

30 May 1984

To : Don Retallack  
From : Bob Payne  
Subject : Grider Scheduling priorities

On the GRIDER mapping system the MAPCON scheduling task scheules mapping tasks based on a set of priorities.

There are four priorities:

- a user setable priority (may be lowered with PROBE)
- a usage priority to penalize heavy users of the system
- an aging priority to insure that a job will eventually be run
- a time usage priority based on an estimate of the time it will take to process the job.

This priority scheduling was implemented in terms of the old MAPPER mapping system and still reflects that system. In particular the Wim Brouw mapping programs using SORTER datasets and the newer version of CLEAN are no longer handled correctly in estimating the time usage priority.

The other three priority assignments are still being set as in the old system and are independent of the mapping tasks themselves.

The time usage can no longer be set correctly without some investment of effort. In the old system the mapping request contained a count of DEC10 visibility data and since sorting dominated the map making it also dominated the time estimate. With the use of SORTER datasets we no longer have a count of visibility data before the request is processed so we can no longer estimate the time correctly. Also the old CLEAN program always verified the existence of the dirty map and beam before processing the request. This is no longer done and so we no longer have accurate information for setting the CLEAN type priorities either.

We can at the addition of some overhead conform to the old pattern of calculating the time estimates. The CLNMAP program could be changed to verify the size of the map to be cleaned.

The map submitting task could in fact check the SORTER datasets and count the visilitites. In any case the priorities would still be wrong since sorting the data may no longer be the dominant time consideration.

\*\*\*\*\* I would suggest that we change the priority based on time estimates to one based on disk useage instead. This would require very little additional code since I now have in MAPCON the total disk requirements for the task.

\*\*\*\*\*

The following code taken from various places in MAPCON sets the priorities.

```

C
C   Time estimate for data transfer
C
N=LENTMP
NVIS=LENTMP
IF (N.EQ. 0) N=20000 * RP added this line
T=1+.0004*N for sorted datasets when W=0
IAP=T
16 CONTINUE
OPRIO=9
PRIAGE=10000
PRITIM=IAP

C
C   Time estimate for a clean request
C
X1=ITRLIM
X2=NSIZ/64
T=X2+X2*X2*X1/1000.
IAP=IAP+T
GO TO 16

C
C   MAPSEL and MAPSRT
C
V=NVIS/1000.
T=T+2+.063*V+.15*V**1.13

C
C   MAPGRD and MAPFFT
C
4 V=NVIS/1000.
T=T+1+.028*V
XN=NX/64
T=T+1+.028*XN*XN
3 MIXPTR=MIXPTR+$PROD(2,mixlength)
GO TO 10
1 Q(priage)=10000
IF (T .LT. 1.) T=1.
IF (T .GT. 1440.) T=1440.
Q(pritim)=T

C
SUBROUTINE SETPRI(QEL)
INTEGER *2 QEL(1)

C
C   This subroutine establishes the overall priority based
C   on:
C   OPRIO      the user settable priority in the range 0 to 9,
C   PRIUIC     a handicap based on the other jobs a user has in the queue
C   PRITIM     a term proportional to the estimated time for completion
C   PRIAGE     a term to account for how long a request has been waiting
C
P1=QEL(origpri)
IF (P1 .LT. 0.) P1=0.
IF (P1 .GT. 9.) P1=9.

```

```
P2=QEL(priuic)/10000.  
P3=QEL(pritim)  
IF (P3 .LE. 0.) P3=1.  
P3=1440./P3  
P4=QEL(priage)  
P4=10000./P4  
P=P1*P2*P3*P4  
IF (P .GT. 10000.) P=10000.  
QEL(priority)=P  
RETURN  
END
```