SGP MEMO NO. 3B

FILE NAMES FOR VLA DATA PROCESSING

Eric Graham

October 17, 1980

DEFINITIONS

For the purpose of this note I will define a "file" to be a named set of related data that is treated as a single entity by a data processing task. The operating systems of the various machines that perform VLA data reduction support file systems. I will call the file systems that they maintain OS-files. Since OS-file naming conventions and structure are different on different machines, and OS-files do not have all the properties that are needed by VLA data reduction tasks, they need only be considered as a tool for implementing a file system on a particular machine.

The term "map" has often been used to describe files used in VLA data reduction processes. It has been correctly pointed out that there need not be a one-to-one correspondence between maps and OS-files. The only objection to the word "map" is that it conotes a picture of something in sky, frequency or U-V space, and does seem appropriate for describing, for example, a file of visibility records.

EXISTING FILE NAMING SYSTEMS

At least six file naming systems are used in VLA data processing:

System name-component	description	example
TOPS-10		
device:	4 character predefined name	DSKF:
name	1-6 alphanumeric characters	3C273
.extension	0-3 alphanumeric characters	.VIS
[directory]	project number (6 octal digits), programmer number (6 octal digits), optionally up to 5 sub-file directories	[14,342]
RSX-11M		
device:	2 alphabetic and 0-2 octal digits	DB0:
[uic]	8 bit group and member numbers in octal	[300,20]
name	1-9 alphanumeric characters	MAPRCV
extension	0-3 alphanumeric characters	.GAI
;version	15 bit number in octal	;13

TABLE OF CONTENTS

DEC-10 SOFTWARE

name	1-6 alphanumeric characters	SS433
.ext	predefined name of 3 characters	.INX
[p	project number)	
,pn	programmer number)	[14,342]
,category]	optional string)	

MAPCON SOFTWARE

number	6 hexadecimal digit map number assigned when the file is created	003C41
.ext	3 character extension	.MAP

POST PROCESSING SOFTWARE

userid	An implicit user identification	
name	1-12 alphanumeric characters	1226PLUS023
.type	1-6 alphanumeric characters	.ICLN
.version	a decimal number in the range 1-30000	.19

IMPS

userid	Implicit user identification	
name	1-12 characters	1751T

DISCUSSION

We see that each software system has a different way of specifying a file. Could a single specification be adopted for all software that a user need encounter?

If so, two approaches are possible: select one system and change all the others to conform to it; or to design a new file naming convention and change all existing software.

A separate directory must be maintained for each user, to avoid name clashes and accidental corruption of other peoples data. If the ID of the file creator is a part of the file specification, then a user can access the files of another user.

The TOPS-10 file name of 6 characters is clearly too short for good mnemonic content. Even the RSX-11M names of 9 characters are really too short. Since both IMPS and the post processing software use 12 character names, this would be a good length to adopt.

The use of a file type to distinguish different kinds of files has several advantages. It can be used to protect users from performing operations which should never be attempted, such as using a gain table as if it were a set of clean components. If the type implies band and/or polarization, then multiple band or Stokes maps could be produced with a single command without the user having to supply

multiple names. The philosophy is that the executing task should generate the types for output files, and to have the type for input files be generated by context or by standard command inputs. File deletion, for example, would require either explicit or wildcard file types to be specified. Since the file type contains only a limited quantity of information, I would favor one character for each independent quality that is being described.

For example the first character could denote the kind of file: map, gain table etc. and the second and third, where appropriate, could describe the band and Stokes parameter.

A standard set of codes would be easy to use, especially with a crib card. The real problem would be to agree on how many independent qualities to support, and what character codes to adopt.

A version number is useful in several ways. If the user specifies a file that already exists as an output file, the system can generate a new version in order to prevent overwriting the old version. If the user does not wish to invent filenames, then he can have all his files with the same (default) name and refer to them only via the version number. The major drawback with version numbers is that they facilitate keeping unwanted files. A purge utility, which deletes all but the latest version is of some help, or the scheme used by the TI-ASC of allowing only a finite number (different for each file type?) of versions. When the limit is reached, the oldest version is automatically deleted.

I would like to propose that when the system deletes the most recent version of a file, version N, and then subsequently creates a new version, that it assign the version number N+1 rather than reuse version N. All files will then be created with unique names, irrespective of the history of deletions. If all files of a given name are deleted, then the record of deleted files can be discarded.

SUMMARY

The following scheme for naming files is proposed for the new MAPPER software. It has been chosen primarily for the convenience of the user, but with an eye to making it as similar as possible to existing naming systems.

A full file name consists of 4 parts:

iii:aaaaaaaaaaaa.eee;nnnnn

where

iii	is the file owner's identification number.
aaaaaaaaaaaa	is a name of up to 12 characters
ttt	is a string of three characters describing the type of the file.
nnnnn	is the version number of the file, it is a decimal number in the range 1 to 30000.

where it is understood that iii will always default to the users own number; ttt may only be specified for input files, and it will normally default by context; nnnnn may only be specified for input files, the