

UNIVERSITY OF MINNESOTA COMPUTER CENTER

Deadstart Systems Newsletter

09 December 1975

Vol. 1, No. 15

Send all comments, criticisms and contributions to the editor: T. W. Lanzatella
University Computer Center, 1925 Sather St., St. Paul, MN 55113

NEW SYSTEM INSTALLATION

Due to the holiday schedule, the next deadstart tape will be installed on Tuesday, 23, December 1975.

NOTICE OF CHANGES TO THE OPERATING SYSTEM

Dennis Lienke converted a 350 card modification to program COPYU on MOMS which enhances user documentation.

Marisa Riviere added a new entry point to RFM named RFMOD. The RFMOD command replaces the command sequences RFPUR,RFSAV and RFPUR,RFDEF. Marisa also updated RFM documentation and fixed several small bugs in RFCAT.

Bill Elliott contributed the following four modifications:

- a) A repair to 1DU which fixes Julian date processing.
- b) A modification to 1DU which adds two new operator drop messages.
- c) Program CERTIFY now writes accounting data to the ACCOUNT file.
- d) A fix for an interlock problem between MAGNET and CERTIFY which prevents erstwhile rewinds fouling the footage counter.

Bob Zalusky completed the proposed changes to program MODVAL (DSN, 1, 12, p. 2) with the following change in specification. The new format of the PASSWOR command is:

PASSWOR,OLDPW,NEWPW/UN=...

Note that /UN=... is the new option. The symbol UN specifies which user number's password to change.

PROPOSED CHANGES TO THE OPERATING SYSTEM

Since PSR 410 is now an impending reality, the entry points OLD and NEW should be removed from CALLPRG since OLD and NEW are reintroduced as BATCH commands in PSR 410.

The following proposal was contributed by N. L. Reddy:

STRUCTURE OF PERMANENT FILE SPACE - N. L. Reddy

All the users in one family are first divided into 8 categories, depending on the last octal digit of their user indices. Permanent file devices are either master devices or not master devices. Each master device carries information about all the files belonging to one or more of the 8 categories

of users, depending on its device mask. Thus we are limited to 8 master devices in a family. Information about a permanent file is contained in 8-words in what is called a catalog entry on the user's master device.

An auxiliary pack is a world all by itself. It belongs to some family in terms of user number - user index relationship, but holds within itself all the catalog entries, indirects and directs of all the 8 categories of users. It can be either removable or non-removable.

A family includes all the master devices and other permanent file devices including auxiliary and removable devices belonging to users in one VALIDUX file.

In Standard Kronos, all the indirect access files and their catalog entries must reside on the master device. In Kronos 2.0, all the direct access files also had the same restriction. Kronos took a big step forward in level 2.1 when it permitted direct access files to reside in any permanent file device in the family, though the catalog entry was still in the master device of the user.

The MERITSS staff found one inconvenience in this concept. For example, one user purges his file and the next day decides to have it back in the system. He calls up UCC and to oblige him, one has to go through the dump of his master device to get at the catalog entry of his file, find its residence and load the file from the dump of the appropriate device. In Kronos 2.0 philosophy, we always know the residence. This was a big factor with MERITSS and they decided to stay with the Kronos 2.0 in this concept.

What is surprising is that we did the same on the Cyber.. The environment in the Cyber is different. We don't back up any unsecured files at all. We always encourage the users to back up even the secured ones on tape. I don't know how many such requests have actually been made by our users. We can either deny this service or request the user to wait until Bob Hursh gets time to load it back. We can also train our operators to go through 2 dump tapes to load a file back. Note that this has nothing to do with disk crashes, in which case, we just load back the affected devices with their own dumps.

The advantages of Kronos 2.1 concept are many.

1. We stay compatible with literature.
2. All the directs are uniformly distributed over all the devices, not requiring surgery if one device gets too full.
3. Better user control, like types of residence.
4. Ability to make a file permanent after it is created.

We also move closer to level 10, where a family includes not only permanent files, but input and output files. We have to spread them out to prevent filling up one device when lots of room remain in other devices.

If we decide to stay totally compatible with Standard Kronos, we have one minor problem with auxiliary packs. Currently we have four DI-2 devices

as our user file base, two DI-1 as auxiliary packs for staff and CALLPRG and one DI-1 as removable device. Our expectation was to use the two auxiliary packs as the main source for temporary file space, though the user file base can also receive temporary files. We have dual channel access to packs SP and STF for this reason. However, staff usage of SP and STF has grown according to Parkinson's law. Auxiliary packs are either private (associated with one user number) or public. Both of ours are public. We don't want any user permanent file to reside on the auxiliary packs. Standard Kronos, unfortunately, does not make any distinction between the main permanent file devices and public packs.

So temporarily at least, we cannot allow users to make a local file permanent after its existence, because most of the temporary files are expected to be on the auxiliary packs.

For consideration in the next few months:

Kronos is not very comfortable with auxiliary packs and a major reason for their creation seems to be to make available removable packs for private pack owners. In fact the same physical pack can be toggled between removable and non-removable status at deadstart. Our own fondness for auxiliary packs goes back to MOMS when we had UCC 101 - UCC 105 for users and UCC 901, etc. for staff. It has served us very well in keeping up two systems, one for production and the other for System time. To achieve the same end, but more in tune with Kronos, we may consider the creation of another family.

The advantages are:

1. We will have only staff account numbers in the second family. No user can get at "sensitive" staff files.
2. Users are prevented from using the packs in the second family. Now any well-informed user can get free space in STF, SP or SYSTEM packs.
3. The 6400 has 2 controllers, with a number of empty ports. The Cyber has a lot of drives with just a single channel access. By connecting the pack for the alternate family to one controller on either machine, we can get access to all staff files from either machine. There will not be any need to keep two copies of OPL-S etc. (some interlock is necessary).
4. Staff usage of 844-drives could decrease. Reconfiguring the three drives with auxiliary (removable and non-removable) packs as one drive for the second family, and two drives for removable packs, we can guarantee the users that at least one removable pack drive will be available all the time (if one drive goes down). This might encourage users to put their long files on removable packs.

SYSTEM MAINTENANCE: PEOPLE AND PROCEDURES

The DSN as a Document of Record - A. B. Mickel

For the first time since I joined UCC in 1972, I have gotten my technical materials not directly relating to my primary work (language processors) in shape; really organized. I have file folders now for:

- Deadstart Systems Newsletters (DSNs) 14 in all
- Systems Group Meeting Notices - 22 for 1975
- Systems Programming Information (including a set of notes from a rare

- short course taught by Larry Liddiard on FTN optimization and the faster than greased lightning SQRT routine)
- Timesharing Information (such as CATLISTS of UN=LIBRARY, XEDIT, and XMIT documentation)
- Minnesota Modification Method
- CALLPRG Information
- CALLPRG Changes
- CDC VIM
- 6600/Cyber 74 hardware
- Etc., etc., etc.

I also have a folder on Systems Group: Memos. For 1975 we in the Systems Group have written (note this does not include anything mentioned above) 76 typewritten pages of memos! Who says we don't like to write?

I suggest that we try to put information that may be a candidate for a memo in the Deadstart Systems Newsletter for several reasons.

1. It is an established forum of discussion and reference document.
2. Its readership and circulation are wider than any memo.
3. It (because of #1) becomes the easiest and fastest means of orienting and educating new UCC programmers to our ways of doing things.

Therefore if memos have to used, the author should at least have a reference pointer placed in the DSN. State the subject and date of the memo.

6400 USAGE - Bob Williams

Because of the recent turnover of Systems staff at UCC and the length of time since we last published them, some of the procedures for operation of the 6400 should be outlined here.

The operating hours for the 6400 are:

7:30 am Monday to 2:00 am Tuesday
8:30 am Tuesday to 2:00 am Wednesday
7:30 am Wednesday to 2:00 am Thursday
8:30 am Thursday to 2:00 am Friday
7:30 am Friday to 11:00 pm Friday
7:30 am Saturday to midnight Saturday
6:00 pm Sunday to Midnight Sunday

During these times, no deadstarts, SYSEDITS's, memory changes, channel commands or other activities not directly related to the operation of the 6400 should be performed. Use of DIS, 026, DIAL and games should be restricted to low usage periods and preferably cleared with MERITSS staff. Procedures which are related to regular 6400 operation are described in the Operator's Guide, Supplement to the Operator's Guide, and Operator Notes found on the console.

In addition, if 6400 operations are over for the day but end-of-operation procedures have not been performed (which can be determined by checking the Daily Report), these should be done before anyone uses the machine. The proper end-of-operations procedures may be found in the 6400 Operator's Guide. Be sure to indicate in the Daily Report that these have been done as indicated in the Guide.

Also, any and all action taken on the 6400 during, or relating to, normal operations should be noted in the Daily Report.

Finally, we would appreciate knowing about any use of the 6400 during non-scheduled hours so that effective distribution of the available time can be made. Thank you for your cooperation.

The following is a revised Systems' programming projects list which includes PSR410 tasks. Our target date for a production version of PSR410 is 01 February 1976.

W. Elliott

Update tape use guide (50%)

Online diagnostics for 844's and 669's (PSR410) (10%)

Rewrite EXAMINE for 9-track tapes

MAGNET, 1MT, BLANK, Tape utilities, DSD, DIS, 026,

1DS, 1DU, UFM to PSR410

~~MF501 - LOW PRIORITY~~

L. LIPPIARD

PASCAL

MNF

D. Hammes

Automatic divert

Queue file manipulation programs to PSR 410

Y. Hwang

M. FRISCH

MNF maintenance

MNF correspondence

B. Johnson

Mini-computer protocol (75%)

SUP10 documentation

A. Johnston

TELEX, 1TA, 1TD, 1TO, job scheduling to PSR 410

~~DSO DOCUMENTATION~~

H. Kurs

S2000 Version 2.40 installation

PSR 410/411 common product installation

T. Lanzatella

DSN

PL/I

Revised mod scheme report and development

MIRJE system maintenance

Machine retrievable documentation study

6400/CYBER 74 O.S. identity

PSR 410 installation coordination

CMRDECKS, IPRDECKS, LIBDECKS, ISF, CPU utilities, 1AJ, HELP to PSR 410.

S. Lenz

Installation of CDC ALGOL

UTALGOL
MIXAL
KCL extension (90%)

K. Matthews

Performance measurement statistics (80%)
Password hashing (80%)
PSR 410 accounting coordination
Revised mod scheme development
Deadstart programs, SET, IMS, MSM, REC, CPUMTR, MTR,
USERECS, C10 to PSR 410

R. Matthews

CYBER loader
LINK to PSR 410

A. Mickel

PASCAL compiler (75%) documentation (50%)
LISP interpreter (80%) documentation (30%)
SNOBOL interpreter (100%) documentation (50%)
MIXAL documentation (0%)
PASCAL users group (20%) CCINDEX (90%)
UCC users manual (10%) WRITEUP, LIBRARY. (20%)
Conversion guide for PSR 410
Unified documentation scheme study

J. Mundstock

ECS ROLLIN/ROLLOUT
1RI, 1RO to PSR 410

N. L. Reddy

New version of EXPORT
Permanent file performance measurement
Training personnel in communications
Modifying communications software - including SUPIO
Banner page revision
Off-line plotter queue file processor
New terminal protocol
Unification of queue file management in QFM
Revised SUBMIT command
Automatic DIVERT
Terminal validation (MODTERM)
COMPASS, SUPIO BATCHIO, EXPORT to PSR 410.
PSR 410 queue protect testing

M. Riviere

CALLPRG maintenance
WRITEUP maintenance
FTN library maintenance
SYSIO library maintenance

LIBRARY tape maintenance
FTN3 to FTN4 conversion guide (80%)
FTN4 permanent file utilites
NOTICE, NOTIFY for MIRJE
FTN, FTN LIBRARY to PSR 410

B. Sackett

Revised SUBMIT command
Terminal validation program (MODTERM)
Timesharing users debugging package

T. Salo

Repair LOADPF/DUMPPF to restore permits
Rewrite LOADQ/DUMPQ for PSR 410
Permanent file utilities, CPMEM, DOCUMENT to PSR 410

B. Stahl

PTR's
MNF testing, modification and maintenance

B. Wells

MERITSS NOTICE/NOTIFY
Timesharing users debugging package

B. Williams

MERITSS System maintenance
New KRONOS levels on MERITSS
6400/CYBER 74 O. S. identity
Deadstart dump analyzers

R. Zalusky

P register sampling programs PSAMP, SAMP

THE REVISED MINNESOTA MODIFICATION METHOD - T. W. Lanzatella

This report details the revised Minnesota Modification Method; a scheme devised to formalize the process of applying local software modifications to vendor supplied operating system software. The need for a consistent, well documented modification scheme is obvious. As long as UCC runs vendor supplied operating system software, local modifications to the software must be maintained as an entity separate from the vendor supplied software. This report is approximately the sixth revision of the Minnesota Modification Method and represents a better mod scheme for the following three reasons:

1. The scheme closer represents that employed by the 6400 group, thus simplifying the problem of modification portability.
2. Whereas on the present scheme, the smallest unit of text is all modifications to a single program, this new scheme provides for the smallest unit of text as

a single modification or feature - again contributing to modification portability.

3. The new scheme calls for a much less redundant documentation conventions; which means less typing when installing a feature which spans many decks on CDC OPL.

Modifications to the CDC operating system are maintained on a MODIFY style OPL named JPL, PN=STF, UN=LIBRARY. Each deck on JPL represents a modification to the CDC OPL. The MODIFY prefix character for JPL is /. Each deck on JPL has the following format.

```
NEWMOD
*/
*/      TYPE      DATE      AUTHOR
*/      Comments describing modification. Use as many cards as are necessary
*IDENT  NEWMOD
*DECK   DECK 1
*I      HISTORY.3
*
*      TYPE      DATE      AUTHOR
*      on line description of modification in columns 11 - 72.
```

modifications to DECK 1

```
*DECK   DECK 2
*I      HISTORY.3
```

On the modification description card, the modification type begins in column 11, the modification date begins in column 18 and the modification author begins in column 30. The modification type can be one of the following:

MOD	- U of M feature
GEN	- generalization cards added for other feature code to modify
PSR	- bug correction to CDC code
TWX	- CDC PSR code received as a TWX
KLUDGE	- Kludge code hopefully added to the system for a short time only.

The first deck on JPL is HISTORY. The HISTORY deck is used much like a generalization mod in that HISTORY simply inserts a single card at an appropriate spot in each deck on CDC OPL modified by U of M modifications. Each card serves as an anchor for insertion of comment cards documenting modification history.

Conventions - necessarily, several conventions regarding the use of this mod scheme must be adhered to.

1. Modnames shall be restricted to six characters or less.

2. Each *DECK card in a modification must be accompanied by comments describing the modification to be inserted after HISTORY.3.
3. Metamods, modifications against modifications on JPL, are strictly discouraged. The types of modifications should be used only when correcting a modification on JPL or when inserting DEPEND cards (see 4 below). Note that metamods can only be used to the extent that subsequent modifications do not depend on the altered modification. For this reason, the best way to repair a modification on JPL is to simply repair and resubmit the entire modification.
4. In the event that a metamod must be used, the following comment should be inserted after the *IDENT card of the modified mod:

11	18	30
*/	DEPEND	MODNAM 14

The comment should be read as, "this modification depends on MODNAM at approximately card number 14." The use of DEPEND cards also applies to generalization modifications. So as not to distort card sequence numbers, DEPEND cards are inserted under an ident name composed of the modname of the modification which modifies preceded by a \$.

5. The following command sequence edits and assembles program CPUMTR under the new mod scheme.

```
ATTACH,OPL=JPL/PL=STF.  
MODIFY,F,J,Z,LO=E,C=IN.+/NOSEQ  
RETURN,OPL  
ATTACH,OPL=LV11OPL/PL=SYSTEM.  
MODIFY,Z,LO=E./*READ IN/*EDIT CPUMTR  
COMPASS,I,U,L=0.
```