ARTICLE FOR HP 3000 USER'S GROUP NEWSLETTER

At the Baltimore User's Group Meeting, questions were submitted to HP concerning the operation of, and enhancements to, the HP 3000. Responses to those questions which deal with how the system works, how to work around an error condition, what an error message means, etc., are given below. HP's response to questions regarding enhancements or new products are either included in the User's Group Questionnaire or are considered confidential information which we do not feel comfortable releasing to our User's Group (and thus potentially our competitors).

SECTION I. Questions on MPE/SYSTEM

1. Some computers log disc hardware/software retries to enable the user or CE to monitor the error rate of a particular disc. In many cases preventative maintenance can be performed before the disc actually fails and crashes the system. Since the Series II has memory chip error logging, would it be possible to expand this concept to other physical devices?

   [Unlike memory, the system devices return error conditions via the controller to the I/O system in MPE. If I/O error logging is enabled, such errors are recorded in the current log file for subsequent listing using the system Utility LISTLOG2. Note that each error is listed, and no analysis performed.]

2. A tape read error during a RELOAD is absolutely fatal, even though only one file is usually involved. Why can't error recovery be included to:
   1. attempt several rereads, and
   2. if the rereads fail, purge the affected file from the directory and go on to the next?

   [During a RELOAD, tape parity errors detected within a user file actuates a purge of the file and a forward space to the next file. Currently, no retries take place but we are intending to perform this function in the near future. Realize that a parity error occurring within the MPE part of the RELOAD will be fatal....always.]

3. OUTCLASS parameter on JOB card. Can this be used to get all output to a file other than $STDOUT? We have not been able to make this work, but the documentation in the manual indicates this should be possible.

   [The OUTCLASS parameter on a :JOB commands provides a means to define $STDLIST for a job. The documentation is not <u>too</u> specific in places - in summary, the device specified must be configured as a serial output device.]

4. A Series II from CX change you cannot drop the outfence to zero in the Series II machine, to print priority one files. Why not print => outfence, and allow the user to put priority zero on his file command for complete deferment of output?

   [The feature to enable users to completely defer output by specifying an OUTPRI=1 was considered to be the simplest mechanism, enabling an operator to distinguish between User and System deferments.]

5. LOG files. Why is old log file not reopened and closed properly after a crash? At least why not a utility program that the System Manager can direct toward the improperly closed file? How does the System Manager reset the numbering of the log files?

   [Log files are not automatically cleaned up after a system crash because of two reasons. First, it is not at all obvious as to which was the last open log file and secondly, the internal structure prevents a simple file open/file close mechanism. We recognize the suggestions to provide this facility, plus a reset of the log file number by the System Manager and both will be implemented.]

6. LISTF: Why do LISTF options 0 and 1 not indicate which files are currently in use, as the manual says they should?

   [A :LISTF with option ≥ indicates by an * which files are currently open.

   Option 0, disagrees with documentation, in not doing the same. The reason is that we decided not to access every file label in order to detect an open file. Disc access overhead is thus the trade-off. If it is universally felt that it should be done, it shall be, otherwise the documentation will be updated.]

7. What is the remedy to the following message on the System Console: (3000 CX)

JOB OVERLOAD, TYPE 4

("no job process count table JPCNT entry available" says the Operator's Manual.)

["Job overload type 4." It appears that this would occur having the job LIMIT set to the maximum configured limit, executing the maximum number of jobs plus at least one HIPRI job. To remedy this, keep the operator established job LIMIT less than the total configured limit with a negative adjustment allowing for the maximum number of concurrent HIPRI jobs.]

8. Is there any way to configure the Card Reader Driver IOCDRDO, so that it will accept punched cards in COLUMN BINARY CODE FORMAT (supported by the hardware with Operating Mode 1)? If not, is there any other way to read binary cards?

[Would need to change Filesize and Driver to do binary open.]

9. Is the Series II multiplexer limited to a max of 2400 baud?

[Yes.]

MPE/SYSTEM

10. I'm running RPG and I get spoofles which go over the 32 extents which are normally allocated. Therefore, my report just quits printing prematurely. How can I stop this from happening?

[The only current solution to prevent reaching the 32 extent limit on spoofles is to reconfigure the system and increment the "# of sectors per spoofile extent". Note that the document is incorrect - there is no upper limit of 1024.]

SECTION II. Communications

1. Is HP considering software to support an IBM link that allows a 3000 to appear like a HASP workstation?

   [HASP workstation is in investigation phase.]

2. European Modems: Compatible with 3000? Will distributed processing work in Europe?

   [Distributed processing project is in development and will be supported in Europe.]

3. On Series II HP 3000, will the running of 3780 Emulator degrade the system as it does on Series CS when it is locked in memory while you wait online for mainframe response?

   [No, a different line driver is used on Series II.]

4. How far will terminal I/O advances go? Many smaller, less powerful systems have faster I/O than the HP 3000. What are your objectives or goals in this area? Some terminals (264X) will go 9600 baud. I've heard rumbles of 19200 baud in the future. Is this high speed I/O being considered for the HP 3000?

   [High speed terminal I/O is considered an important area for HP 3000 enhancement, and several projects are in development phase.]

SECTION III. Data Base Management System

1. Does HP plan in QUERY to chain detail data sets based on a search item?

   [There is no short term plan to modify QUERY.]

2. When will IMAGE be improved to lock only records for update, put, delete? The IMAGE enhancement allowing "read only" accesses not to lock the entire data base was great. Set locking is now under consideration. Locking only particular records is the next logical step. Is this possible? How soon? This is absolutely necessary to make a truly online system run efficiently.

   [There is no plan to add record locking at this time because it would require major modifications to IMAGE. Data set locking is a simpler enhancement, and it is currently under investigation.]

3. IMAGE UTILITY ERRORS: Should display what the error was, i.e., wrong # at data sets. The octal address of where the error occurred is of little use to the customer...he doesn't have the source code/compile list to find what the error was. This problem has cost us many hours in situations where the manual does not indicate whether the proposed change to the schema is allowed or disallowed, especially in common situations such as adding a data set, changing set type or changing set linkages.

   [In some cases, these errors are associated with unsupported capabilities. Check with local HP support personnel.]

4. IMAGE SCHEMA RESTRUCTURE: Experience (a very costly teacher!!!) has proven that many seemingly simple data base instructural modifications are prohibitively costly due to the lack of utility flexibility (i.e., DBLOAD/DBUNLOAD) or sequence/occurrence (i.e., add/delete) sets in schema.

   1. We feel it is a must that these considerations be explained in the Manual.
   2. We feel the vendor should demonstrate greater interest in making the product it has provided more human engineered in terms of the implications of restrictions in real life situations.

   [An attempt will be made to make this information more visible in future manuals. The next enhancement release will provide a utility for quickly modifying password and security scheme.]

5. IMAGE DATA INTEGRITY: In the area of system/product enhancement, it is felt that the subject of Data Integrity subject to system reliability...i.e., the cost of system crash in a data base environment, be reviewed. We feel an integral function to data base maintenance (add/delete/update) should be to log these modifications to a non volatile media (tape/diskette). A major limitation in the approach to user designed/written logging systems would be the inability to interface to QUERY.

   [Transaction logging and recovery are currently in development.]

6. Empirical evidence indicates that QUERY locks the data base while performing a find operation without regard to the mode in which the data base was opened. The implication being that system management while attempting to provide optimum system response concurrent with system accessability is faced within our case an online order entry process accessing the data base in Mode 1 which can be locked out for up to 10 minutes by a user opening the D.B. in mode 5 and performing a serial read.

   1. Why lock the D.B. for a FIND operation?
   2. Why lock the D.B. if your mode of access doesn't require it?

   [QUERY has not been modified to include the latest IMAGE enhancements. These enhancements are currently under consideration.]

7. I have a problem with QUERY/IMAGE - Is there any way to allow different users to update data, find information and report information and report information concurrently? I'm using modes 1 and 5. Mode 1 for update, mode 5 for finds and reports.

   [Modes 1 and 5 allow concurrency if used in a user written program; QUERY enhancements are currently under consideration.]

8. IMAGE schemas have to be changed. How about a DBLOAD program that will load data to a changed schema? (Additions of data sets should pose no problems.)

   [A restructuring enhancement is possible but it requires substantial effort. These enhancements are being considered.].

9. Is there any reason HP cannot write and support a general purpose program that will allow a user to modify a search item in a detail data set?

   [No, however this can be done simply with two IMAGE calls today and it is low on our enhancement list.]

10. Is the data entry language (DEL) project continuing? What enhancements are planned? When can we expect new releases, and what, if any, plans are there to support other than 2640 terminals?

   [Yes, work is continuing on improved data entry facilities. Direct entry to files is being considered.]

SECTION IV. Languages/Utilities

COBOL

1. Will HP improve the object code generated by COBOL? For example, a figurative character comparison always generates at least 21 words of code including a PCAL when 12 or less are sufficient for equal lengths strings. There are more common statements that need simple optimization. (We have COBOL Ver. B).

   [COBOL object code optimization will be released through MIT 1646. The constructs that will produce more efficient object codes include:
   1.1 Character comparisons of the same length,
   1.2 Literal moves,
   1.3 COMP moves of the same size and characteristics.]

2. When can we expect system intrinsics which will interface with COBOL and the provision of good usable examples in the manuals?

   [The COBOL-74 compiler will be able to interface with system intrinsics directly.]

3. When can we expect a COBOL compiler which will produce compile time messages, rather than just ignoring the statement?

   [We need an example that the COBOL compiler ignores an erroneous statement without producing a compile time message.]

4. I understand that the COBOL project is alive and well. When can we expect the next version and what will be the improvements? When can we have ANS compatibility and improved debug features?

   [We are hard at work on a COBOL with ANS 74 compatibilities and the debug feature.]

BASIC/FORTRAN

1. Is there a way, besides using an SPL subroutine, to get the PARM value from the RUN command by FORTRAN and/or BASIC programs? An example in the FORTRAN manual indicates this may be possible but does not actually show how to do it.

   [Compiled BASIC programs are intended to execute exactly as they did in the BASIC Interpreter. For this reason, no new features were added for the singular use by compiled programs. Since it would always be necessary to input programs which are intended to be compiled through the Interpreter, any new language constructs must have equally valid and equivalent meanings in the BASIC Interpreter. This requirement rules out some ostensibly "nice" extensions such as programmable use of control-Y. To permit programmable control over this function would necessitate giving up the debugging capabilities which the Interpreter offers.]

[The compiled BASIC programmer cannot make use of the PARM because the PARM is already being used for another purpose. See the BASIC Compiler manual, p. 2-1.]

[There is no way to get the PARM value in FORTRAN.]

2. Valid need exists for BASIC Interpreter to have the equivalent of the EDITOR's K-files. Feasible?

   [Since the BASIC workspace (roughly equivalent to the EDITOR K-file) is currently maintained in the program data stack rather than in a file, a disc back-up mechanism would impose a significant overhead which would have a very noticable affect on the BASIC response time. For this reason, we feel that a reliable back-up mechanism is not feasible.]

4. BASIC-CONVERT: We have found the bug which permits a "number" like "OA" to be converted to a numeric zero with no error branch taken to be a problem. Is this now being corrected? If not, why not?

   [With regard to the "OA" bug: this is not a BASIC bug. It was first discovered by FORTRAN programmers and was a problem in the Compiler Library. The bug was fixed in Compiler Library 32211C.04.02 dated July 12, 1976.]

5. BASIC-CONVERT: We have had, under some circumstances the conversion of a number to a blank. How can this be possible? Is it a known bug?

   [Has a bug report been submitted? We are not aware this problem has occurred.]

6. Are there any plans for expanding I/O option? e.g., READ USING (at INPUT USING) similar to PRINT USING? Is there a possibility that BASIC will ever be [updated] ?backdated? to use FORTRAN type input/output formatting, as is done on the HP 9830A?

   [There are no plans to incorporate fixed-field or formatted input into BASIC at this time. Nonetheless, we will keep this suggestion on file for consideration with other useful commercial features.]

ETC. (SORT, FCOPY, EDITOR, STAR)

1. STAR: Will STAR be available on Series II? If not, why not? Is there a convenient way of converting ASCII files to STAR files? If currently available, could a subroutine be implemented, callable from any language?

   [We have no plans to make STAR available on HP 3000 II, because of its low usage and generally inferior specification relative to other packages available. Users of STAR are encoureged to contribute utilities to the Users Group Library.]

2. By what date, if at all, does HP plan to enhance EDIT/3000 to include the type of capabilities used at ITEL leasing?

   [Ross Scroggs has contributed an intra-line editing feature to the contributed library which is usable through the PROCEDURE command of the Editor.]

3. What is HP's response to Ross Scroggs' fixes and new features?

   [Editor A.5.01 contains many fixes, including those from Ross Scroggs of Itel.]

4. FCOPY
   a) Why must the program loop forever when copying a variable length record of length zero as produced on the LOG file when there is a system failure?
   b) Why can one not space forward in a variable length record file?
   c) Why can I not fool FCOPY by using a file equation and MR to space forward in a log file?

   [a) Has a bug report been submitted? We are not aware of this problem.]
   [b) FSPACE spaces the blocksize when record is variable, and not record size, viz.

   ```
   BUILD X; REC = -80, 10, V, ASCII

   LISTF X, 1

          REC = -800 BF = 1]
   ```

   c) Probably because of b) above.]