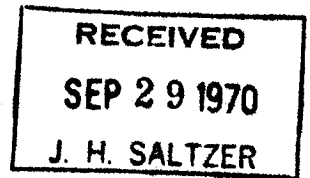




Massachusetts Institute of Technology
Information Processing Center
Cambridge, Massachusetts 02139



To: Wes Burner
From: John M^cManus *JM*
Subject: Multics Software Reliability
Date: September 24, 1970
CC: Multics Administrative Distribution

Realizing that September 15 has come and gone, I started this memo as the "second and final report of the Multics Software Reliability Committee"; but then realized that there would probably not be a consensus of opinion on the progress toward making Multics' software as reliable as required for a service grade system. I decided to express only my own personal, biased views and let the other committee members disagree in separate documents if they desire. This critique of our progress follows point-for-point the committee's report of July 20, 1970.

MSRI

MSRI was a good idea which was impossible to administer. It consumed a large amount of effort, caused a lot of hard feelings and conveyed only slightly more information than Roger Roach's weekly "black-white charts." A side benefit was the list of alleged bugs and suggestions. I recommend that a modified version of this softbug-info file be maintained by User Services similar to the OS bug list which they currently maintain. MSRI will no longer be computed.

Software Improvement

A. System Testing

1. Hardcore/Softcore: There are some encouraging indications in this area. The best example is the new directory control which was thoroughly pre-tested and was installed with no major problems. We now have regularly scheduled time on the service machine which should facilitate additional testing. The problems (8 crashes in one week) with the tty dim indicate that we still have a long way to go, however.
2. Commands: Progress in certain access of the command system is not as apparent as in the hardcore/softcore area. User control is cited as an example. Although better test procedures for PL/1 are now available we are not using them on a regular basis. PL/1 installations are still handled by General Electric.

B. Communications

1. System/User Communication: This has improved by using the info file mechanism and by having mail sent to consultant IPC. The check_info command will help even further. We still need two additional forms of user communication:
 - (a) Advance notice of changes
 - (b) Hardcopy summary of changes
2. User-problem/System Programmer/Resolution Communications: The loop has been shortened even though the answers are not always satisfactory. Softbug-info helps by giving the illusion that problems are receiving attention.
3. IPC/MAC Communication: Worse than ever at the management level. The basic problem is that everyone has too much to do and no one is forcing us to communicate. Certain areas of responsibility (access, quotas, obtaining source, priorities, etc.) still are undefined.

Summary

Although Multics software reliability as viewed from these goals discussed in the July 20 report, does not seem significantly better, other indications seem to contradict this:

1. Multics is still offered (and used) as a service.
2. Overall reliability for the past three weeks has improved from the previous three weeks.

The first reaction to these conflicting analyses is that reliability has improved in spite of the committee - and this is true. It is apparent, however, that we still have a long way to go. We must also begin to be concerned with more than reliability, especially useability and availability.