"PIPELINE" SYSTEM REVIEW

Version 5 - May 27th, 1983

1. INTRODUCTION

The purposes of this document are:

1- To describe the Pipeline system as it is currently planned and, by so doing, to elicit suggestions and criticisms.

2- To describe the current status of the system, the nature of work in progress and the "game plan" for completion.

3- To introduce potential users to system components that are currently useable.

4- To list known problem areas and to report solutions when they have been achieved.

5- To draw attention to design or implimentation issues that have not yet been resolved.

To meet these objectives this review has been divided into the following parts:

1- Introduction.

2- The Pipeline System Plan

3- Planned Use of the Pipeline: Step-by-step

4- Current Status of the System.

5- A User's Guide through the Current System

6- Plans for the Display Subsystem

7- Miscellaneous System Implimentation Details

8- A List of Unresolved Issues

9- Appendix: Changes from the Previous Version

2. THE PIPELINE SYSTEM PLAN

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 the pipeline 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.

2.2. QUANTITATIVE OBJECTIVES

Please note that these are OBJECTIVES, not currently achieved performance figures. For a description of what is available at this time turn to section 4 of this report.

The quantitative goal that has been set for the pipeline is expressed in terms of the number of input data values that should be accepted by the system in a continuous stream while it is concurrently producing, at a specified rate, maps of a given size.
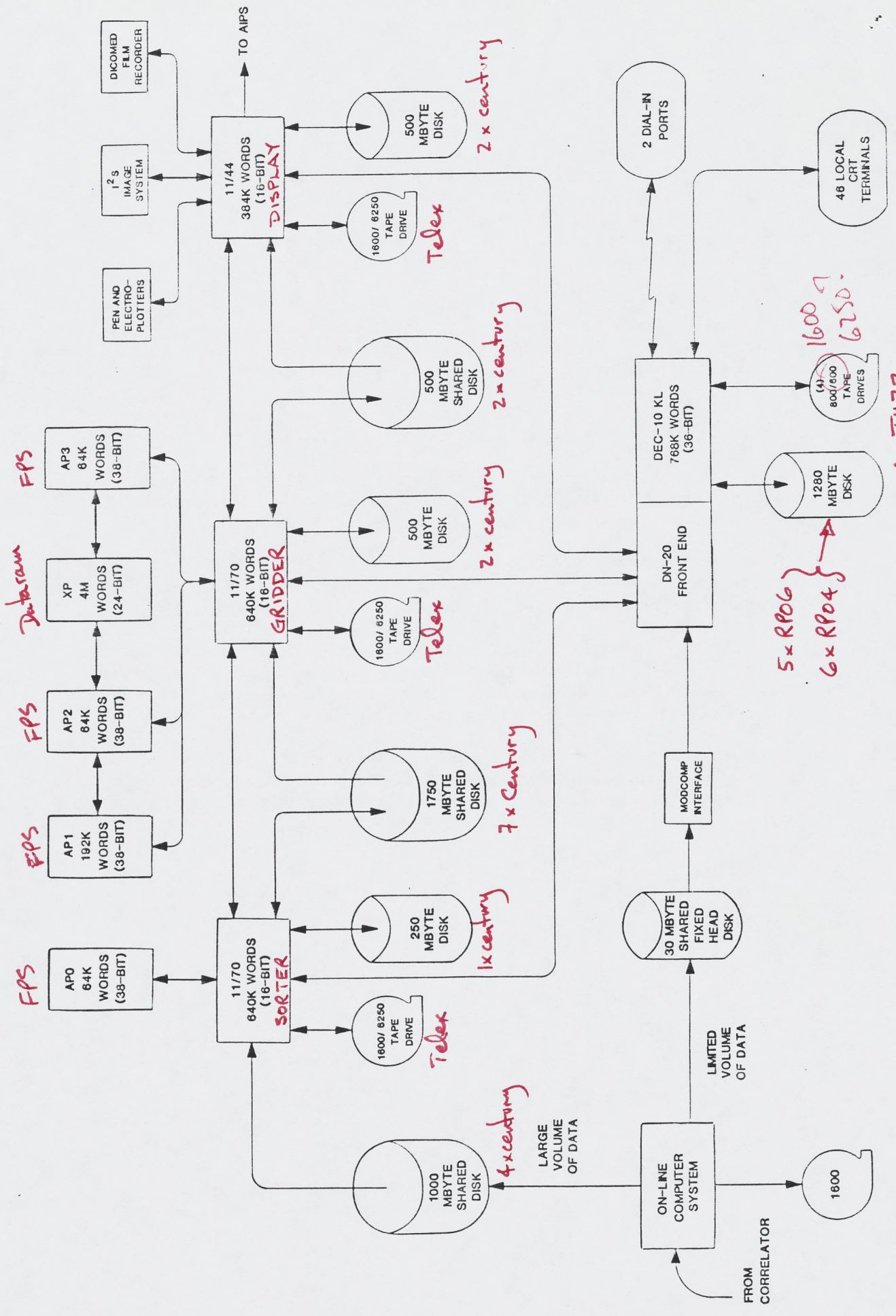
At this time the goal is to be able to accept 6000 complex numbers per second while producing one 2048 by 2048 map per minute.

By way of comparison, 256-channel data from all 351 baselines, averaged over 10 seconds, corresponds to a data rate of 8985.6 complex values per second.

Note that the stated objective does not provide for any other simultaneous activity such as the transfer of data or maps between tape and disc or between the pipeline and any other machine.

The Pipeline system will be capable of producing continuum maps that are up to 4096 x 4096. The user will be able to copy maps of this size or put them onto FITS tapes but most other functions in the system will be limited to processing a 1024x1024 subsection.

Work on 8192x8192 maps can be supported only if it is done rarely. The typical map size for which the system is designed is 1024x1024. The system will be capable of producing rectangular maps.

FIGURE

Diagram labels:

DICOMED FILM RECORDER

TO AIPS

I²S IMAGE SYSTEM

PEN AND ELECTRO-PLOTTERS

11/44 384K WORDS (16-BIT) — DISPLAY

500 MBYTE DISK — 2 x century

1600/6250 TAPE DRIVE — Telex

500 MBYTE SHARED DISK — 2 x century

2 DIAL-IN PORTS

48 LOCAL CRT TERMINALS

AP3 64K WORDS (38-BIT) — FPS

XP 4M WORDS (24-BIT) — Datasam

11/70 640K WORDS (16-BIT) — GRIDDER

500 MBYTE DISK — 2 x century

1600/6250 TAPE DRIVE — Telex

500 MBYTE SHARED DISK — 2 x century

DEC-10 KL 768K WORDS (36-BIT)

(4) 800/500 TAPE DRIVES — 1600 (625D.)

1280 MBYTE DISK

4 × TU77 (2 × TU78)

5 × RP06
6 × RP04

AP2 64K WORDS (38-BIT) — FPS

DN-20 FRONT END

AP1 192K WORDS (38-BIT) — FPS

1750 MBYTE SHARED DISK — 7 x Century

MODCOMP INTERFACE

AP0 64K WORDS (38-BIT) — FPS

11/70 640K WORDS (16-BIT) — SORTER

250 MBYTE DISK — 1x century

1600/6250 TAPE DRIVE — Telex

30 MBYTE SHARED FIXED HEAD DISK

LIMITED VOLUME OF DATA

1000 MBYTE SHARED DISK — 4 x century

LARGE VOLUME OF DATA

ON-LINE COMPUTER SYSTEM

1800

FROM CORRELATOR

2.3. HARDWARE CONFIGURATION

The "pipeline" is a system rather than a piece of hardware. Components of the pipeline software reside on 6 different computers. Figure 1 is a diagram of the Pipeline hardware configuration. The computers involved in the pipeline system are:

1- A Modcomp which serves merely as an interface to the on-line (data-acquisition) system. (The on-line Modcomps also do some preliminary sorting of spectral line data for the pipeline.)

2- A DEC10/KL used for calibration and editing. Control of pipeline operations (scheduling) is exercised by the user at terminals connected to this machine.

3- A PDP11/70 used for sorting the visibility data. This machine is also used for miscellaneous "housekeeping" functions.

4- A PDP11/70 which acts as a host machine for array processors and transpose memory which do the actual gridding and mapping.

5- A PDP11/44 which supports various display devices. Links from this machine to the DEC10/KL let the user participate in an iterative process wherein data is edited, mapped, evaluated, then re-edited for another round of processing.

6- A PDP11/44 (not shown in Figure 1) which is intended for program development and as a DECNET gateway to the outside world

The name "Pipeline" is sometimes used (misleadingly) to refer to just the first three PDP-11's. That practice will NOT be followed in this document. Each of the four PDP-11's is identified by a name that reflects the specialized task it performs within the system. Those names, SORTER, GRIDDER, DISPLAY and WORKER will be used in this document.

The amount of memory on each machine is as follows:

SORTER	PDP-11	640 K 16-bit words	(1.28 Mbytes)
	AP	64 K 38-bit words	
GRIDER	PDP-11	640 K 16-bit words	(1.28 Mbytes)
	AP-1	192 K 38-bit words	
	AP-2	64 K 38-bit words	
	XPOSE MEM	4,000 K 24-bit words	
DISPLAY	PDP-11	384 K 16-bit words	(0.76 Mbytes)
WORKER	PDP-11	704 K 16-bit words	(1.41 Mbytes)
DEC-10		768 K 36-bit words	

The control information that is required to make these individual machines act as one logical unit flows through the system via a set of DECNET links. (See Figure 2.) The DEC10 is linked through its DN20 "front end" with each of the PDP11's. There are also 3 direct DECNET links between GRIDDER and SORTER, DISPLAY and WORKER. Links to the outside world go through WORKER. All of these links operate at 56 kilobaud. (NOTE: It might appear that there are redundant links to and through the DEC10 but the TOPS10 operating system supports only an outdated version of DECNET that does not provide any "routing" facility.)

The connection between the online Modcomp system and SORTER consists of four shared-access disks. One Modcomp processor is dedicated to communication with a DN20 which is the DEC10 "front end" machine. That Modcomp is buffered from the on-line Modcomp machines by a shared-access fixed-head disk. The buffer capacity is several hours worth of data if (as planned) only calibration data is transferred through that route to the DEC10. The bulk of the data moves from the online Modcomp system to the SORTER machine via 4 250-megabyte shared-access disks.

Data flow from SORTER to GRIDDER and from GRIDDER to DISPLAY also uses shared-access disks.

The pipeline system contains a total of 17 disk drives exclusive of the fixed-head disk in the Modcomp system and exclusive of the disk space used on the DEC10. All of the drives are essentially identical. Each has a capacity of approximately 250 megabytes. Four of the disk drives are used for the "mail box" link between the on-line Modcomp system and SORTER. SORTER places the data on seven drives which can be read (but not written) by GRIDDER. GRIDDER is equipped with two disk drives which can be read (but not written) by DISPLAY. The remaining drives are accessible by only one machine each. There is one such drive on SORTER, on GRIDDER, on DISPLAY and on WORKER.

Each of the four PDP-11's has its own 1600/6250 bpi tape drive. MODCOMP tapes will probably not handle archiving at the maximum designed rate of 6000 line-channels/second. No decision has been made on alternative equipment.

Four FPS-120B array processors are used in the pipeline. One is attached to SORTER; the other three are available to GRIDDER. Each of the AP's on GRIDDER contain IOP devices whereby it communicates directly with the transpose memory and with the other APs without host intervention.

A detailed list of the graphics devices intended for the pipeline will be presented later in a section devoted to the display subsystem.
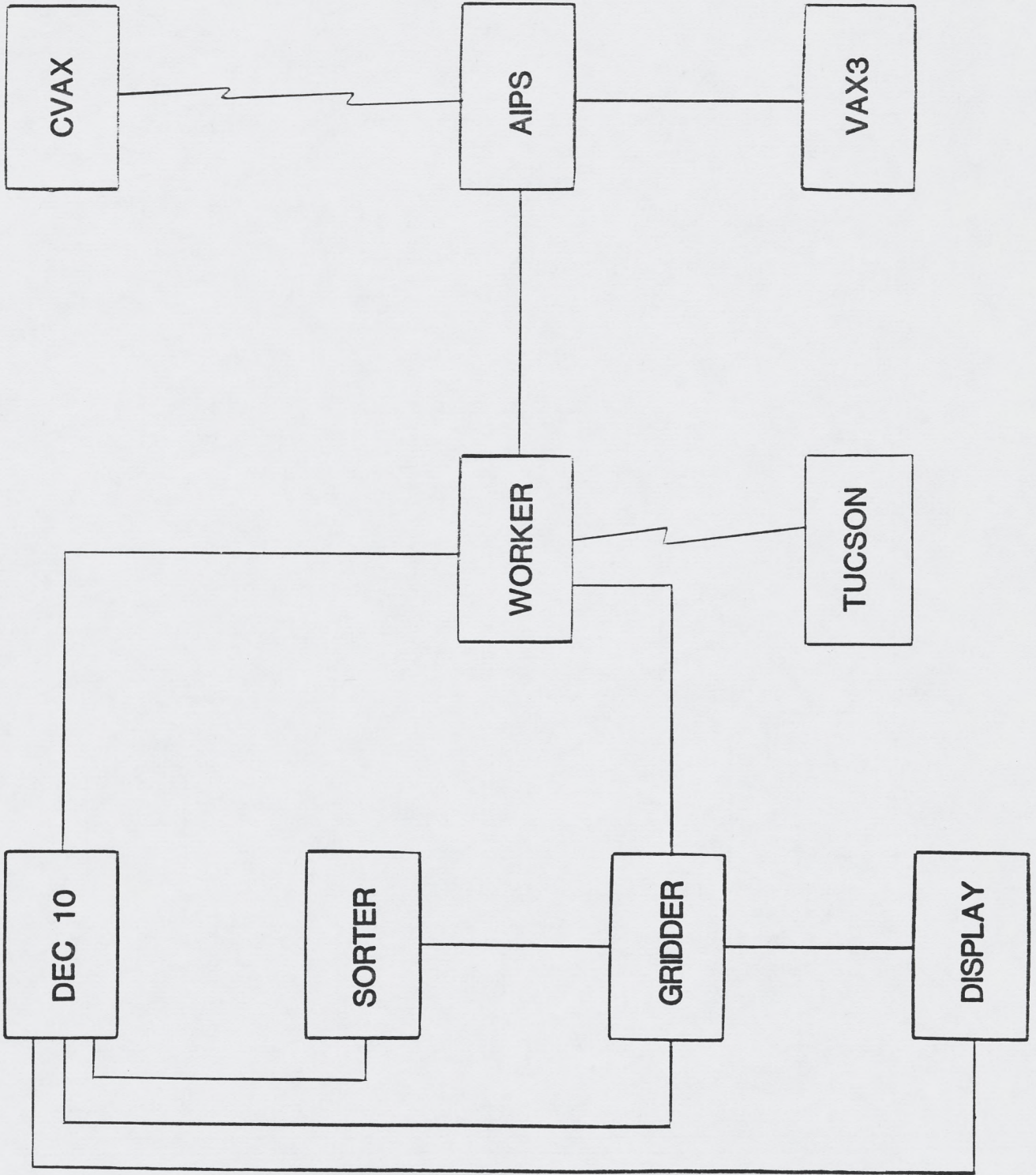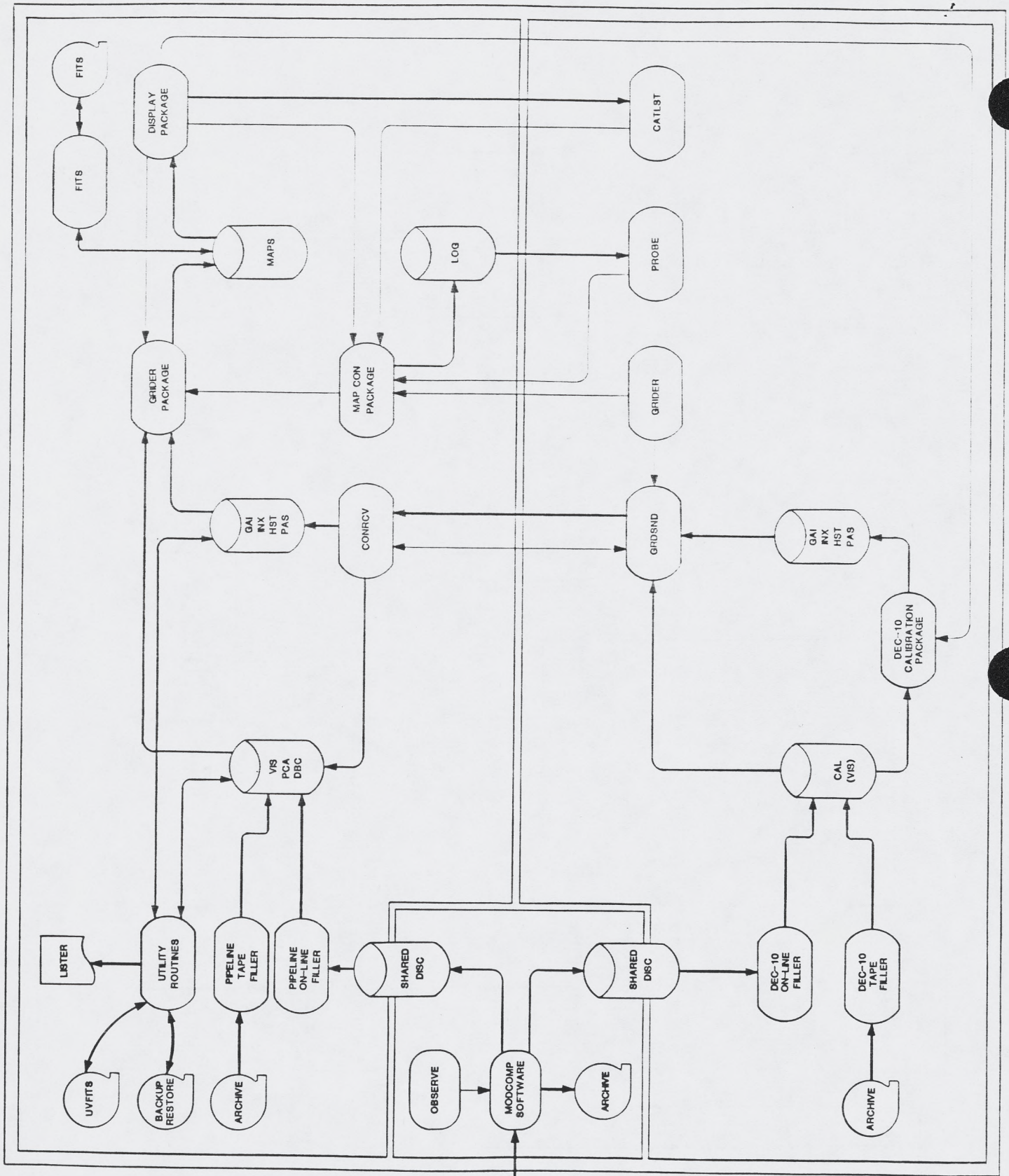
FIGURE 2

FIGURE 3

2.4. OVERVIEW OF DATA FLOW THROUGH THE SYSTEM

The following paragraphs may be more meaningful if the reader refers to Figure 3 which is a schematic representation of data flow through the pipeline. The lighter lines in Figure 3 show the interaction (flow of control) between various pipeline programs.

During observing, all visibility data will be transfered from the online Modcomp system (where it will have been partially ordered) to the pipeline's SORTER machine where further ordering will occur before the data is stored on disks. Visibility data for calibrator observations only will be sent to the DEC10 (as well as to SORTER ?). Some descriptive data on all the other sources that were observed will also be sent to the DEC10.

At the same time that it is transmitting data directly to SORTER and to the DEC10, the Modcomp system will write a copy of all visibility data (for sources and calibrators) on an Archive tape. The tape can be read by programs on the DEC-10 and on the pipeline machines as an alternate (asynchonous) data input source.

Calibration, correction and flagging will be done in the DEC-10 under the observer's direction and initiative (this is NOT an automatic process). The quality of the visibility data stored on SORTER's disks (the data for sources) can be ascertained by displaying various pseudo-maps (baseline vs time plots for example) that can be created in GRIDDER for presentation on DISPLAY. Flagging information can be sent from DISPLAY back to the DEC-10.

The results of calibration and editing are recorded in a set of gain, index and history files which will reside initially on the DEC-10 disks.

When a map is to be made, the observer must send a request to the pipeline scheduler (MAPCON). A DEC-10 program (called GRIDER) is available for this purpose.

The pipeline system will respond to a mapping request by copying the gain, index and history files from the DEC10 to the GRIDDER disks. Actual mapping can begin as soon as those files have been copied but, if the pipeline is busy answering earlier requests, it might not begin until much later. Some requests may tie up the mapping facility for hours on end.

The observer can monitor the progress of mapping requests via a DEC-10 program (called PROBE) that reads the pipeline's log file.

When the pipeline is ready to process a request, the visibility data for sources will be obtained from the disks shared by SORTER and GRIDDER. The appropriate selection of data, like almost all of the gridding and mapping computations, will take place in the array processors that are peripherals on GRIDDER. The maps (and beams) that are produced will be placed on GRIDDER's disk.

The DISPLAY machine has direct (shared) access to the disks upon which GRIDDER stores its output maps. The DISPLAY machine can also be used to submit a new mapping request to the pipeline scheduller thereby closing a loop between data editing, mapping and displaying. Requests for new maps submitted from the DISPLAY machine will have priority over requests originating in the DEC10.

Finally, map data will be written to FITS tapes either on GRIDDER or DISPLAY. Visibility data can be written on UVFITS tapes by a program in the SORTER machine. (It can also be backed up from disk to tape by a utility routine on SORTER.)

There is no selfcalibration program planned for the pipeline system. For spectral line data it might be worthwhile to combine the facilities of the AIPS system with those of the pipeline. Visibility data and a dirty map for one channel could be copied from the pipeline system to the AIPS system. There one could use the "selfcal" procedure to get a gain table which would be returned to the pipeline in order to create maps for all the other channels.

See p.15? re self-cal

9

Figure 4

Pipeline - functional

- ^{IO} ● DECIO
- Pipeline
- Data path
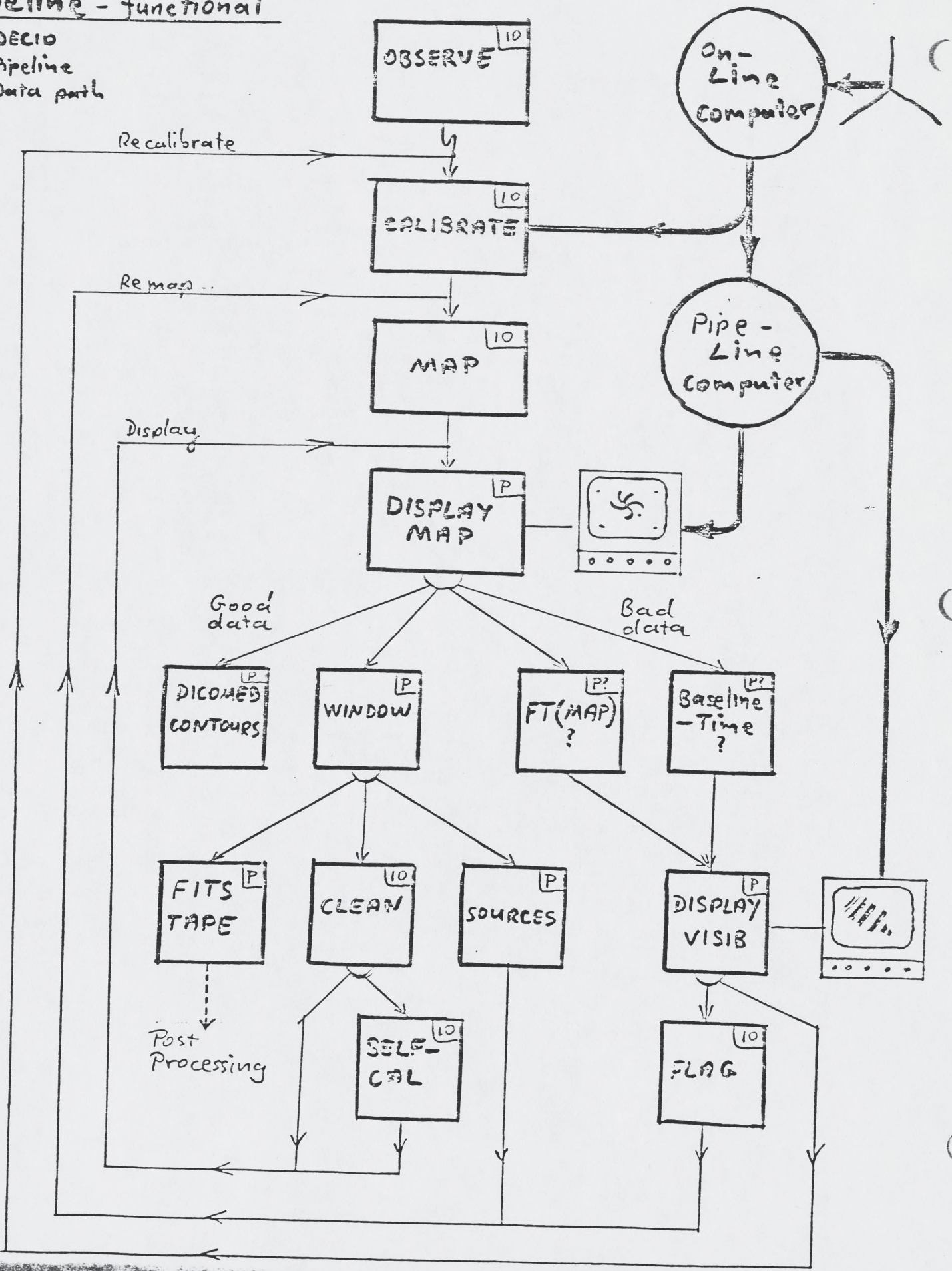
3. PLANNED USE OF THE PIPELINE: STEP-BY-STEP

3.1. OVERVIEW

Figure 4 is a functional overview of the pipeline system from the user's viewpoint. In some ways the Pipeline will appear to be similar to the older 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.

3.2. INPUT TO PROGRAMS (STANDARD COMMANDS)

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

Some Pipeline system functions involve the initiation of programs that run on one of the PDP-11's. This will include such things as FITS tape reading/writing and loading visibility databases from tape. All of these programs will be run from a DEC-10 terminal using a program called PIPELN which will give the user simple and direct communication with the required PDP-11 programs. The pipeline user will never need to use a PDP-11 terminal (except for the DISPLAY system, of course). The PDP11 programs will be written to use a command scanner which provides standard commands that work exactly like the DEC-10 standard commands.

The user interface for display system functions will also use standard commands but it will also contain display-oriented and menu-selection input. 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 may 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. Each visibility catalog record will
describe all the data for a set of scans which have unique values of
..... 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 but
calibrator data may not be ignored. 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 6000
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~~ *to* 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.

First use
of this
"What is it?"
asks a user!

13

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 corrections will be stored in the gain file in exactly the same manner as is currently done.

3.7. PIPELINE CONTROL FROM THE DEC-10

3.7.1. MAPPING 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. 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.

contradicts p. 9!

3.7.3. 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. Log files of the status of various jobs will be created on
each system. This file will be accessed by PROBE.

3.8. 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, although it is not clear where these correlator
flags will be stored. The flagging of individual visibility records
will not be supported.

Flagging

3.9.   REMOVING DATA FROM THE SYSTEM

3.9.1. 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.

The UVFITS program will provide a standard format tape that can
be used to transport calibrated visibility data to other machines at
any site.

3.9.2. REMOVING MAP DATA FROM THE SYSTEM

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

3.10.   MISCELLANEOUS

3.10.1. 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.

3.10.2. MAP NAMING CONVENTION

(See the May 19th, 1983 memo distibuted by R. Ekers on this subject.)

3.10.3. DISPLAY SYSTEM USE

(See the next Section)

3.10.4. 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. CURRENT STATUS OF THE SYSTEM

4.1. HARDWARE

The greatest difference between the pipeline hardware, as described above, and the current configuration is the absence of separate PDP11s for sorting and gridding. All of the GRIDDER and SORTER functions are being carried out by a single PDP11 which is identified as SORTER. The missing PDP11 is on hand but has not been integrated into the system.

Only 6 of the 12 disks drives that are ultimately to be used on SORTER and GRIDDER are available to the one machine that is performing the combined functions.

There is no DECNET link between DISPLAY and the DEC-10. The disks that are to be shared between GRIDDER and DISPLAY are not yet in place. This leaves the DISPLAY machine somewhat isolated from the rest of the system. (There is a DECNET link to the SORTER/GRIDDER combination).

A conversion kit that, it is hoped, will make the Telex tape drives useable (reliably) at 6250 bpi has not yet arrived.

A sporadic problem in the transpose memory has recently been identified as such but has not yet been located.

The image storage system and the control panel for the DISPLAY machine are still under construction.

4.2. SOFTWARE

The pipeline hardware is in a state where it can be used to prepare maps. Even though many important pieces of pipeline software are missing, enough is available that the pipeline actually has been used. To reach that condition it was necessary to write "kludges" as temporary substitutes for some essential software that is not yet available and not expected in the near future.

This section lists the major software elements that await completion. It also describes temporary replacements that have been thrown together. But this section is not meant to be an operations manual. A later section of this document will describe how the conglomeration of existing software can be used in its present state.

The "online" filler programs for SORTER and the DEC10 are not yet available (but they appear to be imminent). This means that data must be filled from tape.

The tape filler for SORTER is working but at a disappointingly slow rate: 2000 visibility values per second for continuum data, and only half of that rate for spectral line data. (Recall that the goal for online filling is 6000 values per second.) The tape filler for the DEC10 is an old program that has not yet been modified to accomodate the needs of the pipeline. In particular, all visibility data - not just calibrator visibilities - must be filled in order to generate proper index and gain files. This turns out to be just as well since, for other reasons, it is convenient to have the source visibility data in the DEC10.

There is no way to flag visibility records in the pipeline data base on SORTER. There is not even a way to get a useful listing of source data on SORTER since the LISTER program has not been written for that machine. A utility program can be used to check the integrity of files but it was never intended as a lister program, it cannot do flagging and it cannot yet patch "broken" files.

The superiority of DEC10 data editing facilities over those currently available for SORTER data bases is an important reason why it is desirable to fill source visibility data into the DEC10 rather than into SORTER. The GRIDER program on the DEC10, whereby the user submits a mapping request to the pipeline scheduler, offers the option of using DEC10 rather than SORTER visibilities for mapping. If that option is exercised the pipeline reads the DEC10 VIS files as well as its gain and index files.

Of course the DEC10 visibilities must be stored on SORTER if they are to be used in the pipeline. The organization of these files differs from the "pigeon hole" organization used for normal SORTER databases. The pipeline gridding routines were expanded to read either type of file.

No part of the DISPLAY software is operational yet. (Some pieces of software left over from the former IMPS system can be used with pipeline data).

Only the MAPCON (scheduler) program creates files that can be accessed by PROBE. Other programs need to send messages to MAPCON and/or they need to write files that PROBE can read. There are no individual log files, merely a system file for programmer use (in debugging).

The UVFITS program has not yet beeen written so pipeline databeases cannot be transported to/from other machines.

There is no way to flag visibility data in the pipeline by creating "history" records. (The DISPLAY system doesn't write into the HISTORY file in any case.)

The FITS program cannot handle real-valued maps. In addition the 32-bit wordsize has never been implimented in FITS. (The entire program needs to be rewritten to use the generalized MAPIO package that has been developed recently.)

The DEC10 database orgaization needs to be revised to accommodate spectral line data. This involves changes to a great deal of software; as many as 12 man-months may be required for this alone.

5. A USER'S GUIDE THROUGH THE CURRENT SYSTEM

1- Data acquistion should NOT BE AFFECTED IN ANY WAY. Prepare for observations exactly as you have always done.

2- Data may be stored on the DEC10 in realtime or with the current DEC10 FILLER program exactly as it is done without the pipeline. (Don't bother trying to use the pipeline filler at this time.) (If you INSIST on using the pipeline filler see step 13 below.)

3- Calibrate and edit the data exactly as you would do without the pipeline. In addition to the usual routines you may also use the pipeline system to produce gridded data displays of baseline versus time in map format. These maps can then be analysed on the IMPS or AIPS systems. To use this pipeline feature you should run the DEC10 program BTMAP which submits baseline-time requests to the mapping system. (This program uses standard commands.) The program BTLST lists the baseline-time displays in the user's map catalog. It will also give a summary listing or plot of seleted antennas from these displays. The IMPS horizontal plot option will also list the antenna pair and time for selected points in these maps.

4- On any DEC10 terminal (logged in from any user number) you should type the following line (exactly as shown but followed with a carriage return):

R PIPELN (DSKSPACE)

The result will tell you several things that you need to know:

A- Is DECNET working between the DEC10 and the pipeline ?
   (If not, you can forget the rest of these instructions.)

B- Is there space for your data set on any of the pipeline disks? If so, on which one ?

At the present time the program that transmits databases from the DEC10 to the pipeline can only use DB1: so that is the absolutely crucial disk.

5- If the disks appear to be full you may be able to remedy the situation by deleteing some of your old databases. (Otherwise you need to get help to find out who is using all the space.) To free up space that you own, you should use the DEC10 programs CATLST and/or PROBE.

6- If DECNET is working and space is available you may now submit a mapping request to the pipeline. From any DEC10 terminal (logged in under your own PPN) run the program GRIDER. This program submits mapping requests to Wim Brouw's mapping routines in the portion of the pipeline that is currently functional. The input requirements (standard commands) are similar to the MAKMAP mapping parameters with two additions. For spectral line data GRIDER has a "CHANNELS" command. It also allows you to specify (via the standard command DBTYPE) whether your dataset is resident on the DEC10 (as it probably will be) or in the pipeline (as it might be for experimental purposes).

7- Go away for a while. Get a cup of coffee - get a couple of cups of coffee! Transmitting a large database from the DEC10 to the pipeline may take several hours. Consider the following calculation: 30,000 DEC10 disk blocks, times 128 words/block, times 5 bytes/word, times 10 bits/byte is 192 million bits. At 56 Kilobaud that is 1 hour's worth of data EXCLUSIVE OF OVERHEAD.

8- You may use the program PROBE from any DEC10 terminal to monitor the state of the pipeline and the progress of your request. This routine also lets the user delete requests that are no longer needed. This is a another useful thing to do to free up disk space since datasets from the DEC10 are kept in the pipeline until the request is deleted.

9- You may also run CATLST from a DEC10 terminal to display the list of maps that has been created for you. (Be sure to set the parameter "INMACHINE" to SORTER).

10- At present you must run a special program to transfer maps from the mapping portion of the pipeline to the display portion. To do this you must run the program GETMAP on the DISPLAY machine (from a terminal in the room that was formerly called the IMPS room).

11- For small sets of maps the GETMAP program is convenient but for large sets of maps it is more efficient and faster to use the standard program FITS. This is what you must do in any case to transfer your maps to the AIPS system. Before starting this program you should get a tape and mount it (with the write-protect ring IN) on the pipeline tape drive. Be advised that the hardware is far more reliable at 1600bpi than at 6250bpi. At any DEC10 terminal (logged in from any user number) you should type the following line (exactly as shown but followed with a carriage return):

R PIPELN (FITS)

You should ALWAYS use the VERIFY option since the tape might be unreadable. If, in fact, the tape IS unreadable try getting someone to clean the oxide off of the tapedrive heads.

12- Once a request has been sent from the DEC10 and the dataset stored on the SORTER disks it is possible to redo some mapping parameters and to resubmit the mapping request without transmitting the dataset from the DEC10 again. Needless to say, this speeds things up a great deal! To submit a new request you should use a DEC10 terminal and run the program REMAP. From there on you should follow steps 7 through 11 as before.

13- If you have already calibrated your data and you have the necessary GAI and INX files on the DEC10 but you do not have the VIS data there, you might want to experiment by filling the VIS data into the pipeline database rather than into the DEC10. (It should be much faster!) To fill a dataset in the pipeline you must first mount the ARCHIVE tape on the pipeline tape drive (Write-protect ring OUT - density 1600 bpi) then go to any DEC10 terminal and type:

R PIPELN (DBFILL)

Don't forget to set the DBTYPE command to SORTER in the GRIDER programs in order to use this dataset in the mapping routines.

***** NOTE *****

All mapping problems should be refered to Bob Payne (294).
Problems with DBFILL or DBUTIL should be refered to Bob Duquet (306)
and problems with FITS or IMPS to Jim Torson (277).

6. PLANS FOR THE DISPLAY SUBSYSTEM

6.1. THE HARDWARE

6.1.1. DEVICE LIST

The DISPLAY computer peripherals include the following graphics devices:

Image Display   IIS Model 70F. Two 512x512 image planes,
                         4 graphic overlays, cursor, look-up tables for
                         color & gray scale, ALU and Videometer.

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 Versatec D-1200A. 200 dots per inch. 10.5 inch
                         wide plots.

Pen Plotter   Zeta Model 1453B. 12 inch wide drum plotter
                         with four pens.

6.1.2. 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 to 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.

6.1.3. 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.

6.2. SOFTWARE

6.2.1. 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.

6.2.2. 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.

6.2.3. 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.

6.2.4. DI3000 Graphics Package

The DI3000 graphics package has implemented the GSK graphics standards. This allows the possibility of writing device independent application software.

6.2.5. 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.

6.3. USER INTERFACE ON THE DISPLAY SYSTEM

6.3.1 GENERAL

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.

6.3.2. 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:

sub-map?
Load a sequence of maps?

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.   *images! Not maps?*

6.3.3. 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.

6.3.4. 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.

6.3.5. 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.

7. MISCELLANEOUS SYSTEM IMPLEMENTATION DETAILS

7.1. 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.

7.2. 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 PIGEON 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 PIGEON 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 DEC-PIGEON will run on SORTER to read the data from the shared disks. At startup, MC-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.

7.3. VISIBILITY DATA SORTING ON SORTER

The PIGEON 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 and splits line databases into 8 chaannels. *groups.*

7.4. VISIBILITY DATABASE FORMAT ON SORTER

For each user that has visibility data on SORTER's disks, there will be a single "Database Control" file. The name of this file will be the four digit user number (leading zeroes are required) and the extension will be .DBC. Each record in the DBC file 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.

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 Number 161, "Floating Point Format for Visibilities" by Barry Clark.

7.5. SORTER/GRIDDER DISK PACKS

It has been recognized for some time that we needed some mechanism to cope with three problems related to disk usage by the pipeline. The first problem is how to make use of more than one pack for a large multi-file database. The same problem crops up as the need for flexibility in utilizing available disk space wherever it might be. The second problem is how to minimize the impact that would result from the loss of one or more disk drives. The third is what to do about private packs.

The problem is solved by giving a pack name to each disk pack and having system utilities to maintain the pack name to physical disk drive translation tables. Then the pipeline programs at the user level deal with physical pack names and inner level routines figure out what physical drive the pack is on.

7.6. 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.

7.7. 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 rescheduled 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. The log of each event does not contain all information in requests.

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. The user

33

interface counterpart tasks currently exists only on the DEC10 (user
CATLST does exist on PDP11).

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

7.7.1. 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.

7.7.2. 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.

7.7.3. 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.

7.8. MAP MAKING

The user will submit a mapping request from the DEC-10 using the
GRIDER 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. Data read from the disks shared by SORTER and
GRIDDER will be will be sent to AP-1, which will perform data
selection, gain corrections, gridding and convolution. 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:

> VLA Computer Memo Number 157   February 1981.
>   Remarks on the GRIDDER Development.
>
> VLA Computer Memo Number 158   September 15, 1981.
>   GRIDDER System. Part 1 - Array Processor Programs.
>
> VLA Computer Memo Number 159   September 15, 1981.
>   GRIDDER System. Part 2 - PDP 11/44 Programs.
>
> VLA Computer Memo Number 160   September 15, 1981.
>   GRIDDER System. Part 3 - Changes made to FPS Software.

7.9. 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 currently be displayed on the IMPS system (and will be displayed on DISPLAY). The program BASPLT on the DEC-10 is used to initiate the generation of baseline-time files.

7.10. MAP CLEANING

The B. Clark CLEAN algorithm is implemented and will be the standard CLEAN available on GRIDDER. The current version does not use the transpose memory.

7.11. SELF CALIBRATION

The self-calibration algorithm currently used in the AIPS system will be implemented. It will be possible to ship gain tables from AIPS to GRIDDER.

7.12. 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 GRIDDER 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 GRIDDER as part of the mapping requests.

No! Delete this section and renumber or else add note of no support for SELFCAL

8. UNRESOLVED ISSUES

8.1. MAPPING DURING OBSERVING.
How will the user be able to make maps while his observation is
still in progress?

8.2. 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?

8.3. 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?

8.4. 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?

8.5. 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?

8.6. 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?

8.7. 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 flaging into the SORTER database, be possible? 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?

8.8. 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?

8.9. 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?

8.10. 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?

8.11. 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?

APPENDIX

CHANGES TO THE PIPELINE PLAN (as of Version 5)

1- The high speed data link between the SORTER and GRIDDER component machines has been eliminated in favor of sharing the discs that previously were accessible only by SORTER.

2- The Decnet links have been reorganized to remove the previous critical dependency upon the MAPPER machine and to provide a connection from the DISPLAY machine to the DEC10. (A new figure has been added to this document to show the Decnet connections in more detail.)

3- An AP has been placed on the SORTER machine to speed up various tasks such as UVFITS.

4- The amount of memory shown in each machine and array processor has been increased to levels which currently exist on each unit.

5- The processor in the GRIDER machine has been changed from an 11/44 to an 11/70 by swapping it with the MAPPER machine which is expected to become redundant.

6- The machine that was known previously as MAPPER has been given two new roles and a new name. It will serve primarily as a workstation for program development and as a central link for DECNET. Henceforth it will be called the WORKER machine. (We can continue to use the "mapper" software in the GRIDDER machine.)

7- A DEC-10 program has been written to give the pipeline user simple and direct access to all PDP-11 programs from a DEC-10 terminal. This means that the user will never need to use a PDP-11 terminal (except for the DISPLAY functions of course) and need not know anything about the PDP-11 operating system commands or conventions.

8- This document will henceforth contain a User's Guide to the currently implimented system. It will also include a list of problems that have been reported and the fixes (if any).

9- The text in this document has been re-organized and almost all of it has been reworded.