

A I P S L E T T E R

Volume XIV, Number 1: January 15, 1994

A newsletter for users of the NRAO
Astronomical Image Processing System

Written by a cast of *AIPS*

PROPERTY OF THE U.S. GOVERNMENT
RADIO ASTRONOMY OBSERVATORY
CHARLOTTESVILLE, VA.

Edited by

Eric W. Greisen

Feb 10 1994

National Radio Astronomy Observatory

520 Edgemont Road, Charlottesville, VA 22903-2475

804-296-0211, x209; TWX 910-997-0174; Fax 804-296-0278

aipsmail@nrao.edu

The Good News ...

The 15JAN94 release of Classic *AIPS* is now available. Contact Ernie Allen at any of the addresses given in the masthead to obtain a copy. As of this writing, 94 copies of the 15JUL93 release have been given out electronically (46 tar.Z, 15 tar.gz) or on magnetic tape (20 8mm, 5 4mm, 4 QIC, 4 9-track). The ports of *AIPS* to DEC's alpha computer (OSF/1 operating system) and HP's 9000 series computers (HP-UX operating system) were cleaned up during the last six months using machines loaned to us by the vendors. A new port to personal computers (80386 and 80486 architectures) using the Linux (a public-domain UNIX-like, i.e., not DOS) operating system was developed for this release as well. (See the article later in this AIPSLETTER.) In addition to the ports, we have written a number of new tasks, enhanced the imaging and Cleaning tasks, improved access to helpful information, and even improved the *POPS* language for the 15JAN94 release.

The NRAO is advertising a support scientist position initially in the Classic *AIPS* group at the Array Operations Center in Socorro, New Mexico. If you are interested, or know anyone who might be interested, please contact Gustaaf van Moorsel at the NRAO in Socorro (505-835-7396 or gvanmoor@nrao.edu). A position for a real-time programmer in the VLBA Correlator group is also available. In addition to the new position, we have begun to receive assistance with VLBA-related software from Leonid Kogan at the AOC. Also, Jeff Uphoff, who did the initial port of *AIPS* to the PC while a student at Virginia Tech, now has a student-level position in Charlottesville and spends some of his time on *AIPS* matters.

... and the Bad

The new position at the AOC is needed because Gustaaf van Moorsel has been promoted to Head of the Computer Division at the AOC. While this may be good news for Gustaaf, it means that the *AIPS* Group will get very little of his time in future. He will continue to look after FILLM and prepare our weekly meeting notes. We congratulate Gustaaf on his promotion, and wish him well in his new position. In Charlottesville, Bill Cotton has resigned from the *AIPS* Group and moved from the Computer to the Scientific Services Division. It is impossible to overestimate Bill's contribution to Classic *AIPS* and he will be sorely missed. (He still advises your Editor concerning which rocks should be overturned in the search for bugs, fortunately.) We wish him luck with the VLA D-Array survey and his other projects.

NRAO's last Convex, the C-1 called yucca, will be decommissioned during the first quarter of 1994. It has not been used much for *AIPS* in some time, and was not used to test the current release. Although the C-1 is no longer an interesting computer in itself, Convex makes very powerful computers used at other *AIPS* sites and we regret that our support for this architecture will not be at its previous level.

Improvements for Users in 15JAN94

VLBI data formatting and fringe fitting

The task **FITLD** not only performs the functions of **INLDD** and **UVLDD**, but also acts to convert the **FITS** format used by the VLBA correlator into the internal **AIPS uv** format. It has acquired also the jobs of correcting for numerous foibles of the correlator. **FITLD** has been substantially improved for this release. It now is better at handling premature termination of the data, changed antenna numbers during concatenation, crossing of midnight, VLBA lower side band data, data out of time order, global data scaling, and reordering of baselines. It has substantially improved table handling routines.

The fringe-fitting program **FRING** was corrected in its initialization for the multi-band delay solution, in its handling of small numbers of channels and IFs, and in its choice of solution interval for short data scans. The baseline-based fringe-fitting task **BLING** was given numerous improvements. It uses improved fitting, minimization, interpolation, and initial estimation methods. It now offers a display of the degree of coherence of the fringes. A number of options were added to provide access to convergence criteria and to control deletion of scans where interpolation fails. The help and explain files were improved. The task **BLAPP** applies the results of **BLING**. It was given options to control the weights and the application of the solutions to the CL table and had a couple of bugs exterminated.

Lesser changes include adding a multi-band delay correction option to **CLCOR**. A minor error in the times used to plot models in **CLPLT** and **VBPLT** was corrected. The warnings about reference day errors in **MK3IN** were strengthened. And the **VLBA** procedure was speeded up by moving where the spectral averaging is done. It was improved to give the user the choice of display devices and of deleting or keeping intermediate results.

UV data manipulation and display

Three new tasks of relevance to *uv* data appear in the 15JAN94 release. **RFI** searches for interference by looking for high values of the amplitude rms. It does a good job of finding regions of bad interference, which is far more prevalent at the VLA at L band than many people realize. The task **UVCRS** looks for places in the *uv* plane where tracks cross and compares the measured, calibrated data from the baselines involved. It determines gain corrections to apply to make the data agree at the crossing points. Finally, **CCEDT** edits a Clean Components file keeping only components inside specified boxes above a specified cutoff flux (after merging). It is used to improve source models before self-calibration.

Significant changes and improvements were made to **DBCON** and **SPLIT**. **DBCON** was improved to correct the times in the flagging table of the second file to match any change in the apparent time (days) made in the concatenation. It was also improved to offer the option to correct phases for position shifts in a correct frequency-dependent manner. **SPLIT** was given the option to average data across IFs in addition to the previous option to average across spectral channels. It was corrected to use the frequencies in the source table and was given the option to store the full 16-character source name as keywords in the data header.

Less visible changes included raising the maximum number of sources allowed in a data set from 300 or 500 to a parameterized 2500. **UVDIF** was given the ability to detect the presence of sort-order swaps to avoid treating all such as serious differences. **SNPLT** had a number of minor annoyances in its display and handling of empty baselines corrected. The **OOPS uv**-data handling routines were corrected to determine the correct actual frequency and *uvw* scaling to use in the presence of IF selection. This error had led to minor errors in the header frequency and image scaling, especially if IF 1 were not included.

Imaging

Substantial changes were made in the handling of data gridding and the Clark Clean algorithm for the current release. These were driven by experiences with the DDT tests; see the article below for those aspects of the matter. Detailed examination of a number of residual images revealed high-frequency (about 1-cell from plus to minus) oscillations. These are caused by data gridded into the outer parts of the *uv* plane. The cell size may appear to be adequate for the fit beam width, but that beam width relates primarily to the inner portions of the *uv* plane where there is the most data. The outer portions of the *uv* plane are not adequately sampled in the image plane and, hence, cannot be cleaned adequately or correctly. The new adverb **GUARD** allows the user to define a "guard band" around the outside of the *uv* plane in which data will be discarded (with a warning message) rather than gridded. This is a serious problem and users are strongly encouraged to use this option rather than to shut it off. Actually, what users really should do is choose smaller cell sizes and, possibly, larger image sizes so that no data samples fall into, or even near, the guard-band area.

During the Clark Clean, the histogram of the residual image is taken and only points in the upper few cells are considered for components during the next major cycle. It was found that the histogram was too coarse for large images, so that only a few cells were used rather than a more reasonable number. All imaging tasks were given much finer histograms. The maximum number of cells to be "loaded" and searched for components in any major cycle controls how well and how fast the Clean progresses. If too many are loaded, then many cells are searched and subtracted and never actually selected as having a component. If too few are loaded, then the major cycle has to cut off very early and the expensive processes (model Fourier transform and subtraction, FFTs, etc.) of a major cycle are repeated too frequently. The adverb **MAXPIXEL** was added to allow the users to control the maximum number of cells to be searched. For small images, a small number would be appropriate; for normal images, the default (20000) is okay; and for large images with extended emission (e.g., Cas A), a rather larger number would be profitable. The Cleaning of each component inside a major cycle is more accurate when more points of the dirty beam are used. All imaging tasks now allow for, and use when possible, a rather larger "beam patch."

The beam width fit to the dirty beam was found to have a small but significant error. This had affected the relative units of the restored components and the residual image and, hence, is of interest. **WFCLN** offers the option to determine and correct for the relative scaling of the clean and residual portions of an image. **WFCLN** received a number of corrections to its handling of default values and channel averaging.

Two new tasks appeared in the 15JAN94 release to determine and correct for the variation of instrumental polarization across the single-dish beam. **MAPBM** uses a VLA raster made in holography mode to compute an image of the beam in the four Stokes parameters. Then **VLAPB** applies this image to sets of I, Q, and U images (as separate images or as a polarization cube). Note that this applies only to snapshot images, where the instrumental polarization has not been smeared by rotation of the parallactic angle.

The option to specify a Gaussian width to the zero-spacing in task **MX** was removed since the implementation available was too approximate and misleading. Other means should be used to add short-spacing data to a *uv* data set. One possibility would be to use **IM2UV** on a low resolution image, flag the outer spacings with **UVFLG**, and then concatenate the short-spacing data with the high-resolution data set. This would allow reasonable weighting and modeling of the short-spacing model, things not allowed by the previous option.

Image handling and display

Walter Jaffe's contributed task **SAD** (for search and destroy) was substantially revised and generalized for use by the VLA D-Array survey. It searches through an image for regions of flux above a cutoff level and then attempts to fit one or more Gaussians automatically to each region. It now prepares an elaborate **MF** file containing the complete results of the fitting in such a form that **SAD** can be restarted with all previously fitted sources removed (and presumably new windows, cutoffs, and the like). **SAD** can write a residual output file, can add components to a **CC** file, and uses the line printer to display its results. Provision for polarization has been made in the **MF** file, but the task to do that part of the fitting has yet to be written. Walter's task **CCNTR** was enhanced to plot either **CC** or **MF** components on top of contours using all the plot symbols available for plotting of "stars."

A new verb, **EPOSWTCH**, was added to change the coordinates in the image header between B1950 and J2000. Such a conversion is of necessity not exact, but should be sufficient for most non-astrometric uses.

Velocity field analysis task MODVF

The new *AIPS* task **MODVF** (MODEL Velocity Field) does what its name implies, using a tilted ring model. This model is specified in a text file, which is read into the task using the adverb **INFILE**. This file contains information about the rotation velocity in each ring and its warping with respect to the central portion of the galaxy — the reference disk. The warping geometry used in **MODVF** is described in the paper “Kinematical modeling of warps in the HI disks of Galaxies” (Christodoulou, *et al.*, 1993, *Ap. J.*, 416, 74-103). The advantage of this approach is that warps are described in the geometry of the underlying galaxy; instead of using the observer-dependent position angle and inclination, the warp is specified in terms of the angular momentum vector of each ring. This approach makes it much easier for users to test their favorite warp model against actual observations.

New history and help verbs

The *AIPS* verbs dealing with history files were changed and enhanced for this release. **PRTHI** now uses adverbs **HISTART** and **HIEND** to control which lines are printed. **HINOTE**, which adds text to a history file, now has new formatting controls and can be used to create a history file if the old one went missing. The new verbs are **HITEXT** to write out portions of the history file to a text file and **STALIN** to delete portions of the history file with a **DOCONFIRM** option.

There are two new verbs to help users find things in *AIPS* now that it is so large. Both of them use the one-line description and the one-line category list typed into help files by the programmers. We have developed programs to build the necessary secondary files from the 946 (\pm) help files in the system. The first of the verbs is **ABOUT** which lists the one-line descriptions from all help files in the user-specified category (*e.g.*, **ABOUT EDITING**, **ABOUT VERB**, **ABOUT TAPE**). The second is called **APROPOS** and lists all one-line descriptions and category lists for all help files containing the user-specified keyword or keywords (*e.g.*, **APROPOS CLEAN** to display all with words beginning with “clean” and **APROPOS UV,PLOT** to display all that contain both words beginning with “uv” and words beginning with “plot”). The help files for both **ABOUT** and **APROPOS** contain several useful examples and the help file for **ABOUT** gives the complete list of recognized categories.

Magnetic tapes

A new task, **TCOPY** appears in the 15JAN94 release. It copies selected contents, in any format including ones otherwise unknown to *AIPS*, from one magnetic tape to another. It is designed particularly to copy many tapes to one tape, an operation which is difficult to do properly with UNIX utilities. Disk FITS files are also supported.

For the 15JAN94 release, many *AIPS* tape functions were made more efficient by removing excess tape operations where possible. This may be a little less safe, but certainly speeds up the writing of ends-of-file at the beginning of **FITTP** and after each FITS file. An extra back-file 0 operation was added at the end of **FITTP** to keep IBM (and perhaps other) systems from appending an extra, unwanted end-of-file mark after the one **FITTP** had just written. The default blocksize in **FITTP** was changed to 10, which is more efficient than smaller blockings. The dangerous adverb **DOTABLE** was removed from **FITTP** and the **DONEWTAB** adverb was changed to refer to the name convention used on FITS binary tables. Older releases of *AIPS* only recognize the non-standard name “A3DTABLE” while the now-standard name for such tables is “BINTABLE”. **FITTP** was also corrected to keep the coordinate projection information on *u*, *v*, and *w* axes. One should also use **DONEW = FALSE** for this change when carrying data to older releases of *AIPS*. A bug in **FITTP** that caused it to write two consecutive ends-of-file when no data were found was corrected.

PRTPP was also improved by having it recognize the binary table *uv* format used by the VLBA and to display various things accordingly. It also acquired the option to do a very brief display to a text file (with **PRTPLEV = -3**). **PRTPP** is currently undergoing development to make it more robust against errors and to have all of its error display appear in the printer file. These improvements were not complete in time to make it to the 15JAN94 version, but will probably be offered in the patch area for 15JAN94.

POPS language processor changes

Two new pseudoverbs were written to declare verb numbers for verbs and pseudoverbs. This will allow new verbs and pseudoverbs to be declared in the **NEWPARMS** run file and thus to let users keep all their environments (**SAVE/GET** files) intact. Use of **RESTORE 0** should now be optional under all normal circumstances. The **RUN** pseudoverb was corrected to use the **VERSION** adverb correctly and to give real precedence to the **\$RUNFIL** area (over the **\$RUNSYS** area). A new verb **CLRTEMP** was written to clear the temporary literal area of all scratch numeric and string data. This verb can be used (carefully) in long looping procedures to keep the use of character strings, substrings and concatenation from gradually eating up all of the available space. If you get a "BLEW TEMP C" message, then you need this verb in your procedures.

Adverb values are no longer found through a complex Fortran equivalence which depended on an exact match between a Fortran include file and the **POPSDAT** file from which the POPS environment is defined. Instead, the language routines are used by subroutines **ADVERB** and **ADVRBS** to get and set, respectively, the adverb values. The language processor was corrected to enforce an upper limit of 132 characters for character strings and to handle such long strings appropriately in such verbs as **INPUTS** and **PRINT**. Subtle bugs affecting **IF** and **SUBSTR** statements were also corrected. All verbs and pseudoverbs were renumbered and rearranged in the **POPSDAT** file to make space for new ones and to make the lists easier to read and process. Thus, all users will have to do one more **RESTORE 0** when they begin using the **15JAN94** release and will need to replace all of their old **SAVE/GET** files.

Miscellaneous changes

Other corrections and improvements made to *AIPS* for the **15JAN94** release include:

- batch** Batch jobs now run at a new, much lower priority under UNIX systems. They will use all the available cpu, but not interfere at all with interactive usage of the computer.
- printing** All printer output now shows the host and release names at the top of each page. In this way, users may determine which computer and release actually produced their hard-copy output.
- printing** The deletion of temporary printer and plotter files is now done by a detached process after a time delay. This should reduce the incidence of lost files due to overburdened printer/plotter queues.
- plotting** Plot files are now selected for display by the new adverb **PLVER** rather than the over-used adverb **INVERS**.
- HOLGR** Numerous options and corrections were made to this antenna holography task.
- UVNOU** Renamed from **UVNOV**, this useful little task deletes data near $u = 0$, where the fringe rate is zero, which may therefore be corrupted by offsets in the correlator and by terrestrial interference.

Improvements Primarily for Programmers in 15JAN94

Coding standards enforced

The subject of coding standards in *AIPS* tends to evoke reactions of either extreme boredom or extreme anger. (For those who have tried to forget, the basic coding standards may be found in *Going AIPS, Volume 1* and the ANSI Fortran standard.) Nonetheless, they are there for a variety of reasons including the need to get through the wide variety of Fortran compilers encountered by *AIPS*. The general items of standards we encountered, and at least began to correct, in this release were:

- **Unused variables:** The enormous number of unused variables in some programs actually caused at least one compiler to fail. In addition, unused variables are reported by compilers to allow us to detect typographical errors and the like. If these messages are taken to be "normal", then the abnormal ones, which we need to detect, will be overlooked.

Level	Meaning
0	Message to file only (user input)
1	Message to terminal only (instructions)
2	Progress messages, input parameter display
3	Nature of computation, lesser informative messages
4	Interesting, useful messages
5	Significant answers, results of computations
6	Minor error messages, warnings (suppressible)
7	Probably serious error messages, warnings (suppressible)
8	Serious errors
9	Major structural failure
10	The building is on fire or under nuclear attack

Table 1: *AIPS* Message Level Meanings

- Routines in wrong subroutine directories:** The Solaris 2.x operating system, which many of us will have on our desks in the next few months, will not run a load module if there are any unresolved external references in the shared libraries. This is a departure from past practice which allowed load modules to be executed so long as the unresolved references were never called — a useful debugging tool among other things. If, for example, a subroutine calls a Y routine, then it must be in `$YSUB` or `$YNOT` and all tasks that call it must be in the appropriate Y program areas. If this routine were placed instead in, for example, `$APLSUB`, then no program in *AIPS* except those in Y program areas would run under Solaris, at least with shared libraries. Several such examples were found and corrected. In another case, the subroutine `VERBSC` had to be moved from `$AIPSUB` into the program file `AIPSC.FOR` itself, since it called routines that were only found in that file.
- Interchanged character and non-character variables:** Fortran `CHARACTER` variables are fundamentally different from all other kinds since they have both a length and a data area. Under VMS, this is implemented as a pointer to a data structure. UNIX systems, however, assume that you might be calling C routines, and expand the `CHARACTER` argument to a routine into two arguments, a length argument and a data argument. Many UNIX systems replace the `CHARACTER` variable with the data and add the length argument at the end of the list of arguments. However, some use other, equally sensible, arrangements. Therefore, even if a `CHARACTER` variable in a particular call will not be used, it is not acceptable to substitute a non-character dummy variable. To do so will change the number of arguments to the called routine from the number expected by it and might even change their position in the call from that expected. The `f2c` used in compiling on PCs was particularly useful in detecting the rather large number of call sequence mismatches involving character variables, and a number of erroneously missing or inserted arguments of other kinds as well.
- Sloppy precursor comments:** *AIPS* is too large to maintain all documentation by hand, and has been for many years. We have programs to generate “shopping lists” of subroutines for programmers. For this release, we also developed user verbs called `APROPOS` and `ABOUT` to make help-file information available to users by keywords and categories. We created tools to build the necessary files, but found that the precursor comments in many of the help files were poorly worded and lacking adequate keywords, or in some cases, plain wrong. The usefulness of these tools to users and programmers is clearly limited by the (lack of) quality of the precursor comments. We are gradually improving them, but it will be a slow process.
- Incorrect message levels:** The message levels used as the argument to `MSGWRT` have meaning and may be used for `PRTMSG` displays which select just the desired types of messages. When programmers misuse the levels, then, for example, displays of error messages show lots of minor informative things and alarm people running the DDT test. We have corrected the tasks used by DDT among others to the message level meanings given in Table 1.

Miscellaneous changes for programmers

The DDT test has been changed and new master data files computed (except for the HUGE test). The nature of these changes is described in a separate article below. As part of this process, the handling of the pseudo array processor has been changed. The size of the "AP" memories is set, for Fortran compilation, by the new parameter include file(s) `$INCS/PAPC.INC`. The OS-dependent version of this file is examined by `ZDCHIN` after it reads the *AIPS* Manager set "secondary AP size" parameter from the system parameter file. The lower of the two values for the AP size is then used. This means that a large AP can be compiled into the code, but only a portion of it used. One can also, for example, compile a large AP size into the TST version and a smaller size into the NEW version of *AIPS* and the correct AP size will be known for both versions even though they use the same system parameter file. The default AP size is now 5 Mbytes. If you have less than 32 Mbytes of memory, you might consider editing the file to use a smaller AP at your site. Numerous examples are given in the file. We have changed all "array processor" programs to use as much of the AP memory as they can reasonably do. This is rather restricted for the Clark Clean algorithm (under control of a new user adverb `MAXPIXEL`) and quite large for most other algorithms. Pseudo-AP code must not use floating point variables to store addresses into the AP memory. This becomes inaccurate (in IEEE floating point) at 16 Megawords and quite a lot sooner in other floating formats. We corrected a number of routines for this usage (which was once required by the rolling of true array processors to disk).

The object-based programming package received numerous improvements and corrections. All memory needed for objects is now allocated dynamically (using a C routine) and deallocated when the object is closed. This raises the limit on the number of simultaneous objects to around 1000 rather than the previous 20. A private D-array survey program which can run for weeks at a time has been used to test for "memory leaks" and we believe no more remain. A printer class was added for this release. Routines which will correct polarization in a spatially-dependent fashion across the single-dish beam were added. The UV data access routine `UVGET` was given an added parameter `INITVS` to specify the first visibility desired in single-source data sets.

The coordinate location common was made multi-dimensional, so that more than one image may be referenced at a time simply by changing a single index variable. This should speed up the conversions done in `HGEOM`, `OHGEO`, and the like. The abort handling routine `ZABORS.C` in its numerous versions was changed to die cleanly on recursion rather than simply returning. Some infinite looping occurred in bad task deaths otherwise. Two new procedures were added to help programmers find things. `PROG` is like `WHICH` to find files by name, but looks only in program directories. `GREP` does a case-sensitive `grep -l` through whole directory trees. List the procedures (from `$SYSUNIX`) to see the full range of options and grammar.

Ports to new operating systems

A number of annoying problems with the Solaris 2.x operating system were uncovered during the past six months. Some of them are mentioned in the article on patches to the 15JUL93 release (later in the AIPSLETTER), including minor C errors in `ZDAOPN`, missing quotes in `ZLPCL2`, and incorrect units in `ZFRE2`. In addition, errors in setting buffer sizes in `ZVTPX2` and `ZVTPX3` and errors in setting the number of adverb values in `CNTR`, `GREYS`, `PROFL`, and `MULTI` affected all systems, but were serious only on Solaris. Hewlett-Packard loaned us a model 9000/755 for a few weeks, which allowed us to test and correct a variety of things. Its C compiler seemed to have trouble with having two subroutines in one file, causing us to split out `ZIGNAL` from `ZABOR2` in `$APLUNIX`. We corrected `XAS` to pick out and use the default visual — HP offers two pseudocolor visuals in the basic window and the second is the default. We also tested out and improved the basic tape handling routine `ZTAP2`. DEC has loaned us an Alpha 3000/300 which we used to complete a port to the OSF/1 operating system and which we continue to use to support that system. The port was relatively easy and we now use `$APLDEC` to mean OSF/1, with a subdirectory for `ULTRIX` below it. Routines that were cleaned up and improved included `ZFRE2`, `ZMOUN2`, and `ZTAP2` — the usual three.

The other new system for us is Linux — a public-domain UNIX-like operating system for personal computers (80386 and 80486 architectures). Again, the port turned out to be relatively straightforward. A couple of `$APLUNIX` routines had minor changes to allow for different spellings of include file names. The Berkeley versions of `ZDELA2` and `ZTQSP2` are needed in `$APLLINUX`. The tape Z routines have not been developed since there are a great many tape devices for personal computers, but none has emerged as a usable standard. The remote tape and FITS disk routines work well, however. The port revealed problems in our Fortran usage (see above) and a bad call sequence to `XShmDetach` in `XAS`. See also the article on this port below.

Computer	$A_m^{(93)}$	$AL_m^{(93)}$	R	T_{LARGE}	T_{MEDIUM}	T_{SMALL}
IBM RS/6000 (580)	3.62	3.94		1104	323	184
IBM RS/6000 (560)	3.29	3.15	4.6	1215	357	193
HP 9000/755	3.06		7.1	1306	337	205
Sun 10/512MP (2.2) *	2.45		4.0	1630	448	179
DEC Alpha 3000/300	2.10	4.83	4.4	1908	501	267
HP 9000/735 *	1.82	2.71	7.1	2199	546	200
IBM RS/6000 (530)	1.50	1.56		2674	683	328
Sun IPX (4.1.2)	1.01	1.00	1.0	3975	918	355
Convex C-1 *	0.99	0.49		4039	1319	770
Sun LX Solaris 2.3	0.95	1.27	1.0	4225	1018	405
Sun LX Solaris 2.2	0.91	1.32	1.0	4376	1135	453
Sun IPC Solaris 2.1	0.56		0.5	7135	1670	595
Sun IPC (4.1.2)	0.54	0.52	0.5	7375	1616	564
Gateway 486DX2-66V	0.51	0.75		7780	1579	508
DECStation 3100	0.38	1.01		10626	2399	849

Table 2: New DDT AIPSMARK⁽⁹³⁾ Results

AIPS Verification Package Changed

The set of *AIPS* procedures and data files known as DDT (for Dirty-Dozen Test) has become a standard for verifying that basic *AIPS* tasks continue to function correctly and for measuring the performance of computer systems. We run it, and its younger siblings VLAC and VLAL, routinely every few weeks in Charlottesville as a continuing quality check on the developing software. Over time, however, the tests had become a bit "shop-worn." The worst problem was that DDT depended heavily on ASCAL, a task we no longer recommend to users. Also, as the software has changed, the "right" output data have also changed slightly. The accumulation of these changes made some of the current output images only agree to 9 bits or so with the previous "master" images. Because of this, minor programming errors were able to slip by undetected.

Initially it was thought that a couple of days in early June would be enough to effect these changes. However, de Sade's principle ("no good deed goes unpunished") struck. Eight months later, the full saga of changes is documented in *AIPS* Memo 85. DDT itself was changed to use CALIB rather than ASCAL, to use FITLD rather than IMLOD and UVLOD, to use IEEE floating-point in its FITS files, to support FITS disk files fully, and to use better imaging windows and cell sizes. The imaging tasks were revised to put correct shift parameters in the image headers, to limit the portion of the *uv* plane into which data are gridded, to allow the user control over the number of cells searched in each minor Clean iteration, to improve fits to the dirty beam, and to use the full pseudo-array processor memory wherever possible, including when that memory exceeds 16777216 words. Even the POPS language and task initiation got some improvements. In doing timing tests, numerous revisions were made to the ports to individual architectures as well.

The revised DDT runs faster than before. The new "*AIPS* Mark", called AIPSMARK⁽⁹³⁾, is defined as

$$A_m^{(93)} \equiv \frac{4000}{T_{LARGE}},$$

where T_{LARGE} is the total run time in seconds of the LARGE test. (In the older test, including only 40% of the ASCAL real time, run times were around 5000 seconds for the IPX and Convex. The constant 4000 was chosen to keep the AIPSMarks for the Sun IPX about 1.0, as in the previous test.) The results for a number of architectures and operating systems are shown in Table 2. The rightmost columns give the total run times in seconds for the LARGE, MEDIUM, and SMALL DDT tests. The column labeled R is the ratio of the SPECfp92 measured for the computer to that of the Sun IPX. This and the column labeled $AL_m^{(93)}$ (defined in *AIPS* Memo 85) tend to measure cpu performance (relative to the IPX) rather than the total system performance measured by $A_m^{(93)}$. All machines were tested with all of the algorithmic improvements described above except those marked with an asterisk. The results for those three machines should be regarded as representative, but not precisely correct. (The HP 9000/755 was no longer available when RTIME, the task

that measures $AL_m^{(93)}$, was written, but it should give about the same $AL_m^{(93)}$ as the HP 9000/735.)

It is interesting to note how, for some of the machines, general performance on a mix of normal *AIPS* tasks (i.e., DDT) is well predicted by their SPECfp92 ratings (or AIPSLoopMarks⁽⁹³⁾) and, for others, it is not. Some of the machines were affected by poor I/O performance in the DDT, particularly the HP 9000/735 and the DECStation 3100. But, even for these machines, poor I/O does not explain all of the differences. Please do not base any procurement decisions on this summary article. Get and read the full details in *AIPS* Memo 85 first.

AIPS Joins the World Wide Web

Recently there has been a virtual explosion in the use of a new protocol on the Internet — that of the *World Wide Web* or *WWW* for short. It is basically a method for sending hypertext over the network. Clients such as *NCSA Mosaic* and *Lynx* have certainly helped this new method of information access. The astronomy community, as usual, is particularly well connected and has been quick to offer many “home pages,” and NRAO is among the institutions working towards this. The current NRAO Newsletter has details on the NRAO home page, and the *AIPS* project also has one of these. The Universal Resource Locator (URL) is:

`ftp://baboon.cv.nrao.edu/pub/aips/aips-home.html`

If your system cannot resolve baboon's address, substitute 192.33.115.103 for baboon.cv.nrao.edu in the URL above. The information accessible here is mostly what was already accessible via anonymous ftp to baboon, but access to items such as the 15JUL93 patches and to *AIPS* documentation is considerably improved. Comments and suggestions for improvement are welcome; send these to `pmurphy@nrao.edu`.

Latest *AIPS* Memos

Below is a list of the latest *AIPS* Memos. Memos 84 and 85 are new in this AIPSLETTER.

MEMO	DATE	TITLE and AUTHOR
83	92/12/14	Dual Libraries and Binaries in AIPS Patrick P. Murphy, NRAO
84	93/11/12	A Proposed Package to Support the Use of the X Window System in AIPS Tasks Chris Flatters, NRAO
85	94/02/01	DDT Revised and AIPSMARK ⁽⁹³⁾ Measurements Eric W. Greisen, NRAO

Note that the version of *AIPS* Memo 83 now available has been modified slightly since it was distributed in December 1992. In addition, a heavily revised edition of *AIPS* Memo 78 (on Object-Oriented Programming in *AIPS*) is available. It appears as file `AIPSOOF.TEX` and, in postscript form, as `AIPSOOF.PS`.

Many of the *AIPS* Memos and AIPSLETTERS may be obtained through the facilities of the World-Wide Web or by anonymous ftp. Since some Memos are not available electronically and others do not yet have computer readable figures, you may wish to write for a paper copy of these. To do so, use an *AIPS* order form or e-mail your request to `aipsmail@nrao.edu`.

To use ftp to retrieve the memos:

1. `ftp baboon.cv.nrao.edu` or `192.33.115.103`
2. Login under user name anonymous and use your e-mail address as a password.
3. `cd pub/aips/TEXT/PUBL`
4. Read `AAAREADME` for more information.
5. Read `AIPSMEMO.LIST` for a full list of *AIPS* Memos.

AIPS Memos from Number 65 through 85 are present in this area as are Numbers 27, 33, 35, 39, 46, 51, 54, 61, and 62. We have been filling in this list gradually, by finding and fixing old files in other areas of the authors' disks, by scanning in text and figures, or by retyping text and redrawing the figures. The AIPSLETTERS from 1991 through the present are also available in this area. Many of the Memos are in both TeX and PostScript forms, with the TeX ones stored in a subdirectory called **TEX**. Note that many, if not all of these may be found on your home *AIPS* system in an area called **\$AIPSPUBL**. All Memos are available in paper form from Ernie Allen at the addresses in the masthead.

Of particular interest to users, the *AIPS* COOKBOOK will soon be available (in the form of PostScript files) in this area as well. Initially the chapters from the 1990 version of the COOKBOOK will be available; whenever one of these chapters is updated, the latest version will be available immediately in this area. We note that the chapter on VLBI processing in *AIPS* is slated for update soon, and updates for others are under consideration. All files in this area are, of course, immediately available via **ftp** and **WWW**.

Patch Distribution

Since *AIPS* is now released only semi-annually (or even less frequently), we make selected, important bug fixes and improvements available via *anonymous ftp* on the NRAO Cpu baboon (192.33.115.103). Documentation about patches to a release is placed in the anonymous-ftp area **pub/aips/release-name** and the code is placed in suitable subdirectories below this. Reports of significant bugs in 15JUL93 *AIPS* have been relatively few; however, the documentation file **pub/aips/15JUL93/README.15JUL93** mentions the following items:

- MX** **\$QNOT/ALGSUB**, and **\$QNOT/ALGSTB** used only the old and small part of the AP memory to decide how big to make the interpolation kernel for gridded modeling. For images of 4096 rows, this was too small and caused inaccurate modeling — namely some of the source did not get subtracted even though it was represented in the Clean components. **\$QYPGNOT/MX** had some faulty logic affecting restarts which would cause it not to recompute images that required recomputation.
- CNTR** **\$YPGM/CNTR**, **\$YPGM/GREYS**, **\$YPGM/PROFL**, and **\$APGNOT/MULTI** give incorrect numbers of parameters to **GTPARM**, causing problems especially on Solaris.
- System** **15OCT92** (yes, the older version) **\$SYSLOCAL** and **\$SYSUNIX** files conflict with those of **15JUL93** if they are installed with the same **\$AIPS_ROOT** area.
- Solaris** **ZDAOPW.C** had a C error which caused compilation to fail on Solaris systems. **ZVTPX2.C** and **ZVTPX3.C** had errors which caused the send and receive buffer sizes to be smaller than intended on all systems and zero on new Solaris systems. All of these are in **\$APLUNIX**.
- Solaris** **\$APLSOL/ZFRE2.C** used incorrect units, producing confusing results.
- Solaris** **\$SYSSOL/LIBR** (and **\$SYSLOCAL/LIBR** via **INSTEP1**) should have been marked as executable in the distribution tar file, but was not.
- Solaris** **\$SYSSOL/ZLPCL2** or **\$SYSLOCAL/ZLPCL2** on Solaris systems have a missing quote on line 31.
- HP** **\$APLHP/ZTAP2.C** corrected to do an advance file after a **BAKF** operation; Affects HP systems only.

Note that we did not revise the original **15JUL93** tape or **tar** files for these patches. No matter when you received your **15JUL93** "tape," you must fetch and install these patches if you require them. See the **15APR92** AIPSLETTER for an example of how to fetch and apply a patch. Information on patches and how to fetch and apply them is also available through the World-Wide Web pages for *AIPS*.

As bugs in **15JAN94** are found, the patches will be placed in the **ftp** area for **15JAN94**. As usual, we will not revise the original **15JAN94** tape or **tar** files for any such patches. No matter when you receive your **15JAN94** "tape," you must fetch and install these patches if you require them.

AIPS On a Personal Computer!

AIPS is now available to people who cannot afford (or have limited access to) the other types of systems on which *AIPS* currently runs, but who do have access to a modern personal computer. The only PC operating system *AIPS* understands currently is Linux, which is a free (public domain) UNIX-like operating system available by anonymous ftp from sunsite.unc.edu (preferred) and tsx-11.mit.edu. See below for more details on versions of the bits and pieces you will need.

Initial benchmarks in November 1993 put *AIPS* on a 486DX2/66V at roughly the same speed as a Sun IPC (see the article on DDT earlier in this AIPSLETTER). A 486DX/33 runs the small DDT just over half as fast as a DX2/66, and a 386DX/33 (even with a 80387 installed) is so slow as to be nearly unusable in any serious applications (about 45 minutes for the Small DDT versus about 10 minutes on the 486DX2/66V).

The minimal and “comfortable” system requirements are as follows:

- A 386, 486 or pentium processor. A 486DX at 33 MHz is probably the lowest “usable” system. On 486SX or any 386 systems, make sure you have the math coprocessor.
- 8 Mbytes of RAM (more if you plan on running X windows; our benchmarks on the 486 were with 16 and 32 Mbytes). 16 Mbytes of swap space is also recommended; you can have multiple swap partitions/files and this will improve performance for larger problems.
- If you want to run X windows, a color monitor is recommended; the Linux X server supports many SVGA card/monitor combinations. The NRAO system we run has an ATI Mach.32 chipset accelerated SVGA card.
- For a full Linux/X11/*AIPS* installation, you will need about 100 Mbytes of disk for the Linux/X11 part, and about 170–200 Mbytes for all of *AIPS*, assuming you strip the *.EXE files. Add to that whatever you want for local user data. A 400-Mbyte disk would give you 100+ Mbytes for *AIPS* data after you are done with everything else. Note that this is a *real* 400-Mbyte disk; there are no “disk doublers” of interest to Linux and *AIPS*. Our test system actually has two 400-Mbyte disks.
- All software needed is in the public domain or under a GNU CopyLeft (except *AIPS* which is of course freely available under the usual user agreement to academic/education/research sites).

Some of the details of the port follow.

- The `/usr/lib/libbsd.a` library was used for the socket library, because of BSD-style `ioctl()` calls used by *AIPS*. This should be transparent to both the installer and the end user.
- `INSTEP2` may hang when compiling large libraries (those containing a large number of object files) due to the passing of a greater number of command line parameters to COMRPL than the environment can handle. A simple `CONTROL-C` and re-execution of `INSTEP2` will restart it where it left off. Likewise for `INSTEP4`.
- Local tape access has not been implemented. Most PC's have either no tape drive or have a QIC (quarter-inch cartridge) or similar drive. The latter is incapable of backspace operations (other than rewind) so is of little interest to *AIPS*. Remote tape access and FITS disk file access both work.
- Two tasks, `TVFLG` and `SPFLG`, need special attention to get around a subtle “feature” of `gcc` and the `f2c` converter. This is handled automatically in `INSTEP1` as long as you have the `patch` program on your system. Otherwise you will have to patch the files by hand.

AIPS has been ported and is known to work on the following configurations:

- 0.99.12 kernel with `libc-4.4.4`, `libm-4.4.4` and `gcc-2.4.5`
- 0.99.14 kernel with `libc-4.5.8`, `libm-4.5.8` and `gcc-2.5.7`

- 0.99.13 kernel with same config as 0.99.12 with one *notable* exception: The file locking in 0.99.13 does not seem to work, so in order to use the 0.99.13 kernel, the `/usr/src/linux/fs/locks.c` kernel source file from the 0.99.12 kernel must be substituted for the `locks.c` file found in the 0.99.13 kernel (and the kernel recompiled).
- 0.99.15 kernel still to be tested (probably will work by the time you read this).

For all configurations, the FORTRAN code was compiled with `f2c` (which outputs C code that is fed to `gcc`). The version of `f2c` used was “*VERSION 21 October 1993 13:46:10*”. The `f2c` shared library used was `libf2c.so.0.9`.

The `f77` shell script front-end to `f2c` was modified locally to perform some *AIPS*-specific functions, and the modified version is placed in your `$SYSLOCAL` by the installation procedure `INSTEP1`.

All kernel sources, libraries, and compilers are in the public domain, covered by the GNU public license. They are available on the sites mentioned above. There are also several Linux “packages” available on these archives that can be used to build a complete, working system. These are not necessarily needed for *AIPS* but you may want them for other purposes.

On the Future of AIPS (Contributed by Richard Simon)

There seems to be some confusion in the community regarding the future of *AIPS* and *AIPS* support. The driver for this confusion, of course, is the *AIPS#* project. *AIPS#* is the project to create a new, object-oriented system for processing radio astronomical data; NRAO is one of seven members of the *AIPS#* Consortium created for the project. The ultimate expectation is that *AIPS#* will replace *AIPS* completely, with full support for all of the current functionality within *AIPS*, as well as new capabilities (improved programability, full support of other interferometers, single dish support, and many other ambitious goals).

NRAO is fully committed to the *AIPS#* project, but also recognizes the vital role that *AIPS* currently plays, and the necessity for maintaining *AIPS* until *AIPS#* is a truly viable alternative.

The key questions are:

- When is *AIPS#* likely to be delivered?
- When is *AIPS#* likely to be fully functional and relatively bug-free, from a user’s perspective?
- How does long does NRAO plan to continue *AIPS* support?

The first question can be answered fairly simply: the *AIPS#* project plans to release a beta version of *AIPS#* in December of this year, followed by a full release in early 1995. The beta version will have some interesting capabilities, including some not currently available within *AIPS*. It will not be a full replacement for *AIPS*. This schedule means that a functional *AIPS#* should be available by the 2nd quarter of 1995. Full functionality for *AIPS#* will be achieved in a subsequent release, perhaps later in 1995. This schedule is admittedly ambitious.

The second question above is harder to answer. Users who are currently happy within *AIPS* may or may not see reasons to switch during 1995. It is difficult if not impossible to predict the amount of time between the release of a large package and its general acceptance by the community. If fully functional is defined as encompassing all of the current capability of *AIPS*, the end of 1995 may be a reasonable target for *AIPS#* to aim at.

As far as the last question, it is clear that *AIPS* will be needed and supported through the end of 1995, and very likely to the end of 1996. In other words, the current lifetime of *AIPS* is about two to three years, with a large margin of error, depending on how optimistic one is about *AIPS#*. It will be important that *AIPS* and *AIPS#* have some overlap, with both packages fully operational, so that new or translated algorithms can be thoroughly tested and compared between the two packages.

AIPS is a very difficult act to top, and the *AIPS#* Consortium has its work cut out for it. It is fair to say that one of the successes of *AIPS#* to date has been to make the radio astronomy community realize how good *AIPS* really is!

AIPS Order Form for 15JAN94 (Unix)

1. Name and address of Contact Person: _____

☐ Check here if address on back is ok. _____

(Include street address for UPS delivery) _____

2. ☐ New order ☐ ReOrder

Do you have an AIPS user agreement? ☐ Yes ☐ No

(The agreement is a prerequisite for obtaining AIPS code)

3. Computer make, model and OS version: _____

(Include manufacturer and model of computer, and Version number of Operating System, e.g.
Sun 4/SunOS 4.1.2, Intel 486DX2/66 Linux 0.99.15, IBM RS/6000-530/AIX 3.2, etc.)

4. DDT test package (15JAN94 version): ☐ (includes small, medium, and large tests;
(FITS tape) 9-track 1600bpi tape does not have large)

5. VLA Calibration test tape:..... ☐

6. Tape "media" desired: ☐ Internet FTP ☐ 4mm DDS DAT ☐ 9-track 6250 bpi
(Unix "tar" format) ☐ Exabyte (5GB) ☐ Exabyte (2.2G) ☐ DEC TK50
☐ QIC 24 (60 MB) ☐ QIC 150 (150MB)

7. Are there "gripes" on the returned tape? ☐ Yes ☐ No

8. Printed Documents Requested: ☐ 15OCT90 AIPS Cookbook
Going AIPS (15APR90) ☐ Vol 1 ☐ Vol 2
☐ AIPS Memos: _____
(See separate list)

9. Custom 3-ring binders Cookbook ☐
Going AIPS ☐ Vol 1 ☐ Vol 2

Send this order form (or an electronic equivalent) to:

AIPS Group
National Radio Astronomy Observatory
520 Edgemont Road
Charlottesville, VA 22903-2475
USA

Network Addresses
Internet: aipsmail@nrao.edu
UUCP: ...uunet!nrao.edu!aipsmail

DATED MATERIAL - DO NOT DELAY

Library
Nat. Radio Astronomy Obs.
520 Edgemont Road
NRAO

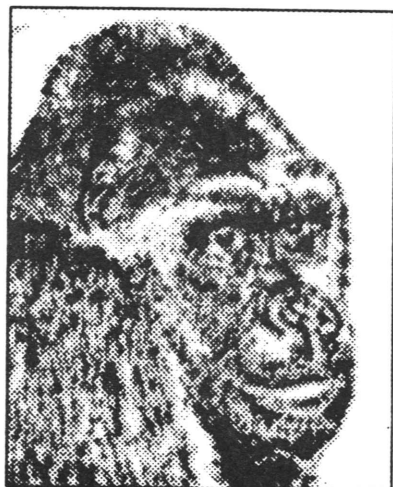
To.

FORWARDING AND RETURN
POSTAGE GUARANTEED
ADDRESS CORRECTION REQUESTED

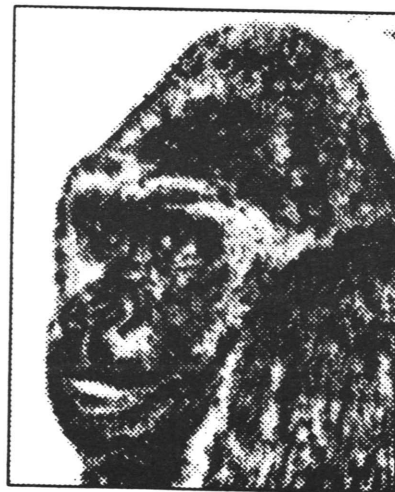
ALPSLETTER
National Radio Astronomy Observatory
520 Edgemont Road
Charlottesville, VA 22903-2475



NON - PROFIT ORG.
U.S. POSTAGE PAID
PERMIT # 373
CHARLOTTESVILLE, VA



January 15, 1994



US/OK OK/

A I P S L E T T E R

Volume XIV, Number 2: July 15, 1994

A newsletter for users of the NRAO
Astronomical Image Processing System

NATIONAL RADIO ASTRONOMY OBSERVATORY
CHARLOTTESVILLE, VA.

JUL 28 1994

Written by a cast of *AIPS*

Edited by

Eric W. Greisen

National Radio Astronomy Observatory

520 Edgemont Road, Charlottesville, VA 22903-2475

804-296-0211, x209; TWX 910-997-0174; Fax 804-296-0278

aipsmail@nrao.edu

The Good News ...

The 15JUL94 release of Classic *AIPS* is now available. Contact Ernie Allen at any of the addresses given in the masthead to obtain a copy. As of this writing, 80 copies of the 15JAN94 release have been given out electronically (32 tar.Z, 31 tar.gz) or on magnetic tape (10 8mm, 4 4mm, 2 QIC, 1 9-track). A new port to Silicon Graphics computers was developed for this release, and the magnetic-tape portions of the ports of *AIPS* to DEC's alpha computer (OSF/1 operating system) and the Linux system on PCs were debugged. In addition to the ports, we have written a number of new tasks, enhanced the transmission and calibration of VLBA data, and improved the automatic source finding and fitting task *SAD*. The large job of rewriting and updating the *CookBook* was begun, with the new and remaining old chapters being made available on-line through the World-Wide Web.

The NRAO is advertising a support scientist position in the AIPS# group at the Array Operations Center in Socorro, New Mexico. If you are interested, or know anyone who might be interested, please contact Ina Cole at the NRAO in Socorro (505-835-7309 or icole@nrao.edu). The position we advertised in the Classic *AIPS* group has been filled by Athol Kemball, an expert in VLBI spectral-line polarimetry and an experienced *AIPS* programmer. We would like to thank all those who applied for the position. The high caliber of the applicants was very gratifying.

The IAU FITS Working Group approved the binary tables extension to the FITS format conventions on June 10, 1994. At long last, this gives official approval at the highest possible level to the tables format we have been using in *AIPS* since about 1987. This format was designed initially by Bill Cotton to cope with the need for greater efficiency than one could obtain with ASCII tables and with the need for columns in tables to contain arrays as well as scalars. Inside *AIPS* these tables have been called *A3DTABLE*, but the new official name is *BINTABLE*. (This change is the current reason for the adverb *DONEWTAB* in *FITTP* since *BINTABLE* is understood only by the 15JAN94 and later versions of *AIPS*.)

The DDT test has been run on a Silicon Graphics Indigo 1 computer (33 MHz MIPS R3000 cpu, 16 Mbyte memory) and an Indigo 2 (150 MHz MIPS R4400 cpu, 32 Mbyte memory) by Jeff Pedelty (GSFC). The AIPSMarks⁽⁹³⁾ achieved were 1.18 and 2.66, respectively. Both of these should increase with larger memory and SCSI-2 fast/wide disks. A new record AIPSMark⁽⁹³⁾ of 3.89 was achieved on a DEC Alpha 3000 model 600 with 128 Mbyte of memory, 250 Mbytes of swap space, and OSF/1 version 1.3. The tests were carried out by Scott E. Aaron and Jing Ping Ge of Brandeis.

... and the Bad

Improvements for Users in 15JUL94

VLBI data processing

Two of the new tasks in 15JUL94 are designed to correct format limitations in VLBA data. **VBGLU** is used to bind together MkIII format VLBA data which were correlated in more than one pass. The result is a data set with a significant number of IFs. **FXPOL** is a task to correct the *AIPS* header and tables for dual-polarization VLBA data sets. The correlator treats the two polarizations as separate IFs, while *AIPS* would prefer to treat them as separate Stokes. **FITLD**, which translates correlator FITS files into *AIPS*, was improved in a number of ways. The format of the **IM** table was updated, a progress message giving source name and times was added, user control over the frequency tolerance for file concatenation was added, and suitable cleanup operations were added to remove unused entries in tables and empty data files. Task **INDXR** was enhanced to use the **IM** table if available when creating a new base calibration table (**CL** table version 1).

The task which solves for residual delays and rates, **FRING**, received considerable attention. Bugs were fixed which would (1) stop the task because the (reported) solution interval of 0.0 was too large, (2) cause bad solutions due to aliasing in default (Nyquist) search windows, (3) bias solutions toward zero in narrow search windows, and (4) stop the task for singularities in the least-squares fit which can actually be managed. Warning messages about failed solutions were also added. **ANCAL**, which enters system temperature and antenna gain corrections in the **CL** table, was enhanced to read in system temperatures for more IFs and to read in antenna temperatures for many more times for each antenna. The option that allows users to specify system temperatures in arbitrary order (**INDEX**) was made operational with several indexing errors corrected. **ACFIT**, which determines antenna gains from spectral-line auto-correlation data, was corrected for an error in comparing the times of the template to the specified time range and was enhanced by new adverbs to allow the user to specify the system temperatures for each IF in the template spectrum. Shifts in the fringe-rate spectra plotted by **FRPLT** were corrected. **WTMOD** was deterred from modifying weights back to positive (no longer flagged) which it was doing in some modes.

UV data calibration and manipulation

15JUL94 *AIPS* will now support 90-antenna arrays rather than mere 45-antenna arrays. This allows *AIPS* to support the new real-time solar imaging array built by the Japanese. The gain normalization was found to have two errors: unsampled antennas were included (at gain 1.0) in the average of the gains (at least when **APARM(9)** = 1) and the square root of the correct factor was applied to the data. Significant bugs in the handling of parallactic angle were found affecting VLBI polarization corrections and the **RAPR** option in **PCAL**. This may explain why the linear approximation solutions did not seem to work well. Significant bugs were also corrected throughout the calibration package in the deselection of antennas, sources, and calibrators. There were modes in which things deselected at the beginning of the task would become the only things selected at later stages. Several tasks were changed to make sure that data in tables are consistent with the data in the headers. **SPLIT** and **DBCON** were changed to change the frequency in the antenna table in the same way that the frequency in the header was changed. **SPLIT** was also changed to fix the **FQ** table for channel averaging and to remove the polarization correction from the antenna table after applying it to the data. This is intended to keep the programs from applying the same correction twice. **UVCOP** now removes unused sources from the output source table.

A bug in **CALIB** caused L1 solutions to fail if one of IFs had no data for an antenna present at other IFs. **UVFIX** was improved to make all phase corrections functions of frequency and to include leap seconds by default. A source-position correction option was added to **CLCOR** and its internal units for the zenith angle gain correction were made consistent with the documentation. **UVFLG** was changed to be able to read more flagging commands and to alter user-specified times by only 0.5 seconds and then only when the two ends of a time range are equal. **LISTR** will now include auto-correlation data, if present, in **MATX** listings. **UVFND** now tests for data flagging in all cases. The interference detection task **RFI** was improved to catch interference of steady amplitude and variable phase and was corrected in its handling of the amplitude-proportional part of the cutoff level. **SOLCL**, which applies solar system temperature measurements to the calibration table, was improved to avoid writing over **CL** table one and make a **CL** version two automatically when needed.

Imaging

The relatively new imaging and deconvolution with corrections task **WFCLN** required a number of corrections for this release. The default handling of clean boxes was not as a user would have it and the meanings of the channel selection and averaging adverbs were not consistent throughout the task. The option to restart with a pre-existing work file was turned off since it appears that it can never be done in a quick mode and it may be the cause of a number of mysterious failures. The imaging task **HORUS** was found to be using only one IF by default, which was a (nasty) surprise to continuum users. It was changed to do **SUM** mode on continuum data by default, to inform users whenever their data selection adverbs are overridden, and to prevent a failure due to channel-based data editing.

A minor bug in **SUBIM** causing it to fail to correct the data extrema in the output header was fixed. Fourier-transform task **FFT** was corrected to stop trashing rectangular images. **VLBP**, which applies wide-field instrumental polarization corrections to VLA snapshot images, was fixed to use the correct geometry in its scratch files and to write its output all in the correct place even when some of the input adverbs have default values. **OHGEO**, which corrects and alters image geometry, had a serious bug corrected. To quote the **CHANGE.DOC** entry: "Apparently correct behavior in the distant past is mystifying."

Holographic imaging of antenna surfaces is supported in *AIPS* by two tasks, **UVPRT** which translates the pseudo-*uv* data into text files and **HOLGR** which reads the text files, grids the data, solves for offset of the antenna vertex plus corrections for pointing, focus, and feed offset, and Fourier transforms the data to make images of the telescope. These programs were substantially revised for this release. **UVPRT** was given a new holography mode with a number of user-controlled and automatic data editing functions to limit the data written to the text files. **HOLGR** was changed to offer new tapering options, to do uniform weighting by default, to compute the weight and observation beam images from all of the weighting and tapering, to ignore samples outside the grid, to parameterize the image dimensions and to use equivalences to avoid using too much virtual memory, to give the user some control over the resolution of the image of the telescope's beam pattern, and to allow images up to 512 on a side.

Image analysis and display

Task **SAD** is intended to "search and destroy" Gaussian components in images. It encountered the first real image from the VLA D-Array Survey and received a variety of improvements to allow it to cope. Corrections were made in the units of the clean beam position angle, in disk access while subtracting components, in the estimate of the error in the position angle, and in the corrections of the component widths and position angle for projective geometry. Solutions were made more reliable by changes to fit the components in descending flux order, to limit initial guesses to reasonable widths, and to determine the extrema and rms of the residuals around each component only after all components are subtracted. New options were added to include a flux-dependent component in warning levels and to allow the user to specify conditions under which components are rejected and to display information about those rejections. Anticipating spectral-line (or time domain) usage, a column for data cube plane number was added to the model fit table file. A new task called **MFPR** was written to display the contents of the model fit file in a format suitable for certain non-*AIPS* modeling software.

To provide external software with better access to values from *AIPS*, task **IMEAN** was changed to write histogram values into its output text file and task **SLICE** was changed to write slice values to a text file. Task **IMERG** uses an interesting FFT-based algorithm to merge images of differing resolution. It was corrected to remove several coding errors which prevented it from doing anything. For single-dish users of *AIPS*, task **PRTSD** was corrected in its handling of spectral channels (it applied **BCHAN** twice) and in its formats. **IMLIN**, which fits and removes "continuum" from data cubes, was corrected to handle missing data gracefully, a condition which arises naturally in single-dish imaging.

A new display verb, **TVSTAR**, was written to plot positions from a "star" (**ST**) file directly on a TV. The color PostScript task **TVCP** was enhanced with options to invert the coloring, to add a user-specified string at the bottom of the picture, and to read the data from disk rather than from the TV (while still using the TV transfer functions). Both it and **LWPLA** were corrected to write only correct encapsulated PostScript. Three-color display tasks **TVHUI** and **TVRGB** were corrected in their enhancement algorithms, in their image coincidence checking, and in their handling of blanked pixels. **KNTR**, which plots multiple contour drawings per page, was corrected to plot the specified axis labeling type.

Magnetic tapes

Because people are now transferring all of their old data tapes to a relatively few DAT or Exabyte tapes, we have improved PRTTP to attempt to report on their contents. In particular, it was changed to skip over null files (if requested), to report on unknown formats, to write any error messages into the print file as well as the message file, to keep going as long as possible despite errors in data, to write something to the output text file (on PRTLEV = -3) for every format, and to handle large numbers of sources in VLA archive files. The tape-mounting software was masking errors from higher-level code, causing the tape daemon **TPMON** to think tapes were mounted when they were not.

Tape routines were tested on a variety of new computer systems. Improved low-level routines were developed for Sun Solaris, DEC, and Linux. The latter two are “byte-swapped” architectures. Errors translating ModComp and FITS-standard floating-point numbers to local floating point were discovered on these while running FILLM and FITLD, respectively. FILLM was also corrected to accept an “end-of-information” return code as a signal to quit normally; it was only accepting end-of-file. A bug in the Sun-specific low-level code caused users at the NRAO to write tapes with no filemarks between FITS files (over a two-week period). Task TCOPY was enhanced to recognize and, optionally, correct this problem.

Miscellaneous changes of interest to users

Other corrections and improvements made to *AIPS* for the 15JUL94 release include:

POSSM Changed task to add an option to reverse the direction of the plot added, to fix a one-channel offset in the axis labeling (yet again), and to fix a weighting problem for IFS > 1.

CHKNAME New verb to determine whether the file specified by **INNAME**, *et al.* exists and to return a status adverb. It should be useful in large procedures.

EXTDEST Changed to require confirmation before deleting version 1 of either a **CL** or an **HI** file. One should almost never do either of these.

TELL Batch tasks may now actually receive parameters from the **TELL** verb. Thus, an interactive user can alter the parameters of, or request a graceful termination of, his/her batch tasks.

helps Updated the general help files to point at **ABOUT** and continued to keep current the files used by **ABOUT** and **APROPOS**.

Improvements Primarily for Programmers in 15JUL94

Ports to new operating systems

The new operating system for this release is Silicon Graphics. We extend our thanks to Jeffrey A. Pedelty of the Goddard Space Flight Center for doing the initial port and then making his computer available to us for later testing. It is a relatively standard Bell port with a few differences. In particular, clock ticks are represented by a different standard symbol (HZ) and value (100), the Unix pipes routine **popen** has a non-standard type requiring a cast to the usual type, and terminal characteristics are set by methods other than those used previously. So far as we know, all of the port is working, but we have not tested the magnetic tape routines. These routines have been brought up to the form and standards of the best supported systems using the SGI includes and data structure. However, they are likely to have minor problems and will probably appear in due course in the patch area for 15JUL94.

Miscellaneous changes for programmers

Byte-swapped computers are making a renaissance, so programmers should remain aware of the difficulties they pose and the *AIPS* parameter (**BYTFLP** in Fortran, **Z_bytflp** in C) carried around to describe byte and word swapping. The relatively generic routines for converting FITS-standard (**ZR64RL**) and ModComp (**ZRM2RL**, **ZDM2DL**) floating-point binary numbers to local format all had to be corrected for this release.

Magnetic tapes remain a software porting nightmare. Most versions of **ZTAP2** were improved for this release, removing excess operations. This can be dangerous as EOFs were removed excessively for 2 weeks on Suns. The Solaris Z routines were upgraded to current standards. The DEC versions of the tape routines were made to work, corrected to handle `errno 28` which DEC uses for EOI, corrected in their error handling on back-files, and enhanced to use another `ioctl` to obtain additional status information. The Linux Z routines were also made to work. In that case, there are a number of caveats. Linux systems are really opposed to writing more than one file and virtually require that to be at the previous end of information. By adding a variety of slow and normally un-needed tape operations, we have been able to overcome most of the limitations of the Linux tape driver. Unfortunately, we cannot prevent it from writing an extra end of file whenever a tape opened for write is closed. Fortunately, in *AIPS*, such closes only come when the tape is actually at EOI already. If we rewound the tape before the close, Linux would put an EOF at the start of the tape, destroying its entire contents. As it is, Linux users should either avoid `DOEOT TRUE`, or learn to tolerate a large number of null files on their tapes.

The area of printing also received some attention for this release. To remove our dependence on non-standard filters to convert Fortran print files to plain text and then to PostScript, we have written our own filter **F2PS** to convert Fortran or plain text to PostScript. We have also written a filter called **F2TEXT** to convert Fortran output to plain text printers. These filters take, as an argument, the number of lines per page assumed by the *AIPS* task and attempt to do something reasonable with it. *AIPS* deletes print and plot files from the `/tmp` area after a delay interval sufficient to get the file plotted. Since "sufficient" varies, we changed the system to make this interval available to the *AIPS* Manager via `SETPAR`. The delay may then be set and changed simply for all hosts in your network together or individually.

Other matters of interest to programmers in 15JUL94 include

large disks Several versions of **ZCREA2** were fixed to avoid integer overflow problems on disk systems larger than 2 Gbytes. Disk free space can no longer be measured in bytes!

disk access The disk-use permission parameters have been extended to include *AIPS* disk 1. Previously, all users were allowed to read and write disk 1, but the new network-wise *AIPS* allows users to define an otherwise forbidden disk as disk 1. *AIPS* user number 1 now has privilege on all disks and no longer needs to be listed in `$NETO/NETSP`.

standards One of the compilers has forced us to require that commas separate all format items in **FORMAT** statements. This is the ANSI standard, but most compilers have been forgiving.

image size The verbs **GO**, **INPUTS**, and friends were changed to apply an upper limit of 8192 for image corners, ignoring any limits typed in the help files. Code should anticipate images of this size in the future.

AIX The print routines **ZLASC2** and **ZLPCL2** were not interpreting error code from pipes correctly on IBM systems. This led the `/tmp` area to overflow.

swap space *AIPS* tasks are now requiring large amounts of swap space, partly from large arrays of their own using. When "shared libraries" are used, however, the swap requirements increase to include all subroutines in the shared libraries. Some attempts to remove subroutines from libraries has been made and other attempts should be considered. Large buffers should be coded into tasks and passed down to subroutines. Otherwise those buffers are included in the swap-space requirements of all tasks that use that subroutine library. The current swap-space requirements for **TVFLG** under Sun OS 4.1.2, for example, are 13.24 Mbytes when linked statically and 24.76 Mbytes when linked with shared libraries. For **FITTP**, which uses neither the TV libraries nor the calibration parts of **\$APLNOT**, the difference is even more extreme, with the static binary requiring only 0.85 Mbytes while the dynamic binary requires 21.4 Mbytes. These numbers show that the saving in disk space for the load modules, currently about 50 Mbytes, is less than the amount of extra swap space required for *AIPS* on each workstation in the system.

AIPS Publications and the World-Wide Web

There has been a virtual explosion in the use of the *World-Wide Web* (WWW) protocol on the Internet. It is a method for sending hypertext over the network and has been made easy to use by clients such as *NCSA Mosaic* and *Lynx*. NRAO is among the many institutions which now offer informative “home pages” and networks of additional information. The NRAO home page is at the Universal Resource Locator (URL) address

<http://info.aoc.nrao.edu/>

This page can lead you to information about each of the NRAO’s sites and telescopes, library system, major new initiatives, software packages, phone directory, and Newsletter as well as information about other astronomy resources on the Internet and about *Mosaic* and its mark-up language called *html*. The *AIPS* group home page may be found from the NRAO home page or addressed directly at URL

<http://info.cv.nrao.edu/aips/>

This page points at basic information (“What is AIPS?” and a “FAQ”), news items about *AIPS* (such as “15JUL94 Release imminent ...”), the PostScript text of recent *AIPS Letters*, patch information for all releases after 15JAN91, information about known bugs, the latest *AIPS* benchmark data from various computer systems, copies of *CHANGE.DOC* for every release since 15JAN90, and *all* relevant *AIPS* Memos, *every* chapter of the *CookBook*, and all recent quarterly reports to the NSF. We recommend that you check this area occasionally. As we correct and update the *CookBook* and as we write new *AIPS* Memos and reports, we place the documents in PostScript forms in the area used by WWW and we have the *html* listings, indices, tables of contents, and the like updated to reflect the additions. In this way, you do not have to wait until 15JAN95 to get the latest *CookBook* chapters. We expect, for example, to have revised versions of the Spectral-line and VLBI chapters in the relatively near future.

Below is a list of the latest *AIPS* Memos. Memos 86, 87, and 88 are new in this *AIPS Letter*, although 88 is simply a *TEX*-set and slightly updated version of several old internal memos.

Memo	Date	Title and author
84	93/11/12	A Proposed Package to Support the Use of the X Window System in AIPS Tasks Chris Flatters, NRAO
85	94/02/01	DDT Revised and AIPSMark ⁽⁹³⁾ Measurements Eric W. Greisen, NRAO
86	94/03/16	Wide-field Polarization Correction of VLA Snapshot Images at 1.4 GHz W. D. Cotton, NRAO
87	94/04/05	The NRAO <i>AIPS</i> Project — a Summary Alan H. Bridle, Eric W. Greisen, NRAO
88	94/05/16	The <i>AIPS</i> Gripes Database W. D. Cotton, Dean Schlemmer, NRAO

A heavily revised edition of the Memo on Object-Oriented Programming in *AIPS* is available as file *AIPSOOF.TEX* and, in PostScript form, as *AIPSOOF.PS*.

Since some Memos are not available electronically and others do not yet have computer readable figures, you may wish to write for a paper copy of these. To do so, use an *AIPS* order form or e-mail your request to aipsmail@nrao.edu.

To use *ftp* to retrieve the Memos, *CookBook* chapters, etc.:

1. *ftp* [baboon.cv.nrao.edu](ftp://baboon.cv.nrao.edu) or 192.33.115.103
2. Login under user name anonymous and use your e-mail address as a password.
3. *cd* pub/aips/TEXT/PUBL
4. Read *AAAREADME* for more information.
5. Read *AIPSMEMO.LIST* for a full list of *AIPS* Memos.

AIPS Memos from Number 65 through 88 are present in this area as are Numbers 27, 33, 35, 39, 46, 51, 54, 61, and 62. We have been filling in this list gradually, by finding and fixing old files in other areas of the authors’ disks, by scanning in text and figures, or by retyping text and redrawing the figures. The

AIPS Letters from 1991 through the present are also available in this area. Many of the Memos are in both TeX and PostScript forms, with the TeX ones stored in a subdirectory called **TEX**. Note that many, if not all of these may be found on your home *AIPS* system in an area called **\$AIPSPUBL**. All Memos are available in paper form from Ernie Allen at the addresses in the masthead.

The latest version of the *AIPS Cookbook* is also available (in the form of PostScript files) in this area. Initially the chapters from the 1990 version of the *CookBook* were placed in this area; whenever one of these chapters is updated, the latest version will be available immediately in this area. Updated so far are chapters 1 (Introduction), 2 (Starting Up *AIPS*), 3 (Basic *AIPS* Utilities), 4 (Calibrating Interferometer Data), 15 (Current *AIPS* Software), and Z (System-Dependent *AIPS* Tips). Chapter 10 (Spectral-Line Software) is nearly ready as well. The remaining old chapters were revised to include figures in the PostScript and improve the typesetting, but are full of outdated information in addition to the good stuff.

Patch Distribution

Since *AIPS* is now released only semi-annually (or even less frequently), we make selected, important bug fixes and improvements available via *anonymous ftp* on the NRAO Cpu **baboon** (192.33.115.103). Documentation about patches to a release is placed in the anonymous-ftp area **pub/aips/release-name** and the code is placed in suitable subdirectories below this. (The patches and their documentation are also available on-line via the World-Wide Web.) Reports of significant bugs in 15JAN94 *AIPS* were more numerous than recent previous releases. The documentation file **pub/aips/15JAN94/README.15JAN94** mentions the following items:

FRING **FRING.FOR** was corrected to be consistent in its use of buffers and to avoid telling the user incorrectly to use a **SOLINT** of 0.0.

FRING.FOR was corrected to avoid aliasing in the default (Nyquist) search windows. This aliasing appears to have been responsible for bad solutions for residual delay and rate.

FRING.FOR was corrected for a bias toward zero in the solutions for delay using narrow search windows. This caused non-closing errors in the amplitudes degrading the dynamic ranges of the resulting images.

ANCAL **KEYIN.FOR** was changed to increase the line length in text files. This is needed in **ANCAL** to read T_{sys} information for large numbers of IFs.

ANCAL.FOR was corrected for a number of indexing errors in the processing of T_{sys} and T_{ant} entries. **ANCAL.HLP** was corrected to fix misleading information about T_{sys} groups.

SPLIT **SPLIT.FOR** was changed to keep frequencies consistent in all parts of the output (the antenna tables were not changed previously) and to reset the polarization corrections to prevent them from being applied twice.

MX **MX.FOR** was corrected for a data-selection bug in making spectral-line images while "averaging" multiple IFs. The wrong data were selected in all but the first output channel. Fortunately, this is not a likely mode of imaging.

FFT **FFT.FOR** was corrected to stop mangling rectangular images.

KNTR **KNTR.FOR** was corrected to use all the input adverbs and to do the desired type of axis labeling.

true color **TVHUI.FOR** and **TVRGB.FOR** were corrected to use the TV cursor for image enhancement in the usual way, to be more correct and complete in checking image alignment, and to be more forgiving about blanked pixels.

IMERG **IMERG.FOR** was not previously able to run.

PRTSD **PRTSD.FOR** was corrected to display the requested channel (it was offsetting twice) and to improve minor formatting problems.

FILLM **FILLM.FOR** failed to test for end-of-information, depending on end-of-file only. SUN returns EOI for double end-of-files on 9-track tapes, leading to an infinite loop of error messages.

PRTTP PRTTP.FOR was enhanced to handle null files, to do something useful with unrecognized formats, to force all error messages to the printer file, to try to continue despite data-format errors, and to be rather more bomb proof in general.

DEC ZTAP2.C, ZTPWA2.C, and ZMOUN2.C for DEC Ultrix and DEC Alpha systems only were changed to recognize beginning and end of tape correctly and to do a few other things better. Affects all tape tasks (including AIPS), but only for DEC systems.

DEC, PC ZDM2DL.C and ZRM2RL.C were corrected to run correctly on byte-swapped (little-endian) computers. FILLM.FOR was corrected to assure correct alignment of buffers.

big disks ZCREA2.C for several systems was changed to handle disk systems with more than 2 Gbytes of free space. There was no problem for more reasonable systems and in some of other versions of ZCREA2.

printing ZLASC2.C and ZLPCL2.C did not work correctly on IBM AIX systems, causing /tmp to fill up among other things.

HP LIBR.DAT for HP systems only needed a small addition in order to link edit the \$AIPNOT area correctly.

Note that we did not revise the original 15JAN94 tape or tar files for these patches. No matter when you received your 15JAN94 “tape,” you must fetch and install these patches if you require them. See the 15APR92 *AIPS Letter* for an example of how to fetch and apply a patch. Information on patches and how to fetch and apply them is also available through the World-Wide Web pages for *AIPS*. As bugs in 15JUL94 are found, the patches will be placed in the ftp area for 15JUL94. As usual, we will not revise the original 15JUL94 tape or tar files for any such patches. No matter when you receive your 15JUL94 “tape,” you must fetch and install these patches if you require them.

CookBook Update Begun

The *AIPS CookBook* was last updated for the 15OCT90 release. Because a lot has changed in *AIPS* since then, we have decided to upgrade the *CookBook*. We are doing this one chapter at a time and are making each chapter available via the World-Wide Web as soon as it is ready. For details of the Web, see the publications article in this *AIPS Letter*. The chapters changed so far are

- 1 — *Introduction* — New sections giving a project summary and a diagram of the structure of *AIPS* were added.
- 2 — *Starting Up AIPS* — Changed to describe workstation use, *AIPS* in networked environments, and managing the TV server XAS.
- 3 — *Basic AIPS Utilities* — Updated information about history files and disk allocation, added ABOUT and APROPOS to the help section, moved and updated tape mounting, and added a discussion on external disk files (Fits, text, ...).
- 4 — *Calibrating Interferometer Data* — With much help from Rick Perley and Alan Bridle, rearranged and corrected everything, adding a substantial discussion of when and how to edit and bringing the description of TVFLG up to date including a picture.
- 15 — *Current AIPS Software* — Replaced old lists with new ones produced for the ABOUT verb.
- Z — *System-Dependent AIPS Tips* — Replaced with whole new discussions including color printers, screen copying, film recorders, workstation environments. A method for people to have NRAO make slides for them is described.

There exist drafts of both Chapters 10 *Spectral-Line Software* and 11 *Reducing VLBI Data in AIPS* which will be released in the relatively near future. Elias Brinks has helped with the spectral-line chapter, which will be more up to date with less emphasis on HORUS and more on MX and WFCLN, and more attention for the IMLIN and UVLIN type of continuum subtraction. The VLBI chapter is a major rewrite to account for the advent of the VLBA. It is our hope to update the other chapters, add a chapter on single-dish data in *AIPS*, and even to add an index to the *CookBook*.

AIPS Order Form for 15JUL94 (Unix)

1. Name and address of Contact Person: _____

☐ Check here if address on back is ok. _____

(Include street address for UPS delivery) _____

2. ☐ New order ☐ ReOrder

Do you have an AIPS user agreement? ☐ Yes ☐ No

(The agreement is a prerequisite for obtaining AIPS code)

3. Computer make, model and OS version: _____

(Include manufacturer and model of computer, and Version number of Operating System, e.g. Sun 4/SunOS 4.1.2, Intel 486DX2/66 Linux 0.99.15, IBM RS/6000-530/AIX 3.2, etc.)

4. DDT test package (15JAN94 version): ☐ (includes small, medium, and large tests;
(FITS tape) 9-track 1600bpi tape does not have large)

5. VLA Calibration test tape:..... ☐

6. Tape "media" desired: ☐ Internet FTP ☐ 4mm DDS DAT ☐ 9-track 6250 bpi
(Unix "tar" format) ☐ Exabyte (5GB) ☐ Exabyte (2.2G) ☐ DEC TK50
☐ QIC 24 (60 MB) ☐ QIC 150 (150MB)

7. Are there "gripes" on the returned tape? ☐ Yes ☐ No

8. Printed Documents Requested: ☐ 15OCT90/15JUL94 AIPS Cookbook
Going AIPS (15APR90) ☐ Vol 1 ☐ Vol 2
☐ AIPS Memos: _____
(See separate list)

9. Custom 3-ring binders Cookbook ☐
Going AIPS ☐ Vol 1 ☐ Vol 2

Send this order form (or an electronic equivalent) to:

AIPS Group
National Radio Astronomy Observatory
520 Edgemont Road
Charlottesville, VA 22903-2475
USA

Network Addresses
Internet: aipsmail@nrao.edu
UUCP: ...uunet!nrao.edu!aipsmail

July 15, 1994

CookBook revision begun:



Panel of *ATPS* users tests new recipes



AIPSLETTER

National Radio Astronomy Observatory
520 Edgemont Road
Charlottesville, VA 22903-2475

NON-PROFIT ORG.
U.S. POSTAGE PAID
PERMIT # 373
CHARLOTTESVILLE, VA

ADDRESS CORRECTION REQUESTED

TO:

Library
Nat. Radio Astronomy Obs.
520 Edgemont Road
NRAO

DATED MATERIAL-DO NOT DELAY