

UNIVERSITY OF MINNESOTA COMPUTER CENTER
Deadstart Systems Newsletter

23 June 1981

Vol. 7, No. 12

Send all comments, criticisms and contributions to the editor: T. W. Lanzatella,
University Computer Center, 2520 Broadway Drive, Lauderdale, MN 55113.
The University of Minnesota is an equal opportunity educator and employer.

TABLE OF CONTENTS

NOTICE OF CHANGES TO THE SYSTEM	77
PROPOSED CHANGES TO THE SYSTEM	78
B-DISPLAY REVISITED - A. B. Hastings	78
NOS 541 - D. W. Mears	79
AUTO DISCONNECT - D. W. Mears	80
SYSTEM MAINTENANCE	82
LAST WEEKS SYSTEMS GROUP MEETING - T. W. Lanzatella	82
CALLPRG AND LIBRARY TAPE NEWS - M. Riviere	82

NOTICE OF CHANGES TO THE SYSTEM

Marisa Riviere installed a small change to Callprg which changes the conditions on which Callprg routes an output file to her bin after a tape copy failure.

Jeff Woolsey applied a correction to multi-user job processing which causes the MID's for all MMF members to be stored in the system sector of all multi-user jobs.

Kevin Matthews repaired an error in DUMPPF which caused problems if the NP-option is used along with a selection list. The NP-option means do not purge the file before loading. The effect is that any file existing at the time of the load will be skipped if it also exists on tape.

Jeff Drummond installed X and N carriage control character support into BATCHIO. These characters were originally installed so that EXPORT could distinguish between normal and extended mode print files. This was necessary because EXPORT does not know about EC=A9 or A6. The neat thing about X and N support in BATCHIO is that banner pages, which always contain the X or N control character, will now be spaced properly when they print at Lauderdale. Jeff also installed documentation and coding standard changes into 19 of our local common decks.

I would like to point out that this batch of code is Jeff's last hurrah at UCC. After four years of timely and frequently witty contributions to the the system group, Jeff is departing for greener pastures at Cray Research.

Andy Hastings corrected several typographical errors in COMCMUM, a local common deck.

Don Mears added three new sites to SUPIO and corrected his recently installed X-OFF processing in LTD so that S and I keys are not ignored during end-of-line, carriage return or line feed processing.

Marshall Midden corrected LAJ and DSP so that the location of binary sequence errors and hollerith code errors are displayed as decimal instead of octal numbers. Marshall also installed some small changes into the Xerox 9700 queue file processor, XEROX, which straighten out SRU charge processing.

PROPOSED CHANGES TO THE SYSTEM

B-Display Revisited - by A. B. Hastings

The B-display message buffer is usually used to display the current status of a job. A local modification to the timesharing executive makes it possible for interactive users to look at the B-display by interrupting the job (with the I-key or break key) and entering the letter B followed by a carriage return. Unfortunately, it is not possible to look at the B-display using this method while a program which disables terminal control is executing (the interrupt is processed by the program). I find this most irritating when I'm using a procedure file, and I'm waiting for CALLPRG to load something. I hit the break key so I can look at the B-display and see how far the procedure has progressed, and CALLPRG immediately aborts because of the terminal interrupt. Also, certain terminals (i.e., a Terak operating under COM) do not have a convenient break key.

To remedy this situation, I would like to implement some CDC feature code which displays the B-display message in the output from ENQUIRE. This makes it possible to look at the B-display message merely by entering an "E" command, and without interrupting an executing program. A sample output might be as follows:

```
TERMINAL:    107, P 103/TTY
MACHINE ID:  72
CSET:        NORMAL
SYSTEM:      BATCH
FILE NAME:
STATUS:      EXECUTE
MESSAGE:     ASSEMBLING TELEX  PASS 2
```

If this format is not acceptable, I am open to suggestions as to how to rearrange it.

//////////

NOS 541 - by D. W. Mears

We should convert the operating system and product sets to the 541 release. As in previous conversions this represents a change to the last release of a level - NOS level 1.4. It will bring us up to the same level as MECC, Duluth, and HCS. It will bring us up to the level of most other NOS sites and thus make it easier to get and distribute NOS software. It will fix bugs and give us new features which we have already paid for. The conversion will give the new people in the systems staff experience in, and understanding of many parts of the system.

Here is a list of improvements that will result from converting to 541.

- 1) Deadstart from disk. This will allow faster and easier deadstarts.
- 2) More efficient disk channel utilization. PP's will be able to perform a seek and a data transfer on different channels.
- 3) BATCHIO will be able to print 8 in 12 ASCII files.
- 4) Exchange dumps from aborts will be written to a file (ZZZDUMP) instead of to the timesharing terminal.
- 5) CATLIST will be greatly improved by, 1) listing the direct and indirect access files separately, 2) giving a summary of indirect and direct access file counts and PRU's, and 3) allowing the specification of a subset of file names to be CATLISTed. E.G., "CATLIST, FN=DWM****." will produce a list of all files which start with "DWM".
- 6) REWIND, RETURN, and UNLOAD will permit the rewinding, returning, and unloading of all files except those in a list.
- 7) ENQUIRE will display the current B-display message.
- 8) Operators will be able to "up" and "down" mass storage channels through DSD commands.
- 9) 6DP (the DDP driver) will use the same ECS error recovery technique as CPUMTR and COMCECS. It will also issue better ECS error log messages.
- 10) CIO will enforce direct access file size limits.
- 11) Dayfile messages will be up to 80 characters long.
- 12) Reconfiguration of 885 drives and channels and swapping of 885 HDA head disk assemblies will be supported.
- 13) Reconfiguration of 844 drives across channels will be supported.

In addition to these new features, many bugs have been fixed. There are too many fixes to list here.

As far as I can tell, there should be no major changes which are not upward compatible. Thus, we will not need to provide a lot of advance warning to the users and this should not take a major user education effort. This means that we can work on the conversion and then decide at the end of August if we will be ready for fall quarter break. If we don't think we will be ready then, we can delay the change until Christmas break.

//////////

Auto Disconnect - by D. W. Mears

We occasionally have system hangs caused by malfunctioning hardware which makes PP's hang on the channel. The basic problem is that, due to a deficiency in the PP instruction set, it is impossible for a PP to protect itself against malfunctioning hardware during a high speed block transfer on the channel. Once the PP starts a block transfer, if the controller it is talking to stops transferring data, the PP will hang forever waiting for the transfer to complete. Once a PP becomes hung in this way, the only way to get it running again is to abort the transfer by forcing the channel inactive.

There are two ways of forcing the channel inactive. The newer CDC controllers have an automatic deadman timer which watches the traffic on the channel and when it looks like the channel might be hung, it makes the channel inactive. The problem is that we have many controllers without deadman timers (e.g., PDP11 Links, 6676 Mux's, 6681 3000 channel adapters, and the DC145 DDP). Also, there is no guarantee that a deadman timer will always work. In the last two months there have been several instances of 844 and 885 disk controllers which have deadman timers causing channel hangs.

The other way channels get unhung is through the operator entering the DSD "DCNxx." command which forces the channel inactive. This requires the operator to determine that the system is hung due to a hung channel, to determine which channel is hung, to get DSD to accept commands from the console keyboard, and then enter the DCN command before the lack of available PP's and memory starts causing problems with the subsystems. Each of these requirements presents a problem.

It is not easy to determine that the system is hung due to a hung channel. The symptoms are similar to the usual unfixable software or hardware crashes. Everything stops running and the console keyboard commands stop working. The only indication of a channel being hung is that one of the 12 or 24 channel status indicators at the upper right of the 12 or 24 channel status indicators at the upper right of the right display is a constant "E" or "F" rather than a constant "D" (indicating disconnected). Since this one letter does not stand out from all the other stuff on the B-display and since channel hangs do not occur very frequently compared to other types of hangs, it is easy to neglect to check for hung channels when the system grinds to a halt.

Once the operator determines that the system is hung due to a hung channel, the operator has to figure out which channel is hung. The channel status at the upper right of the right display is a list of 12 or 24 letters. There is no labeling to indicate which letter corresponds to which channel number. To compute the channel number, the operator must either memorize which indicators correspond to which channel or count the indicators from left to right in octal starting at zero and skipping channels 14 to 17 (which are not displayed). For example, the following status line is displayed when channel 27 is hung full:

```
DDDD DDDD EDDD DDDD DDDF DDDD 0000
```

The "E" for channel 10 does not indicate that that channel is hung. Channel 10 is the console display channel and it is always active and empty at the time the channel status line is being formatted.

Once the channel number is computed, the operator has to get DSD to accept the DCN command. Unfortunately, when the system hangs up, normal DSD command processing stops working. This is because DSD command and syntax overlays are kept on the disk. To process command syntax DSD must call a PP to load the syntax overlays off the disk. The operator can usually get DSD working again by clearing any pending commands and keyboard error messages with the left-blank key. The "99." command can then be used to disable DSD's normal command syntax processing. If the systems people have remembered to keep the DCN command overlay in CM, the DSD DCN command should work. However, there have been times when everyone did everything correctly, but DSD would not accept the DCN command.

Here is my proposal:

The current method of handling channel hangs is unreliable and is costing us downtime unnecessarily. My proposal is to install some code I have written (based on my MERITSS unhang the DDP mod, circa 1973) which makes MTR scan the channels looking for hung channels and either inactivate the bad channel and issue an errorlog message or put up a blinky message on the console B-display saying "CHxx HUNG full/empty."

The way it works is: every time through MTR's main loop it will check one channel. If the channel is inactive or changing between full and empty, it will advance to the next channel on the next cycle. If the channel stays active and the same empty/full status for 4096 consecutive cycles (or about 2 to 4 seconds), MTR then monitors the channel constantly for the next 16 milliseconds (8 milliseconds on a 2X PP). If there is still no change, MTR declares the channel to be hung and writes the channel number and channel full/empty status to a low core word (MHCL). If AUTODCN is enabled, MTR inactivates the channel with a "DCN". Later ISP will pick up the hung channel flag from low core and issue a blinky message to the B-display if AUTODCN is enabled. There will be DSD "DISABLE, AUTODCN." and "ENABLE, AUTODCN." commands to turn the inactivating off and on.

I have tested this code on the worst cases I can think of. MST sequential write never exceeds 100 consecutive MTR cycles of the same status. The STIMULATOR never exceeds 1000 consecutive MTR cycles of the same status.

556 BPI L-tape of 184000 character blocks never exceeds 300 consecutive MTR cycles of the same status. The Link test doing full speed writes to nonexistent PDP11 memory (which generates a bus timeout on each word) never exceeds 2000 consecutive MTR cycles of the same status.

SYSTEM MAINTENANCE: People and Procedures

Last Week's Systems Group Meeting - by T. W. Lanzatella

Our entire meeting, held at IPC to accomodate the crowd, was spent discussing the pros and cons of UNIX and VMS on the VAX's. For anyone who is interested I have three pages of notes which describe just about all the opinions expressed.

//////////

Callprg and Library Tape Changes - by M. Riviere

On June 30, Steve Reisman will be introducing System 2000 version 2.80 products as future Callprg packages. The new System 2000 set consists of the following products: PLICOB, PLICOB, PLICOB, PLIFOR, PLIFORL, PLIFORS, PLILGO, PLILGOL, PLILGOS, PLXCOB, PLXCOBL, PLXCOBS, PLXFOR, PLXFORS, PLXLGO, PLXLGOS, RWEXIT, RWEXIT, RWEXITL, RWEXITS, S2000L and S2000S. The new products will be introduced on the C74/172 and C720 indices.

In addition, Steve will be removing the Data Description Language (DDL) and the Query Update products (QU) from the C172 Library Tape. These products will not be available at UCC any longer.

Also on June 30, the entry for FETCH,M77 will be removed from both Callprg indices. The version of M77 referenced by that statement is now the current one.

The next set of Callprg and Library Tape changes will be taking place on July 14. The deadline for requests for that date is July 3, by noon.