UNIVERSITY OF MINNESOTA COMPUTER CENTER
Deadstart Systems Newsletter

18 June 1975 Vol. 1 No. 4

Send all comments, criticisms and contributions to the editor
T.W. Lanzatella

NOTICE OF CHANGES TO THE OPERATING SYSTEM

The following changes were brought into production on 1 May 1975.

Bruce Johnson installed a new message in DSD, VP BUSY, which appears in the I display whenever a transfer is in progress between the CYBER and the PDP-11. Bruce also changed DMQ and LDQ so that QT = PL is recognized on the command,thus allowing dumping and loading of PLOT type files.

Kevin Matthews modified 1AJ so that MS PRU's transferred by LDR during program loading are logged in MSUW in the control point area.

Tom Lanzatella implemented modification LINK28A from PSR 393. The mod fixes a bug in CDC's implementation of extended replication table processing in LINK. Without the correction, several RUN23 programs do not load correctly.

N.L. Reddy added more code to BATCHIO, this time to 1IO which further enhances the plotter interface. The program 1IO now treats PQ as a controller and searches the queue for PLOT type files. PQ is really a PDP-11.

Stuart Lenz implemented the following changes to MF501:

1. A test and message for empty input file with a new parameter on the control card, E, which directs MF501 to abort on empty input file.

2. A new parameter on the control card which directs MF501 to perform an internal tape request.

3. Repair various sundry documentation errors.

Bill Elliott added the following modifications:

1. A C option to TDUMP which allows dumping of even parity S, SI, L and F made tapes.

2. A new DSD display E,U which displays MAGNETS's UDT's.

3. An assembly constant was added to COMP3XD to accomodate a new version of CERTIFY. The constant changes treatment of the tape controller by COMP3XD.

James Mundstock added code to ISF so that the CALLPRG index is treated as a fast attach permanent file. Jim also made the following changes to CALLPRG:

1. The index file, SCR1, is now returned in all instances.

2. The L = lfn option now works on WRITEUP.

3. WRITEUP no longer writes EOF's on file OUTPUT.

4. CALLPRG now copies tapes to the auxiliary device SP.

Alan Johnston contributed the following changes:

1. A new version of the PP common deck COMPUPS which unpacks a control card to a character buffer. The current CDC version of the common deck has many errors and is not reentrant. The new version of COMPUPS will be sent to CDC as suggested corrective code against the current version.

2. A modification from PSR 404 (level 9) which fixes the use of the AM parameter (maximum field length) in the IPRDECK when it was not set to 7777.

3. A generalization of one character file types for DSD commands. This modification introduces a micro ≠QFNM≠ and a macro QINDER. The micro contains one character for each queue type. The macro generates table entries for DSD with the appropriate symbols and values for the tables. In the future, additon or deletion of queue types will require only a change to the micro. This modification enables the operators to get the plotter, batcher and XMIT queues.

4. Word 43B in the installation area of the control point area in PPTEXT is now tagged SOBW. The word will be used by future U of M modifications.

5. Bit 58 of SOBW is now used to indicate that all files with 74B ID attached to the job should be returned at the next job step. All SSJ = jobs with a parameter block set all created files with 74B ID and set bit 58 of SOBW. SSJ = jobs without a parameter block do not have this feature but LFM now sets bit 58 of SOBW whenever a SETID with 74B ID is processed. Additionally, the attach control statement function in LFM now sets this bit in SOBW. This used to be a way of getting passwords since whenever CONTROL aborted it left the control statement file entry in the FNT in tact with tracks released thus leaving the disk open to random examination.

6. The clear core function before program load, CPM function 72, now uses bit 59 of SOBW instead of LDCW, the loader control word. This change is for U of M generalization and apparent changes CDC may make in LDCW. Also, this function was moved to CPU MONITOR instead of placing it in the PP program CPM.

Marisa Reviere changed MODIFY, VFYLIB, VERIFY and LIBEDIT so that a Q is not output as first character of the OUTPUT file if the file is of time sharing type. Marisa also changed MODIFY to suppress page skipping when only one line was written on the previous page.

The following changes will go into production on Tuesday, 1 July 1975.

James Mundstock contributed the following modifications:

1. A modification to CPM which prevents user initiated rollouts of less than eight seconds. All user initiated rollouts of less than eight seconds are set to eight seconds.
2. A fix to 1AJ which repairs time limit error processing. 1AJ was reprieving time limit errors by adding eight times the validated time limit to the job.

3. A modification to CALLPRG so that if the fast attach copy of the CALLPRG INDEX is not available the indirect access copy is used.

4. A modification to CALLPRG which prevents the program from loading tape files to the device SP but attempting to purge the files from the default system device.

5. Program WRITEUP now outputs a message when a user requests a non-existent writeup.

T. W. Lanzatella added the following modifications:

1. Program 1AJ no longer allows internal user requests to load programs with an SSJ = entry point.

2. Program MODVAL no longer allows a user to execute a PASSWOR control card unless the user number which password is to be changed has password change permission. This change is necessary because certain malicious users were changing passwords of class user numbers.

N. L. Reddy has fixed the DISPOSE entry in FILES to be reentrant. Program FILES, like PFILES, has a look ahead feature. Previously, two sequential DISPOSE commands did not work correctly.

Bill Elliott modified the DENSITY control card to accept mnemonics HY, LO, etc. rather than numeric values for density. Bill also changed PFILES as follows:

1. The ACQUIRE and RETAIN control cards are now installed. RETAIN is the new name for FSAVE.

2. The PG parameter is now recognized on the DEFINE command. This parameter replaces the old FDEFINE command. PG means that the file will be purged before being defined.

Marisa Riviere has altered DOCUMENT to accept the symbols .PDP11 and .END as keywords indicating the start and end, respectively, of PDP-11 documentation. Marisa also changed the origin of program MODIFY to allow use of the Z parameter when MODIFY is called from a local file.

PROPOSED CHANGES TO THE OPERATING SYSTEM

Marisa Riviere proposes a modification to PFILES which would allow the user control word (UCW) field of a permanent file to be set by the control cards SAVE, DEFINE, RETAIN and CHANGE. Currently, this is possible only through internal calls to PFM. The utility of such a change would lie in the ability of a local version of CATLIST (for example) to group files according to the UCW field. The program RFM currently uses the UCW field for just such a purpose.

Alan Johnston contributed the following objections to the proposed change in the micro NMOT.

> As stated in the Deadstart Newsletter and Yuccs, there is a proposal to change the batch two character name in the NMOT micro from 'BC' to 'LD'. We believe that such a change should not be instigated for the following reasons:
>
> 1. Many users, including ourselves, are accustomed to the use of 'BC' as a site code for the parameters on control cards. Changing the site to

'LD' invokes confusion to users and to the parameters on control cards.

2. The KRONOS reference manual states that the central batch site is 'BC' (pages 5-16, 5-32, 7-29). Conflicts will occur with users who read the reference manual and what actually is.

3. Changing the batch site 'LD', which I assume means Lauderdale, is ambigious. There exists two I/O stations at Lauderdale, the Univac 1004 and the CDC on site equipment. Some users may become confused as to which place 'LD' refers to, and not know where to pick up their output.

4. The micro NMOT is used to generate a two character name which designates the origin of the job. The site codes are determined with the first character being the origin and the second the terminal identifier. But the central site 'BC' is always the same, since there is only one central I/O station at the present. The current symbol 'BC' is sufficient and is generalized for future central batch sites.

In the previous issue of the DEADSTART SYSTEMS NEWSLETTER, attention was brought to a small dispute between KCM and NLR over whether or not maintenance routines should be internally initiated and scheduled by KRONOS. Several discussions ensued and a formidable list of advantages of internal maintenance job scheduling was the result. The list, due to N.L. Reddy, is presented here.

ADVANTAGES OF SCHEDULER RUNNING MAINTENANCE JOBS

1. This is a natural extension of the functions of the scheduler. No "bending" of the operating system is involved.

2. The routines will be scheduled evenly, like one every 5 minutes, instead of running them all at once every 17 minutes. Errors will be caught sooner.

3. Far, far less overhead to the system. No rolling priority evalution, calling other PP's is involved. No disk activity is involved in bringing a routine to a control point, other than loading the routine. Procedure calls are very expensive.

4. Some of the routines need a different seed every run, so that they may test a different section of the main frame in a more random fashion. Only the scheduler can maintain and generate a new seed for every call.

5. In the procedure-oriented scheme, the routines are initiated by ISF (Initiate System Files). ISF clears out all resource files like ResexDF, ResexVF, etc. which hold information about current resource allocation. So ISF can be run only when there is no job in the system requesting or holding a resource. So if for some reason, the routines have to be restarted, it can be done only under "DIS" from the console.

   In the scheduler controlled scheme, the jobs will be run as long as the scheduler is active, in other words, as long as the system is up.

6. If the procedure file is lost in a disk crash, no maintenance job will be initiated.

7. If a routine does find a hardware problem, it can be allowed to end normally giving all the information in its output file.

   No restarting will be necessary.

8. In the procedure-call, the job has a funny name like 555AAE2, which does not mean much to operators, accounting or anybody else. This is true whether it is running at a CP or is in the rollout queue.

9. Maintenance routines are truly system-origin jobs, and in the scheduler controlled scheme they will be system jobs with meaningful job names. In procedure call, they are submitted jobs with a BATCH origin.

10. Operator will be able to control the running with DISABLE, MAINTENANCE and ENABLE, MAINTENANCE. The interval for calling the routines can also be set from the console. This is possible only with the scheduler.

11. If CDC ever implements the scheme under NOS, it is more likely to be in the scheduler than anywhere else. In fact, scheduler is the most natural place to put this stuff in.

12. If the rolled-out job under the Procedure Scheme disappears or gets dropped by the operator etc., re-starting is not easy. But the scheduler is not subject to these.

13. Adding a new routine to the list means a sysedit under the scheduler scheme. One has to go under "DIS" to replace a permanent file and submit it under the procedure scheme. During production, doing a sysedit from a card deck is simpler and less of an inconvenience to Operations then use of the console. Anyway I don't anticipate adding routines in a hurry.

14. Effort involved is not much, as all the work to the scheduler has been done, only DSD commands have to be added along with suitable changes to PPTEXT.

15. When we add the ECS test routine EC3 to the system, there has to be a request for ECS FL. If some user has got it tied up for a couple of hours, no maintenance routine would run during that period under the procedure-scheme.

    But under the scheduler-scheme EC3 would not be scheduled at all, if that resource was not available, and other routines would be scheduled as usual. When ECS became available, it would be scheduled in the proper order. This can be extended to other resources, if necessary in the future.

16. Procedure files are one of the most convenient tools in KRONOS, but they can never compete with a system routine, in stability, efficiency, inexpensiveness of system utilization, etc.

SYSTEM MAINTENANCE: PROCEDURES AND PEOPLE

A correction to the 16 April 1975 DEADSTART SYSTEMS NEWSLETTER: Please send a rough draft of all CDC PSR's to Betty Stahl. She will have it typed in final form, submit it and maintain a record of date published and reply.

The following is a control card responsiblity list compiled by Andy Mickel. The list is intended to indicate roughly what people are working with. Often one program (especially a popular systems routine) is affected by many people - not just the persons indicated.

CONTROL CARD RESPONSIBILITY LIST - ABM 6/25/75

[-C = compilers Ø = operating system]

J. Cosgrove: BUILD, S2000, PLIFOR, PLICOB

Ø W. Elliott: BLANK, DENSITY, LABEL LISTLB, NEXTVSN, PACK, PREVIEW, RESOURC, UNLOAD, VSN, EXAMINE, RETAIN, ACQUIRE

M. Frisch: FORPREP (with K. Frankowski), TXTEDT (with M. Riviere), ARTHUR (with C. Liedke), PERT66

-C D. Hamnes: BASIC, DISPOSE, MIMIC, SIMS, SIMULA

R. Hotchkiss: EMULATE, SAP, UMRPG

Ø B. Johnson: DMPCOR, DMPECS

Ø A. Johnston: CLEAR

-C H. Kurs: COBOL, FILE, SORTMRG, SORTMR3, CREATE, IXGEN, SISTAT, ESTIMATE

Ø T. Lanzatella: CKP, DOCUMENT, RESTART, SETDFL, SYSEDIT, PLONE (see also G.S.) EDIT, STRATEN, UPDATE

-C S. Lenz: MF501, MIXAL, UTALGOL, MIXBYTE

D. Lienke: APEX, DRESS, GPSS, LPKODE

S. Lin: OMNITAB

Ø K. Matthews: CALL, CATLIST, CHANGE, DEFINE, DAYFILE, DUMPPF, ECS, LOADPF, LOADSD, SAVE (see also G.S.), OPLEDIT, KRONREF, UPMOD

Ø R. Matthews: LINK, LOAD, MODIFY, MAP

L. May: ISIS

-C A. Mickel: LISP, PASCAL, PSCXREF, SNOBOL, SNOBOLC, SPRUCE, TYPESET

-C E. Mundstock: BATCHER, FETCH, FUTURE, MNF, PAST, WRITEUP, BASIC, FTN, RUN (with CDC)

R. Myster: UMTIMER

S. Nachtsheim: BUILD, COPYCL, KWIC, PLICOB, PLIFOR, S2000, S2000RW, TRIAL

Ø N. Reddy: ACCOUNT, BIN, COMPASS, DISPOSE, LIMITS, OUT, RENAME, ROLLOUT, SEND, SUBMIT, CHARGE, FAMILY, MODVAL, PROFILE, SETID, MODUP

Ø M. Riviere: ALGOL, LIBEDIT, TXTEDT

-C W. Sackett: APL

Ø T. Salo: AMEND, COST, PACKMS, REBLOCK

-C B. Stahl: MNF, TIDY

-C J. Strait: PASCAL, CNOBOLC, UNPAGE

B. Williams: XEDIT

S. Yen: BCTRY, all BMDnmX, CTAB, ISIS (with L. May), SPSS, all UMSTnmO

Ø GENERAL SYSTEMS (G.S.)

jobcard, APPEND, ATTACH, BKSP, CATALOG, COMMENT, COPY, COPYBF, COPYBR, COPYCF, COPYCR, COPYEI, COPYSBF, COPUU, COPYX, CTIME, DISPLAY, DMD, DMP, EVICT, EXECUTE, EXIT, GET, GOTO, GTR, IF, LBC, LDI, LDSET, LIBGEN, LIBRARY, LIST80, LOC, LOCK, LO72, MODE, NOEXIT, NOGO, NOMAP, OFFSW, ONEXIT, ONSW, OUT, PACK, PACKNAM, PASSWOR, PBC, PERMIT, PURGALL, PURGE, RBR, REDUCE, REPLACE, RESEQ, RETURN, REWIND, RFL, RTIME, SET, SETCORE, SETPR, SETTL, SKIPEI, SKIPF, SKIPFB, SKIPR, SORT, SWITCH, TDUMP, UNLOCK, VERIFY, VFYLIB, WBR, WRITEF, WRITER, ASSIGN, COMMON, RELEASE, REQUEST, STAGE, SUI, USECPU.

This memo about CALLPRG discusses five unrelated topics. Part I and the references to CALLPRG in the DEADSTART SYSTEMS NEWSLETTER of April 17, 1975 should be read first to understand the following parts.

1. NEW CALLPRG PROCEDURES

Please read the description about the CALLPRG procedures on the DEADSTART SYSTEMS NEWSLETTER of April 17.

I plan to implement the procedures that are described there by the middle of July. This will give you enough time to become familiar with the new procedures. If you have any questions or if you would like to have a meeting to discuss the new procedures, please tell me to arrange it.

2. PACKNAME FOR THE CALLPRG FILES

Most of the UCC CALLPRG packages have been moved to the SP pack to improve distribution of disk space.

The CALLPRG index (CALLPRG), the new version of the index (NEWCP) and the OPL that shows all the NEWCP modifications (CPOPL) remain on the STF pack.

3. CALLPRG INDEX MODIFICATIONS

I modify the CALLPRG index once a week. Every request submitted before Thursday noon will go on the system the following weekend.

To avoid introducing errors on the CALLPRG index cards you should test the new cards before sending them to me. The new index cards can be tested in a local CALLPRG index. The following example illustrates a new index card

test job.

```
CBF (INPUT, CALLPRG)
FETCH,OPL.
7/8/9
.LOCAL CALLPRG INDEX TEST FILE
OPL,TY = FETCH, UN = YZE6008, PN = SP, IA = OPL.
```

Note that for the test job, the CALLPRG index is your own local file and the job does not need to be run under the CALLPRG account number. Also note that at the time you test the new card, the file must be already saved, defined or written on tape, as it will be once the card is included on the UCC CALLPRG index.

4. THE CALLPRG INDEX AS A SOURCE OF USER'S INFORMATION - PASSWORDS ON CALLPRG FILES

The CALLPRG index is a public file and therefore any user that knows the CALLPRG account number can list it.

There is no harm in users obtaining lists of the CALLPRG index. However, the CALLPRG index is not an official source of information about what is available on the UCC KRONOS System. (CALLPRG may include some special private user's packages and some UCC staff packages still in debugging stage.)

The use of passwords on the CALLPRG files is meaningless, since the passwords are included on the list of the index.

5. PRIVATE PACKAGES AND CALLPRG

The CALLPRG index should include mainly UCC staff packages. CALLPRG can handle local CALLPRG indices that may be supplied by private users to access their own packages. The card a user must add to access a private CALLPRG index is:

```
GET, CALLPRG/UN = nnnnnnn.
```

This card has to precede the cards that reference the private packages. CALLPRG searches the local CALLPRG index to find a requested package. If the package is not included on the local index, CALLPRG proceeds to search the UCC CALLPRG index. If the user wants to suppress the searching of the UCC CALLPRG index, the local index has to include the parameter SW = . on the DEFAULT card.

In the case that later in the job the user needs to reference a UCC CALLPRG package which name is also included in the local CALLPRG index, he should insert a second card returning the local CALLPRG file. This will allow the CALLPRG program to scan the UCC CALLPRG index for the UCC version of that package.

Exceptions to include private packages in the UCC index can be made, if the need is really justified. (e.g., private packages used by very large groups of computer class students at an elementary level when the use of extra control cards may become confusing.) In these cases, please contact me for the arrangements.

M. Riviere
April, 1975

LETTERS TO THE EDITOR

To the Editor, Deadstart Systems Newsletter:

The purpose of this letter is to comment upon recent discussions concerning "policies" in the Systems Group. Originally I wanted to comment on certain specific proposed changes to the operating system (FSAVE and FDEFINE) but these issues have now been resolved, and I will instead address general concerns. Hopefully this letter will serve at least to get some things written down in the DEADSTART SYSTEMS NEWSLETTER, but more importantly to elicit comment from other members of the Systems Group (the Kronos Kronie Klan).

There are four sections following: example problems, stated policies, change, and documentation.

Example Problems

1. The DEADSTART SYSTEMS NEWSLETTER has become a valuable source of information concerning changes to the operating system. But the DEADSTART SYSTEMS NEWSLETTER has only existed since February, 1975. I would like to see printed in a future issue, a list (or at least a summary) of all changes to KRONOS before February, 1975. Examples include the infamous change of the TXOT symbol in PPTEXT. (And I just found out last week that PPTEXT itself is local to the U of M!)

2. SUBMIT as is currently implemented is not standard and the documentation of changes are not easily available. Just one control card of many, I suspect.

3. Our documentation efforts seem to be on track according to our original plans when we went to KRONOS. But it is very apparent that we need to review our scheme! (see sction on documentation)

These problems and others need to be attacked.

Stated Policies

Concerning the most recently stated policies from Systems Group meetings:

1. All proposed changes must be brought before a Systems Group meeting for open discussion and the proposed change should be explained fully and candidly in the DEADSTART SYSTEMS NEWSLETTER prior to the meeting.

2. There exist rules of etiquette for using the console on systems time.

3. Staff persons should keep all but a few procedure files on the Staff pack (PN = STF) and keep the total number of sectors used below 1 or 2 thousand sectors. PN = SP should not be used by staff for private files, and PN = SYSTEM (the removable pack) should be used for very long direct access files (such as OPL's).

Other policies are:

1. The current documentation scheme calls for trying to stay with standard KRONOS but information for any changes should be placed under WRITEUP, in SYSNOTES, and in the UCC NEWSLETTER. DOCMENTed files of standard routines should have ≡

signs in the left margin by the stated differences.

2. Changes for language processors are phased in by the FUTURE, current, and PAST hierarchy and no change takes place during a school quarter.

3. All CDC PSR's must be routed through Betty Stahl for coordination and followup.

4. And to resurrect an old opinion (which might become a policy), see the attached memo from Roger Gulbranson to Larry Liddiard from June of last year concerning System Initialization after Normal Deadstart.

Change

When we decided to go to KRONOS in November of 1973, we took a not-so-solemn vow to avoid another MOMS (the Mine, Ours, MACE, and SCOPE local system). MOMS was poorly documented and unwieldly. Now in May, 1975 after having made some substantial changes to KRONOS we should be consciously aware of preventing the evolution of a POPS (Polluted Operating System) of a similar nature - son of MOMS grown up.

Of course, certain changes are highly desirable in all respects as we customize KRONOS to fit the needs of our installation. But changes must be made with the knowledge of all the consequences, including:

1. Making the change in a way that fits into the unifying underlaying concepts of KRONOS (uniformity, internal documentation standards, easy to use, consistent in syntax,etc.)

2. Informing users of the change initially.

3. Documenting the change adequately and permanently.

4. Implementing the change via programming.

It may well be the case that a change which seems to be a natural thing to do on the basis of programming or execution efficiency may be outweighed by the factors of cost and bother of fully carrying out the change all the way through user services so that it will be used effectively by programmers. Since according to Steve Nachtsheim, the Information System Group has estimated that each new control card costs UCC about $2000 all told, we should carefully consider our actions before plunging in.

Half of the Systems Group is at Experimental Engineering and their experience with users is certainly different from that of the half at Lauderdale. We don't want to give users the view that 1) we have priorities for change not matching theirs and 2) we have time on our hands for implementing frivolous features.

On the subject of esthetics in change and systematic overall design of the operating system, Ben Sheiderman of the University of Indiana (who was here at Minnesota to speak in March) in a recent technical report entitled "Experimental Testing in Programming Languages, Stylistic Considerations and Design Techniques" states: "As the issues become more subtle, it is no longer acceptable for developers and implementors to make highly subjective and personalized statements concerning the worthiness of a particular language feature or stylistic technique. The rampant proliferation of new dialects or entirely new languages is counter-productive since it limits the sharing of programs. A concerted effort must be made to ensure that new features, dialects, languages, and techniques are truly improvements. Control cards for operating systems and utility programs could also be improved by proper experiments with users. As computer utilization

becomes more widespread, large numbers of programming amateurs in diverse disciplines will demand facilities which are simple to use."

Documentation

How do we introduce change and how do we expect users to find out information about these changes? What is your definiton of the term "users"? From my consulting experience, at least half of the users oriented to batch processing do not know what a control card is. The situation seems to be much better with timesharing users. We need to aim for a single source of information, kept up-to-date that contains at least pointers to the places for documentation for our system. The level of explanation in such a document has to be for the user who does not know what a control card is. This document can educate the user to remedy this deficiency - can you guess now? It's the User's Manual. I sincerely believe that when it appears the User's Manual will go a long way to prevent staff harassment by users wanting to know trivial information. This implies that the staff will be able to save time and simply point to the User's Manual for the answer.

But what do we do for interim documentation? We have to consider that question from the low-level user's point of view. Just think of trying to explain to a person, who knows how to program in an applications system, some aspects of the operating system. Right now if such a person wants to try out a new (for him or her) routine like LIBEDIT, the procedure followed might be:

1. Look in the KRONOS Reference Manual.

2. Look on the Control Card Index to find out if there have been any changes made to LIBEDIT locally, and if so,

3. Run a job with a WRITEUP, LIBEDIT.

This is perhaps requiring too much of ordinary users. Also DOCMENTed files under WRITEUP repeat information in the KRONOS Reference Manual. Jim Mundstock has therefore proposed that changes be changed à la the Minnesota Modification Method. For an example see WRITEUP,CALLPRG. In this list changes are classified as features or corrections (PSR's). This scheme would do wonders for WRITEUP,CATALOG, for example.

If you still think that the first procedure is adequate, consider the case in which we revise a WRITEUP file and there are old copies still extant in user-land. The user must do a WRITEUP. to determine a date for the file. For this situation, Steve Nachtsheim suggests that the path a user should take is as follows:

1. Reference a stable source of documentation. This is the User's Manual. Updates to the User's Manual are first announced in SYSNOTES and exist on file for a temporary period - so as to not clog the disks and because the update will eventually be printed as a handout. In fact, all information under WRITEUP not of a very detailed nature should not have to remain there indefinitely. Otherwise we will run out of storage. It should eventually appear in the User's Manual.

2. For any changes, the user could access a program which will answer the following question: List all the changes to COBOL since April 1. (If the query were for COBOL.)

   Steve indicated that in his opinion the DISPOSE writeup is good: For example, it lists what pages it affects in the KRONOS Reference Manual. He says that documentation of this kind should be written as if it could be the last paragraph of the appropriate discussion in the KRONOS Reference Manual. Now, Steve is on record as having made some of these suggestions in August 1973! Opinions anybody?

Andy Mickel
May 24, 1975

June 3, 1974

MEMO TO: L. A. Liddiard

FROM: R. L. Gulbranson

SUBJECT: System Initialization After Normal Deadstarts

The KRONOS operating system allows for a clean method of initializing non-deadstart routines and special files. This is done via the program ISF (Initialize System Files), and a special procedure file 'SYSPROC.' This procedure file runs as system origin, and at a system account number (377770 - 377777). These system account numbers are only accessable via a system origin job.

It has recently been suggested that SYSPROC will call other procedure files from non-system account numbers (i.e., staff account numbers), as well as possibly getting new program binaries from these same account numbers. Reasons for doing this are as follows:

1) It is inconvenient to run a system origin job to change part of the initialization procedure.

2) The systems people are responsible people and will not do anything that might cause system instability.

I believe that both of these reason are totally invalid.

As to reason 1, surely it is inconvenient. However, to maintain a system we must have controlled change. If we allow more than one person to maintain this procedure we may discover that things do not work as planned. These non-system files can be changed on-the-fly. A change at 2:00 p.m. may cause a deadstart at 2:00 a.m. the next day to fail. As with all things concerning tapes and permanent files, the only complete test is to run them under the correct account numbers to verify that they work. Since we do not allow untested binaries to be placed in the system why should we allow untested procedure files to be run under the all-allowing system account number?

With regard to the responsibility of the Systems Group, we should all remember that all of us have done some dumb things in the past. Even well meant changes have been known to fall flat on their faces within 1/2 hour of their inception.

Finally, I do not see any reason why the Systems Group should take a step backward with regard to system integrity. We should remember past mistakes, and learn from them. Given the opportunity to make a reliable, automatic procedure to perform system initialization. We should not allow this to be modified without the necessary testing procedures that have always been mandatory. If the System's Group cannot police itself we may find someone doing it for us. We should remember past occurrences of this concerning the deadstart tape procedure and the sysdeck.

RLG/js