UNIVERSITY OF MINNESOTA COMPUTER CENTER
Deadstart Systems Newsletter

21 April 1981 Vol. 7, No. 2

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

Kevin Matthews changed COMSSRU by defining a multiplier for the Cyber 730. The multiplier defines the speed of the CPU relative to a 6600 type CPU. The Cyber 730 has a multiplier of .7.

Don Mears added a new site to SUPIO. Site 35 now corresponds to SSRFC. Don also added a direct limit of 200 PRU's for site 24.

Marisa Riviere added code to RFM making the program suitable as a subroutine in a FORTRAN program.

Bill Sackett corrected PFPACK processing in PFU so that the new location of the catalog overflow bit in the MST is taken into account.

Paul Thompson corrected an error in USERS which caused certain types of timed-event rollout files to be mistaken for low priority input queue files.

Andy Hastings installed the following changes:

1) Armed with an auspicious correction from CDC, Andy will again try the AUTOPACK facility (see DSN 6,10 P. 77).

2) Andy installed some PSR 518 code which moves the ENTER command into CONTROL. This change corrects problems in control language loops. Kevin Fjelsted proposed this exact change but never got around to installing it (see DSN 5,21 P. 170).

3) A typographical error in the mod installing secure parameter processing was corrected. The error could have caused ROUTE commands with certain types of errors to loop to time limit.

4) Andy corrected a problem in ROUTE=. This is the version of ROUTE used when a UN, PW, CN or PN parameter is specified. Previously, the program did not work right if a ten character charge number or a project number greater than ten characters in length was entered.

5) A small timing problem related to secure parameter processing in COMCESP was corrected.

6) Program PREVIEW was altered slightly to correct some typographical errors and to speed up the program.

Steve Collins installed a new version of CCL which uses ACQUIRE to get procedure files (see DSN 7,6 P. 38).

PROPOSED CHANGES TO THE SYSTEM

The Cray Station Installed - by T. W. Lanzatella

In a previous proposal, I suggested that turnaround time for Cray jobs was too large and was a deterrent to convenient use. In order to improve turnaround, I suggested that we run the Cray Station, a package under development at Cray which allows a Cyber to talk directly to a Cray via NSC adaptors (see DSN 6,21 P. 180). The proposal was accepted.

I would now like to discuss some station implementation details. I will advise that we make a few changes to the station in order to tailor its performance to our site requirements. I would also like to point out some of the differences between the Cray service rendered via the station and via the CQDUMP/CQLOAD mechanism as far as users are concerned. Suggested changes are noted with [n].

The Cray Station is made up of three components:

1) A CF-program, CRSTAT, written in Pascal which supervises all traffic between the Cyber and Cray. CRSTAT is initiated by operator entry. The package drives a K-display which illustrates station activity. Once CRSTAT has been initiated, it occupies a control point and about 30K of memory.

2) A PP-program, CSD, which talks to a Cray-1 via NSC adaptor and transfers data to and from various buffers on the Cyber.

3) A CP-program, CTASK, written in Compass which has the special purpose of delivering files, once assembled, to their intended destination. CTASK has many overlays. Each overlay handles a different file format or destination. Two overlays handle coded transfers to or from the Cray. Two handle binary transfers. One handles input files bound for the Cray and one handles output (unit record) files returning from the Cray. Coded transfer means that an ASCII/Display code translation occurs. Binary transfer, also called transparent, means that no translation is performed. The roll of each overlay is called a task. The tasks which handle input

jobs and output files are called spooling tasks. The tasks which handle tape and permanent file transfers are called staging tasks.

When CRSTAT is initiated, the spooling tasks are also initiated. These tasks begin by searching the queues for files to transfer. If none are found, they roll out and search again later. When an input file is found CTASK asks CRSTAT to open a data stream to the Cray. The task then translates the job into ASCII and sends it across. For output files, the reverse happens. When a file is found, a stream is opened, data is read and translated back into display code and finally routed to the output queue.

CRSTAT is the hub of the Cray Station. The program manages all file transfers between the Cyber and Cray. This is done in a rather general way because of the different types of transfers which can occur. Actually, CRSTAT only manages a number of logical data paths between the two computers. Functional manipulation of data is handled by the staging and spooling tasks.

Apart from bringing the Cray on-line to the Cybers, the Cray Station provides numerous capabilities beyond the scope of CQDUMP/CQLOAD. The station serves as a file transfer facility. A Cray job which originates on the Cyber host can access tape or disk files on the Cyber. The transfer capability works in both directions. Tapes can be read or written from the Cray as can permanent files. Cray jobs can even execute procedure files on the Cyber.

Two commands which are part of the Cray Operating System, COS, are used to effect file transfers. The commands are ACQUIRE and DISPOSE. The ACQUIRE command is used to retrieve a tape or permanent file from the Cyber. The DISPOSE command is used to send tape, permanent or print (punch or plot) files to the Cyber.

In one sense, the general format of the DISPOSE and ACQUIRE commands is an important implementation detail because they need to be documented. Referring users to the Cray manuals for instruction is not an acceptable option. This isn't a reflection on the quality of Cray manuals. The manual is a reference document and does not give examples of usage or warn of non-obvious hazards. [1] I suggest that we start a new indexed writeup named CRAY which can serve as a repository for user documentation on COS. Writeup CRAY would be patterned after writeup CONTROL except that it would have complete (rather than abridged) information on each control statement.

For this discussion, I am concerned with only a small number of file transfers: input jobs moving to the Cray and output (print, punch or plot) files coming from the Cray, the spooling transfers. The other transfers are not important because they are invisible to users.

Currently, a Cray-bound job is indicated by using a MICR parameter on the job statement. The format of the job is: Job statement, User statement, Bin statement (optional), COS commands. A job like this would be placed in the input queue with a low priority and would eventually be removed by CQDUMP. Assuming that we continue to provide Cray service via CQDUMP, how will Cray Station jobs be distinguished from CQDUMP jobs? [2] I propose that we pick a new MID for Cray-bound via station jobs and modify the input spooling task to select jobs of a certain priority (corresponding to the new MID). A similar

change was made to CQDUMP when it was installed. The stock station uses selection criteria tailored to Boeing Computer Services which are not suitable to our site.

An alternative to this suggestion and one which may prove more versatile in the future was suggested to me by KCM. The alternative would be to remove from the job statement any indication of what machine the job is supposed to execute on. Instead, a new command would be invented, say EXPORT, which would have the following meaning. Whenever a EXPORT command is encountered, everything following the command would be treated as a job, packaged up and queued as per options specified on the EXPORT command. This plan has two advantages over the MIxx scheme. First, because of very limited space for job card processing in OVJ, not very much can be done as far as job administration. For instance, we may have a hard time with a nofrills job class on the Cray using the MIxx scheme. This plan would also allow plenty of expansion room for incorporating other remote mainframes.

The format of station jobs is slightly different from that of CQDUMP jobs. For the stock station, the job structure is: NOS job statement, User statement, COS job statement, COS commands. This would certainly be confusing and bothersome to any user who wanted to switch between the two Cray access methods. This structure also does not allow for an optional Bin statement. -[3] I suggest that we change the input spooling task so that the job structure for station jobs is identical to that used for CQDUMP. This change has the disadvantage of invalidating the Cray Station Reference Manual, however because of other changes related to output file routing, the station manual will have to be excerpted and placed as a writeup anyway. Writeup Service=Cray will be revised for this purpose.

As a warning, I should point out that job structure will probably change when our own Cray is installed. The reason is accounting and validation. When a Cray job is processed by CQDUMP, a Cray job and ACCOUNT statement is generated and prepended to the submitted job. The account number used on the ACCOUNT statement is the NOS user number with an eight character prefix. The ACCOUNT statement will allow a password but the Cray system at Mendota is not using them - an installation option. UCC will undoubtedly use passwords. Hence, Cray job structure will change so that at least ACCOUNT statements are required among the COS commands.

Output from the Cray can have print, punch or plot disposition. Ensuring that output files return to their site of origin is an important consideration. Jobs which run on the Cray can produce multiple output files just as jobs run on the Cyber. The COS-DISPOSE command is used to "route" any file produced in the course of job execution. The DISPOSE command has a forms code option (SF) and a TID option. These options are precisely analogous to their NOS counterparts.

In the case of CQDUMP, output files from card input jobs are returned to their site of origin by using an elaborate mechanism involving a fast-attach file (MERCURY). When a job is dumped to tape, a copy of the input file system sector is placed on the file MERCURY. When completed Cray jobs are read back from tape, the file is searched for an entry with an identical job name. If an entry is found, all routing information is obtained and the file is routed. The entry is then deleted. If the input job is a submit job, the resulting

output is not disposed but rather retained as a permanent file, named after the job, under the submitting user's user number.

The MERCURY mechanism is not acceptable for use with the Cray Station. Multiple output files present the greatest problem. We can never determine when to remove an entry since each job can spawn lots of output files. Other problems are that the mechanism is high in overhead and potentially unreliable. What do we do if file MERCURY is ever wrecked while twenty jobs are executing on the Cray?

An alternative method of disposing output from the Cray is to attempt to carry routing information along with every job. This method makes use of a property of all files which move between the Cray and Cyber. Each file is prefixed with a 38-word block called a data set header. The header is partitioned into various fields. Every field in the header is used currently and no space is available as an installation area. The header is a useful place to store job routing information because the header is replicated, in part, for every file transferred back to the Cyber. The problem is in finding space. We need 22 characters in order to guarantee that output will return to its site of origin.

By adopting suggestion 3, we are appropriating the job name parameter on the Cray job statement. We are saying that any Cray-bound job executes on the Cray under the same job name that it had on the Cyber. I see no operational disadvantages to this restriction. That's the first 7 characters. [4] To get the other 15 characters, I suggest that we appropriate the US option on the Cray job statement. The US-option is a 15 character parameter used to specify, using Cray terminology, a user number. The COS manual says this parameter is provided to facilitate site accounting and can be used in any way by the site. It need not have anything to do with accounting. In fact, job accounting is done with a utility named ACCOUNT. The ACCOUNT utility uses a 15 character identifier for the account number which is different from the US option. Using the US field will not interfere with our plans for job flow or classification - yet.

The 15 character US field would be used as follows:

Originating TID or UI	4 characters. If the job is a submit job, UI would be the user index hash.
Bin number	4 characters
Originating MID	2 characters
Job origin mnemonic ordinal	1 character
Bin card site	4 characters

By now you have undoubtedly noticed that this design has a rather precarious fit. We are making some dangerous assumptions about the future of Cray service if we stipulate that these changes are permanent. For now, we can provide a reliable, documented service. When we asked Cray to add an installation area to the data set header, they indicated that one is planned. We are not sure when it will be installed. However, when it is available we will have the option of removing the constraints on job name and the use of the US option.

The activity suddenly ceases and 1TA looks alarmed. It holds up a hand, motioning for silence.

PROGRAMMER: What is it?
1TA: A crash. 1RI is hung in PP6 (points to the side and moves in that direction).
PROGRAMMER: Let's clear the output register.
1TA (disturbed): Who are you anyway? Don't you know about drop tracks?
PROGRAMMER: Sorry, I guess I wasn't thinking.
1TA (examining 1RI): Doesn't look good (shakes head). Bit two to the forty - forth is picked on every eighth word.
PROGRAMMER: A bad load off ECS?
1TA: Yep. People shouldn't reside out there. It's a fast commute but too dangerous.
PROGRAMMER: But the users.....
1TA (preachy): Look, crashes like this cost a lot. You save a few nanoseconds here and there. Then something like this wipes those savings out in one stroke. It's insane!
PROGRAMMER: I think you're being emotional.
1TA: Maybe so. What's this accounting thing?
PROGRAMMER (they both sit down on a channel cable): Well, near as I can figure it, every time a person logs off, you dump all their accounting data to the account file.
1TA (nodding in the affirmative): Yea, as a continuation type message if they timed-out or hung-up.
PROGRAMMER: Right. Or if the system crashed. (1TA scowls) But if they recover, you pick up their accumulators where they left off and continue them.
1TA (annoyed): What do you want me to do?
PROGRAMMER: Well, the problem is that accounting has to go to a great deal of effort. If a person recovers, we don't want to charge them for the original session since that data is included in the final accounting for the session after recovery.
1TA: Because accumulators are continued.
PROGRAMMER (excited): Right, but if they don't recover, they should be charged for the session that was hung-up or timed-out.
1TA: Or was aborted due to a system software error. (PROGRAMMER scowls). Fair enough. So what's the big deal?
PROGRAMMER: Well, you could make our life a lot easier, it seems to me, if you didn't dump accounting for recoverable sessions to the account file until you drop them from SALVARE.
1TA: The recovery file?
PROGRAMMER: Right. Then, if someone recovers, their partial session charges never appear but, if they don't recover, you dump them to the account file before purging their recovery data.
1TA: So, when a person hangs-up, times-out...
PROGRAMMER (enthusiastically): or is dropped due to a hardware crash.
1TA (glaring): or software crash. I place them in the recovery file but issue no ASSR, USMS, USMT, and like that.
PROGRAMMER (unsure): Yea, I think those are the messages. I get confused.
1TA: Trust me (puts hand on Programmer's shoulder). Then, if they recover, I still don't issue any accounting but restart them just as I do now?

PROGRAMMER: Right.
1TA: And, if ten minutes expires and they haven't recovered, I issue the accounting at that time so they get charged for the unrecovered session. Then I drop their rollin recovery file as usual. (pauses, then stubbornly) What's in it for me?
PROGRAMMER (stammers): Ah, ah,....
1TA (smiling): Figures. Absolutely nothing, as usual. I'll consider it. Gotta go. This field length reduction request is gettin' pretty stale. (Leaves).
PROGRAMMER (shouting): Please give it some thought. It'll save some CPU time.

The curtain is lowered as the PROGRAMMER, looking crestfallen, lets the words trail off to silence.

ACT II

The curtain rises to show a room, opulently decorated with scarlet draperies, Louis XIV furniture, and a table, holding a sumptuous feast. Candles adorn the table and an ornate chandelier is suspended directly above it. In one corner of the room a huge pop-art orange crayon sits atop the Steinway grand. Seated at the table are PROGRAMMER and 1TA.

1TA: Nice of you to have me over.
PROGRAMMER: My pleasure. I wish to continue our discourse on accounting.
1TA: (eyebrows raised): I knew there was a catch --
PROGRAMMER (changing the subject): You certainly are consuming a lot.
1TA (hurt): Excuse me. Next time ask a zero-level overlay to dinner. Ever eaten with S2000? (pause) Do you see me criticizing your decor? I mean this pad is o.k. except for that (motions in the direction of the crayon).
PROGRAMMER: It was a gift. (pause) Now let's talk about accounting. Care for another Apple? (offers a plate)
1TA: No, thank you.
PROGRAMMER: When someone recovers...
1TA (holds up hand): We covered that.
PROGRAMMER: No, no. This is different. When a user with CHARGE set recovers.
1TA: (shocked): You use PROFILE? Now I've heard everything. (Gets up as if to leave).
PROGRAMMER: (shrugs): It wasn't my idea. But listen. If a CHARGE user recovers, their user number is entered in the account file...
1TA (sits again and wipes mouth with napkin): Sure, at the login.
PROGRAMMER (continuing): but the charge and project aren't.
1TA: So, pick them up from the partial session before recovery.
PROGRAMMER: We do, but wouldn't it be easier for you to reissue them to the account file, via ABCN, when the user recovers?
1TA (sipping from a wine glass): Yeah, I s'pose so. Before restarting the user we could issue a message by pulling the data out of the control point area. We could do it.
PROGRAMMER (happy): Great, now I just have to get the system group to approve these proposals.
1TA (in thought): Wait a microsecond! How many bytes is this gonna take? I can barely fit into those PP's now.
PROGRAMMER (reassuringly): Don't worry. No problem. We'll just shorten a few messages.

1TA (looking at watch): I better go. Almost time for the system to come up. When do you think that stuff'll be implemented?
PROGRAMMER (enthusiastic): By the end of the month.
1TA (chuckles): I'll look for it be fall. Thanks for the repast.
PROGRAMMER: Thank you for joining me. May I use your endorsement at the Systems Meeting?
1TA (opening door): Sure. (Leaves, then pauses and returns). Oh, by the way, on this system name thing.
PROGRAMMER: Not you too. Make any suggestion you please, as long as it's serious.
1TA: Of course. It seems that, taxonomically speaking, one should strive to select names that convey the nature of relationships between entities in the set to be labeled. What do you think?
PROGRAMMER: Do you get philosophical very often?
1TA: Remember that dump last August that you never figured out?
PROGRAMMER: (nodding and stroking chin): Yea....
1TA: Seriously, think of how streets are named. By number, letter, seasons, famous past leaders.
PROGRAMMER: I see what you mean. (Thoughtful) What do you mean?
1TA: They don't relate to function, but to relative location.
PROGRAMMER (puzzles): What does that imply for us?
1TA (shrugs): Nothing, it just sounded good. Name them after famous past mainframes - 6400, 6600, like that. (Leaves)

The CURTAIN comes down.

DISCUSSION TOPICS

Archive Enhancements - by R. T. Franta

A user has requested that the next version of ARCHIVE incorporate a feature to preserve the user-control-word when it is dumped and reloaded. I wish that the following points be discussed:

1) Selection of files to load or dump by user-control-word.

2) How and where the UCW should appear on the ARCHIVE output.

3) Other file attributes which should be recognized by ARCHIVE.

4) If and where UCW will appear on CATALOG of ARCHIVE tapes.

SYSTEM MAINTENANCE: People and Procedures:

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

TWL made an announcement in reference to the emergency SYSNOTE procedure. For lack of adaquate security, the SYSNOTE procedure can only be executed by approaching the operator and making the request in person rather than by phoning the operator.

The following proposals were discussed:

1) Jeff Drummond's proposal to establish a name for each computer provided by UCC was tediously discussed (see DSN 7,6 P. 38). First, the suggestion of planetary names was vigorously opposed by the executive committee. Colors were also rejected. The reason given was that names derived from planets or colors do not have a professional bearing. We should instead come up with names, possibly based on acronyms which reflect on the function of the computer. At this point, the meeting bogged-down. The whole topic was sent to committee. Membership was not specified but JJD will chair.

2) Steve Collins' proposal to change the CCL GET to an ACQUIRE was accepted (see DSN 7,6 P. 38).

3) Andy Hastings proposed changes to secure entry of passwords was accepted but we stipulated that the change should go in when the default families are given names and all time sharing is moved to the C730 (see DSN 7,6 P. 39).

Bill Wells' discussion topic was taken up with nearly equal number in or not in agreement with the approach. Nothing was resolved, however, the Xerox 9700 committee will decide whether to change the direction of the X9700 utility design.

Larry Liddiard announced a job classification change for all civil service positions reporting to him. John Sell spoke briefly about the building modifications which are planned for the Cray installation. Larry also took comments on a draft copy of a letter to be presented to the Computer Advisory regarding our intention to purchase a Cray-1.

//////////

Callprg and Library Tape News - by M. Riviere

There are no Callprg index or Library Tape changes scheduled for April 28. The next set of changes take place on May 19. Requests for that date should be submitted before noon on May 7.

//////////

MERITSS (Cyber 720) Deadstart Dump Analysis from 4/1-4/16 - by B. E. Blasing

Saturday 4/4, 01:01 No Dump
A power fluctuation downed all three systems. The PMS file had a linkage error when a deadstart was performed. This was fixed by RAW and the system was brought back up.

Saturday 4/4, 06:50 DD1
A power fluctuation caused 1TD to hang. A level 3 deadstart was attempted to clear it. The level 3 failed due to an ECS error.