PLANS FOR THE "PIPELINE" SYSTEM

Written by Jim Torson
Version 3
February 7, 1983

1. INTRODUCTION

The purpose of this document is to provide an overall description of the Pipeline system. This is intended to allow potential users of the system to gain an understanding of what is being planned so that they will be able to provide constructive comments. This document is also intended to serve as a design document for the people who are working on implementing the system. This should increase the chance that the various pieces will fit together and produce a properly functioning system.

2. OVERVIEW OF THE SYSTEM

2.1. PURPOSE OF THE PIPELINE

The main objective of the VLA Pipeline System is to provide the mapping throughput necessary to operate the VLA at full efficiency, including the full spectral line capacity designed into the hardware. The most important factor governing its throughput is obviously the computer capacity as measured in I/O bandwidth and in compute cycles. The major concern in the design of the Pipeline has been to provide this capacity. However, this alone is not enough to make the Pipeline successful. If the Pipeline is to have its full impact on the amount and the quality of science done with the VLA it also has to be one of the astronomer's tools. It will have to be able to cope with less than perfect data. It must have the flexibilty to perform the kind of mapping functions desired by the astronomer based on his assessment of the data and his objectives. Finally, it must enable him to select a subset of data to take away for further analysis.   This subset should contain all the relevant information but be sufficiently reduced in volume to take away on a reasonable number of magnetic tapes. For example, a spectral line observer should be able to see the full field observed in all of the channels observed with all significant instrumental errors removed before he has to make the (unfortunately) necessary decision to reduce the database until it only contains the object of interest. If the VLA cannot meet objectives such as these we severely limit our chance of making new discoveries and we will be removing a critical segment of the research path in which the astronomer should participate.

2.2. HARDWARE CONFIGURATION

Figure 1 is a diagram of the Pipeline hardware configuration. All of the disk drives in the system (except the fixed head disk) are essentially identical with a capacity of approximately 250 megabytes each.

The connection between the online Modcomp system and SORTER consists

of four shared access disks. The connection between SORTER and GRIDDER is not part of the DECNET system although it uses a DA11, which is the same as the hardware used for the other DECNET connections between PDP-11's. The DA11 is currently set to operate at about 100K baud. The connections between the DEC-10 and SORTER and GRIDDER are DECNET connections that are currently set to operate at 56K baud. In addition to the connections shown, both GRIDDER and DISPLAY have DECNET connections to the MAPPER PDP-11/70.

GRIDDER has four disk drives, two of which can be read (but not written) by DISPLAY.

SORTER, GRIDDER, and DISPLAY each have a single 1600/6250 bpi tape drive.

The following is a summary of the graphics devices connected to DISPLAY:

Image Display	IIS Model 70F. Two 512x512 image planes, graphic overlay, cursor, look-up tables for color & gray scale.
Image Storage System	NRAO. Storage for 600 512x512 images with film loop playback on IIS display.
Refresh Vector Display	DEC VT-11. Line drawing graphics.
Data Tablet	Summagraphics. IIS and VT-11 cursor positioning.
Control Panel	NRAO. Switches, knobs, etc. for controlling image storage system and IIS look-up tables etc.
Gray Scale Copier	Honeywell VGR-4000. Gray scale copy of IIS image without use of software.
Film Recorder	Dicomed. 4Kx4K resolution color or gray scale images on 4x5 film. 1024x1536 resolution on 35mm film.
Dot Matrix Plotter	Versatec D-1200A. 200 dots per inch. 11 inch wide plots.
Pen Plotter	Zeta Model 1453B. 12 inch wide drum plotter with four pens.

2.3. OVERALL FLOW OF DATA THROUGH THE SYSTEM

During observing, all visibility data will be trasferred from the online Modcomp system to the SORTER system where it will be partially sorted and stored on SORTER's disks. A copy of all visibility data will

also be written to tape on the Modcomp system. In addition, all visibility data for calibrator observations will be sent to the DEC-10 along with a description of the other sources that were observed. Calibration and correction will be done in the DEC-10 and will produce gain files on the DEC-10 disk. When a map is to be made, the gain files will be transferred from the DEC-10 to GRIDDER and the visibility data will be trasferred from SORTER to GRIDDER. The visibility data and gain data will be processed by the array processors and the transpose memory to produce the map, which will be put onto GRIDDER's disk. The display system will read map data from the disk that is shared with GRIDDER in order to display the maps that have been produced. Finally, map data will be written to FITS tapes either on GRIDDER or DISPLAY. Visibility data will be saved on tape on the SORTER system.

OVERALL SYSTEM THROUGHPUT

The current design goal of the system is to acheive a data throughput rate of about 6000 channels per second. Each channel is considered a complex number.

3. THE USER'S VIEW OF THE PIPELINE SYSTEM

Figure 2 shows a functional overview of how the user will use the Pipeline system. In some ways the Pipleine will appear to be similar to the current MAPPER system. The user will run a program on the DEC-10 to request a map making operation. The resulting map will be placed on the disk connected to a PDP-11 (GRIDDER). In some ways displaying the maps that are produced by the Pipeline will be similar to the use of the current IMPS system. However, use of the Pipeline system will be different from use of MAPPER/IMPS in several ways. It will not be necessary to transfer maps from one PDP-11 to another via DECNET links or FITS tapes in order to display the maps since the display system will have direct shared access to the GRIDDER disks which store the finished maps. Also, after looking at a map with the display system, the user will be able to submit additional mapping requests directly from the display system rather than being required to go back to the DEC-10. The following sections describe the appearance of the Pipeline system from the point of view of the user.

3.1. DEC-10 STANDARD COMMANDS

Pipeline system functions which are invoked by running programs on the DEC-10 will use the DEC-10 standard commands.

3.2. STANDARD COMMANDS IN THE PDP-11'S

Some Pipeline system functions will be invoked by running a program on one of the PDP-11's. These programs will use a command scanner which provides standard commands that work exactly like the DEC-10 standard commands. This will include such things as FITS tape reading/writing and loading visibility databases from tape. The user interface for display system functions will be somewhat different. This is described in another section.

3.3. PREPARATION OF OBSERVING SPECIFICATION

The user will prepare his observing specification by running the DEC-10 OBSERV program just as is currently done.

3.4. GETTING VISIBILITY DATA INTO THE PIPELINE

3.4.1. DEC-10 AND SORTER VISIBILITY DATABASES

The Pipeline system will make use of visibility databases on both the DEC-10 and the SORTER computers. These databases will differ in both content and organization.

The user's visibility database on the DEC-10 will be very similar to the current DEC-10 visibility database. It will consist of an index file, a gain file, and a visibility file. The index and gain information will be identical to the current format. Complete visibility data for calibrators will be stored just as in the current system. However, for non-calibrator sources, no visibility data (or perhaps just scan averages) will be stored. The data will be stored in baseline-time order as is currently done on the DEC-10.

Complete visibility data for both calibrators and non-calibrators will be stored in SORTER. Also, the data stored in SORTER will be partially sorted according to the absolute value of the "U" coordinate. The database for each user will consist of a visibility catalog file (which is different from a map catalog file) and a set of visibility data files. For a continuum observation, each visibility catalog record will describe all the data for the observation even if multiple "scans" are involved. For spectral line data, each visibility catalog record will describe a group of eight frequency channels. Thus, a spectral line observation may require multiple entries in the visibility catalog. Note that this organization will make it possible to delete visibility data that pertains to just a single source.

3.4.2. FILLING VISIBILITY DATABASE DURING OBSERVING

During normal operations, all of the observed visibility data will be automatically sent from the Modcomp system to SORTER. Also, the visibility data for calibrator observations will be automatically sent from the Modcomp system to the user's disk area on the DEC-10.

The full data rate produced by the Modcomps will be about 8000 channel-baseline measurements per second, where each measurement is a four byte number. At this rate, SORTER's disks can be filled in about ten hours. It is expected that 20 second or 30 second averaging will often be done in the Modcomps, which will reduce the data rate by a factor of two or three over the maximum rate.

3.4.3. FILLING VISIBILITY DATABASE FROM TAPE

If for some reason the on-line visibility database filling did not occur, then the database can be filled from tape. To fill the DEC-10 database, the user will run a modified version of the current DEC-10 tape FILLER program. To fill the SORTER database, the user will run a SORTER program called DBFILL. This program will not be able the do data averaging as it fills the database. (This program currently exists. See the document written by Bob Duquet for details.) In the full spectral line case with 10-second averaging, the data rate may be too high to be written out on Modcomp archive tapes. In this case the data will have to be sent to SORTER in real time. In this situation, SORTER can be considered to be an essential part of the on-line system. Although the

Modcomps will be able to do data averaging if desired, they will not be able to send unaveraged data to SORTER at the same time averaged data is being written to tape, i.e., the same averaging interval must be used for data sent to SORTER and data sent to tape.

3.4.4. DELAYED DATA PROCESSING

For observations involving very large databases, processing will have to be done immediately after the observed data is transferred from the Modcomps to SORTER. However, for small databases, it may be possible to dump the raw visibility data from SORTER to tape and then restore the data to SORTER's disks and process it at a later time.

3.5. LISTING VISIBILITY DATA

The user will be capable of listing visibility data stored in the Pigeon Hole database. Currently it is possible to list parts of the SORTER visibility database by running the SORTER program DBUTIL. The listing capability of this program is designed for program debugging and not for listing in a useable fashion for astronomers. It is currently possible to display baseline-time data from IMPS.

3.6. CALIBRATION AND CORRECTION

3.6.1. APPLYING KNOWN CORRECTIONS - GTBCOR

Applying known corrections to the visibility data will be done by running the DEC-10 program GTBCOR just as is currently done. This will modify the gain file stored on the DEC-10 disk to correct for time errors, position errors, frequency errors, shadowing, etc.

3.6.2. DETERMINING ANTENNA GAINS - ANTSOL

Determining antenna gains will be done by running the DEC-10 program ANTSOL just as is currently done. This will examine the calibrator visibility data stored on the DEC-10 disk and produce a .CAL file on the DEC-10 disk.

3.6.3. APPLYING CALIBRATION - GTBCAL

The calibration information contained in the DEC-10 .CAL file will be applied to the DEC-10 gain file by running the DEC-10 program GTBCAL just as is currently done. The gain table will be shipped to SORTER at the time map requests are made. Application of the gains will occur on the fly during gridding in the APs.

3.6.4. PASSBAND CORRECTION

The passband corrections for spectral line data will be calculated by SPECTRE.

3.6.5. POLARIZATION CALIBRATION - POLCAL

The polarization corrections will be calculated from calibrators stored in the DEC10 database. The correections will be stored in the gain file in exactly the same manner as is currently done.

3.7. MAPPING REQUESTS FROM THE DEC-10

3.7.1. MAKING REQUESTS

The user will request that maps be made by the Pipeline system by running a program on the DEC-10. This will be very similar to the way mapping operations on MAPPER are currently requested. There will of course be some differences in mapping options that are available. The mapping request will be entered into a queue of requests to be processed. After the request has been processed, the finished map will be stored on GRIDDER's disk.

In a single request, the system will be capable of producing just a map, just a beam (antenna pattern), or both a map and a beam. Also, a single request will be able to produce all polarization maps. For the spectral line case, a single request will make an entire spectral line map data cube.

The mapping requests shipped to SORTER will result in the transfer of the gain table (file) to SORTER for subsequent application to the data.

3.7.2. MAP SIZE LIMITATIONS

The Pipeline system will be capable of producing continuum maps that are up to 8096x8096, although the typical size will be much smaller (less than 1024x1024). The user will be able to copy maps of this size or put them onto FITS tapes. However, most other functions in the system such as Versatec contouring will be limited to processing a 1024x1024 subsection. The system will be capable of producing rectangular maps.

3.7.3. MAPPING SPEED

The mapping speed of the Pipeline system will be significantly greater than the speed of MAPPER or AIPS. There are several reasons for this. First, the visibility data stored in SORTER is partially sorted. This will eliminate the need for sorting when a map is to be made. Second, the system will be able to make multiple maps at the same time, e.g., a set of spectral line channel maps. This will result in various

efficiencies. Third, since three array processors will be involved in processing a mapping request, separate parts of the mapping process will be done concurrently.

3.8. OTHER PROCESSING REQUESTS FROM THE DEC-10

The user will be able to make other requests for Pipeline processing by running programs on the DEC-10. This will include such things as cleaning and self calibration.

3.9. CHECKING PIPELINE SYSTEM STATUS - PROBE

The DEC-10 PROBE program will allow the user to check on the status of the Pipeline system (i.e., GRIDDER processing) just as PROBE currently allows the user to check on the status of the MAPPER system.

3.10. LOGS OF PIPELINE SYSTEM PROCESSING

When the DEC-10 user makes requests for Pipeline system operations, entries will be made in a log file in his DEC-10 disk area. This will allow the user to easily see what requests have been processed. If some error condition was encountered in processing the request, then appropriate information will be included in the log file.

3.11. PDP-11 DISK USAGE

Within each of the PDP-11's (SORTER, GRIDDER, and DISPLAY), the data belonging to a given user will be kept separate from the data belonging to all other users. Although each computer will have more than one disk drive, the user will not be concerned with keeping track of which drive contains his data. From the user's point of view, the collection of disk drives on a single computer will appear to be a single large pool of disk space.

The user of the display system will of course be aware of the fact that some maps are being stored on GRIDDER's disk pool and some maps are being stored on DISPLAY's disk pool.

Unlike AIPS, removable "private" disk packs will not be supported.

3.12. MAP NAMING CONVENTION

3.13. DISPLAY SYSTEM USE

3.13.1. USER INTERFACE ON THE DISPLAY SYSTEM

The user interface on the Pipeline display system will be an improved version of the IMPS user interface. The system will display a tree-structured menu of available functions on the VT-11 display screen. The user will point to the desired function with a cursor that is controlled by the data tablet stylus. In addition to the tree-structured menu, there will be an "expert" mode of operation in which a single page (or a small number of pages) on the screen will briefly list all of the available functions. This will allow experienced users to by-pass the tree structure.

After the user has selected a function, a list of current parameters pertaining to that function will be displayed on the VT-11. This will be similar to the "inputs" listed by the DEC-10 standard commands. The current parameters will be the values that were used on the previous execution of the function by the current user. For parameters which can take on only a discrete set of values, all of the possible values will be listed, and the currently selected value will be highlighted. To select a different value for the parameter, all the user will have to do is point to the desired value with the cursor. If the user points to a parameter involving map coordinates, a cursor will appear on the IIS image display and the user will be able to interactively specify the image coordinates. This will be useful for entering such things as the map subsection to be contoured. To change the value of one of the remaining types of parameters, the user will point to the parameter and then type in the desired new value. It will also be possible to reset all parameters to their default values and to save/restore sets of parameter values in named files.

Another part of the user interface on the display system will be the control panel which is being built in Charlottesville. This device will consist of a variety of switches, knobs and buttons for interactively controlling the IIS display functions such as look-up table modification for image enhancement, blinking, zoom, roam, etc. The control panel will also be used for controlling the image storage system which is being built in Charlottesville. This will include such things as speed and direction of movie loop play-back of stored images.

3.13.2. INTERACTIVE IMAGE DISPLAY

The display system user will of course be able to access maps that are stored on the disks associated with the DISPLAY PDP-11. However, the display system user will also have read-only access to maps stored on GRIDDER's disks. This will give the user the ability to immediately see the maps produced by the Pipeline without transferring maps over DECNET connections or FITS tapes. The following is a summary of the display functions that will be available:

1. Load a map into an IIS image plane for display.
2. Load a reduced map (skipping an integral number of pixels) into an IIS image plane.
3. Generate axis labelling and annotated wedge.
4. Intensity-hue display of two images.
5. Compare two images using blink and/or split screen.
6. Display horizontal, vertical, or arbitrary orientation cross sections on the VT-11.
7. Display spectrum at selected sky coordinate on VT-11.

The image storage system will allow the user to see a sequence of stored images displayed on the IIS as a movie loop presentation. This will be useful for looking at a sequence of spectral line channel maps. The image storage system will also allow the user to rapidly save and restore selected IIS images.

3.13.3. HARD COPY OUTPUT

The following is a summary of the types of hard copy output that the system will produce:

1. Versatec or pen plotter contour plot.
2. Honeywell gray-scale copy of IIS image.
3. Versatec copy of VT-11 image.
4. Dicomed gray-scale or color film output.
5. Versatec or line printer print-out of numerical values for a map subsection.

3.13.4. IMAGE PROCESSING AND ANALYSIS

The following is a summary of the functions that will be available for extracting information from maps:

1. Interactively display coordinates and pixel value of the IIS cursor location.
2. Fit discrete source parameters (amplitude, position, size).
3. Calculate mean, rms, histogram for a map subsection.

The system will also allow the user to process maps to produce new maps. The following is a summary of the functions that will be available:

1. Produce map subsection.
2. Map arithmetic, e.g., +, -, *, /, spectral index, etc.
3. Average or sum of a group of spectral line channels.

3.13.5. MAPPING/CLEANING/SELFCAL REQUESTS FROM THE DISPLAY SYSTEM

The various display and analysis functions listed above will be executed entirely within the DISPLAY PDP-11/44. In addition to these functions, the display system user will be able to request the same Pipeline processing operations that the DEC-10 user can request, e.g., mapping, cleaning, self calibration. These requests will be entered into the processing queue along with requests from DEC-10 users. However, the requests from the display system user will be given a higher priority than the DEC-10 requests.

3.13.6. VISIBILITY DATA EDITING

Another function of the display system will be to allow the user to display and interactively edit visibility data. To do this the user will first make a "mapping" request which specifies that a baseline-time array is to be produced rather than a map array. This will result in an array of data in which each row of the array will correspond to a different correlator. The pixels across a row will be the visibility values as a function of time for that correlator. This array can then be displayed on the IIS with intensity-hue encoding, i.e., phase will be encoded as the displayed hue and amplitude will be encoded as the displayed intensity. A bad correlator will then show up as a horizontal pattern on the display. Interference will show up as a vertical pattern on the display. The user will then be able to edit out the bad data by pointing to it with the IIS cursor. It will be possible to flag individual correlators on a scan basis. The flagging of individual visibility records will not be supported.

3.14. REMOVING VISIBILITY DATA FROM THE SYSTEM

It will be possible to dump raw uncalibrated SORTER visibility data to tape by running the SORTER program DBUTIL. (This program currently exists. See the document written by Bob Duquet for details.) This program will also allow deleting data from SORTER's disk.

3.15. REMOVING MAP DATA FROM THE SYSTEM

Map data will be written to tape by running the FITS program on GRIDDER or DISPLAY.

3.16. ADDITIONAL MAP PROCESSING

The Pipeline system is not expected to provide all of the processing and analysis that will be needed by the users. Thus, it is expected that users will write their maps (or appropriate map subsections) onto FITS tapes and then continue their processing at their home institution or on

one of the AIPS systems at NRAO.

4. IMPLEMENTATION DETAILS OF THE SYSTEM

4.1. MODCOMP TO DEC-10 CONNECTION

Visibility data will be passed from the Modcomp system to the DEC-10 through the fixed-head disk just as is currently done. The program on the Modcomp side (which runs in BOSS) will be modified so that it does not send complete visibility data for non-calibrator sources to the DEC-10. The DEC-10 real-time FILLER will create the index, gain and visibility files on the DEC-10 disk just as is currently done.

The DEC-10 tape FILLER program will be modified so that it will not put complete visibility data for non-calibrators onto the DEC-10 disk when it reads a Modcomp archive tape.

4.2. MODCOMP TO SORTER CONNECTION

The connection between the Modcomp system and SORTER will consist of four shared access disk drives. On the Modcomp side, two of these will be connected to CORA and two will be connected to CORBIN. On the SORTER side, all four will be connected to SORTER.

The files on the shared disks will not be in standard PDP-11 FILES-11 format. Instead, blocks of data will be written and read at pre-defined physical locations on the disks. The data items on the disks will also not be in PDP-11 format. The byte order will be the Modcomp byte order and floating point numbers will be in Modcomp format. The SORTER program will do the byte swap and conversion to PDP-11 floating point format.

The data transfer system will be initialized by running the PGI program in CORA and CORBIN. This will clear two special locations at the beginning of each disk which are used for communicating control information between CORA/CORBIN and SORTER.

A program called PIG will run in CORA and CORBIN to write the visibility data out to one of the shared disks. CORA will output one group of spectral line channels and CORBIN will output the others. Each time there is a source change PIG will check the special location on the other disk to see if SORTER is done reading the data from the disk. If so, it will set the special location on the first disk to indicate that it is done writing to that disk and then begin writing to the second disk. A program called PIGEON will run on SORTER to read the data from the shared disks. At startup, PIGEON will check the special location on the first disk connected to CORA which indicates when it is done writing to the disk. When PIGEON sees that CORA is done writing, PIGEON will begin reading and processing from the first disk. When it finishes processing the data it will set the location on the disk which indicate that it is done reading. It will then process the data from CORBIN in a similar manner.

4.3. VISIBILITY DATA SORTING ON SORTER

The PIG programs will write the visibility data onto the shared disks in a partially sorted order. There will be 64 "pigeonholes," where a pigeonhole consists of a range of values in the absolute value of the "U" coordinate. The data within a pigeonhole will not be in any particular order. The PIGEON program in SORTER will further sort the data so that there will be a total of 256 pigeonholes in the SORTER disk database.

4.4. VISIBILITY DATABASE FORMAT ON SORTER

PCC

For each user that has visibility data on SORTER's disks, there will be a single visibility catalog file. (This is different from a map catalog.) The filename of this file will be the four digit user number (leading zeroes are required) and the extension will be .CAT. Each visibility catalog record will contain header information and a "pointer" to a separate file in which the actual visibility data is stored. Each visibility data file will be entirely contained on one disk, but a single user's catalog will be able to point to visibility files that are stored on any of SORTER's eight disks.

For continuum data, each catalog record and associated visibility data file will pertain to all of the data for a single source, even if it includes more than one "scan."

For spectral line data, each catalog record and associated data file will pertain to a group of eight channels for a single source. Thus, spectral line data will usually require multiple catalog records and visibility data files.

For each visibility data file, there is an associated file called a pigeonhole control area. This file stores book-keeping information that is used for accessing the 256 pigeonholes that are contained in the visibility file.

Each visibility record contains the real and imaginary values for the set of eight spectral line channels (or the appropriate number of continuum correlators) plus the corresponding U, V, W values, the time, the baseline identification and the flag information. The actual visibility values will be stored in a special 16-bit floating point format. *but not currently (now 16 b fixed point)*

A detailed description of the SORTER visibility database format can be found in a document written by Bob Duquet. A discussion of the visibility data format can be found in Computer Memo #161, "Floating Point Format for Visibilities" by Barry Clark.

4.5. MAP DATABASE FORMAT

4.6. MAP I/O

There will be a set of standard subroutines for accessing map data and its associated header information. All programs in the system will use these subroutines for accessing maps.

4.7. MAPCON - CONTROL OF OPERATIONS IN GRIDDER

MAPCON   is the name given to a set of tasks that control map processing activity on SORTER. MAPCON was originally developed on MAPPER with the following objectives:

1)   To accept map processing requests from other machines.

2)   To invoke whatever tasks are needed to fulfill the request. For a given operation the user may specify what class of software is used: SYS,   a mature set that has been tested for some time; NEW, software that has been recently changed to provide some new feature or correct some recently discovered bug; or EXP, those tasks that are radically different from SYS or NEW.

3)   To schedule tasks to make best use of CPU and disk resources, with turnaround enhancement for users who have used the system least in the recent past. Users may specify a relative priority for their requests, in order to achieve a particular execution order, or to minimise their impact on others,   if they do not need the results for a day or two.
MAPCON   provides a measure of disk space management. It always tries to clean the disk of unwanted or temporary files, and if one disk is full it will try and use another. Application tasks need not be concerned with disk space management.

4)   To prevent different tasks from using the same file or some other non-sharable resource,   such as an array processor at the same time. Different tasks communicate with NQTASK which arbitrates contentions as well as handling disk assignments.

5)   To provide error logging and recovery for tasks that fail, while attempting to maintain useful throughput without human intervention. If a task fails for a reason that MAPCON determines is correctable, the task will be reschedule with lower priority. If this is not possible the request is dispatched to a queue of bombed jobs. The status of bombed jobs may be examined in great detail, facilitating location of the problem that caused the failure.

6)   To provide a statistics gathering facility. The execution times

for each task invoked to service a request is maintained, as well as a log of significant events that occur while MAPCON is running.

7) To provide a utility for other machines to interrogate the system and change its operation. This is provided by the tasks CATLST and PROBE. CATLST permits the transfer of maps to other machines and may be used to delete maps from SORTER. PROBE may be used to examine the queues of requests. It can provide detailed information on the parameters used to define a request and on the causes of an error. PROBE can also be used to delete an unexecuted request or to abort an executing one. It may also be used to gracefully terminate a clean operation.

8) To provide management utilities for manipulating priorities, debugging, and keeping the disks clear of unwanted files. Requests that have failed can often be manually restarted after the cause of the problem has been removed. The management task is called CONCON.

MAPCON is a flexible framework and is subject to occasional changes as new need become apparent.

4.7.1. THE REQUEST QUEUE SYSTEM

MAPCON maintains a multiple queue system for scheduling requests. There are (at present) four distinct queues. The first queue is for requests that need, but have not completed database transfer from another machine. The second is for requests that are in the process of executing or are ready to execute. The third is for requests that have successfully been completed. Requests are automatically aged off this queue at a rate that depends upon the activity of the particular user. The last queue is for jobs that have failed for reasons that MAPCON is unable to rectify.

4.7.2. THE MESSAGE AND LOG SYSTEM

Tasks within the MAPCON system can make use of its message and error logging system. When application tasks complete, either successfully, or after detecting an error in input data, or even catastrophically, such as when aborted by the console terminal, status is returned to MAPCON so that appropriate action may be taken. Tasks can also send a number of different types of messages, for example clean requests return data from time to time that may be interrogated by the DEC 10 via the probe utility. Other messages may be useful in debugging application tasks. By providing this service MAPCON removes the need for application tasks to perform FORTRAN formatted output, with a substantial saving in code size. Messages are accumulated by MAPCON with timestamps and are stored in a large circular file that may be interrogated by CONCON.

4.7.3. THE MANAGEMENT SYSTEM

CONCON may be used to interrogate the MAPCON system at a very low level. It can be used to examine and modify both the queue system and the individual request files. In addition, CONCON can turn the individual request streams on and off and initiate shutdowns and startups of the MAPCON system. If the queue system, which is a multiply linked disk resident list structure, becomes tangled as the result of a hardware error, or a system boot at an inappropriate time, MAPCON will normally rectify the problem when it restarts. If this fails, CONCON can strip the queue system down to its individual elements and then relink them, although the original ordering within queues will be lost.

4.8. MAP MAKING

The user will submit a mapping request from the DEC-10 using the GRIDDER program. The request will then be enqueued by MAPCON. GRIDDER will obtain the appropriate gain file from the DEC-10(MAPCON initiates the transfer of the gain file prior to the time when the array processors become available to make the map).

After the gain file is transfered into GRIDDER's disk and the array processors are available, MAPCON will initiate the mapping program in GRIDDER. This program will request that the visibility data for the source to be mapped be sent to GRIDDER from SORTER. As the data is received from SORTER it will be sent to AP-1, which will perform data selection, gain corrections, gridding and convolution. Note that the visibility data will not get stored on GRIDDER's disks. When AP-1 finishes the gridding of a line in the UV plane it will send the line to AP-2, which will perform the FFT on the line and write the result out to the transpose memory. When enough data has been written out to the transpose memory, AP-3 will begin reading columns from the transpose memory and doing the column FFT's. As output map lines become available, they will be transferred to GRIDDER and written out to the disk.

The above description applies to the case where the entire gain table can be loaded into the memory of AP-1. If the gain table is too big, it will be necessary to transfer it from GRIDDER's disk to AP-1 multiple times. The above description also applies to the case where the entire map will fit into the transpose memory, i.e., the map is 2048x2048 or smaller. For larger maps, multiple passes through the system will be required. This will involve transferring the visibility data from SORTER to GRIDDER to AP-1 multiple times.

Additional details on the mapping system can be found in the following Computer Memos by Wim Brouw:

#157 Remarks on the GRIDDER Development. February 1981.

157 Remarks?

#158 GRIDDER System. Part 1 - Array Processor
      Programs. September 15, 1981.
#159 GRIDDER System. Part 2 - PDP 11/44 Programs.
      September 15, 1981.
#160 GRIDDER System. Part 3 - Changes made to Floating
      Point Systems Software. September 15, 1981.

4.9. MAKING BASELINE-TIME ARRAYS

Producing baseline-time arrays of visibility data will be similar
to producing maps. The data will be gridded onto a baseline-time plane
instead of a UV plane. This gridded array will then be output without
the need for any FFT's. The B-T files can be displayed on the IMPS
system. The program BASPLT on the DEC-10 is used to initiate the
generation of baseline-time files.

4.10. MAP CLEANING

The B. Clark CLEAN algorithm is implemented and will be the standard
CLEAN available on SORTER.

→ currently no xPose mem.

4.11. SELF CALIBRATION

The self-calibration algorithm currently used in the AIPS system
will be implemented.

4.12. VISIBILITY DATA EDITING

Visibility data will be flagged in two ways. The FLAGER program on
the DEC10 can be used to flag antennas by scan. This flagging
information is stored in the index record which is transfered to SORTER
as part of the map requests made from the DEC10. Secondly, it will be
possible to flag data by correlator from the DISPLAY system. These flags
will be shipped to SORTER as part of the mapping requests.

4.13. THE DISPLAY SYSTEM

4.13.1. IIS SOFTWARE

We will make use of the IIS interface driver for RSX-11M which is
supplied with the hardware. We will also make use of the set of basic
Fortran callable interface routines which is supplied with the hardware.
And, we will make use of the set of "Primitives" routines which was
purchased as a separate option. These routines provide higher level
applications oriented functions. Some of these routines will need to be
modified since we will be using our data tablet rather than the IIS
trackball.

4.13.2. VT-11 SOFTWARE

We will make use of DEC's subroutine package for handling the VT-11 display. Although this is not a very good package, there seems to be no practical alternative to using it. We have ordered the latest version of this package. Perhaps this will provide an improvement over the version that we currently have in terms of functionality or implementation efficiency.

4.13.3. VERSATEC SOFTWARE

We will make use of Versatec's subroutine package for handling the Versatec electrostatic printer/plotter. We have ordered the latest version of this package. It should be functionally identical to the version that we currently have, but it will hopefully include a fix for the bug that causes the output task to die occasionally.

4.13.4. DATA TABLET SOFTWARE

We will make use of the data tablet driver that Al Braun implemented years ago but which was never incorporated into IMPS. This will provide improved response to the data tablet compared with that of IMPS.

4.13.5. IMAGE STORAGE SYSTEM

The image storage system is being designed and built by Ray Escoffier in Charlottesville. The image data will be stored digitally on two Winchester disks. There will be a programmable microcomputer (a Motorola 6803) to control the transfer of data between the disks and the PDP-11/44 host computer or the IIS. The image storage system will be connected to the IIS in such a way that data can be transferred in either direction between the PDP-11 and the image storage system, between the PDP-11 and the IIS, and between the image storage system and the IIS. The program in the PDP-11 will be able to send commands to the image storage system to do such things as load a given set of images into the IIS as a movie presentation with a given frame rate. The microcomputer will then control the loading without the need of constant program execution in the PDP-11. The system will be able to store 600 512x512 images and play them back at a frame rate of up to seven frames per second. Since the system will be controlled by the programmable microcomputer, we will be able add new functions to the system as they are needed.

The image storage system will initially be used in one of the AIPS systems. At some later time, a copy of the hardware will be built for use with the Pipeline display system.

4.13.6. CONTROL PANEL

The control panel is currently being designed by Don Wells in Charlottesville. It will be connected to the same microcomputer that is incorporated into the image storage system. This microcomputer will monitor the control panel and then send appropriate commands or data to the IIS when the user changes something on the panel. This organization will thus provide smooth and rapid response to the user without putting a burden on the host computer. By changing the microcomputer program we will be able to modify the functions of the devices on the control panel. Also, the control panel itself will consist of modules which can be replaced or modified. Thus, we will be able to modify the control panel as new ideas for its use are developed.

In addition to controlling the IIS display functions, the control panel will also control the functions of the image storage system.

The control panel will initially be used in one of the AIPS systems. At some later time, a copy of the hardware will be built for use with the Pipeline display system.

5. SOME QUESTIONS TO BE ANSWERED

5.1. MAPPING DURING OBSERVING.

Will the user be able to make maps while his observation is still in progress? ANS: Database organization should not preclude this.

5.2. SORTER/GRIDDER PROCESSING FUNCTIONS.

What operations will SORTER/GRIDDER be able to perform in addition to making maps? Cleaning? Self calibration? Source subtraction? Transforming a map or map subsection back to the UV plane?

5.3. SHARED DISK ACCESS - USER'S VIEW.

If all the "public" disks on a given PDP-11 are seen as a single storage unit by the user, how will the user of DISPLAY view the shared access to GRIDDER's disks? ANS: These disks should have the same name on both systems. Probably simplest if catalog contains the unit name.

5.4. SHARED DISK ACCESS - IMPLEMENTAION.

How will a program running in DISPLAY specify that a file is to be read from GRIDDER's disk? How will DISPLAY's reading of GRIDDER's disk be coordinated with GRIDDER's reading and writing? ANS: Catalog could contain a busy bit. However, it is not a problem if not coordinated.

5.5. GRIDDER DISK USAGE.

The current plan calls for GRIDDER to have four disks, only two of which can be read by DISPLAY. If this is the case, what will the two unshared GRIDDER disks be used for? Should we give DISPLAY read-only access to all of GRIDDER's disks?

5.6. PRIVATE DISKS.

Should the system allow "private" disks in a manner similar to AIPS? If so, how will this be done? E.g., will the private disks be connected to SORTER for storing visibility data? Will they be connected to GRIDDER? Will they be connected to DISPLAY?

5.7. MAP DATABASE FORMAT.

What will the physical map data base format on disk look like? Will it look similar to the current PDP-11 map data base format, or will it be the same as the AIPS format? Will pixels be stored as 16-bit scaled integers or 32-bit floating point? ANS: Modular I/O makes irrelevant.

5.8. DISK DRIVE FAILURE.

Will the system be able to run if one of the disk drives is down? Will a user have a single map catalog file which points to map files on all of the disks or a separate map catalog file on each disk? If a disk goes down and a user creates a new map with the same name as a map on the down disk, what happens when the disk comes back up?

5.9. DEFINITION OF MAP HEADER ITEMS.

Will some of the current PDP-11 map header items be redefined, e.g.,

items dealing with keeping track of map coordinates?

5.10. STORAGE OF UNGRIDDED VISIBILITY DATA.
Will ungridded visibility data be stored in the GRIDDER database? Will ungridded visibility data be stored in the DISPLAY database? If so, will these be listed in the map catalogs? How will these data files get into GRIDDER or DISPLAY?

5.11. STORAGE OF BASELINE-TIME VISIBILITY DATA.
Will baseline-time visibility data arrays be listed in map catalogs? What new header items will be needed for keeping track of these arrays?

5.12. SORTING OF BASELINES IN BASELINE-TIME VISIBILITY ARRAYS.
It will probably be useful for baseline-time arrays to be available with several different ordering of baselines. Will these different orderings be produced as separate arrays by GRIDDER or will a "standard" ordering be sorted by DISPLAY?

5.13. EXPORT OF VISIBILITY DATA.
Will it be possible to write calibrated visibility data to tape in EXPORT format? Will some new format be used? Will it be UV-FITS?

5.14. STORAGE OF ANTENNAS FILES.
Will antennas files be stored in the DEC-10 as is currently done? Will they be stored in SORTER? In GRIDDER? In DISPLAY?

5.15. GENERALIZED LISTS OF COORDINATES.
Should the database design include provision for using a generalized list of coordinates for uniformly storing such things as clean components, fitted sources, star positions, etc?

5.16. MAP SLICES.
Should the database design include the capability for storing one dimensional slice data arrays that are associated with a map file?

5.17. MAP HISTOGRAMS.
Should the database design include the capability for storing a histogram of the distribution of pixel values in a map?

5.18. TRANSPOSED SPECTRAL LINE DATA CUBES.
Should the display system be able to handle transposed spectral line cubes that are read in from a FITS tape? Should the Pipeline be able to transpose a spectral line cube? Should the Pipeline be able to make a transposed spectral line cube?

5.19. ROTATED MAPS.
Will the system be capable of making rotated maps?

5.20. TRANSFER OF SAVED MAPPING PARAMETERS.
If the user makes a mapping request from the DEC-10, looks at the resulting map on the display system, and then wants to submit another

mapping request from the display system, it would be desirable for him to be able to easily submit a request which has just a single parameter changed. How will the previous parameter values from the DEC-10 mapping request be communicated to the display system? Should this be automatic, or should we require the user to enter a specific command to get parameters from the DEC-10? What about going back to DEC-10 and getting the last parameters that were used in a request from the display system?

5.21. USER LOGS OF PIPELINE OPERATIONS.
If the user makes a mapping request from the display system, should the log of that operation be put into the user's "disk area" on the display system or on the DEC-10? Should a copy be put into both places? Or should the user at one location be able to look at the log stored in the other location that applies to requests made from that other location?

5.22. LOCATION OF MAPPING REQUESTS.
Should the user be able to initiate a mapping request by running a program directly on GRIDDER that uses a DEC-10 standard commands type of interface? Should a user be able to run the same program on DISPLAY to initiate a mapping request without using the graphics-oriented interface?

5.23. MAPPING PRIORITY FOR DISPLAY SYSTEM USER.
Should the system be capable of suspending a mapping (or cleaning, etc.) operation that was requested from the DEC-10 in order to immediately do an operation that was requested from the display system?

5.24. DEC-10 LISTING/DELETING OF VISIBILITY DATA.
Will the user be able to get summary listings and detailed listings of SORTER (GRIDDER? DISPLAY?) visibility databases by running a DEC-10 program? Will he be able to run a DEC-10 program to delete this data?

5.25. DISPLAY SYSTEM LISTING/DELETING OF VISIBILITY DATA.
Will the display system user be able to list SORTER visibility data? Will he be able to run a DISPLAY program to delete this data?

5.26. IDENTIFICATION OF SORTER VISIBILITY DATABASE.
Should the user be required to identify his SORTER visibility database by specifying a four digit user number with leading zeroes? (Leading zeroes are not required in other parts of the system.)

5.27. VISIBILITY DATA EDITING.
If the user does visibility data editing on the display system, how will the flag information become associated the the visibility data? Will scan level flags be incorporated into the index records in the DEC-10? Will visibility record flagging be supported? Will visibility record flags be incorporated into the SORTER database? Or, will flags from DEC-10 index records, flags from the display system, and flags from the SORTER database just be collected together and OR'd at map making time?

5.28. ANTSOL SOLUTION DISPLAY.

Will the display system need to be able to display the ANTSOL solutions? If so, how will it get access to this data?

5.29. MONITOR DATA DISPLAY.

Will the display system need to be able to display monitor data? If so, how will it get access to this data?

5.30. CONCURRENT DISPLAY SYSTEM OPERATIONS.

Should the display system allow concurrent operations such as making a Versatec contour plot at the same time the user is looking at maps on the IIS or is one thing at a time like in IMPS sufficient?

5.31. BATCH MODE DISPLAY SYSTEM OPERATIONS.

Should the display system allow concurrent execution of a queue of batch mode operations while the user is doing interactive things? If so, how flexible should this capability be? Should it be possible for a user who is using the display devices to make entries in the queue at the same time that another user is making entries in the queue by running a program on DISPLAY which uses standard commands? Should a log of batch queue processing be produced?

5.32. INITIATION OF BATCH DISPLAY FUNCTIONS FROM THE DEC-10.

Should the user be able to initiate batch mode display functions such as making Versatec contour plots or making Dicomed images from the DEC-10? If so, how should this be done? Should the DEC-10 user also be able to examine and modify the display system batch queue? Should a log of display system batch processing be put into the user's disk area on the DEC-10?

5.33. PROBE.

Should the display system user be able to PROBE the status of GRIDDER processing? Should the DEC-10 user be able to PROBE the status of display system batch processing? ANS: Yes.

5.34. AUTOMATIC HARD COPY OF MAPS.

Should we automatically produce a hard copy output of each map that is produced by the system? If so, what kind, e.g., Versatec gray-scale, Versatec contour, Dicomed gray-scale, Dicomed color, Honeywell gray-scale? Should the user be able to specify that he wants a hard copy produced when he makes the mapping request?

5.35. RUNNING DEC-10 PROGRAMS FROM THE DISPLAY SYSTEM.

Should the display system provide the capability to logically connect the user to the DEC-10 so that the user can run DEC-10 programs without going and finding a DEC-10 terminal? ANS: DEC-10 terminals are everywhere. Could you run DISPLAY from a DEC-10 terminal.

5.36. FITS.

Where will the FITS tape writing program run? Just in GRIDDER? Just in DISPLAY? Both? If running in DISPLAY, will it get data from

DISPLAY's disk(s)? From the disk(s) shared with GRIDDER? Where will the user control the FITS program from? E.g., will he run a DEC-10 program which uses DEC-10 standard commands or will he run a program directly on the computer which runs the tape writing program? ANS: Talking to the computer which runs the program has the advantage that it is easy to reserve the tape drive.

5.37. DICOMED.

How will the Dicomed be handled? E.g., like things are currently done in IMPS? What about getting Dicomed output from AIPS on a VAX? Should the Dicomed be run in a batch mode with image data obtained from tape and/or disk files? Should the user of the display system be able to request a Dicomed copy of the current IIS image and have the image saved somewhere for spooling to the Dicomed at a later time?

5.38. MULTIPLE DISPLAY SYSTEMS.

Should the system architecture allow the future addition of one or more additional display stations? If so, how should this be done?

5.39. DECNET CONNECTIONS.

Would it be better for the DECNET connection from DISPLAY to go directly to GRIDDER rather than to MAPPER?

5.40. ALTERNATE HARDWARE CONFIGURATIONS.

Alternate hardware configurations have been suggested to reduce the possibility of I/O bottle-necks in the system. For example, there could be a direct connection from SORTER to AP-1 so that visibility data would not have to pass through GRIDDER memory. A second suggestion is to have the map data from AP-3 go directly to disk without going through ANS: We need to make the pipeline work before bringing another improvement along. GRIDDER memory. A third suggestion is to have map data from AP-3 go to DISPLAY rather than to GRIDDER. A fourth suggestion is to have map data from AP-3 go to an additional computer which would function as a database handler, e.g., pass data to DISPLAY to be looked at and write map data to FITS tapes. Should we seriously consider any of these proposals?

5.41. ARRAY PROCESSOR MEMORY.

Should we increase the memory in the array processors and eliminate the transpose memory?

5.42. PDP-11 AIPS.

If AIPS is to be used on the PDP-11's, what exactly will it be used for? Just initiating a mapping request from DISPLAY? Reading/writing FITS tapes? Initiating Versatec contour plotting? Will we also use AIPS on the DEC-10 for initiating mapping requests from DEC-10 terminals?

5.43. MAP TRANSFER TO VAX'S.

Should it be possible to transfer maps from GRIDDER or DISPLAY to one of the VAX's through the DECNET connections? If so, how should this be initiated? Should it be possible to initiate a mapping request from the DEC-10 and have the map automatically sent to a VAX?

5.44. VAX INITIATION OF PIPELINE FUNCTIONS.

Should it be possible to initiate Pipeline system functions from one of the VAX/AIPS systems?

5.45. FUTURE SYSTEM CONVERSION.

How much effort should be expended in implementing pieces of the system in such a manner that will facilitate converting the system to run on different hardware (e.g., VAX) in the future?
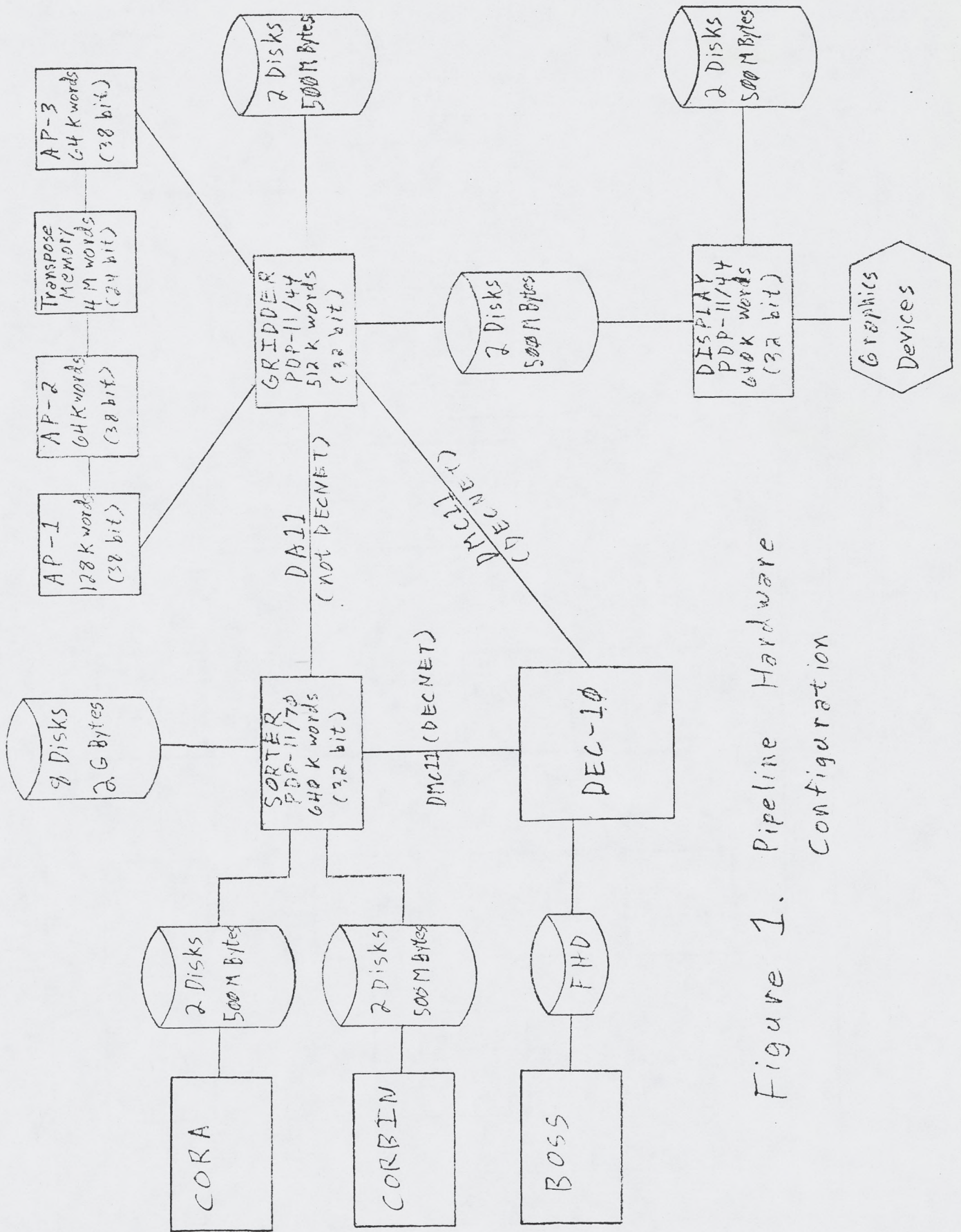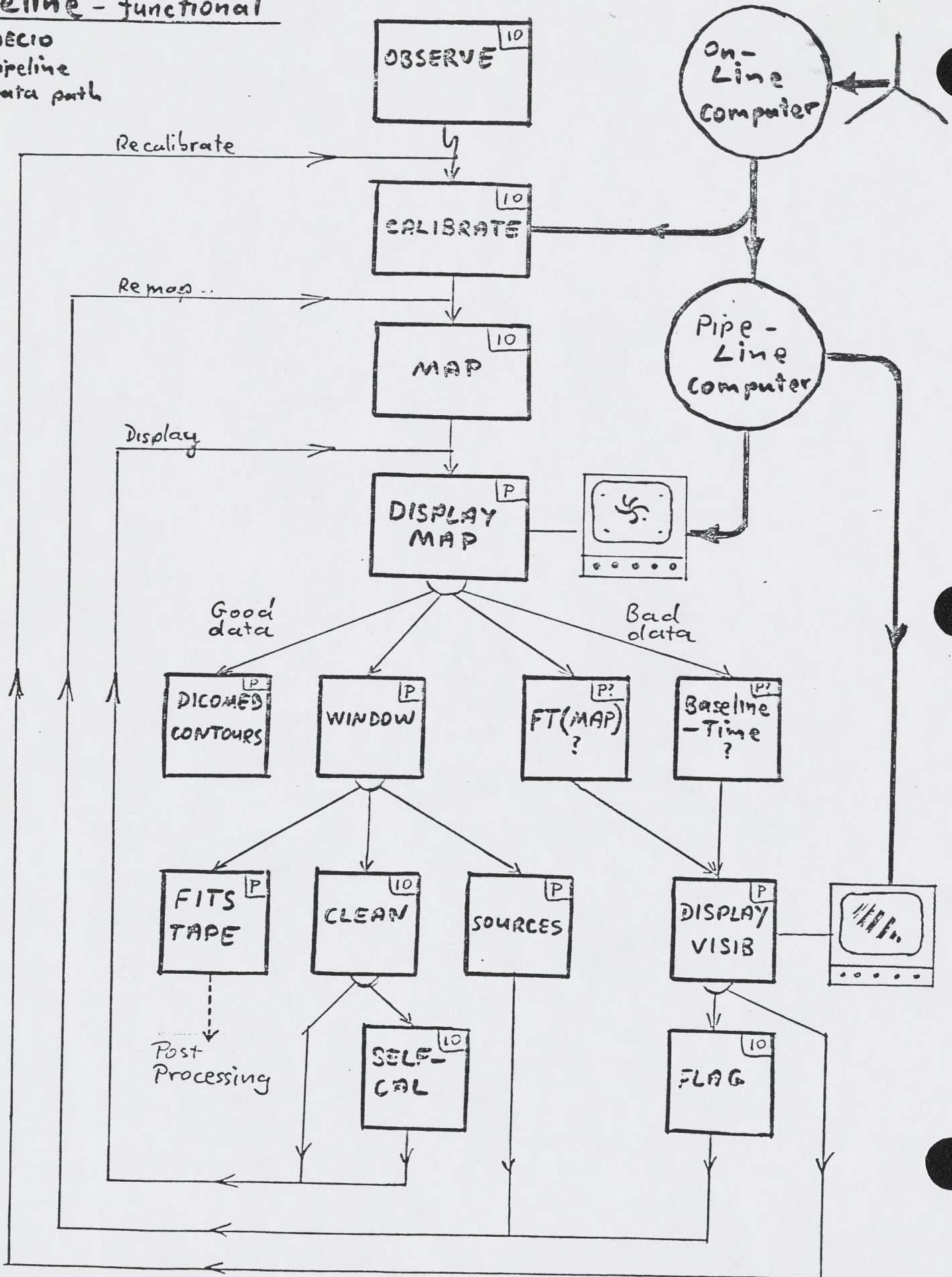
Figure 1. Pipeline Hardware Configuration

Figure 2.

Pipeline - functional

IO DECIO
P   Pipeline
⬅   Data path

Suggested Answers to Jim Torson's Pipeline Questions
(refer to Section 5 of SGP memo #26)

Don Retallack 2/10/83

5. 1 Observer should have _limited_ map making capability for quick
look and ability to submit a job to completely map sources which
are finished.

5. 2 Source subtraction before map making will become very important
for 327 MHz; I expect users may want to subtract several hundred
sources if experience with the WSRT 327 MHz and 608 MHz systems
can be used as a guide.

5. 6 This would create large problems since the pipeline, unlike AIPS
on a VAX, will not necessarily be dedicated to one user. It
also doubles our costs and effort to buy and administer
removable packs.

5. 8 An overall plan for hardware failure should be devised. It
should include scenarios for replacing disk drives quickly in
case of failure.

5.11 Either in map catalog or database catalog, I don't think it
make much difference provided it can be made self-identifying.

5.12 Produce one "standard" ordering and let DISPLAY sort for
different orders.

5.13 Transport tape will contain calibrated data only and will be in
UVFITS format.

5.16, 5.17 No. Call these items maps and put them into a user's map
catalog as one dimensional maps.

5.18, 5.19 DISPLAY should be able to handle rotated and transposed
map cubes and also make them.

5.20 Re-submit mapping request from DEC-10. Allow recall of
previous mapping parameters with "get commands" as now existing.
DISPLAY should not be able to submit map requests. Map should
have sufficient information in its header to define all mapping
parameters used to make it; thus DISPLAY will not need to query
the DEC-10.

5.21 Irrelevant. See above.

5.22 GRIDER maybe, DISPLAY no.

5.23 No.

5.24 Yes; yes.

5.25   No; no.

5.26   No. Allow up to 4 digit user no. We <u>can</u> have 1 and 2 digit user
       numbers too! Omit requirement for leading zeroes.

5.29   Maybe later.

5.30   Multi-tasking is available - let's use it.

5.34   No! WSRT reduction systems do this and the maps are very seldom
       used. In general, one cannot choose a set of display parameters
       appropriate for every map. Let the user do it the way he wants.

5.35   No. Let user log on to DEC-10 terminal.

5.36   Both. FITS should be runnable on DISPLAY to get DISPLAY maps
       only and on GRIDER to get GRIDER maps only.

5.37   Dicomed should be run from DISPLAY only. Other machines can
       send images via Decnet or (preferably) tape. Have a sign up
       sheet and let computer operator make the Dicomed images but
       allow users to do it themselves if they want.

5.38   Yes.

5.42   No AIPS on PDP-11's.

5.43   Possible, yes. Routine, no! No automatic or easy way for
       users to tie up DECnet with map transfers.

5.44   No.

DR/bmg