDATA package, and numerous bug fixes. Sites that have already installed IRAF V2.9 do not have to do a full installation to install the V2.9.1 update; a patch is available which can be installed in a few minutes without modifying the site dependent files. Updates are available for all host systems supported by V2.9. See the accompanying article in this newsletter for details about the IRAF V2.9.1 release.

A second patch release, V2.9.2, is planned for early 1991, with the next major release being V2.10, due out in the late spring or early summer of 1991 (work on V2.10 has been underway for some time; these patches are still essentially V2.9, with only minor changes). The V2.9.2 patch will be required to run the first production version of PROS (V1.0), scheduled for release early next year, when ROSAT starts returning the first pointed data to general observers. Currently there is no firm date for the V2.10 IRAF release.

The network archive used for IRAF distributions, documentation, etc. has a new Internet host address as of August. If you have had trouble accessing the archive recently it may be because this number has changed. The new Internet number is 140.252.1.1. The host name, tucana.tuc.noao.edu is unchanged; if you have the Internet name server running locally you can access tucana (or any other node on the Internet!) directly by name. We can track access to the archive more easily if you give your last name as the password when you access the archive.

The limited support for Exabyte drives under SunOS available in the V2.9.1 patch is only a short term measure as we continue to revamp the entire IRAF magtape system to support all the new types of devices now available. The IRAF project has recently acquired a DAT drive and we will be evaluating this as an alternative to the Exabyte. A new magtape driver has been developed for IRAF which will support DAT, Exabyte, and 1/4 inch cartridge tapes as well as conventional 1/2 inch reel tape, both fixed and variable block devices, and which is "programmable" via a device file to support a wide range of devices and host systems. Due to the extent of the changes required, we will probably not be able to release this until the full IRAF V2.10 system ships.

Two new IRAF ports were completed this summer. An IRAF port to the Apollo DN-3500 was completed, and Domain/IRAF SR10.2 for the DN-3500 and the DN-10000 is now in distribution. The second IRAF port was to a MIPS RC2030 computer running RISC/os 4.10. The MIPS system is on indefinite loan to NOAO and the IRAF project for the purpose of supporting MIPS/IRAF. MIPS/IRAF is now in distribution. The A/UX (Macintosh) port is still on hold pending a host of system upgrades, but everything needed has finally arrived and the port will go forward as soon as the system upgrades are completed.

The engineering group at NOAO-Tucson has kindly consented to let us use their HP9000/370 workstation to maintain HP-UX/IRAF. An upgrade of HP-UX/IRAF to IRAF version 2.9.1 for both the Motorola and RISC architectures is in the final stages of testing and should be available for distribution by the time this newsletter is printed. Mike Cobb at the Naval Research Lab in Washington, in collaboration with Steve Rooke in the IRAF group, has been porting IRAF V2.9.1 to the Alliant computer at NRL. The system is nearly ready for internal testing and will be made available for distribution before long.

Two new VAXstation 3100 computers arrived in July, loaned to us by the LDP group at DEC for the purpose of better supporting VMS/IRAF in the workstation environment. These systems are being used to develop and test IRAF with X11 (DECwindows) operating in the VMS environment. Experimental versions of SAOIMAGE and XTERM, specially modified for VMS, are currently being tested on these systems.

Silicon Graphics Inc. has granted an indefinite loan of a Personal IRIS 2 workstation (4D25/GTX) to the IRAF project for an IRAF port and subsequent support. Unique features of this workstation include a 32 bit frame buffer (24 bits RGB, 8 overlay), a 24 bit Z-buffer, and enhanced 3D graphics hardware. We plan to use this machine to develop support for 24 bit (true color) image display under X11, and to investigate the potential of the advanced graphics capabilities of workstations such as the SGI for astronomical applications.

IBM has also given us an indefinite loan of an RS/6000 Model 520, their new RISC workstation running AIX. This is reported to be a very fast machine, especially for floating point. Again the purpose of the grant is to port IRAF to AIX and be able to provide ongoing support and maintenance. This machine is also equipped with an accelerated 24 bit frame buffer (IBM actually uses the SGI hardware and display software, which should make life easier for us). We also plan to buy a 24 bit frame buffer for the Macintosh (thanks to NASA who has given us much needed R&D funds), so along with the other facilities we already have such as the GX boards for our Suns, we will shortly be in a very good position to develop and test our new X based graphics and image display software.

The IRAF group (primarily Doug Tody, Frank Valdes and Lindsey Davis) began work this fall on a new IRAF based data acquisition program called CCDPHOT. CCDPHOT simulates a conventional photoelectric photometer in software, using digital aperture photometry and fast subregion readout of a CCD to replace the hardware apertures and photomultiplier tube used in the classical photoelectric photometer. The CCDPHOT project was initiated by the Kitt Peak scientific staff in response to the realization that bright time could be better utilized by having a CCD photometer, that a CCD based photometer could significantly increase the efficiency with which certain types of programs are run at the telescope, and to reduce dependence on the conventional photoelectric photometers which may be faced with decreasing support in the future. The project will also serve as a pilot project to help learn how to make better use of IRAF in the mountain environment.

Work continued throughout the summer on other application software projects as well. Lindsey Davis continues her work with Pedro Gigoux of CTIO on a new photometric calibration package for IRAF, and with Peter Stetson of DAO on incorporating new techniques for fitting the PSF for undersampled data into the IRAF/DAOPHOT package. Lindsey has also been working on interactive tools for examining the output catalogues generated from both the APPHOT and DAOPHOT photometry packages. Frank Valdes has completed his work on modifications to several spectroscopic packages including ONEDSPEC, NEWIMRED, and APEXTRACT (see the accompanying articles in this newsletter). Mike Fitzpatrick continues his work on the radial velocity package (see accompanying article in this newsletter).

As mentioned in the last IRAF Newsletter, planning is currently underway for a data analysis software conference to be sponsored by the major IRAF centers (NOAO, STScI, and SAO, with help from Berkeley/SSL and DAO). The conference, the *First Annual Conference on Astronomical Data Analysis Software and Systems*, will be held in Tucson November 6-8, 1991. A preliminary announcement and preregistration form is attached to the back of this newsletter. Anyone interested in attending is asked to please fill out the form on the back of this announcement and return it to us so that we may include you on our conference mailing list. You can send the information via electronic mail if you wish. Registration packets will be mailed sometime next year to everyone on the conference mailing list. Anyone who uses or develops astronomical data analysis software - not just those who use IRAF - would probably find this conference worth attending.

Doug Tody

Release of IRAF Version 2.9.1

An upgrade to IRAF Version 2.9, IRAF Version 2.9.1, was made available in early September. IRAF V2.9.1 replaced IRAF V2.9 in the network archive on September 6 and is also being distributed with all mailed distributions after that date. For sites that are already running IRAF V2.9 a patch is available, minimizing the amount of effort necessary to upgrade to IRAF V2.9.1, for those sites that wish to do so.

IRAF V2.9.1 includes the following bug fixes or enhancements to V2.9.

- Support for the beta test release of PROS has been added. This involved minor changes to the system interfaces QPOE, MWCS, and PLIO. Any site wishing to run PROS should first upgrade to V2.9.1.
- On Sun/IRAF systems only, the IRAF magtape driver has been hacked to add support for Exabyte tape drives. The following combinations of SunOS systems and host drivers are supported by this patch:
  - Sun-3, Sun-4 running 4.0.3 with the Sun SCSI tape (ST) driver.
  - Sun-3 running 4.0.3 with the Ciprico Rimfire (RT) driver.
  - Sun-3 running 4.1 with the Sun SCSI tape (ST) driver. It is possible that a Sun-4 under this combination will work as well, although this was not tested.

  Exabyte on a sparcstation is not currently supported, due to serious bugs in the Sun SCSI driver on this system. 1/4inch cartridge tape drives are not supported. 1/2inch reel tape on the Sun SCSI driver is not supported under 4.0.3, although there is a chance it may work under 4.1. 1/2inch reel tape on a non-SCSI driver is supported in all versions of IRAF. [NOTE - IRAF version 2.10 will include support for all of the above, plus DAT drives.]

  The Exabyte is interfaced in IRAF as is any other tape device (you will need to make a new logical device entry in *dev$devices*). You can do anything with the Exabyte that you can with 1/2 inch reel tape, except truncate the tape and overwrite an existing file, as the Sun drivers permit files to be written only at BOT or EOT with this device. You can randomly position to and read any file on the tape. Tape records are variable size as for 1/2 inch reel tape (and unlike cartridge tapes, which have a fixed block size of 512 bytes). Beware that although the i/o transfer rate is not bad, tape positioning operations execute VERY slowly on the Exabyte. If you need to interrupt a tape operation you may have to wait up to a minute for the task to respond to the interrupt. Seeking to EOT to add a file to an existing tape containing many files can take up to an hour or longer.
- MTIO and the DATAIO package underwent some revisions, partially motivated by the addition of Exabyte support. The maximum tape blocksize is increased to 65535 bytes and the maximum FITS blocking factor is increased to 22 (anything over 10 is not standard FITS tape, however, and may not be readable on other systems).
- The VMS/IRAF process main has been updated to permit passing the name of the IRAF task to be executed on the command line, along with any arguments. This allows most IRAF tasks to be called from the host system as if they were ordinary host tasks, without entering the iraf main command interpreter.
- A bug in the IMDKERN graphics kernel (graphics overlays on image displays) which could cause the kernel to crash on a segmentation violation has been fixed. (Bug #137)
- Support has been added to Sun/IRAF for the new Sun compilers, in particular the Fortran 1.3 compiler. This appears to work, however our experience with this compiler to this point is minimal. (Bug #134)

- A bug in IMFORT which could prevent access to images not in the current directory has been fixed. (Bug #128)
- A bug in ONEDSPEC.CALIBRATE involving an error of one pixel in the wavelength scale for extinction correction has been fixed. (Bug #129)
- The entire ARTDATA package (first introduced in V2.9) has been replaced by a newer version.
- Support for output to the null image (as in "imcopy image dev$null") has been added to the OIF image kernel.

Doug Tody

Changes to the ARTDATA Package: Version 1.1

Since the first release of the ARTDATA package with V2.9 there have been several bug fixes and some new features and tasks have been added. This note summarizes these changes.

The bugs (buglog entries 127, 130, 131, and 135) are all in the main task MKOBJECTS. Two of these concern the size of the PSF as specified by the "radius" parameter. The radius parameter for the gaussian PSF is interpreted as a FWHM instead of the intended HWHM (radius at half maximum). This leads to very narrow stars until one compensates for the error. The size of the moffat PSF is off by a small factor (~10%) depending on the beta parameter. Another bug causes objects near the edge to be off by 1 pixel from the specified coordinates. The last bug prevents use of a user supplied profile function for the PSF. Thanks go to Lindsey Davis and Steven Perry for the careful studies which identified the first three bugs.

The artificial one dimensional spectra produced by MK1DSPEC have new parameters allowing addition of a velocity shift specified either as a velocity or redshift. This is a useful addition for tests of radial velocity programs.

The initial version of the package provided minimal headers; mostly just the wavelengths for 1D spectra and the dispersion axis for 2D spectra. People then need to HEDIT keywords to make the images work well with some packages. Four things have been done to make useful headers for the artificial data. First, a detailed log of parameters used to generate the images is included as COMMENT cards. This includes task parameters and information from data files such as those in the star and galaxy list files produced by STARLIST and GALLIST. Second, the gain, read noise, exposure time, and dispersion correction flag have been added to the header when appropriate. Third, each task has a new parameter for specifying a header keyword data file containing a list of keywords and values to be automatically added. A default file is supplied but any set for a particular type of data may be substituted. Finally, a new task called MKHEADER has been added which applies header keyword data files to images.

A new task called MKEXAMPLES has been added. Given the name of an example from a menu an image is created. This task is intended to provide examples for the ARTDATA package as well as test and demonstration images for the various packages. The initial menu includes long slit and multifiber spectra, a globular cluster, a star field, a galaxy field, and a galaxy cluster. Additional examples will be added as demonstrations and test procedures are developed for various packages.

This new version of the ARTDATA package is included in IRAF V2.9.1.

Frank Valdes

CCDRED News

There have been a few complaints that CCDPROC does not check for division by zero when flat fielding. The considerations behind this are that such checking, since it has to be done for every pixel, would impact the processing efficiency which was one of the prime goals of the package. Also it seemed to me that to arrive at flat fields with zero values required that something go wrong and so aborting would require the user to take some kind of corrective action. However, it was pointed out that CCD defects could cause zero values which are no fault of the observer.

I have added a new parameter to CCDPROC (IRAF Version 2.10), called "minreplace", that sets a lower limit to any flat field pixel value produced by the task. The advantages of this approach are that the flat field checking is only done once when the flat field is initially processed for bias, etc. (it is an assumption that any flat fields used by CCDPROC are also processed by CCDPROC). The actual processing of the observations is unchanged with no efficiency concerns due to explicit zero division checks. The flat field correction is such that other tasks outside of the package benefit. The default of 1 will prevent division by zero during processing but users may adjust it to anything they like from some very large negative value to avoid any changes to the data to higher positive values to minimize flat field noise in regions of low signal such as outside spectroscopic apertures.

A second item of interest is that we are beginning to think about extensions for reducing multiple amplifier and multiple chip read outs which produce a single image (multiple images could, of course, be reduced with the current package). Conceptually it will most likely consist of some way to define multiple data, bias, and trim sections which CCDPROC would apply efficiently, in a single pass through the image, in much the same way as is currently done. One new wrinkle would be how to describe putting together different pieces after trimming independent overscan regions. Anyone who would like to contribute ideas and experiences and review plans is encouraged to contact me (fvaldes@noao.edu).

Finally, a new task is currently being written for the next release which allows reviewing and editing header translation files. It is common for people trying to setup such files to have difficulties getting the format and image type translations correct. The new task makes it much easier to verify or create translation files.

Frank Valdes

IMRED Package Revisions Summary: IRAF Version 2.10

A number of new specialized image reduction packages and new versions of the generic ECHELLE and multiobject spectroscopy packages have been added to the IMRED package in IRAF Version 2.10. The new subpackages have been made available as part of the NEWIMRED add-on package prior to the release of Version 2.10. The major changes are:

- New multifiber reduction packages ARGUS, NESSIE, and KPCOUDE.FIBER.
- New spectrophotometric slit spectra reduction packages GOLDCAM, SPECRED, and KPCOUDE.SLIT.
- New versions of the MSRED and ECHELLE packages based on the new versions of APEXTRACT and ONEDSPEC.
- Addition of an iterative rejection capability in ECIDENTIFY and ECREIDENTIFY.

In addition there have been some minor changes in the other spectroscopy packages required by changes in the ONEDSPEC package.

The new packages are specialized to specific instruments or types of data. They contain tasks collected from the various general spectroscopy packages which are appropriate for a particular type of data. However, the most important contribution of these packages are special reduction tasks which are streamlined to perform the complete calibration and reduction of the data in as simple and automated manner as possible. The tasks combine operations from both two dimensional extraction and one dimensional spectral calibrations and collect all the useful parameters in two parameter sets while fixing and hiding parameters which are irrelevant.

The new packages are as follows. The ARGUS package is for the flat fielding, throughput correction, extraction, dispersion correction, and sky correction of data from the CTIO *Argus* multifiber instrument. The NESSIE package is similar and is for the KPNO *Nessie* multifiber plugboard instrument. The KPCOUDE.FIBER package is specialized for the three fiber (two arc and one object) instrument at the KPNO Coude. It is similar to the other multifiber packages except there is no sky subtraction.

The other three packages are for sky subtracted extraction, dispersion correction, extinction correction, and flux calibration of slit instruments. The packages are for the KPNO *Goldcam*, the KPNO Coude, and for the CTIO *2DFRUTTI*. They are all fairly general and could be used for other instruments. They are distinguished by choices of default parameters.

There are user's guides for the powerful new reduction tasks in the new packages. These are available both as nicely typeset documents and as online IRAF manual pages. PostScript copies of the guides are available in the IRAF network archive.

Tasks from the revised APEXTRACT and ONEDSPEC packages appear in many of the IMRED packages. In particular the ECHELLE and MSRED packages are now based on this new software.

Some minor changes are the replacement of the SPECPHOT package by SPECRED and the renaming and reorganization of the old COUDE package.

Frank Valdes

ONEDSPEC Package Revisions Summary: IRAF Version 2.10

Though much of the ONEDSPEC package is unchanged there have been significant changes to a number of the commonly used tasks for the next release of IRAF (Version 2.10). The new software is currently available as part of the NEWIMRED add-on package. The changes primarily apply to multispec or echelle format spectra. This note simply summarizes the major new features and changes. A more detailed revisions summary is available in the IRAF network archive as file *iraf/docs/onedv3.ps.Z*.

- IDENTIFY and REIDENTIFY now treat multispec format spectra and two dimensional images as a unit allowing easy movement between different image lines or columns. The database is only updated upon exiting the image.
- REIDENTIFY supports both tracing (the old method) and always starting with the primary reference vector when REIDENTIFYing other vectors in a two dimensional reference image.
- REIDENTIFY matches reference lines or apertures when REIDENTIFYing those vectors in different images rather than tracing.
- REIDENTIFY has an interactive capability to review suspect reidentifications.

- REIDENTIFY provides the capability to add new features.
- The task MSDISPCOR allows using auxiliary reference spectra to provide a shift in the wavelength zero point to the primary dispersion functions. This includes spatial interpolation of simultaneous arc spectra in multifiber spectrographs.
- The new task SCOPY copies subsets of apertures and does format conversions between the different spectrum formats.
- The new task SAPERTURES adds or modifies beam numbers and aperture titles for selected apertures based on an aperture identification file.
- The new task SFIT fits spectra and outputs the fits in various ways. This includes a new feature to replace deviant points by the fit. Apertures in multispec and echelle format are fit independently.
- The task CONTINUUM now does independent fits for multispec and echelle format spectra.
- SPLOT now allows deblending of any number of components and allows simultaneous fitting of a linear background.
- The new task FITPROFS fits 1D gaussian profiles to spectral lines or features in an image line or column. This is done noninteractively and driven by an input list of feature positions.

The following changes are of a more minor nature than those above.

- The IDENTIFY database format uses aperture numbers rather than image sections for multispec format spectra.
- The apertures in multispec format images need not be in the same order or even have the same number of apertures as the reference image in REIDENTIFY or MSDISPCOR.
- An automatic write parameter has been added to IDENTIFY.
- The tasks MSDISPCOR and SPECPLOT support the extra information in the third dimension of multispec format spectra which is optionally output by the APEXTRACT package.
- MSDISPCOR and SPECPLOT now include a logfile.
- SPLOT selects spectra from multispec or echelle format by their aperture number. Also a new keystroke was added to select a new line/aperture without having to enter the image name again.
- The task SPECPLOT may select apertures from a multispec or echelle format spectrum.
- The aperture identification in multispec format is used, if present, for labeling in SPLOT, SPECPLOT, and STANDARD.

Frank Valdes

APEXTRACT Package Revisions Summary: IRAF Version 2.10

A new version, version 3, of the IRAF APEXTRACT package has been completed. It will be part of the next IRAF release (Version 2.10). The revisions are currently available as part of the NEWIMRED add-on package. This note simply summarizes the changes and new features of the package. A more detailed revisions summary is available in the IRAF network archive (file *iraf/docs/apexv3.ps.Z*).

There were three goals for the new package: new and improved cleaning and variance weighting (optimal extraction) algorithms, the addition of recommended or desirable new tasks and algorithms (particularly to support large numbers of spectra from fiber and aperture mask

instruments), and special support for the new image reduction scripts.

The following summarizes the major new features and changes.

- New techniques for cleaning and variance weighting extracted spectra.
- A new task, APALL, which integrates all the parameters used for one dimensional extraction of spectra.
- A new extended output format for recording both weighted and unweighted extractions, subtracted background, and variance information.
- Special features for automatically numbering and identifying large numbers of apertures.
- New tasks and algorithms, APRECENTER and APRESIZE, for automatically recentering and resizing aperture definitions.
- A new task, APFLATTEN, for creating flat fields from fiber and slitlet spectra.
- A new task, APFIT, providing various types of fitting for two dimensional multiobject spectra.
- A new task, APMASK, for creating mask images from aperture definitions.

Frank Valdes

Radial Velocity Analysis Package Being Revised

A meeting was held several months ago between members of the IRAF group and interested members of the NOAO scientific staff to discuss the future development of the RV package. The strengths and shortcomings of the RV prototype were discussed, and a list of priorities for the final package was derived. Based on this list and the concerns expressed by the staff, we have come up with the following plan for the next phase in the development of the RV package. This plan is intended to serve our immediate needs for radial velocity analysis as quickly as possible, while providing a fully featured package in the longer term.

Briefly, here is what is planned:

- **Level Zero Package**. This is intended to provide a basic radial velocity capability, providing the most needed functions but avoiding the more ambitious features planned for the eventual full IRAF package. Once completed, the level zero package will be frozen and will continue to be available indefinitely, without change, providing a stable tool for basic radial velocity analysis while development and testing of the full package goes forward. Although the level zero package will provide limited functionality, those algorithms and features provided are intended to be about as good as can be done, and will have undergone extensive testing with both real and artificial data by the time the software is released.

  The current makeup of the level zero package consists of a revised RVXCOR task and psets to control Fourier filtering, continuum subtraction, and header keyword translation. This is just reaching the user test stage at NOAO.
- **Baseline Package**. This will be more or less what is currently available in the prototype version of RV now released, most likely with many changes reflecting what we learn from the level zero package.
- **Full Package**. This will be a second version of the baseline release, adding all the desirable but lower priority functions. Many changes to the baseline package are sure to be needed once user testing begins, and these will be incorporated into the full package along with the remaining tasks not planned for the baseline package.

The level zero package provides Fourier cross correlation for both raw pixel arrays and wavelength calibrated data, including those features we thought were necessary for even the level zero task for it to be useful. These include: automatic log-lambda mapping of the input data, continuum fitting and subtraction, Fourier filtering, weighting of the correlation peak fit to minimize sampling errors, and optional correction to heliocentric velocities. Important changes to the user interface have also taken place. The database capabilities (RVSELECT) have been left out, but will be provided in the baseline version of the package (the LISTS.FIELDS task can be used to make lists of selected quantities from the output of the level zero task). Other important but less essential functions such as the Fourier quotient algorithm, redshift and deredshift functions, etc., will also be deferred to the baseline package.

In addition to the above, the current prototype RV package will continue to be supported and available for use while the new software is under development. Important bug fixes will still be provided to users, but no new features will be added to the prototype RV package as they will be available in the newer versions.

Comments on our plans for further development of the package are most welcome. Users who would like to review the task specifications or who are willing to test the level zero package should contact Mike Fitzpatrick (fitz@noao.edu, 5355::fitz, or (602)-325-9387).

Mike Fitzpatrick
Doug Tody

Photometry Package, BASPHOTC, Available from Hawaii

There is a new photometry package available that has been written in C and which may be called from IRAF. The program, BASPHOTC, was written by Marc W. Buie (Space Telescope Science Institute) and has been modified by Frank Valdes to run on the SUN workstation using IMTOOL or SAOimage. Although the routine performs much the same function as the IRAF APPHOT photometry package, Marc has written the BASPHOTC photometry routine with a difference which is important for data which contains extended objects (e.g. comets). While it is possible using the APPHOT routines to set the sky level at some pre-determined value far from the extended object in order to subtract this value from the object in the photometry aperture, this is not the most convenient manner in which to perform photometry with the package. However, at my request, Marc has added a "comet" feature to BASPHOTC which allows the user to roam around the CCD frame to select the regions where the sky is uncontaminated by the extended source rather than using the sky in an annulus centered on the photometry aperture.

Unlike other IRAF functions, BASPHOTC does not have a parameter file; instead all of the required parameters are placed on the command line at the time of calling the function:

```
basphotc gain array exptime radius sky1 sky2
```

where gain is the gain of the system in photons per ADU, array is the image name, exptime is the exposure time in seconds, radius is the radius of the object aperture in pixels, and sky1 and sky2 are the inner and outer radii of the sky annulus, respectively. The routine performs a weighted sum of all pixels contained in a circular aperture, to compute robust statistical moments of the data. Outliers from the data are removed if exceeding a threshold of 3 sigma, and the process is iterated until the change in successive means converges to within 0.5 times the standard deviation of the mean. The sky background subtracted is determined either in the annulus surrounding the object (for point sources) or as an average of the user selected sky patches. Although the routine does not have as many options as the APPHOT photometry routines which can be set by the user, and it is not designed to utilize the keyword information

which is present in the FITS headers, the routine is simple to use on a large number of frames by creating a simple IRAF script.

For further information about the BASPHOTC package contact Karen Meech (meech@galileo.IFA.Hawaii.Edu) at the University of Hawaii.

Karen Meech
University of Hawaii

Problems with Long Image Headers

IRAF users should be aware that there is a potential for loss of image header information when running any IRAF task which creates or changes the image header. Up until now most NOAO package users have not experienced significant problems because typical image headers were small. However with the advent of the multi-fiber reduction packages, such as NESSIE and ARGUS, which can generate very large image headers, the likelihood of this happening has increased.

The amount of space in the image header that the applications programs can read and write into when an image is mapped into memory is controlled by the IRAF environment variable "min_lenuserarea", whose default value of 8000 characters is roughly equivalent to space for 100 image header keywords. Headers larger than this will be silently truncated when, for example, an existing image is copied to a new image with IMCOPY or written to tape with WFITS. For similar reasons it is possible for users to be deceived into thinking that they have lost image header information when in fact the disk image is intact, because the IMHEADER task only displays the first "min_lenuserarea" characters.

The best approach to avoiding these problems at the current time is to increase "min_lenuserarea" to a number larger than the largest header normally required. For NESSIE and ARGUS a good value might be 28000 or space for 300 or so keywords. A simple "set min_lenuserarea = 28000" command in the user's *loginuser.cl* file will avoid further problems. The only penalty in this approach is a slight increase in the memory required when the image is read or written. To decrease the chances of problems further the RFITS and WFITS tasks have been modified in IRAF versions 2.9.1 and greater to set the userarea to a maximum of "min_lenuserarea" or the current default value of 28000.

There are two situations where long image headers might arise. One is when importing images created elsewhere. This should not cause problems since the FITS reader produces a warning about the long header and the user can then increase the maximum header length parameter. However, note the possible problem described below, which can arise if one logs out.

A more serious problem is when IRAF tasks extend the image header beyond the maximum. There is no warning in this case. The current example which produces large headers is the multiobject spectral extractions. The new reduction packages for NESSIE and ARGUS avoid most of the possible problems by increasing the min_lenuserarea parameter automatically when the package is loaded. Where troubles arise (and this has actually happened) is when after extracting the spectra the user logs out. Later when the user logs in and, without reloading one of the reduction packages, uses tasks such as HEDIT, IMCOPY, and WFITS with the default header size; the header information may be truncated. Prior to the changes described above, particularly to WFITS, this could lead to an irrecoverable loss of information.

Lindsey Davis
Frank Valdes

Getting Help in IRAF

Some new online help facilities were added in the last two releases of IRAF but it appears that many users may not yet be aware of them.

A task called PHELP was added in IRAF Version 2.9. PHELP is a front end script to the HELP task that allows the user to page *forward* and *backward* through the online manual pages. Its use is basically the same as that for HELP. Type `?` while you are in PHELP to get a listing of the paging options.

```
cl> phelp phelp
```

You may also page the help for entire packages as you might page a list of files, using the 'N' and 'P' keystrokes to move to the next or previous help topic for the package. By this means you can interactively review all the help text for an entire package with a single command.

Both HELP and PHELP can be used with any files ending in ".hlp" that the user may find throughout the IRAF directories that are not part of the help database. For example,

```
cl> cd mwcs
cl> phelp MWCS.hlp fi+
```

The HELP and PHELP tasks have an "option" parameter that allows the user to look at other online documents associated with a task other than just its "help" pages. There is also a "section" parameter that allows the user to scan through a particular section of the "help" pages. In the following examples PHELP is used to 1) scan through the source code of a task, 2) scan through the REVISIONS file for a package, and 3) scan through only the EXAMPLES section of the task's "help" pages.

```
cl> phelp imcombine opt=source
cl> phelp dataio.revisions opt=sysdoc
cl> phelp rfits sec=examples
```

A new task was introduced with IRAF Version 2.8 called REFERENCES. This task scans through the online description for the tasks in the help database printing out a list of those tasks that "pattern match" a string used as input to the task. Add-on packages will be included in this search if they were installed properly.

```
cl> references "flat field"
cl> references dispersion
```

Jeannette Barnes

Parameter Sets or Psets

In IRAF version 2.5 the IRAF CL was modified to support named, sharable parameter sets, breaking the one-to-one relationship between tasks and what were formerly called parameter files. In subsequent releases of IRAF users have seen this new CL facility called **psets** used by many of the new packages. Currently psets are used in the IMEXAMINE task, the APPHOT package, and the test versions of the DAOPHOT and RV packages. This article provides an introduction to psets and summarizes how they are used. The details of how to examine, edit, save and recall pset parameters can be found in the help pages for LPARAM and EPARAM and in the help for those tasks which use named psets. More complete information may be found in the article "*Named External Parameter Sets in the CL*" by Doug Tody in Volume 3A of the

IRAF documentation or in the IRAF network archive in the file *iraf/docs/pset.ps.Z*.

The main uses of psets are to group logically related sets of parameters and to allow a single set of parameters to be shared by any number of tasks. A named pset might serve as a simple database of global parameters used by all the tasks in a package, for example a pset "datapars" could be defined within a package to describe the characteristics of the data that all the tasks in the package operate upon. Since these parameters refer to the data rather than to the tasks which access the data, it makes little sense to duplicate the parameters in each task. By breaking the association of parameter sets with tasks, psets also allow objects other than tasks to have parameters, e.g., a centering algorithm used by several of the tasks in a package can now have its own set of parameters, independently of the tasks which use the centering algorithm.

At the CL level a pset appears as a simple task which calls up EPARAM when run. Like any other IRAF task, a **pset-task** may be visible or hidden to the user. Visible pset-tasks may be distinguished from compiled or script tasks by the trailing "@" which, if the CL parameter `showtype=yes` is set, appears after the pset-task name when a package is loaded or listed. For example, if one types `cl.showtype=yes` and loads the APPHOT package one will see five pset-tasks. The parameters for each of these tasks can be edited with EPARAM and viewed with LPARAM in the same manner as those of any other IRAF task. The only distinction is that typing a pset-task name automatically calls up EPARAM to edit the pset. For example, the APPHOT DATAPARS pset-task can be listed by typing `lpar datapars` and edited by typing `datapars` or `epar datapars`.

At the task level a **pset-parameter** is a string parameter whose value is the name of a file containing the stored pset. If this value is "" (the null string) the default pset for the pset-task in the user's *uparm* directory is used. For example the APPHOT package PHOT task has a pset parameter "datapars" whose default value is "". In this case PHOT reads the DATAPARS pset-task parameters from the default set in the user's uparm directory, or from the builtin package defaults if no uparm copy of the pset exists. When the PHOT task is run, keyword=value overrides may be given on the command line for the individual parameters in the DATAPARS pset, just as if the parameters were part of the main PHOT task pset.

Psets can be edited indirectly, while editing the parameters for a task which uses the pset, as well as directly using EPARAM or by running the pset task as discussed above. For example to edit the DATAPARS pset while using EPARAM to edit the PHOT task parameters, one would move the cursor to the "datapars" pset-parameter and type `:e`, which calls up EPARAM to edit the DATAPARS pset. Quitting EPARAM with `:q` will then save the edited DATAPARS pset and resume editing of the main PHOT parameters. This process can be repeated to edit all the other psets used by the PHOT task, without ever leaving EPARAM.

By default when a pset is edited the results are saved in a file in the user's uparm directory. If desired the contents of any pset can instead be saved in any user specified file. Continuing with our DATAPARS example, to save the DATAPARS pset in the file "dpars.par" one would type `:w dpars.par` while editing the DATAPARS pset. Conversely, one can read a saved pset from a file with a command such as `:r dpars.par`, allowing several versions of the same pset to be prepared and later used without having to do any editing. To use a saved pset without having to load it with `:r`, the value of the pset-parameter can be set to the name of the pset file to be loaded when the task is run. In our example, one would change the value of the "datapars" parameter in the PHOT task from the null string to "dpars.par".

Lindsey Davis
Doug Tody

Saving IRAF Plots in PostScript Files

Users have frequently asked us how to save IRAF plots into PostScript format files. This is very easy to do, although there are some subtleties to worry about. In the simplest case of a Unix host with IRAF V2.8 or greater, just set the output device to *psdump*† and make a hard-copy as usual. The PostScript output file will show up (after a GFLUSH, if necessary) in the Unix */tmp* directory with the name *irafdmp$$.ps*, where the *$$* will be replaced by the current process id. You should now be able to print this file out on a LaserWriter or other PostScript printer with the command:

```
cl> lprint /tmp/irafdmp$$.ps map_cc=no paginate=no
```

or:

```
cl> !lpr /tmp/irafdmp$$.ps
```

or with some other site dependent command.

Of course, if you are interested in producing a PostScript plotfile in the first place, it is probably not just so you can print it out. You probably want to include the plot into some document. There are some problems with using the *psdump* device for this purpose. Most obviously, a normal IRAF plot is printed in *landscape* orientation while a plot included in a document should usually be in *portrait* orientation on the page. Another problem is how to position the plot on the page. A third problem is the tiny user and host identification string that IRAF places along the right hand edge of the paper.

The first and third of these problems are easy to deal with by writing a new graphcap entry. The problem of positioning is, unfortunately, hard to define in a general manner. A revised graphcap entry might be:

```
pshalf|Half page (at the top) UNIX Postscript dump:\
        :XO#1680:XW#1290:YO#285:YW#1930:PW#1.2:RO:YF:\
        :DD=pshalf,tmp$sgk,!{ sgidispatch sgi2uapl $F -t -l$(XO) \
        -w$(XW) -b$(YO) -h$(YW) -p$(PW) > pshalf.ps ; rm $F; }&:\
        :tc=sgi_apl:
```

For a description of the graphcap parameters see the help file, *gio$doc/gio.hlp*. This can be printed out with the command `help gio$doc/gio.hlp file+ | lprint`. Briefly, the parameters *RO* (rotate) and *YF* (yflip) produce a portrait plot with labels that read correctly. The *-t* command line switch of the sgi translator is used to turn off the identification string. Another parameter to note is *PW*, which is set for a thinner line width than the default since the plot is smaller than usual.

You may wish to tinker with the placement and size of the plot on the page using the *XO*, *XW*, *YO*, and *YW* parameters. Due to IRAF's default landscape orientation, the X axis is the *long* dimension of the page. It is hard to comment further on positioning since that is the responsibility of the importing document. For an example of importing an IRAF plot into NOAO's particular *troff* implementation, see the file *misc/psmemo.ps.Z* in the IRAF network archive.

To install the *pshalf* graphcap entry, first make a copy of the *dev$graphcap* file in your own directory. This allows you to modify the graphcap while you test it, without affecting the normal operation of the system. Next, edit the entry into the file, near the top. To access your copy of the graphcap, enter the command:

```
cl> set graphcap = home$graphcap
```

assuming you placed the file in your IRAF login directory. This command can be placed in your *loginuser.cl* file to access your copy of graphcap whenever you log in.

---

†The *psdump* graphcap entry was contributed by Joe Harrington at MIT.

You should now be able to generate PostScript plot files in any of the ways that you normally generate hardcopy plots, for instance with a `:.snap pshalf` from within cursor mode. Each time you create a PostScript file, it will have the name *pshalf.ps* and will overwrite previous versions. If you are making multiple plots, don't forget to rename each file in turn! Modifying the graphcap entry to read the filename from a Unix environment variable is left as an exercise for the interested student.

The final hurdle blocking our way is that the PostScript produced by the *sgi2uapl* translator (or *sgi2vapl* under VMS, note that there is no *psdump* entry for VMS) is nonconforming and nonencapsulated. This may present large difficulties when importing a plot into commercial packages (*e.g.*, FrameMaker) that rely on features of standard Encapsulated PostScript. In practice, however, the PostScript is bland and easily digestible with the following caveats:

- The Document Structuring Conventions are completely ignored. The most serious omission is the *%%BoundingBox* comment, meaning that the importing document has no way to know where the plot is located on the page. The simplest way around this is to print the plot and measure its location with a ruler. The correct units are points (1/72 of an inch) measured from the lower left hand corner of the page.
- The *erasepage* and *initgraphics* operators are used explicitly. These can be redefined to be null operators (as with *showpage*) with no ill effects:

```
%!PS-Adobe-2.0 EPSF-2.0
%%BoundingBox: LLx LLy URx URy
/IRAFSAVE save def
    /erasepage { } def
    /initgraphics { } def
    /showpage { } def
    < the plotfile goes here >
IRAFSAVE restore
```

  Note that, except for the redefinitions of *erasepage* and *initgraphics*, this is the normal way to import an encapsulated document.

Rob Seaman

Using PostScript Fonts with IRAF Plots

We are often asked about IRAF support for publication quality graphics. One promising option that is available with the current system makes use of the TranScript software package (from Adobe Systems Incorporated) that some sites use for accessing their PostScript printers. TranScript includes a utility program, *ps4014*, that can be used to improve IRAF plot quality, albeit by entering through the back door.

The *ps4014* command will translate a Tektronix 4014 format file (*i.e.*, the format that is understood by GTERM and Xterm) and convert it to PostScript, using PostScript fonts, of course. The trick is simply to convince IRAF to produce such a format file and then to pass that file to *ps4014*. An example (a Unix host is assumed) shows how easy this is:

```
cl> noao
no> imhistogram dev$pix >G pix.mc
no> task $ps4014 = "$ps4014 -S 6 | lpr -Plw5"
no> stdgraph pix.mc dev=4012 | ps4014
```

The three steps are:

- save the output of the desired plotting task (in this case, IMHISTOGRAM) into a metacode file,
- declare a foreign task, *ps4014*, to allow the Unix command, *ps4014*, to be accessed from within IRAF (the `-S` flag sets the size of the plot in inches and may be omitted),
- convert the metacode into Tektronix format, and finally into PostScript format to send to a LaserWriter, in our example "lw5".

By default, the plot will be labeled using the `Courier` typeface, which is a fixed width font. This can be changed, with a bit more work, to any of the fonts supported by TranScript. For a list of these, see the Unix manual page for *psfonts*. The font is specified in a PostScript prologue file, *ps4014.pro*, which should be located in the *lib* subdirectory of the TranScript root directory on your system. **Do not modify the original file.** Copy the file into a convenient subdirectory of your own area, for example, *~/pslibdir* (where ~ is shorthand for your Unix login directory). Next, edit the file *~/pslibdir/ps4014.pro* and change the definition:

```
/typeface /Courier def
```

to:

```
/typeface /Helvetica def
```

(as an example) to set the font to Helvetica.

To access the new prologue, the Unix environment variable *PSLIBDIR* must be set to the new directory before entering IRAF:

```
% setenv PSLIBDIR '~/pslibdir'
```

The STDGRAPH example from above can now be repeated. Remember to issue the TASK statement again if you had to log out of IRAF and back in to define *PSLIBDIR*. Alternatively, the TASK statement can be put into your *login.cl* or *loginuser.cl* file. You may reuse the metacode file, *pix.mc*, from last time.

Subtleties remain: first, most PostScript fonts are proportional width fonts like Helvetica, not fixed width like `Courier`. The Tektronix format has no knowledge about proportional width fonts and text on the plot will not be precisely centered as a result. The IRAF graphcap mechanism can be tweaked to partially compensate for this by setting the graphcap character width parameter *cw* to agree with the average width of the characters in the font. In the distributed graphcap, the character width is set to 0.0125 (or 80 characters per line). To center the text more exactly, this should be reduced a small amount (to about 0.0110).

Second, there is no way to control the line drawing width. This should only be a problem for plots that are going to be photoreduced for reproduction. To offset this effect, the plot can be scaled downward by *ps4014* itself so that further reduction is not needed.

Third, you will note that the graphcap device *4012* was used by STDGRAPH, rather than *4014* itself. This was done since many of the graphcap parameters have an effect on the appearance of the output and the *4012* entry is a better common denominator than the *4014* entry. Specifically, the characters are wider in the *4014* entry than in the *4012* entry.

Lastly, not all IRAF plotting tasks allow complete control over the labeling of the plot. The usual complaint is that the "sysid" string at the top of the plot cannot be turned off. Many tasks in the NOAO packages do allow this string to be turned off, *e.g.*, by using the command `:/sysid no` while in the task's cursor mode. In all cases, however, the user has control over the version and user parts of the sysid string (but not over the host, date and time), by setting the *version* and *userid* IRAF environment variables.

Rob Seaman

FOCAS News #4

The most important recent development concerns the MATCH program which matches objects in multiple "single" catalogs by the RA and DEC fields into "matched" catalogs. At various times there have been reports of occasional failures to make obvious matches or incorrect matches (these are related). This happens to only a few percent of the objects and in no clearly discernible pattern except that mismatches seem to increase with the number of catalogs.

Starting with a hint from Jeff Munn (Yerkes) I took a close look at the algorithm and discovered a problem. Adding a new catalog to the preceding matched catalogs assumes a certain DEC ordering so that the matching can use a scrolling algorithm not requiring the catalogs to be entirely in memory or checking all possible combinations. The single catalogs to be added are always first sorted. It then follows that when the first catalog is entered in the matched catalog it has the proper sorting.

However, on adding additional catalogs the previous matched catalog gets out of sort, basically it must be sorted by the maximum DEC of all objects in a matched entry rather than by the DEC of the first entry. So when a new catalog object entry has a DEC less than any of the other objects in a group things are fine but when it is greater then the entry starts to get incorrectly sorted. This allows candidate groups to be dropped from consideration before they might still match. Also in what order an unmatched object is added to the matched catalog, that is an entry with only one object, makes a difference in subsequent matching. These effects explain why only a fraction of the matches have problems and why the mismatches occur more frequently with the number of catalogs matched.

The solution is to re-sort the matched catalog as it is updated. This can be done fairly efficiently since the amount of missorting introduced by matching with a new catalog is small and localized. It is difficult to study this problem but I have done some tests with artificial catalogs which show improved matching with the new version. The new version, dated 10/16/90 is available in the IRAF network archive.

People with Sun workstations are probably aware of the confusion caused by the latest distribution of Sun OS4.1 and the reorganization and unbundling of libraries and compilers. This has plagued the IRAF compiler, XC, which is used to compile the FOCAS version using IRAF image format. A change was made to XC, available with IRAF V2.9.1 and as a patch to V2.9, to handle the libraries and various reorganization of directories and compilers. Another recent discovery by Suzanne Jacoby is that Makefile now automatically adds certain flags, particularly something like "-target sun4", which XC misinterprets. Symptoms are errors such as:

```
Warning: file ‘sun4’ not found
cc: Warning: -O conflicts with -a. -O turned off.
ld: Undefined symbol
___bb_init_func
```

The resolution to this problem is to add the following line to the Makefile:

```
COMPILE.c=$(CC) $(CFLAGS) $(CPPFLAGS) -c
```

This has been done in the current distribution of FOCAS in the IRAF network archive and does not affect previous versions of the OS since the definition is ignored.

The interface between FOCAS and IRAF images has been changed to use a C language binding to the IRAF IMFORT procedures used for image access. This eliminates a number of Fortran procedures in FOCAS and is more in keeping with the C nature of the package. While on the topic of image formats, there are two new programs to directly convert between IRAF and the standalone FOCAS image formats rather than going through FITS. This is for sites who wish to use the old FOCAS image format but have occasion to use IRAF at times. I don’t understand the reasons for this rather than simply using the IRAF format but there are sites which have chosen do it.

Also concerning data formats, there is now a full set of programs to convert the original VAX catalog, area, and image files to Sun (or other workstations with the same byte and floating point formats). This is of interest for those with archived data from the period when FOCAS ran just on Unix VAXes and the machine adaptable formats had not been introduced.

Finally, there are the usual small bug fixes arising in rare circumstances. These are summarized in the Revisions file which accompanies the source.

As described elsewhere in this newsletter, there will be an astronomical software conference in Tucson in the fall of 1991. As part of the conference there will be meeting time set aside for special interest groups. If there is sufficient interest and attendance the second FOCAS users meeting could be planned as part of the conference. Please let me know (fvaldes@noao.edu or noao::fvaldes) if you would be likely to attend such a meeting.

Frank Valdes

Astronomical Calculator and Calendar Programs

John Thorstensen of Dartmouth University has contributed two useful C language astronomical programs. The programs compute and print astronomical quantities of interest for planning observations such as Sun and Moon rising and setting times, times of twilight, Julian dates, precessions, local sidereal times at particular local civil times, and considerably more. These would be particularly useful to nighttime observers right at the telescope.

One program is called "skycalc". It is interactive and provides calculations for times and coordinates of interest. The second program is called "skycalendar" and produces monthly tables similar in nature to the Kitt Peak Observer's Calendar distributed in the NOAO newsletter. The output of this program would generally be printed for reference. Both programs are nicely self-prompting and also have documentary material.

As a service to the astronomical community and since we have a well developed and easily accessible network archive, these programs are being made available through the *misc* directory of our network archive with the consent of the author. As contributed software, complaints and comments should be directed to the author (i.e., to John Thorstensen).

Frank Valdes

IRAF Documentation

Most of the IRAF documentation is now available in the network archive (*iraf/docs*) as compressed PostScript or compressed ASCII text files. This includes past issues of the *IRAF Newsletter*. A few of the user's guides do not have the figures included yet but we hope to have that remedied in the near future. The *iraf/docs/README* file contains a complete listing of what is available in the archive.

The TOC (Table of Contents) files for the *IRAF User Handbooks* (Volumes 1A and 2B) and for the *IRAF System Handbooks* (Volumes 3A and 3B) are also in the network archive. Sites may find these useful if they are considering purchasing these volumes directly from us.

A number of new IRAF documents or manuals have become available since the last issue of the newsletter. Most of these were produced as part of the recently completed NEWIMRED

package. In the following, the file name in the network archive is given at the end in parentheses.

- *APEXTRACT Package Revisions Summary: IRAF Version 2.10*, by Francisco Valdes, September 1990. (apexv3.ps.Z)
- *Guide to the ARGUS Reduction Task DOARGUS*, by Francisco Valdes, September 1990. (doargus.ps.Z)
- *Guide to the Kitt Peak Coude Fiber Reduction Task DOFIBERS*, by Francisco Valdes, September 1990. (dofibers.ps.Z)
- *Guide to the IRAF GOLDCAM Reduction Task DOGOLDCAM*, by Francisco Valdes, September 1990. (dogoldcam.ps.Z)
- *Guide to the NESSIE Reduction Task DONESSIE*, by Francisco Valdes, September 1990. (donessie.ps.Z)
- *Guide to the Kitt Peak Coude Slit Reduction Task DOSLIT*, by Francisco Valdes, September 1990. (doslit.ps.Z)
- *ONEDSPEC Package Revisions Summary: IRAF Version 2.10*, by Francisco Valdes, July 1990. (onedv3.ps.Z)
- *Guide to the CTIO Slit Spectra Reduction Task SPECPROC*, by Francisco Valdes, September 1990. (specproc.ps.Z)
- *IMRED Package Revisions Summary: IRAF Version 2.10*, by Francisco Valdes, September 1990. (imredv2.ps.Z)

Please contact me if you would like to receive printed copies of any of the IRAF documents (jbarnes@noao.edu, 5355::jbarnes).

Jeannette Barnes

Add-on Software Available for IRAF Versions 2.8/2.9/2.9.1

The following software packages are available as add-ons to IRAF versions 2.8, 2.9, and 2.9.1. All packages are available in the IRAF network archive (node tucana.tuc.noao.edu, or 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).

- ARTDATA - the new artificial data package available as an add-on to IRAF version 2.8 only as it is included with IRAF versions 2.9/2.9.1. See the accompanying article in this newsletter.
- 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 - the first complete release of the new spectroscopic reduction packages for a variety of KPNO/CTIO instruments. See accompanying articles in this newsletter. Contact Frank Valdes for further information (fvaldes@noao.edu, 5355::fvaldes).
- DAOPHOT/IRAF - the IRAF DAOPHOT digital stellar photometry package. See a discussion of this software in IRAF Newsletter Number 8 (October 1989).

- Radial Velocity Analysis package - available for user testing only and should not be installed for routine scientific use. The package is currently under revision (see accompanying article in this issue of the newsletter). 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 Newsletters (Number 5 October 1988 and Number 6 February 1989). Contact Steve Rooke (rooke@noao.edu, 5355::rooke) for further information.
- IRAF demos - a set of IRAF demos for tasks in the IRAF and NOAO packages as well as an IMTOOL tutorial. An updated package was made available in early September. 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 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).

The IRAF Group