IRAF NEWSLETTER

February/June 1990 Number 9

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

Table of Contents

* 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 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 major systems efforts undertaken since the last IRAF Newsletter were the development and installation in IRAF of a new VOS interface (MWCS) addressing the general multidimensional nonlinear world coordinate system problem, and preparation of the IRAF version 2.9 release. Both of these efforts are documented extensively elsewhere in this newsletter.

More recently, we have set up a new IRAF network archive containing the full IRAF binary release (currently only for selected systems), an IRAF documentation archive, and all IRAF and user contributed layered or "add-on" software. This should make it much easier for users to browse the available IRAF software and documentation and take what they wish, without having to bother with the hassle and delay associated with a mailed order. Early indications are that the national networks do indeed have the bandwidth necessary to distribute even the full IRAF binary distribution this way (as much as 25-55 Mb compressed per system). More information on the network archive is given elsewhere in this newsletter, but probably the best way to learn about the archive is to log on (via anonymous ftp to 192.31.165.1), transfer the README file in the login directory, and proceed based on what you read therein. Development of the network archive is still in progress, but it already contains much of potential interest (e.g., the IRAF V2.9 distribution!).

In the coming months we will be putting into operation a related facility, the IRAF mail network. This will improve communication between the IRAF group, outside groups developing software for IRAF, and the general user community. Likely topics for discussion are IRAF news and announcements, bug notices, discussions between developers and interested scientists within NOAO and throughout the community during the design phase of new applications, common user problems and questions that come up while installing or using IRAF (and the answers), and discussions pertaining to programming in IRAF.

An IRAF conference is planned for sometime during 1991. Like the IRAF mail, this will help foster improved communication within the greater IRAF community. It will also help provide a forum within the U.S. for technical papers on astronomical software and algorithms, something which is sorely lacking at this time. The conference should also help get the greater astronomical community more involved in developing software for IRAF. The current plans for the IRAF conference are discussed in more detail elsewhere in this newsletter.

The systems projects currently planned for the near term include better support for the X Window system and some badly needed documentation, namely an IRAF intro (introduction to using IRAF), and some new documentation for the IRAF programming environment.

To provide a good X11 development environment for IRAF, as well as more incentive to develop for X, we are currently in the process of getting a number of new systems for use within IRAF which use the X window system, including workstations from several vendors and an X terminal. We have had X11R4 installed and running on the IRAF Suns and a DECstation since January, and although most people are still doing their development work under SunView, several of us now use X daily. We are investigating a number of window managers and toolkits. The next major development effort for X will be an improved virtual graphics terminal to replace the standard XTERM. In the longer term we will be improving our X based image display facilities as well, either by enhancing the current SAOIMAGE X image display server, or by replacing both IMTOOL and SAOIMAGE by a more fully featured display server.

In the past six months the IRAF project has acquired a DECstation 3100 and a Macintosh IIx (both running the X window system). IRAF was ported to the DECstation last year and the DECstation is fully supported beginning with IRAF V2.9. The Macintosh, which was donated to the IRAF project by Apple Computer, is currently running A/UX 1.1.1 (Apple's version of UNIX), and the A/UX version of X11R3. We will be upgrading to A/UX 2.0 and X11R4 this summer and expect to have an alpha release of IRAF available for the Macintosh under A/UX 2.0 sometime this fall. We would like to thank Apple for donating this system, not only so that we can support IRAF on it for the community, but also because it gives us another X Window development system and possibly the opportunity to learn a thing or two from studying some of the ever-popular Mac applications.

A new hire, Chris Biemesderfer, has joined the IRAF programming group as a systems programmer, working in systems support. This position is being funded as part of a three year grant from the NASA space astrophysics division to accelerate IRAF core system development. Chris brings with him the experience gained from having worked for NRAO, STScI (in the STSDAS group), and before that Goddard.

Members of the IRAF group attended the AAS meeting in Washington in January to participate in a joint demo/display with the Space Telescope Science Institute (STSDAS) and the Center for Astrophysics (PROS). We enjoyed this opportunity to meet with our user community and discuss IRAF-related questions, problems, and suggestions. We would like to thank the folks at STScI for providing the computer hardware (a DECstation 3100 and a SPARCstation I) for this presentation, and in particular, Bob Hanisch and Dennis Crabtree for arranging the computer logistics. Several of us are also planning to attend the WGAS sessions at the summer AAS meeting in Albuquerque in June.

As a final note, we have combined the February and June issues of the IRAF newsletter. The next issue of the newsletter is due out this fall, nominally in October.

Doug Tody

IRAF Version 2.9 Now Available

IRAF version 2.9 was completed in early April and is now in distribution. Version 2.9 is primarily a development release intended to support applications software development. Because it is a limited release it has been installed on only the more popular platforms, currently all Suns running SunOS-4, VAX systems running VMS-5, DECstations running Ultrix, and Data General systems running AOS/VS (thanks to Skip Schaller of Steward Observatory). The V2.9 release will be made available on other supported systems only if there is sufficient user demand. The next major IRAF release will be available on all supported systems.

Sites should review the *IRAF Version 2.9 Revisions Summary* article in this newsletter before requesting IRAF version 2.9 to determine if updating their IRAF system is beneficial for them at this time. Note however, that sites requesting STSDAS V1.1 must update to IRAF version 2.9 before installing this latest release of STSDAS.

Due to the tight budget situation at NOAO and the increasing number of IRAF sites, it has become necessary for us to try to recover the costs of mailed IRAF distributions. A new order form and price list is included with this newsletter. Billing information must be provided with all orders. We will continue distributing IRAF version 2.8 for those host systems not supported by the new release, but the same billing procedure as for V2.9 will apply.

As an alternative to a mailed distribution, the IRAF software and most documentation is now available at no charge from the new IRAF network archive, via anonymous FTP to tucana.tuc.noao.edu, node 192.31.165.1 on the internet. Further information about the new network archive is given elsewhere in this newsletter, and in the archive itself. For sites with good access to the internet, getting IRAF from the network archive should prove not only cheaper than a mailed distribution, but faster and more convenient, since you don't have to mess with tapes. We encourage you to give it a try.

For further information concerning the IRAF distribution please send mail to IRAF at *iraf@noao.edu* or *5355::iraf.*

Jeannette Barnes
Doug Tody

Plans for an IRAF Conference in 1991

Planning is now underway for an IRAF conference to be held in Tucson in 1991. A program organizing committee is being formed to provide an agenda for the conference. The conference will be a joint effort of the NOAO IRAF group and the major outside groups developing software for IRAF, including the Space Telescope Science Institute and the Center for Astrophysics. The conference will provide an opportunity for the general IRAF user community and all groups and individuals developing software for IRAF to meet to present technical papers on new software and algorithms, and to participate in user's group meetings for the various projects using IRAF. The duration of the conference is expected to be about 3 days, consisting of 1 to 2 days of talks in general session, followed by a day devoted to various user's group meetings and any special sessions. Further information about the conference will be available after the planning committee has met and decided on a date and an agenda.

Doug Tody
Jeannette Barnes

The New World Coordinate System (MWCS) Interface

A new VOS interface MWCS has been added in IRAF version 2.9. MWCS is a program interface providing general support for linear and nonlinear world coordinate systems, for use both in applications and as an embedded interface in system code. The current version of the interface supports linear coordinate transformations, sampled nonlinear transformations, and the sky projections, e.g., the tangent plane or gnomonic projection. Multidimensional systems with distinct coordinate systems on different axes or axis groups are supported. Both the old FITS CDELT/CROTA representation and the new CD matrix representation are supported, as described in the draft extension to FITS, *World Coordinate Systems Representations Within the FITS Format*, Hanisch and Wells, 1988. A second version of MWCS providing support for additional classes of nonlinear world coordinate systems is planned.

MWCS supports three classes of coordinate systems: 1) "logical" coordinates (typically image pixel coordinates relative to the current image section), 2) "physical" coordinates (typically image pixel coordinates relative to the image of origin, independent of any linear transformations or sectioning performed on the image matrix), and 3) "world" coordinates, which can be anything. Multiple world coordinate systems can be simultaneously defined for the same physical reference system.

The image cursor list task RIMCURSOR in the LISTS package has been modified to return image coordinates in any of the logical, physical, or application defined world coordinate systems; logical coordinates are returned by default for backwards compatibility. Consult the revisions notes and the manual page for details. All other tasks which read the image cursor still return coordinates in logical or image pixel units.

The following tasks in the IMAGES package (all those which geometrically transform an image) have been modified to update the world coordinate system in the image header of the output image if the MWCS has been enabled: IMCOPY, SHIFTLINES, IMSHIFT, MAGNIFY, BLKAVG, BLKREP, ROTATE, IMLINTRAN, REGISTER, and GEOTRAN.

Several tasks in the NOAO spectroscopic packages were modified to support the new CD matrix notation used by the MWCS. These keywords are added to the image header even if the MWCS header editing is disabled. This was done to allow use of NOAO spectroscopy tasks such as SPLOT and SPECPLOT with HST data (STSDAS), which uses the CD matrix representation to represent spectral dispersions. This change should not affect the general IRAF/NOAO user.

Since MWCS is a new interface and integration with the existing IRAF applications software has only just begun, IRAF version 2.9 is distributed with MWCS header editing disabled by default. In particular, it is not recommended that MWCS be enabled when using the IRAF spectroscopy reduction packages in IRAF V2.9. Here, "disabled" does not mean that MWCS cannot be used, merely that IMIO and the IMAGES tasks will not automatically update the WCS description in image headers when images are geometrically transformed. Instructions for enabling MWCS header editing are given in the V2.9 Revisions Summary.

Doug Tody
Lindsey Davis
Frank Valdes

Support for IEEE Floating Point Added to FITS

Support for the IEEE floating point data formats has been added to the RFITS and WFITS tasks in the DATAIO package.

RFITS now recognizes -32 and -64 as legal values of BITPIX on input and automatically converts the corresponding single or double precision IEEE floating point input values to native single or double precision floating point values in the output IRAF images. RFITS will still apply the scaling parameters BSCALE and BZERO if present in the FITS header, however, sites writing IEEE floating FITS files are strongly encouraged to avoid using BSCALE and BZERO, as these keywords are obsoleted now that FITS directly supports a floating point format.

By default, WFITS still outputs floating point data in integer format with autoscaling. The user must explicitly set BITPIX to -32 or -64 to generate an IEEE floating format on output. IEEE floating will become the default output format for floating point images in a later release, once use of this extension has become more widespread.

Users should answer the following questions before deciding to write IEEE floating point FITS files. First, does the destination computer or image processing system support it? If the destination computer runs IRAF then it must be running IRAF version 2.9 or a later version. If the destination computer is running some other software with its own FITS reader then the user should carefully check that the reader supports the IEEE floating point extensions. Second, how big are the output FITS files? Users used to scaling their single precision floating point images into 16 bit integers will require twice as much tape or disk space to store single precision IEEE floating point FITS images. Third, how important is preserving the precision? The IEEE floating point format preserves the precision of the data and removes the necessity for scaling floating point numbers to fit into a 16 or 32 bit integer.

Lindsey Davis

The IRAF Artificial Data Package, ARTDATA

The first version of the IRAF artificial data package, ARTDATA, has been completed, tested, and installed in V2.9. In addition it is available as an add-on package for V2.8 (it is not necessary to order V2.9 in order to obtain this package). The package consists of tasks for generating test patterns, one and two dimensional spectra, images of star and galaxy fields, and adding or creating fixed Gaussian or readout noise, Poisson noise, and cosmic rays. As part of the stellar and galaxy field generation there are tasks for making lists of stars and galaxies having standard spatial and magnitude distributions. Below is the menu summary for the package.

gallist	-	Make an artificial galaxies list
mk1dspec	-	Make/add artificial 1D spectra
mk2dspec	-	Make/add artificial 2D spectra using 1D spectra templates
mknoise	-	Make/add noise and cosmic rays to 1D/2D images
mkobjects	-	Make/add artificial stars and galaxies to 2D images
mkpattern	-	Make/add patterns to images
starlist	-	Make an artificial star list

The pattern task is useful for both generating patterns such as grids and ramps and for testing display oriented tasks but is also the recommended way for generating empty (zero or some other valued) images for making mosaics using IMCOPY.

The one dimensional artificial spectra task can produce blackbody or constant continua with Gaussian emission and absorption lines. The lines are fully subsampled and can be binned to the desired resolution for testing the effects of undersampling. Line lists may be generated randomly or from a user specified line list.

The two dimensional artificial spectra are produced from one dimensional template spectra; either from real observations or from artificial one dimensional spectra. The spectra are convolved with either a Gaussian or a slit profile function to produce two dimensional format spectra. This is done with subsampling to provide tests of undersampling in the spatial direction. The spectra may have a tilt and variable width in the two dimensional format. Any number of spectra may be added to a two dimensional image providing for generating multifiber, echelle, slitlets, and long slit types of artificial data. Noise and cosmic rays are added with the MKNOISE task to complete the creation of realistic artificial two dimensional spectra.

The artificial star and galaxy field generating tasks have many options for modelling both object profiles and seeing. The stellar objects and the seeing may be gaussian or Moffat profiles or a user specified template. The point spread function may be elliptical and of an arbitrary size (from space quality resolutions to poor ground based quality resolutions). The galaxies may have De Vaucouleurs or exponential profiles as well as user specified profiles and templates. The galaxies are given a specified axial ratio and position angle. The profiles are subsampled and binned to give realistic sampling properties.

The artificial images are created from a text list of parameters; x, y, and magnitude for both stars and galaxies, and profile type and shape for galaxies. The user may provide their own object lists, or use the builtin list generating tasks. Some features of these tasks include the ability to generate realistic spatial profiles for globular and galactic clusters, random spatial distributions of varying density, and magnitude distributions for clusters and a homogeneous field. The galaxy list task also generates realistic shapes and scale sizes including a cosmological (1+z)**4 trend.

Potentially, generating realistic images can be very compute intensive. Various special algorithms and trade offs are made to allow generating good fields in a reasonable amount of time on typical workstations. For example, one can generate a globular cluster field containing several thousand stars in a few minutes of cpu time. Galaxy fields are somewhat slower but a 512x512 field consisting of 1000 galaxies and stars takes only on the order of 5-10 minutes to generate with a SPARCstation.

We encourage you to play around with the tasks and run tests with the artificial images using your usual software. We are open to any feedback.

Frank Valdes
Lindsey Davis

APPHOT Update

Several new features have been added to the APPHOT package in IRAF version 2.9. Most of these were discussed in the October 1989 IRAF Newsletter but all are briefly summarized again here. Users should refer to the previous newsletter article for more details.

1. The "datamin" and "datamax" parameters are now active and can be used to identify bad pixels in the centering and photometry apertures, to reject bad pixels from the sky fits, and to reject pixels from the radial profile and psf modeling fits. A common use of this feature would be to identify saturated stars.
2. The APPHOT photometry routines can now pick up the airmass and filter id from the image headers and write them out to the APPHOT database files. This feature was added to make the preparation of files for photometric calibration easier.
3. A new parameter "update" has been added to all the tasks to permit users to save any changes they make in verify mode when running noninteractively.
4. A new sky fitting algorithm "mean" has been added for the benefit of users with very sparsely populated sky background regions.
5. The default help page no longer comes up when the user enters an APPHOT task in interactive mode although it is still available with the "?" keystroke command. This was causing annoyance to users running interactively from contour plots.
6. A new cursor setup menu has been added to all the APPHOT tasks. Although the default setup menu is still provided as an option, it is now possible to set any single parameter graphically or set some of them up on a bright star and others on a faint star. An example where this is useful is the DAOFIND task where one might want to set up the "fwhmpsf" parameter on a bright star, but set up the parameter "threshold" on a faint star. The user is now asked to verify the parameter after it has been set.

Lindsey Davis

IRRED Package Notes

There have been some significant changes to the IRRED package tasks IRMOSAIC, IRALIGN, IRMATCH1D and IRMATCH2D. The execution time and memory requirements of all four tasks have been significantly reduced. Users who are trying to mosaic large input images in particular should notice a significant improvement in performance.

Some new features have been added to the alignment and matching routines. Users may now trim a specified number of columns and rows off each input subraster before it is inserted into the output image. The old "refsection" parameter has been replaced by "xref" and "yref" making it easier for the entire mosaic to be moved around in the output image. Finally a third alignment option "file" has been added to IRALIGN, IRMATCH1D and IRMATCH2D wherein the user can specify the x and y shifts with respect to the reference subraster and the intensity offset explicitly.

Lindsey Davis

New Calibration Data Files

The calibration data used by some of the tasks in the TWODSPEC, ONEDSPEC, and many of the IMRED packages are kept in a directory called `onedstds` in `noao$lib/`. The current contents of this directory are best summarized by paging through its README file, i.e. (from IRAF):

```
cl> page noao$lib/onedstds/README
```

A new directory of calibration files, `spec50redcal`, has been added to the `onedstds` directory. The data is this new directory are from Massey and Gronwall, Ap. J., July 20, 1990.

Jeannette Barnes

Equivalent Widths in IRAF SPLOT

The various equivalent width measurements performed by SPLOT contain approximations to the standard definition. This note describes one change to SPLOT to improve the approximation as well as a discussion of the algorithms.

First, the true equivalent width definition is

$$EW = \int S(\lambda)/C(\lambda)d\lambda \qquad (1)$$

where λ is the wavelength, $C(\lambda)$ is the continuum, and $S(\lambda)$ is the continuum subtracted flux (defined to be positive for both emission and absorption). If the continuum is constant, either by assumption or by first normalizing the spectrum by the continuum, then (1) becomes

$$EW = \int S(\lambda)d\lambda/C = F/C \qquad (2)$$

where F is the flux in the line.

In SPLOT the equivalent widths may be measured using the d, e, h, k, and v keys. Except for e the methods use a Gaussian approximation to the line. Equation (2) then becomes

$$EW = \sqrt{2\pi}I_c\sigma/C \qquad (3)$$

where I_c is the peak intensity and σ is the Gaussian sigma. The methods differ in how the peak intensity, sigma, and continuum are determined.

Now consider approximating the continuum by a linear function rather than a constant.

$$C(\lambda) = C(\lambda_c)(1+AW) \qquad (4)$$

where λ_c is a reference wavelength, $C(\lambda_c)$ is the continuum at the reference wavelength, W is the wavelength relative to the reference wavelength, and A is a normalized slope. Equation (1) can then be written as

$$EW = \int S(W)/(1+AW)dW/C(\lambda_c) \qquad (5)$$

Remember that the continuum is also part of $S(W)$, that is the continuum subtracted intensity.

Using a binomial expansion this can be written

$$EW = F/C(\lambda_c)(1-A\int WS(W)dW/F+...) \qquad (6)$$

where F is defined as in equation (2). If A is zero then (5) is identical with (2) and if A is small then (2) is a good approximation if we choose λ_c as the center of the line. Further, if $S(W)$ is symmetric, such as the Gaussian profile approximation, then the odd terms disappear and the next non-zero term has a coefficient of order A^2. For the Gaussian approximation with λ_c being the line center gives

$$EW = F/C(\lambda_c)(1+A^2\sigma/\sqrt{2}+...) \qquad (7)$$

The reason for all this mathematical discussion is that in general SPLOT uses (2) rather than (1). The following considers consequences of this and announces a change to one of the algorithms.

The h, k, and v keys provide a quick way to measure many lines with just one keystroke (or two for the one-sided variants). The k and h/l and h/r keys assume a continuum of 1 and the cursor selects a flux level at which the width is measured in order to determine the sigma. The v and h/a and h/b keys use the cursor to define a constant continuum level with the sigma determined by the half flux width. The main point here is that the continuum is assumed constant so that (1) and (2) are equivalent.

The d key uses a linear continuum and fits multiple Gaussian profiles. The equivalent width is then given by the first term of (7). Unless the slope is very great the error is quite small.

The e key does not use a Gaussian approximation but does use a sloped continuum. In IRAF versions 2.8 and earlier, equation (2) was used with C being the continuum level at the midpoint between the two cursor positions used to mark the sloped continuum. This introduces the types of errors identified above plus the continuum point will not be at the line center if asymmetric continuum points are used. The magnitude of the errors will depend on the slope of the continuum and whether the profile is symmetric.

In version 2.9 the e key algorithm has been modified to compute (1) using a discrete approximation to the integral. In order to estimate the errors made with the previous version of IRAF, both the new and old method equivalent widths are recorded in the log file. The new value replaces the old value and the old value is printed at the end of the log line in parenthesis. From some simple tests I've made the error is usually extremely small (~0.1%). However, a user reports "most of our errors (are) in the 3 to 5 percent range, with errors of 15 percent common in asymmetric lines, like P-cygni lines". The things to watch out for are large slopes and asymmetric lines. If you usually continuum normalize the spectra then there will be no difference between (1) and (2) except for noise and precise cursor placement.

Even when a linear continuum is used correctly this is still an approximation since a real continuum will be some curve. If one really wants to measure equivalent widths based on definition (1) then the spectra need to be continuum normalized. The usual way to do this is with the CONTINUUM task. Of course even this is often just an approximation to the true continuum.

Thanks go to Stephen Voels for questioning the approximation used by the general e algorithm which led to the thought that there is no reason equation (1) shouldn't be calculated rather than using (2).

Frank Valdes

Status of the NEWIMRED Packages

The set of packages for reducing multifiber and stellar long slit spectra comprising the NEWIMRED package has been available on an "as is" add-on basis since January. The main demand for this software has been for reducing Nessie and Argus multifiber spectra and KPNO coude spectra, both longslit and three fiber. Versions of the Nessie and Coude packages are currently available and in use on Kitt Peak and in Tucson. These packages have user guides for running their main reduction procedure. Parts of the new software have also been successfully used to reduce the similar MX data from Steward Observatory. The main limitation right now is that the individual tasks and new features, including some new tasks in APEXTRACT and ONEDSPEC, are not documented.

One reason the documentation has not been completed is that work is still being done on the new version of APEXTRACT upon which much of NEWIMRED depends. I think this is worth the wait because good progress is being made on improving the cosmic ray rejection and optimal extraction algorithms. These improvements will be implemented in the new APEXTRACT and then the process of documentation can begin. There will be more about this in future newsletters.

One interesting development was initiated by a visitor using Nessie for measuring velocity dispersions. His feeling was that even across a single fiber covering 6-8 pixels there were significant wavelength shifts due to distortions. The request was to be able to make independent dispersion solutions across a fiber and combine the flux after wavelength calibration rather than during the initial extraction. The capability of defining subapertures for extraction was added to the development version of APEXTRACT and modifications made to the Nessie reduction script. There did appear to be a systematic effect across a fiber though much of it may be due to the fact that APEXTRACT type extraction is done along columns or rows and not perpendicular to the local path of the spectrum, which varies due to alignment and distortions. I await a report from the user on whether this new capability was useful and if there are important variations across a fiber.

Frank Valdes

Escaping to the OS and More

Sometimes it is useful to type a single host level command while you are running IRAF. This can be done by prefixing the host level command with the ! keystroke. After the host level command is executed control is passed back to the CL.

```
cl> !dir/date                    # VMS directory listing
cl> !ls -l                       # UNIX directory listing
```

A little-known feature also lets you execute commands on remote nodes in a network by prefacing the ! with !*node*, provided the remote system can be accessed via IRAF networking (see your IRAF system manager if you are unsure about this). Since IRAF networking is not currently supported *into* VMS systems, we give only a UNIX example:

```
cl> !argo!mailq                  # list argo's mail queue
```

At other times it may be useful to be able to suspend your IRAF process and work at the host level for awhile and then go back to IRAF, but all of this without actually logging out of the CL. This mechanism is host system dependent.

VMS systems:

```
cl> !!                           # suspend IRAF
$ ....                           # execute VMS commands
$ logout                         # go back to IRAF session
cl>                              # continue in IRAF
```

UNIX systems:

```
cl> !stty                     # look for suspend character
cl> ^X                        # suspend IRAF process, e.g. crtl/X
% ...                         # execute UNIX commands
% fg                          # go back to IRAF session
cl>                           # continue in IRAF
```

Now look what we have started! Let's go a little beyond and give a few extra hints to VMS users, in particular. Sometimes it can be useful to "mix and match" host level commands with IRAF commands. For example, in VMS one usually has to create temporary files just to search through the output of VMS host commands for some particular character string. Two examples are presented below to demonstrate this "mixing and matching". The first example defines a VMS task as an IRAF foreign task, executes the task and then pipes the output to the IRAF task MATCH that then only lists to the terminal the lines containing the "string" (note that {} braces are used to make the search case-insensitive). The second example is similar except that an arbitrary VMS user program is used as the foreign task.

Host level command known to VMS:

```
cl> task $shq = "$show queue/full/all/batch"
cl> shq | match "{jbarnes}"
```

Arbitrary VMS user program (e.g. IMFORT):

```
cl> task $mtask = "$mtask:==\$usr\$0:[rooke]mtask.exe!mtask"
cl> mtask | match "whatever"
```

Jeannette Barnes
Steve Rooke

News from Steward Observatory

Steward Observatory's IRAF data acquisition software is now available as a beta release from `as.arizona.edu` via anonymous FTP. This software is described in the following paper. For further information please contact me at Steward Observatory, University of Arizona, Tucson, AZ 85721, (602) 621-3091.

> Schaller, S., 1990, in *CCDs In Astronomy*, ed. G. H. Jacoby, A.S.P. Conference Series No. 8 (San Francisco: A.S.P.), p. 206.

Skip Schaller

IRAF on the Solbourne

The Solbourne is a Sun compatible SPARC architecture computer that may be used to run IRAF. Solbourne claims compatibility with software compiled for Sun's SPARC machines, and indeed Sun/IRAF runs unmodified on the Solbourne with no known bugs. With a 33 MHz cpu, the Solbourne 4-500 at the University of Arizona's Lunar and Planetary Lab compares favorably to NOAO’s 25 MHz SPARCstation 370 server, Gemini, and to the IRAF group’s 20 MHz SPARCstation 1, Coma. The Solbourne and Coma each have 16 Mb of memory, while Gemini, which is configured as a server, has 54 Mb. The Solbourne’s SCSI disk also seems to hold up well to the competition.

For comparison, here is a table of benchmarks selected from *A Set of Benchmarks for Measuring IRAF System Performance* by Doug Tody, revised July 1987. See that document for a description of each benchmark and for a comparison of other machines (under IRAF V2.5). Note that the three machines are configured rather differently so that the benchmarks should only be compared as ballpark figures. It is difficult to overemphasize that benchmarks such as these should be used as only the roughest guides. In a "real world" environment (i.e., one that is of interest to anyone choosing a workstation to purchase), figures such as these may or may not indicate even the correct *relative* performance of the machines listed.

Selected Benchmarks

	IMADDS[a]		IMADDR		IMADDL[a]		IMSTATR		IMSHIFTR	
Solbourne[b]	.35	*0:01*	.22	*0:01*	.21	*0:02*	.32	*0:01*	3.93	*0:05*
Gemini[c]	.47	*0:01*	.35	*0:02*	.30	*0:02*	.58	*0:01*	5.74	*0:11*
Coma[c]	.54	*0:02*	.35	*0:02*	.30	*0:02*	1.80	*0:02*	7.06	*0:09*

	IMTRAN		WBIN		RBIN		WTEXT		RTEXT	
Solbourne	.50	*0:01*	.52	*0:02*	.46	*0:01*	1.42	*0:02*	3.22	*0:03*
Gemini	.76	*0:02*	.86	*0:06*	.88	*0:04*	2.24	*0:04*	4.70	*0:08*
Coma	.67	*0:01*	.82	*0:07*	.87	*0:03*	2.75	*0:06*	5.54	*0:06*

cpu time in seconds, *clock time in italics*

NOTES:
a) Short integer (two byte) arithmetic may be more costly than long integer (four byte) arithmetic.
b) IRAF version 2.8 (export)
c) IRAF version 2.9 (beta)

Rob Seaman (NOAO)
Kent Wells (LPL)

FOCAS Update

The two major developments since the last FOCAS news are support for bigger image display windows and verification that FOCAS compiles and runs on DECstations and with the X Window display server SAOIMAGE. Prior to the introduction and general use of bit-mapped workstations capable of image display, image processing was tied to expensive image processing hardware. FOCAS was first used with a RAMTEK display system and then moved to IIS displays. For many years this required the availability of an IIS to "see" what FOCAS was doing. The first step to break this dependence was modification for an IIS emulating display server on a workstation. This was done in 1988 using the IMTOOL display server. However, the display server is capable of dynamically resizing frame buffers and display windows, a feature not found in a device like the IIS. As a result FOCAS was limited to using a 512 square frame even with the display server due to specific dependencies in the code. As everyone is aware modern CCD imagery is exceeding this size and the technique of subsampling to fit a 512 square window is somewhat irritating. FOCAS has now been modified to allow use of any of the IMTOOL frame buffer configurations, such as an 800x800. It will also reset the display to the desired configuration, thus avoiding the confusion when the display is also being used by IRAF in a different configuration. A limitation is that the configuration has to be defined in the code. However, this is simple to do. The default has been set to 800x800. For users with an IIS things will work as before.

Astronomical computing must follow the trends in (relatively low cost) computer hardware. One strong trend is towards widespread use of X Window based window managers which spans many types of computers and vendors. The availability of the SAOIMAGE X Window display server, which uses the same communications protocol as the SunView based IMTOOL display server, allows FOCAS to be used immediately in this environment. SAOIMAGE has many nice display features but it is also limited in one important area: it supports only one active frame buffer. This means some of the functionality used by the FOCAS display programs is not available. However, some display capability is far better than no display capability. The only minor modification made in FOCAS derived from work with SAOIMAGE is that the default color isophotes were changed to use the smaller selection available with SAOIMAGE. This allows the code to remain the same for both display servers and visually the same too. As a final note, a display server having much the same functionality (call it a port) of IMTOOL is expected for X Windows sometime in the future. The FOCAS user will then have a choice of servers.

The IRAF group acquired a DECstation 3100 around the beginning of the year. This will probably be a good competitor with Sun for NOAO/IRAF/FOCAS. While FOCAS is supposedly portable to any BSD compatible UNIX system there are always small differences in the strictness of compilers concerning some C usage which is not fully correct. Compiling FOCAS over the recent years from VAXes to Suns and Alliants have found most of these. The verification of FOCAS on the DECstation also found some minor problems that were easily fixed and remain acceptable to the other systems. Thus, FOCAS can be used on DECstations running Ultrix and using SAOIMAGE as the display device. The only limitations are those imposed by the display server. Note that with use of IRAF networking and the basic networking aspect of X Windows a broad range of system configurations are possible. For example, I can run FOCAS on the DECstation but use my Sun workstation running Sunview with GTERM/IMTOOL for I/O. The reverse is, of course, also possible.

Two new programs have been written. One is called ARTCAT. It allows generation of an artificial catalog, including an area file, from an input list of objects: x, y, magnitude, and classification. One initializes a new catalog as usual with SETCAT and then fills in the entries with ARTCAT rather than with DETECT, etc. This program was written and will be potentially very useful in conjunction with the new IRAF artificial data package. One can now generate a very realistic artificial image, create a catalog using the precisely known positions, magnitudes, and classifications, create an actual catalog by processing the image, and then match the two catalogs in order to analyze the accuracy of detection, photometry and classification. I intend to specify the parameters for some standard test cases, generate artificial images, and carefully process and analyze them. This will be an important extension of the standard test images processing described earlier. Given IRAF and the parameters anyone will be able to produce the same image or any particular type of image for testing and verification. The standard artificial images will also be submitted to the standard images archive. This all sounds good but it may be sometime before I have the time to do this. A possible resource will be use of a summer student who will be working with Lindsey Davis and me in this area of testing. However, this also includes tests of IRAF/DAOPHOT so it will be up to the student’s interest.

FOCAS is primarily an analysis package. However, it provides some image processing capabilities and is used by some people for this purpose. One new task was ported from the work of

Rick Wenk and the Bell Laboratories group for using the image display to mark objects and then register and median a set of observations. This task, called MEDIAN, might be useful to you. IRAF does provide the tools for doing the same thing, though a full, nicely packaged, image display oriented task is not yet available.

Recent bug summary excerpted from the Revisions file

---

convolve: The 5x5 convolution case had an indexing bug. 4/5/90, Wenk

display:
review: There was an off by 1 error in y between the display coordinates and the image coordinates. This has to do with the way IMTOOL computes coordinates. It does not affect data or catalogs. This is now compensated for. 3/8/90

irafimageio/Makefile: Imacck.o was being left out of the imfort library. This was a transient bug not likely to affect anyone. 1/26/90, Valdes

rfits: Fixed a bug in which an 80 character FITS card would cause the next card to be skipped in the listing. 1/15/89, reported by Bernstein

opstrm: Changed the calculation of Fmag from -1.086*log(x) to -2.5*log10(x). The answers differed due to the roundoff of the conversion constant. 1/15/89, reported by Majewski

rotate: Images were not being closed properly (close instead of imclose) resulting in the last few lines being missing for IRAF format images. 1/15/89

FOCAS is available upon request (fvaldes@noao.edu, 5355::fvaldes).

Frank Valdes

Summary of New Documentation Available

A number of new IRAF documents or manuals have become available within the past year. The first three items listed below were mailed to all sites in December with the last issue of the IRAF Newsletter (the two user's guides were additions to Volume 2B of the IRAF User Handbooks), but are listed here just in case someone did not receive the update packet.

- *An Introductory User's Guide to IRAF Scripts*, revised by Rob Seaman, September 1989.
- *A User's Guide to Multislit Spectroscopic Reductions with IRAF*, by E. Ellingson, November 1989.
- Overlays for Sun Keyboards, by Mark Stevens (identifies the important SunView, GTERM, and IMTOOL function keys).
- *Guide to the Nessie Reduction Package*, by Francisco Valdes, January 1990.
- *Future Development of the DAOPHOT Crowded-Field Photometry Package*, by Peter B. Stetson, Lindsey E. Davis, and Dennis R. Crabtree, in *CCDs In Astronomy*, ed. G. H. Jacoby, A.S.P. Conference Series No. 8 (San Francisco: A.S.P.), p. 289.
- *IRAF Reduction of Coude CCD Data*, by Daryl Willmarth, February 1990.
- *A User's Guide to Stellar CCD Photometry with IRAF*, by Philip Massey and Lindsey E. Davis, March 1990.

- *The IRAF Pixel List Package and IMIO Extensions to Support Image Masks*, by Doug Tody, revised June 1988.
- *Quick-POE (Position Ordered Event File) Interface Design*, by Doug Tody, July 1988 (documents the system interface used to store data from event counting detectors).
- *Mini-WCS Interface*, by Doug Tody, October 1989 (design of the new world coordinate system interface).

Many of these documents are also available as PostScript or straight text files in the new IRAF network archive, in the `iraf/docs` subdirectory. Please see the README file in that directory for further information and a complete list of what is available; new documents are being added to the documentation archive daily. Please contact me if you would like to receive printed copies of any of these documents (jbarnes@noao.edu, 5355::jbarnes).

Jeannette Barnes

Add-on Software Available for IRAF Versions 2.8/2.9

The following software packages are available as add-ons to IRAF versions 2.8 and 2.9.

- ARTDATA - the new artificial data package is available as an add-on to IRAF version 2.8 only as it is included with IRAF version 2.9. See the accompanying article in this newsletter. Contact the IRAF HOTLINE for further information.
- 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 - a test release of the new spectroscopic reduction packages for a variety of KPNO/CTIO instruments. See accompanying article in this newsletter. Contact Frank Valdes for further information (fvaldes@noao.edu, 5355::fvaldes).
- DAOPHOT/IRAF - a beta release of the IRAF DAOPHOT package. See a discussion of this software in IRAF Newsletter Number 8 (October 1989). Contact the IRAF HOTLINE for information about obtaining the package.
- Radial Velocity Analysis package - available for user testing only (see article in IRAF Newsletter Number 8 October 1989). This software has only just reached the user test stage and should not be requested for routine scientific use. On the other hand, if you are concerned about the detailed capabilities of the package this is the ideal time to try the software out and let us know what you would like to change or add. 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 NOAO package as well as an IMTOOL tutorial. Contact Jeannette Barnes for further information (jbarnes@noao.edu, 5355::jbarnes).
- Kernel server kits - those files necessary to access tapes drives on a non-IRAF host or those files necessary to utilize a Sun workstation as a smart terminal/display without installing the full IRAF system (for UNIX hosts only). Contact the IRAF HOTLINE for further information.

- SAOimage - an X Window System based display server developed by Mike VanHilst at the Center for Astrophysics for workstations running X11 (see article in IRAF Newsletter Number 8 October 1989). Contact the IRAF HOTLINE for further information.
- 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) - contact the IRAF HOTLINE.

All add-on or layered software is available upon request or from the IRAF network archive, via anonymous ftp to node 192.31.165.1 on the internet.

The IRAF Group

IRAF Version 2.9 Revisions Summary

April 10, 1990

1. Introduction

This document summarizes the changes in IRAF version 2.9. This was primarily a development release intended to support applications software development, hence the major changes were in the programming environment, although there are important new features of interest to general users too. Since IRAF V2.9 is primarily a development release, it is not being released on all platforms, and it is expected that many sites will not need to upgrade until IRAF V2.10 is available. Sites interested in obtaining IRAF V2.9 should contact the IRAF project to determine if the release is available for a particular host system. At the present time, the release is being made available for all Sun systems, for VAX/VMS, and for the DECstation running Ultrix.

What follows is a brief description of some of the new features available in IRAF Version 2.9. This is not intended to be an exhaustive list, but rather a brief summary of the major changes since the last release of IRAF, Version 2.8, released in July 1989. More detailed revisions notes are available in the system notes file, `iraf$local/notes.v29`, as well as in the online revisions notes for the various packages.

Users looking for information on a particular new package should note that if the package is not mentioned in these release notes and therefore is not included in IRAF V2.9, that does not necessarily mean that it is not available. Most major reduction and analysis packages are now made available for testing as user installable layered packages before they are included in the standard distribution. For information on the available add-on packages, contact the IRAF group, or check the latest *IRAF Newsletter*.

This revisions summary is organized as follows:

2. IRAF System Revisions

2.1. IEEE to native floating point conversions

Support has been added to the programming interfaces (§4.2.3) for converting between the IEEE floating point and native floating point data formats, including both single and double precision. The FITS programs in DATAIO (§3.1.1) make use of this, allowing floating point data to be exchanged in FITS format without having to convert to type integer.

2.2. World coordinate system support

A major new VOS interface MWCS has been added to support general world coordinate systems (WCS) and transformations thereon (§4.2.1). This includes support for linear, piecewise linear or sampled WCS, and general nonlinear WCS such as the tangent plane or gnomonic projection.

If a FITS image is read into the system which has WCS information in the header, the WCS will be retained in the IRAF image header and can be used in coordinate transformations. The IMAGES tasks which move pixels around have been modified to edit the WCS to reflect the transformation (§3.1.2). The image i/o system will automatically propagate the WCS of an image to a new copy of the image, and will edit the WCS as necessary if an image section is copied (this applies to all IRAF tasks which operate upon images). The task RIMCURSOR in the LISTS package has been rewritten to add support for coordinate transformations (§3.1.3), and can be used, e.g., to read out the RA and DEC of objects on the image display using the image cursor, if the image has the necessary WCS information in the image header.

Full integration of the new world coordinate facilities into all the IRAF applications, e.g., the graphics tasks and the spectral reduction packages, will take a year or longer due to the amount of software involved. In V2.9 the IRAF spectral packages have not yet been converted to use MWCS, and if MWCS is enabled it could alter the normal behavior of these packages. *IRAF V2.9 is therefore shipped with MWCS disabled.* What "disabled" means is that WCS information in the image headers is not edited to reflect operations involving image sections, or geometric transformations of images. Tasks such as RIMCURSOR which use an already existing WCS will still work whether or not header editing is disabled. If the spectral tasks will not be used and it is desired that world coordinates be propagated correctly in image transformations, MWCS header editing can be enabled in either of the following ways.

The MWCS transformations are disabled by defining the variable "`nomwcs`" in the IRAF environment. To globally enable MWCS by default for everyone using the system, edit the file "`hlib$zzsetenv.def`" and comment out the following line as shown (you want to add the leading #, which will be missing in the distributed version):

```
    #set    nomwcs          = yes
```

To enable MWCS header editing temporarily, for the current IRAF run:

```
    cl> reset nomwcs = no
```

Detailed information on the coordinate systems defined by MWCS can be obtained in the online system with the command

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

Additional information is also given in the help page for RIMCURSOR.

2.3. IMFORT changes

The IMFORT interface (host level Fortran or C interface to the IRAF image format) has undergone the following bug fixes and enhancements:

- A couple of bugs associated with the IMDIR (image pixel-file directory) feature introduced in IRAF V2.8 have been fixed.
- Image clobber checking has been added. By default, if you create a new image and another image with the same name already exists, the image create will now return an error code leaving the existing image unchanged. To override clobber checking in IMFORT programs, restoring the previous behavior of the interface, define "`clobber`" in your host environment.

- IMFORT will now perform a limited filename translation service using the IRAF VOS filename translation code. This should allow most IRAF filenames to be used as input to host level IMFORT programs. Full VOS filename mapping is not provided, but filenames containing upper case characters and multiple "." delimited fields should be translated as in IRAF programs.
- On systems with multiple architecture support (e.g., Sun, Convex) the FC task, used to compile and link IMFORT programs from within the IRAF environment, is now a script rather than a simple foreign task front end to XC. The purpose of the script is to see that all the necessary IRAF and host level command line switches and environment definitions (`IRAFARCH`, `FLOAT_OPTION`, etc.) are used. Previously, users had to make these environment definitions manually, and if they forgot the IMFORT program could fail to link or execute.
- On most UNIX/IRAF systems, the host library `-lU77` is now searched automatically by FC when an IMFORT program is linked. This library is not used by any of the IRAF code, but is required to link some Fortran programs that might want to use IMFORT.

Users are encouraged to use FC to link their IMFORT programs. It is possible to manually link against the IRAF libraries if you know what you are doing, but the location of the libraries and the required host level command line switches vary for different systems and for different architectures of a single system, and it is easy to make mistakes.

2.4. MKIRAF now copies login.cl to login.cl.OLD

On UNIX/IRAF systems, the MKIRAF command will now copy any existing `login.cl` file to `login.cl.OLD`, so that, for example, you can more easily merge any custom changes back in after running MKIRAF. On VMS/IRAF systems a new file version is created, as before.

2.5. Local additions to termcap/graphcap

The termcap and graphcap device capability files have been reorganized with a section at the top for local additions. It is recommended that any locally added entries be made in this area, to simplify future system updates. The local additions can then be simply transferred to the new version of the file when a new version of IRAF is installed (any entries which are modified versions of standard entries should always be checked to see if anything has changed in the distributed version).

2.6. BIN directories now smaller

On systems with multiple architecture support, the architecture save file `OBJS.arc` stored in the BIN directory for each architecture is now maintained as a compressed file. In a typical case this reduces the size of the file by about a factor of two, saving 1-2 Mb of disk space in each BIN directory.

2.7. Various system buffers increased in size

The layered software support in IRAF V2.8 (`extern.pkg` and all that) had a problem with very long `helpdb` environment strings, limiting the number of external packages which could be defined. To fix this problem, various buffers were increased in size all over the system. The maximum length of an environment variable such as `helpdb` is now 960 characters (12 80 character lines of text). String parameters to tasks can also be larger, and the system is more resistant to problems when size limits are exceeded. Foreign task commands, OS escapes, etc., can all be larger now. The current limit on such strings is about 1024 characters, and is defined at sysgen time by the new system parameter `SZ_COMMAND` in `hlib$iraf.h`.

2.8. Shared library versions

The Sun/IRAF shared library mechanism was modified to add support for shared library versions. The result is that when you install IRAF V2.9, which has a different shared library than V2.8, any local programs or other layered software linked under V2.8 will continue to run, because both the old V2.8 shared library and the new V2.9 shared library are included in V2.9 (with different version numbers). Although old programs will continue to run with V2.9, it is recommended that they be relinked eventually to take advantage of the many features and bug fixes provided by V2.9. In the case of very large packages, e.g., STSDAS 1.0, it may be wise to wait until the latest release can be obtained and installed before relinking, as the old version will not have been tested under IRAF V2.9 (which of course, didn't exist back then).

2.9. File pager enhancements

The system file pager, used in the PAGE task, the new PHELP task, and other places, has undergone the following enhancements.

- The N and P keys, used to move to the next or previous file when paging a list of files, now have a dual meaning: when paging a *single* file containing multiple formfeed delimited pages, the keys will move to the next or previous *page* in the file. This feature is used in the new PHELP task to page a large file containing, e.g., all the HELP pages for a package.
- A limited upscrolling capability is now supported, e.g., if you hit the 'k' key while in the pager, the screen will be scrolled up one line in the file being paged. This feature may not be supported for some terminals, in which case the entire screen is redrawn at the new file location.

2.10. STF image kernel enhancements

Extensive work has been done on the STF image kernel in this release (the STF kernel allows IRAF to access the Space Telescope image format directly). The changes included the following.

- Header file caching. STF images often have quite large FITS headers which can be time consuming to read. A header file caching scheme is now used to optimize the kernel in cases where the same imagefile is repeatedly accessed, e.g., when successively reading each element of a large group format image. By default up to 3 header files will be cached; this default should be fine for most applications. If necessary the number of cache slots can be changed by defining the integer variable "`stfcache`" in the IRAF environment (the builtin maximum is 5 cached headers per process).
- *The semantics of the kernel regarding header updates have changed.* STF images differ from other IRAF images in that they may consist of a group of images all in the same file, with each individual image having its own header (the group header), plus a single global FITS header shared by all images in the group. This is no problem in a read operation, but in a write or update operation there can be problems since parameters cannot be added to or deleted from the individual group headers. The new semantics regarding STF image header updates are as follows: 1) when updating the header of a multigroup image (not recommended) only the group header is updated, and attempts to add new parameters are ignored; 2) when updating the header of an image containing a single group, both the group header and the FITS header are updated.

  As a result of these changes, the behavior of a single group STF image is now identical to that of a regular IRAF image. It is recommended that multigroup STF images be treated as read only if possible, creating only single group images during interactive processing (except when running a program that is explicitly designed to create multigroup images).

- The kernel was modified to work with the new MWCS (world coordinate system) interface. The image section transformation is now performed by MWCS rather than by the STF kernel.
- A number of minor changes were made to the way the group parameter block (GPB) cards are maintained in the IRAF image descriptor. The comments on GPB definition cards are now preserved. Restrictions on the grouping of GPB cards in the header have been removed.
- A number of bugs were fixed and restrictions removed, e.g., the size of a header is no longer limited to 32767 characters (404 lines).

The IRAF core system and NOAO science applications were extensively tested with both single and multigroup STF images using the new kernel, and we now feel that it is safe to use the STF image format with these tasks, although the regular format is preferred if there is no special reason to use the STF format (the regular format is more efficient).

2.11. QPOE (event list image format) enhancements

The QPOE image kernel, used for event list data (photon counting detectors, e.g., X-ray satellites such as ROSAT) underwent the following changes.

- MWCS (world coordinate system) support has been added (§4.2.2). This provides a consistent coordinate system despite, e.g., the blocking factor, rect, or image section used to construct an image matrix from an event list.
- When opening a QPOE file as an IRAF image, the runtime filter expression used to create the image matrix is now saved in the parameter `QPFILT`*n* in the image header (multiple cards are used for long expressions).
- Region masks of arbitrary complexity and size can now be used to mask the event list when reading time-ordered or unordered (unindexed) event lists. This is done using the new PLRIO package (§4.2.5) which provides the capability to efficiently random access large image masks of arbitrary complexity.
- Unmatched brackets, braces, or parentheses are now reported as an error by the filter expression parser (this can occur even with a valid expression, e.g., due to truncation of the expression string). A reference to an undefined keyword, e.g., due to a spelling error, is now detected and reported as an error. Any errors occurring during expression parsing will now result in termination of the calling task, unless caught in an error handler.
- A number of bugs were fixed.

2.12. Changes affecting image display in VMS/IRAF

A new version of Nigel Sharp's UISDISPLAY program, for image display on VMS systems running UIS, has been installed in "`iraf$vms/uis`". An executable for an early version of the SAOIMAGE display program for the X window system, written by Mike VanHilst (SAO), and ported to VMS by Jay Travisano (STScI) has been placed in the directory "`iraf$vms/x11`". An executable for a VMS version of XTERM (the X window terminal emulator, ported to VMS by Stephan Jansen), is also in this directory. We wanted our VMS users to have access to these programs, although more development work and testing is needed before we can offer good support for X window based image display and graphics on VMS. A more comprehensive package providing enhanced capabilities should be available as an add-on later this year.

3. IRAF Package Revisions

The most notable changes to the tasks in the IRAF packages are summarized below. Further information may be obtained by reading the help page for each task, or by paging the revisions file for a particular package. For example, to page the revisions for the DATAIO package:

```
    cl> phelp dataio.revisions op=sys
```

3.1. Changes to the System Packages

3.1.1. Modifications to tasks in the DATAIO package

- The RFITS and WFITS tasks have been modified to add support for the IEEE floating point format. The "bitpix" parameter in WFITS can be set to -32 or -64 to specify real or double precision IEEE floating numbers on output. RFITS recognizes these same values in the bitpix keyword in the FITS header on input and converts the data accordingly. Note that this option must be selected by the user as the defaults for writing a FITS tape have not changed. The user is cautioned that support for the IEEE floating formats is a new feature of FITS and may not be supported by all FITS readers.
- RFITS was modified so that the "iraf_file" parameter can be a list of output images or a image root name.

3.1.2. Modifications to tasks in the IMAGES package

- MWCS (world coordinate system) support was added to those tasks in the IMAGES package which change the geometry of an image, i.e., IMSHIFT, SHIFTLINES, MAGNIFY, IMTRANSPOSE, IMCOPY, BLKREP, BLKAVG, ROTATE, IMLINTRAN, REGISTER, and GEOTRAN (REGISTER and GEOTRAN only support simple linear transformations). If one of these tasks is used to linearly transform an image, the world coordinate system (WCS) in the image header will be updated to reflect the transformation. Note that MWCS is disabled by default in IRAF V2.9, and must be explicitly enabled to allow these tasks to edit the image header to update the WCS (see §2.2).
- The IMSTATISTICS task was modified. The "verbose" parameter was renamed "format" with the default being set to "yes" (fixed format with column labels). Otherwise the fields are printed in free format with 2 blanks separating the fields. The name of the median field has been changed to "midpt".
- The IMHISTOGRAM task has a new parameter called "hist_type" that gives the user the option of plotting the integral, first derivative, or second derivative of the histogram instead of the normal histogram.

3.1.3. Modifications to tasks in the LISTS package

- The RIMCURSOR task in the LISTS package was completely rewritten to add MWCS support, so that coordinates may be output in any user specified coordinate system defined by the WCS information in the image header of the reference image. For example, if an image with a TAN projection WCS is loaded into the image display, RIMCURSOR may be used to print the right ascension and declination at the location defined by the image cursor. Refer to the help page for details.

3.1.4. Modifications to tasks in the PLOT package

- A new graphics kernel task IMDKERN (written by Zolt Levay at STScI) has been added to the PLOT package. The new graphics kernel allows the graphics output of any task to be plotted as a graphics overlay on the image display. As with the other graphics kernels, this may be done by calling the IMDKERN task directly, but is more often done by specifying the image display (e.g., device "`imd`") as the output device when running a graphics task. Refer to the help page for details.

- The CONTOUR task was modified so that it could be used with IMDKERN to overlay contour plots on the image display. If the parameters `fill=yes` and `perimeter=no` are set the contour plot is scaled to fill the entire device viewport and all axis and plot labeling is disabled. If the image being displayed also fills the entire device viewport (display frame) then the contour plot will be drawn to the same scale as the displayed image. Refer to the help page for details.
- Several tasks in the PLOT package were modified to allow use with image specifications containing brackets, e.g., group format images, QPOE filter expressions, and image sections. The tasks modified were PROW, PROWS, PCOL, PCOLS, SURFACE, and CONTOUR.
- An option was added to the PVECTOR task to output the vector (cut through the image at an arbitrary angle and center) as a text file or image, rather than plotting the vector.

3.1.5. Modifications to tasks in the SYSTEM package

- A new task PHELP (paged help) was added to the SYSTEM package. PHELP is a script task front end to HELP which collects the output of HELP in a scratch file and pages it with the system pager, allowing one to randomly skip around to read the help text. Note that paging of all the help pages in a package is supported, e.g.,

  ```
  cl> phelp images.*
  ```

  would page all the help files for the IMAGES package.
- The NEWS task was completely rewritten, and is now used to page the revisions summary for the current and previous releases. In other words, one can now type NEWS to find out what is new in the current release.
- The GRIPES task was modified to send mail to `iraf@noao.edu` or `5355::iraf`. The IRAF site administrator may want to check this script for compatibility with the local mail system.

3.2. Glossary of New Tasks in the IRAF System Packages

Task		Package		Description
imdkern	-	plot	-	Image display device (IMD) graphics kernel
news	-	system	-	Summarize what is new in the current release
phelp	-	system	-	Paged HELP: collects and pages the output of HELP
rimcursor	-	lists	-	Read image cursor position in world coordinates

3.3. Changes to the NOAO Packages

3.3.1. New NOAO Packages

A new package ARTDATA, used to generate artificial data, has been added to the NOAO packages. ARTDATA includes tasks for the generation of star fields, optionally containing galaxies, and one and two dimensional spectra as well as simple test pattern images. The tasks GALLIST and STARLIST provide many options for producing lists of galaxies or stars that can then be used by the task MKOBJECTS to produce output images. The tasks MK1DSPEC and MK2DSPEC provide tools for making artificial spectral data. The task MKNOISE allows the user to add readout noise, poisson noise and/or cosmic ray events to new or already existing images. The task MKPATTERN allows the user to make images from a choice of patterns.

3.3.2. Modifications to Existing NOAO Packages

3.3.2.1. The ASTUTIL package

- The task SETAIRMASS in the ASTUTIL package was modified so that it now precesses the coordinates to the epoch of the observation.

3.3.2.2. The DIGIPHOT.APPHOT package

- A new task APTEST was added to the DIGIPHOT.APPHOT package that tests the execution of the package. Output files are generated that the user can review.
- Two new parameters were added to DATAPARS, "datamin" and "datamax". Pixels outside this range are rejected from the sky fitting algorithms and from the non-linear least square fits in FITPSF and RADPROF.
- An "update" parameter was added to all of the APPHOT tasks. If the "verify" parameter is set to "yes" and the task is run in noninteractive mode `update=yes` will update the critical parameters in their respective parameter sets.
- Four new parameters, "airmass", "xairmass", "filter", and "ifilter", were added to the DATAPARS task. These parameters provide the user the option of having the filter and airmass quantities from the image headers to be carried over into the APPHOT database files for later transmission to calibration programs.
- A new algorithm "mean" was added to the sky fitting options.
- A setup menu mode was added to all the APPHOT tasks. When the user types "i" in interactive mode a setup menu is presented rather than a fixed set of predefined commands.

3.3.2.3. The IMRED.IRRED package

- The APSELECT task (from the APPHOT package) has been made visible.
- The image i/o for IRMOSAIC, IRALIGN, IRMATCH1D, and IRMATCH2D has been optimized so things should run much faster. There is now an option to trim each section before insertion into the output image. The actions of these tasks can now optionally be output to the terminal.

3.3.2.4. The IMRED.MSRED package

- A task called MSBPLOT was added to the IMRED.MSRED package. This task allows the user to plot a range of lines in multispec images in batch mode.

3.3.2.5. The ONEDSPEC package

- Several modifications were made to the ONEDSPEC package. These changes affect all of the IMRED packages that include these tasks as well.
- The equivalent width measurement using the "e" keystroke in SPLOT is now computed using the ratio of the spectrum to the continuum. The previous approximation is included in the logfile for comparison.
- The DISPERSION task will now add CD*i_j* (CD matrix) keywords to the image header as an alternative way of expressing the dispersion function. If the keywords W0 and WPC or CRVALn and CDELTn are not in the image header the tasks reading this information for setting the wavelength (IDENTIFY, SENSFUNC, SPLOT, and SPECPLOT) will look for the CD*i_j* keywords. This change should have no affect on the NOAO applications but provides compatibility with STSDAS applications using the new MWCS interface provided with IRAF version 2.9.

- The call to the CALIBRATE task in the script task BATCHRED was modified so that the "extinct" parameter is always set to "yes". Since CALIBRATE checks to be sure the data has not been previously extinction corrected this simple change provides more flexibility.

3.3.2.6. The PROTO package

- Two new tasks, IMALIGN and IMCENTROID, were added to the package. IMCENTROID computes a set of relative shifts required to register a set of images. The task IMALIGN both computes the shifts and aligns the images.
- The JOIN task (previously a simple script) has been replaced by a compiled version which removes many of the restrictions of the previous version.
- The IR tasks have been modified as mentioned above under the IMRED.IRRED section (§3.3.2.3).
- The TVMARK task was modified to permit deletion (the "u" key) as well as addition of objects to the coordinate file. Another cursor keystroke, the "f" key, was added allowing the user to draw a filled rectangle.

3.3.2.7. The TWODSPEC.LONGSLIT package

- Tasks in the TWODSPEC.LONGSLIT package that are used for setting wavelength information (EXTINCTION, FLUXCALIB, and TRANSFORM) were modified for the CD*i_j* keywords as outlined above for ONEDSPEC.

3.4. Modifications and Additions to Calibration Data

The calibration data used by some of the tasks in the TWODSPEC, ONEDSPEC, and many of the IMRED packages are kept in a directory called ONEDSTDS in `noao$lib`. The current contents of this directory are best summarized by paging through its README file, e.g.,

```
cl> page noao$lib/onedstds/README
```

A new directory spec50redcal in "noao$lib/onedstds" has been added containing flux information for standard stars. The data in this list are from Massey and Gronwall, Ap. J., July 20, 1990.

3.5. Glossary of New Tasks in the NOAO Packages

Task		Package		Description
aptest	-	apphot	-	Run basic tests on the apphot package tasks
gallist	-	artdata	-	Make an artificial galaxies list
imalign	-	proto	-	Register and shift a list of images
imcentroid	-	proto	-	Compute relative shifts for a list of images
mk1dspec	-	artdata	-	Make/add artificial 1D spectra
mk2dspec	-	artdata	-	Make/add artificial 2D spectra using 1D spectra templates
mknoise	-	artdata	-	Make/add noise and cosmic rays to 1D/2D images
mkobjects	-	artdata	-	Make/add artificial stars and galaxies to 2D images
mkpattern	-	artdata	-	Make/add patterns to images
msbplot	-	msred	-	Batch plots of multispec spectra using SPLOT
starlist	-	artdata	-	Make an artificial star list

4. Programming Environment Revisions

4.1. Changes to the Programming Utilities

4.1.1. MKPKG changes

The MKPKG utility can now substitute the contents of a file back into the input stream, as a special case of the macro replacement syntax. For example, the sequence

```
abc$(@file)def
```

would be translated as

```
abc10def
```

if the file "*file*" contained the string "10". The replacement is performed by inserting the contents of the file back into the input stream, replacing sequences of newlines, spaces, or tabs by a single space, and omitting any trailing whitespace.

The "-p <pkg>" argument to MKPKG, XC, and so on loads the environment of the named package *pkg*, to define the package environment variables, load the mkpkg special file list, define the directories to be searched for global include files and libraries, and so on. Multiple "-p" arguments may be given to load multiple package environments. What is new is that if `pkglibs` is defined in the environment of a package to list the package library directories to be searched (the usual case), and multiple package environments are loaded, successive redefinitions of `pkglibs` will *add* to the list of directories to be searched, rather than redefining the old list as each new package environment is loaded. For example, if two package environments are loaded, and each defines its own library, both libraries will be searched.

4.1.2. Generic preprocessor

A minor change was made to the generic preprocessor which affects how strings such as "FOO_PIXEL" are translated. In the usual case, the preprocessor replaces all occurrences of "PIXEL" by "int", "real", or whatever the actual datatype is. The translation is now context sensitive. Rather than translating "FOO_PIXEL" as "FOO_int" (e.g., "MII_PIXEL" -> "MII_int"), the type name will now be output in upper case if the rest of the name in which it occurs is upper case. Hence, a string such as "MII_PIXEL" will now be translated as "MII_INT". This allows the use of generic constructs to symbolize SPP macros.

4.1.3. SPP changes

The language constant ARB, formerly defined as 32767, is now treated differently depending upon how it is used. In a declaration of an array argument, ARB is replaced in the output Fortran by a "*", e.g., "`int data[ARB]`" becomes "`INTEGER DATA(*)`". In an executable statement, ARB is replaced by a very large ("arbitrarily" large) integer value, e.g., to define a DO-loop which is to loop an arbitrary number of times. If ARB is mistakenly used to dimension an array which is a local variable rather than an argument, the SPP translator will now detect and report the error.

4.1.4. Interactive development and the process cache

Whenever a CL task is run and the process containing the task is already idling in the CL process cache, the CL will now check to see if the modify date on the process executable has changed, and restart the process if the executable has been modified. For example, when doing software development from within the CL and a process is alternately relinked and tested, the CL will now automatically detect that the process has been relinked and will run the new process, without any need to manually flush the process cache.

4.2. Programming Interface Changes

4.2.1. New MWCS interface (world coordinate system support)

A major new VOS interface MWCS, providing general facilities for linear and nonlinear world coordinate systems, has been added to the programming environment and is used in IRAF V2.9 in IMIO, IMAGES, and other parts of the system. MWCS is intended for use in scientific applications as well as in system code such as IMIO, hence is of potential interest to anyone developing software within the IRAF environment. The source directory is "mwcs" and the interface is documented in the file "mwcs$MWCS.hlp". Users should be aware that, although the new interface addresses the general WCS problem and has been carefully designed, a second version of the interface is planned and the current interface is not yet a "frozen" interface.

4.2.2. QPOE interface changes

The QPOE (event list image) interface has been extended to add routines to store MWCS objects in the QPOE header. By default, there is one MWCS per QPOE file, stored encoded in a machine independent binary format in a variable length array *qpwcs* of type *opaque*. The new routines are as follows:

```
  mw = qp_loadwcs (qp)
       qp_savewcs (qp, mw)
mw = qpio_loadwcs (io)
```

The routines *qp_savewcs* and *qp_loadwcs* merely save a MWCS in the QPOE header, or load a previously saved one. The QPIO (event i/o) routine *qpio_loadwcs* is like *qp_loadwcs*, except that it will also modify the Lterm of the MWCS to reflect any blocking factor or "rect" specified in the filtering expression when the event list was opened. The new routine is called automatically by QPF and IMIO whenever a QPOE event list is opened under image i/o, making the physical coordinate system of the image matrix the same as physical event coordinates.

The calling sequences of the *qp_add* and *qp_astr* routines, used to conditionally add or update header parameters, have been changed (so far as we could determine very few programs exist yet which use these routines, so we decided to risk an interface change). The change made was to add a *comment* argument. This change was motivated by the observation that people would not use the routines but would instead use lower level routines, in order to be able to set the comment field if the parameter has to be added to the header.

4.2.3. IEEE support routines added

Routines for IEEE floating to native floating conversions have been added to the MII and OSB interfaces. The new MII routines are as follows:

```
nelem = miiread[rd] (fd, spp, maxelem)
        miiwrite[rd] (fd, spp, nelem)
          miipak[rd] (spp, mii, nelems, spp_datatype)
          miiupk[rd] (mii, spp, nelems, spp_datatype)
```

The *miiread* and miiwrite routines are like their FIO counterparts, except that they are used only with data of the indicated type, and perform the IEEE to native floating conversion (or vice versa) as part of the i/o operation. The *miipak* and *miiupk* routines pack (native→IEEE) and unpack (IEEE→native) arrays of the indicated type.

The lowest level conversion routines are the OSB routines, which are what the MII routines use to perform the lowest level translation.

```
    ieepak[rd] (datum)
    ieeupk[rd] (datum)
   ieevpak[rd] (native, ieee, nelem)
   ieevupk[rd] (ieee, native, nelem)
iee[sg]nan[rd] (NaN)
```

The *ieepak* and *ieeupk* routines transform a single scalar value in place, while the *ieevpak* and *ieevupk* routines transform vectors (note that the package prefix is "iee", not "ieee"). In-place vector conversions are permitted. Since IRAF does not support the IEEE not-a-number formats, *NaN*, *Inf* etc. values are converted to a legal native floating value on input. The native floating value to which *NaN*s are mapped (default zero) may be globally set with *ieesnan*.

On some systems, e.g., the VAX, the low level conversion routines may be written in assembler or machine dependent C. If so, the source file actually used by the system will be found in the "`host$as`" directory.

4.2.4. New routine GETLLINE added to FIO

A new routine *getlline* (get long line) has been added to FIO. This is similar to *getline*, except that it will reconstruct arbitrarily long newline delimited lines of text, whereas *getline* returns at most SZ_LINE characters.

```
nchars = getlline (fd, outstr, maxch)
```

The new routine should not be confused with the old routine *getlongline*, a higher level routine which performs a similar function, but which also ignores comment lines and help blocks, and maintains a line counter.

4.2.5. Modifications to PLIO/PMIO

A new routine *p[lm]_sectnotconst* has been added to PLIO and PMIO (the pixel list and image mask interfaces). As the name suggests, the routine tests whether a given rectangular section of the mask is all at the same value, and if so returns the mask value as an output argument.

```
bool = pl_sectnotconst (pl_src, v1, v2, ndim, mval)
```

A new subpackage PLRIO has been added. This is used to efficiently random access any 2D plane of an existing pixel list or image mask.

```
   plr = plr_open (pl, plane, buflimit)
       plr_setrect (plr, x1,y1, x2,y2)
mval = plr_getpix (plr, x, y)
        plr_getlut (plr, bufp, xsize,ysize, xblock,yblock)
         plr_close (plr)
```

The mask is opened for random access on a special descriptor which incorporates a scaled, active 2D lookup table. Most subsequent *plr_getpix* calls will return the given mask value directly from the table with very little overhead; only if the referenced pixel occurs in a region too complex to be described by a single table entry is the value computed by direct evaluation of the mask. A special 2D binary recursive algorithm (using *pl_sectnotconst* above) with log2(N) performance is used to calculate the scaled lookup table. These algorithms provide efficient table generation and random mask pixel access even for very large masks.