AIPS MEMO NO. 15

To: Scientific Staff, Programming staff

From: B. Clark

Subj: Mapnames

Date: August 12, 1982

For th sake of discussion, I would like to make a suggestion about mapnames.

1). Maps (and map related files) ould be accessed only by their names. This implies that names are unique.

2). An inherent part of any useful system is a powerful wildcard facility. This is to be provided at the lowest level in the I/O system, so that it is automatically incorporated in all programs.

3). The general mapname would be

(useridnumber):(usergenerated p rt).(programgenerated part);(version)

The characters colon, period, and semicolon would be reserved characters and could not appear within the usergenerated part of the name. (If you really want a mapname of 3C454.3, we ould implement the DEC-20 convention that a period preceded by a ^V is part of the string rather than a delimiter). The whole name should be long enough to allow users to make their own conventions with reasonably mnemonic names in subfields. I suggest 40 characters.

4). The wildcard convention would be slightly different for input and output fields. For input files:

a). Wildcards are not permitted in useridnumber.
b). ? matches any single character
c). * matches a string of any length up to the first occurance of the character followir~ it.
d). " matches the last string matched by a * or a ?
e). If the version field is missing or blank, this matches the highest version number of the first name matching in the other parts.

For output files:

a). Wildcards are not permitted in useridnumber.
b). The only wildcard permitted in the usergenerated part is the ".
c). Programgenerated part and version will be ignored if specified. The program will put its own string in the programgenerated part with the guideline that in the usual progress of a single observation through the system, these tags are sufficient to distinguish all maps (and related files) made. The version will be one higher than the highest currently matching the other descriptors (1 if no other map matches).

Examples:

An observer looks at 3C48, 3C196, and 3C454.3 at 2, 6, and 20 cm. When he makes maps (and beams, and Stokes parameter maps) he gives user names on the system 3C48(2cm), 3C196(20cm), 3C454(6cm), etc. Makmap assigns program generated parts of .I, .B, .Q, and .U.

When he later wants a selective listing of his catalog, he can get all his 2cm intensity maps with a mapname of

```
*(2cm).I;*
```

Or all maps, latest version, of 3C48, any polarization, with a mapname

```
3C48(*).*
```

Or all intensity maps latest version with a mapname

```
*.I
```

He could tell FITS to save all of his 6cm data with an input mapname of

```
*(6cm).*
```

and later restore the intensity maps under a different name with an input mapname of

```
*(6cm).I
```

and an output mapname of

```
"(6cm)old.I
```

5). It is a somewhat touchy subject, but I think I would have the use of wildcards imply iteration. That is, in the above example, the observer could clean all of his 20 cm maps by specifying to CLEAN

```
Dirty   *(20cm).I
Beam    "(20cm).B
Clean   "(20cm)
```

The adverse feature of having wildcards imply iteration is that an observer can thereby monopolise the system for hours by typing only a few characters. If iteration is implied, it might be useful to allow specification of a version in the output, where a match implies no action to betaken, so that if the above example aborts in the middle it can be restarted with an output specification

```
Clean   "(20cm);1
```

to avoid redoing things which have already been done.

Meeting on Map Naming corrections August 6, 1982

In this meeting we discussed the user interface only.

General Principles

We agreed on a few principles

1. Not to be tightly coupled to the system file structure.
2. The name is a mechanism to specify the map the user wants. It does need to also function as a complete description of the map the users wants. The header holds this information and convenient programs should be provided to display i
3. The same basic structure should be used for both batch and interactive use.
4. The name resulting from an operation generating a new map should be predictable "closely enough". Closely enough could imply, for example, the highest version number rather than the actual version number.
5. The map name should include some structure to be used as an aid to the users organization and bookkeeping.
6. For systems which have an interactive screen based display accessible to them map selection can be made by pointing to a list of all maps. In this case any map naming convention is satisfactory.
7. Users should have their own ID and catalogue. A user should be able to access (but not write) into another persons area. Although occasionally useful, mixed specification of maps from more than one area is not a requirement. If it is difficult to implement a copy utility would be satisfactory for the few occasions when this is needed.

Proposals discussed

Proposal 1.

All aspects of the name encoded into one character string and allow the flexible use of wild cards (BGC+EG)

III: aaaaaaaaaaaa.ttttttttttt;nnnnnnnnnnn

This the same as the proposal in Eric Graham's document of October 17, 1980, excepting that both the name and type field are unrestricted length and the version number is defined to default the highest on read and to highest existing +1 on write. (ie don't fill in gaps if maps are deleted).
In addition wild characters will be used extensively following the DEC10 conventions with some additional elaborations.

Proposal 2.
Essentially the same as proposal 1 except that the components of the name string are separate input variables. (Similar to AIPS convention).
ID = III
NAME = aaaaaaaaaaa
TYPE = ttttttttttt
VERSION = nnnnnnnnnnn

Although structurely similar to AIPS the main AIPS disadvantages could be removed in a new implementation. For example no variable disk at user level, common no global use of variables, type defaulted on input when unambiguous, and wild cards allowed as in proposal 1.

Proposal 3. Catalogue numbers.
A simple integer, which is essential a substitute for a pointer in an interactive system, is used to indicate the required map. This is similar to the AIPS use of catalogue number but with two major exceptions: the catalogue number can be specified on output and multiple maps specifications are allowed, eg 13,14 or 5:10 to indicate all numbers from 5 to 10. The feeling at the meeting was that proposal 3 could be an additional feature to either proposal 1 or 2. For example the catalogue number, with suitable identifier, could be given as an alternative to the character string.

Problems
Two main areas of uncertainty appear to exist. Neither proposal 1 nor 2 provide an easy mechanism for the specification of multiple maps. Proposal 2 is especially awkward since all input variables have to made available by the application program for all possible input or output maps. The second problem relates to the definition use for the map type field. This is provided to give additional user convenience and bookkeeping but how fine a discrimination is required?