AIPS MEMO NO. 5

To: Ed Fomalont, Eric Greisen, Bill Cotton, Walter Jaffe, Arnold Rots and Jim Torson

From: Thijs van der Hulst

Subject: Experiences with Aips

During the present (September 1981) visit to the VLA I have been using Aips and the Vax system quite extensively. During my data reduction session I developed the habit of writing notes on the last pages of my note book about all the features of Aips that I would have used had they been available. I wrote a few things in the official note book, but decided to generate a somewhat more extensive list of points and be more descriptive than I might otherwise have been.

The feature of Aips that gave me the most trouble all the time is the fact that parameters are global. The possibility of having the save and get files helps a lot because you can save and recall the inputs that you defined specifically for one or another step in the data reduction, but you still have to keep your own bookkeeping of these files. If you are doing a lot of processing like on snapshots, different programs will run at the same time, e.g. UVLOD, UVSRT, UVMAP, APCLN, CNTR and things like IMEAN and PRTHI. Each program will have different parameters because they pertain to different fields in a different stage of the reduction. Although getting and saving the individual get files for the individual programs is certainly possible (provided you remember where you put things) it would be a million times easier if the system did this for you.

I understand that task shedding is really task shedding, so that (at least the way things are set up now) it becomes very cumbersome if at all possible to have tasks communicate back to the user (i.e. Aips1 and Aips2) about their status other than dumping every thing one one screen. I found it very awkward to have to try to find the result of a source fit amidst the dozens of lines that the task that rane for the other Aips user produced.If there is a way to give each user his own task status monitor than please implement it. I am sure it would make many people very happy.

When many people are on the machine, or many tasks and verbs are running, the response of the TV for things like TVTRANS and TVPSEUDO becomes awkwardly slow. Whether one can do something about that I do not know, but I found it very hard to "fine tune" a transfer function when the display does not almost immediately respond to the trackball.

To stay with the display system: a few suggestions are

- make a "hole" in the center of the cursor so you can see where you point it
- give the user the option to have a cursor with lines that run over the full screen: handy for defining large windows
- use one of the overlay planes to give the cursor's position directly on the screen close to the cursor itself so that you do not loose that information when you zoom the image.
- built in the possibility to steer the cursor to a specified position
- to make GREYS smart enough to read the transfer function so that you can get a copy of the currently displayed map in more or less the same fashion as you see it on the screen

A row of other options that I would have liked to have at my

- the possibility to plot fiducial marks (star positions etc.) on contour plots. The Imps plot version has that option which I would use quite frequently.
- a little more information on the contour plots would be welcome: the scale in "/mm, one more decimal place for the contours, the date and maybe even the time.
- since every map gets its own beam, why not have an entry in the map header that tells you which beam goes with it. The advantage is that programs like APCLN could be made smart, read the header and use the appropriate beam unless maybe the user indicates otherwise. Another advantage is that the IMEAN program can be improved to calculate fluxes from dirty maps by reading the beam, integrating over the appropriate area and dividing the dirty map integral by it.
- APCLN might have a little subprogram that calculates the rms after each minor cycle. What about making APCLN really interactive and have the user "push a button" when he is satisfied with the residual map on the display?
- IMFIT is a very helpful addition that I used a lot. Two remarks: its output does not indicate which map it worked on, and could it be made smart enough to find the strongest source, fit it, subtract it and then go on if wanted to the next strongest source?
- an option in PRTCC to print only from component n to m and if necessary skip every k'th component
- since the clean beam and and the number if clean iterations is put in the map header, why not also the total cleaned flux?
- let PRIO default to a higher value. If the user wants his message file he can always set PRIO to a lower value. Or ask the user when he exits whether he wants his messages, and destroy them when he says no.
- a program to list fits headers from the tape on the printer to have an index of your tape

I hope that this list of some of the things I missed the most helps to further develop the Aips system. (I do not like the name but that is not important)