PROJECT MAC

February 19, 1975

Computer Systems Research Division

Request for Comments No. 69

CSR TASK REPORT
by David D. Clark

The attached report covers progress of the Computer Systems Research Division of Project MAC in the period 1 October 1974 to 31 December 1974.

Certification Tasks.

I. Removal of Linker from Ring 0.

The status of the user ring linker is that it will be installed, perhaps as early as this April, as part of the system which contains the mechanism to pre-link the system libraries. This pre-linking will eliminate the special case pre-linking of the user ring linker, which will make the mechanism less complex. Drew Mason continues to study the linker, discovering improvements which could be made to the code itself, but we have suspended major development activities on the linker until have an installed version to modify.

II. Removal of Name Space Management from Ring 0.

Dick Bratt has completed coding and checkout of a version of his proposed KST manager which splits the function into two components, but which leaves both of these components in ring 0. This implementation demonstrates the validity of his proposed KST restructuring, without requiring the modifications to all those programs which use pathnames inside ring 0. The final step of this project, moving the name space manager into the user ring, will require that all of these ring 0 programs using pathnames be modified so that instead they use the segment number of the directory. Allen Luniewski has been investigating the magnitude of this job, and has determined that in almost every case the modification required to the ring 0 procedure is trivial in nature. He has published an RFC which lists all of the required modifications in ring 0.

III. Removal of the Storage Hierarchy from Ring 0.

The removal of the linker from ring 0 and the removal of name space management from ring 0 can be considered the first two steps in restructuring the file system. Drew Mason is considering the next logical step, the removal of directories from ring 0. Drew has proposed that segment names be managed in an outer ring, but that ring 0 still contain the concept of the "parent" of a segment, whose extended ACL can be used to control access to the attributes of the segment. This particular approach seems to eliminate many of the problems associated with this restructuring.

IV. Support of User Defined Object Types.

A directory can be considered as an object defined in terms of a lower level object, the segment. If directories are moved to an outer ring, the mechanism supporting them could be generalized to support other new

kinds of objects. Doug Hunt and Phil Janson are both interested in devising a mechanism which allows the user to define new types of data objects which can be manipulated and protected as an entity. Such mechanisms have been proposed in the context of systems using capabilities for granting access, but not into an access control list (ACL) environment. Doug is attempting to do this, while Phil is attempting to devise a felicitous combination of the two mechanisms.

V. Restructuring of Page Control.

Andy Huber is continuing his research into possible ways of decomposing page control into a number of cooperating processes. It is hoped that he will soon have available a description of his version of page control coded in the GSPL language.

VI. Restructuring of Traffic Control.

Dave Reed is continuing his research in the restructuring of the Multics traffic controller. He has proposed that the traffic control function should be implemented in two levels. The bottom level should multiplex the processor among a small bounded number of processes; a higher level should multiplex this small number of assignable processes among all the processes existing on the machine. In his scheme, the communication of messages between processes is done using segments, so the standard file system access controls will apply to interprocess signals. Dave recently published an RFC which describes this structure. His current plan is to implement an experimental version of this bottom level process scheduler.

VII. Restructuring of the Answering Service.

Warren Montgomery is continuing his study of the question of what components of the answering service must, in principle, be certified. He has concluded that actuating a process is analogous to entering a new domain; this insight will lead to a simpler view of both. Hopefully, a document describing his current thinking will be available soon.

VIII. Fast Processes in Ring 0.

Bob Mabee has completed the implementation of an experimental environment which provides fast inexpensive processes of limited capability within ring 0. We are currently awaiting the publication of an MTB he has written which describes this mechanism. He has completed the implementation of the tty_dim interrupt handler as a fast process, and the resulting system seems to run with acceptable performance.

IX. Multitasking in the User Ring.

This project, which provides a user ring environment in which the user's process is shared among several tasks, has been proceeding very slowly due to lack of time. As was noted in the last task report, Doug Wells has completed the implementation of a version of user ring IPC which allows several tasks to abandon the processor by calling block. Currently, he is testing this implementation by incorporating it into an experimental version of the Network_Server process. We will, as time permits, use this mechanism to experiment with cleaner ways to deal with quits.

X. A Methodology for Designing Certifiably Correct Computer Systems

Rich Feiertag is proposing a method for designing complex computer systems that will make them easier to certify as correct. He is currently developing a Ph. D. thesis proposal in which he outlines a methodology for determining what functions go in a kernel and how the kernel itself should be structured. His intention is to design a secure system as a demonstration of the validity of his approach.

XI. Study of Multics System Initialization.

If one is to certify that a system works correctly, clearly one must begin by certifying the "initial state" of that system. For this reason, it is very important to understand how the Multics system initializes itself. The current initialization procedure is relatively unstructured and confusing, and is clearly not amenable to certification. Allen Luniewski is commencing a study of this component of the system, in the hopes of proposing an initialization strategy which is simpler, less prone to error, and more easily understood and certifiable. His initial approach is that much of what is now considered initialization ought rather to be considered as reconfiguration. What follows from this

approach is a decomposition of initialization into two phases. The first phase involves getting a minimal Multics up and running on any configuration, the second phase involves a series of reconfigurations based on the contents of the configuration deck. The advantage of this strategy is that all of the reconfigurations run in a complete operational Multics environment, which is much easier to understand than the partial and ever changing environment presented at various stages of initialization. Further, since the minimal Multics is essentially independent of configuration, it can be largely generated as the system tape is created.

XII. Study of Effect of Security on Reliability.

Harry Forsdick is beginning a study of the relationship between the security of a system and the reliability of the system. In the Multics system it is presumed that a system failure may have an unknown effect on the security status of the system; this is the reason that the system is shut down, salvaged, and restarted after every unexplained system failure. It is not obvious, however, that security is directly dependent on reliability. If it were possible to determine that certain classes of computer failure could not influence the security state of the system, then the two functions would have nothing to do with each other. More strongly, it is possible that a highly reliable system contains mechanisms which are not desireable from the viewpoint of security. For example, one way to increase the reliable storage of information on a system is to make several copies of that information; many copies, however, increase the probability that the information may be insecure. This is a new research effort; a better developed statement of the problem should develop in the future.

XIII. Multics Performance Benchmark on Development Machine.

The standard Multics performance benchmark developed by Roger Roach cannot be used on the development machine because it takes far too much time. Thus, some development tool is needed to evaluate the performance of new systems. Drew Mason is attempting to produce a version of the benchmark which is reduced in scale and which shows a high repeatability, when run on the development machine. By reducing the number of processes in the metering run, and fixing a bug in the absentee process environment, Drew has produced a metering run with improved characteristics. This modified metering test is described in an RFC currently in preparation.

Technology Transfer and Network Related Tasks.

I. New Version of IMP DIM.

Raj Kanodia has coded and tested a new version of the IMP DIM, which is essentially ready for installation now. This new version fixes certain bugs, and provides certain new features, the most important of which is an increase in throughput for the transmission of large network messages. The change which produced this increased output rate was a modification to the buffering strategy such that large messages were placed in large buffers rather than being subdivided into many small buffers. The output rate for a single process increased from nine kilobits per second to twenty-seven kilobits per second as a result of this change, and the maximum total output rate for the system rose to 40 kilobits per second. An RFC has been produced which details this particular change to the IMP DIM.

II. Modification to Host-Host Protocol.

Raj Kanodia has proposed a modification to the ARPANET Host-Host Protocol which allows for detection of lost messages while increasing the bandwidth on a single link. The modification involves the redefinition of a field in the message headers so that message sequence numbers are available to the receiving host; these numbers may be used to detect and request retransmission of a lost message. If this scheme were adopted, it would extend error recovery capability to the case in which more than one message was outstanding on a given link at any one time, which would increase the rate at which messages could be sent reliably through a single link. An RFC has been prepared which details this proposed modification.

III. Production of MPM Network User's Supplement.

As a result of a great deal of work on the part of Ken Pogran and others in the Network group, a preliminary version of the Network User's Supplement has been prepared and has been distributed to members of our division. Comments received on this preliminary edition are being incorporated into a version which will be distributed within the next few days to a larger set of interested parties. Our intention is to issue an "official" first edition sometime in the middle of February.

IV. Implementation of RSEXEC on Multics.

Jerry Rudisin is beginning a bachelors thesis which will be the implementation on Multics of an RSEXEC Server. Our intention is to produce a fairly complete server,

which will handle the file transfer requests as well as the other more simple RSEXEC functions. David Clark has produced a document, currently in draft form, which proposes an implementation of a user RSEXEC for the Multics System. The implementation for the user's interface is not immediately possible, because it depends on the installation of Dick Bratt's KST manager. The successful implementation of these RSEXEC facilities would give a Multics user the capability to view all of the files in the systems on the ARPA Network which support RSEXEC as belonging to one large hierarchy. In particular, one would be able to link to a file on any system, one would be able to put a directory on any system in one's search rules, and one would be able to change one's working directory so that it was on any system on the Network. As an example, this would provide a very easy way to use both the development and service machines simultaneously.

V. Installation of Network Software in Network Library.

Over the past several months, we have been moving the various user ring components of the Network support software to the Network library. This move is now essentially complete. This will make it easier to transmit to the Multics at RADC, Rome, New York, a complete set of the software required to bring their s stem onto the Network.

VI. Proposal for Extensions To iox_ Interfaces.

For purposes of experimentation with new I/O capabilities and of interfacing to the ARPA Network, we have felt the need for a standard I/O interface with more capabilities than those provided by iox_. In particular, we have the need for a variable byte-size interface, and a variable delimiter capability. We feel that it will be possible to include a standard representation of these capabilities within the iox_ framework without compromising any aspects of the currently defined interface. Dave Clark and Doug Wells are currently preparing a document with a proposal for some experimental extensions to iox_.

VII. Preparation of Network Program Logic Manual.

Ken Pogran, the editor of the Network PLM, has been prevented from working on his task by several other projects which have occupied all his time, especially the preparation of the NUS. Hopefully, these tasks will be completed in the near future, so that he can once again devote some amount of time to the PLM.

VIII. New Version of User TELNET Command.

Doug Wells is just installing an updated version of the telnet command which fixes certain bugs and removes from the code various obsolete features which we do not intend to support in the officially documented version of the command.

IX. New Version of The User FTP Command.

Doug Wells is currently preparing for installation an official version of the user_ftp command, which allows files to be moved from one site to another using the File Transfer Protocol. An unofficial version of this command has existed for some time but is not appropriate for installation because it does not deal as well as it could with various messages which are sent to it as part of the file transfer procedure. The new user_ftp command is the one which will be officially documented in the distributed version of the Network User's Supplement.

X. Implementation of File Access Protocol.

We have implemented an experimental version of the Network File Access Protocol (FAP), an unofficial protocol which allows one to reference portions of a file, rather than the entire file. Our particular interest in having this protocol available to us at the present time is that it will allow us to implement a cross-network version of the interpret_fdump command, which has been implemented by Bernie Greenberg. The interpret_fdump command, which takes the output from a Multics crash and attempts to explain the cause of the crash, can thus be used to diagnose crashes at other Multics sites on the ARPA Network.

XI. Redesign of ABSI by Electronic Systems Laboratory.

John Ward and his associates at Electronics Systems Laboratory at M.I.T. are continuing their development of an alternative implementation of the Asynchronous Bit-Serial Interface, (ABSI). They are currently producing a prototype version of their ABSI, and it is expected that they will commence to test this in the fairly near future. It is expected that this new version of the Multics/ARPANET interface hardware will be used both at IPC and at RADC.

XII. Study of the Burroughs Operating System.

Ben Williams is continuing his comparison of the error recovery mechanisms in Burroughs Operating Systems and Multics. Hopefully, a document detailing the comparison between these two systems will be available in the near future.

XIII. Distributed Data Base Management.

Art Benjamin has been studying the area of distributed data base management in network computers, and in particular, the idea of a distributed file system. As a particular application of these ideas, he is planning to design a backup mechanism for Multics that uses the ARPA network to provide file storage and retrieval facilities for protection against loss of on-line information. This approach will allow a more automatic operation of backup functions, and will provide a better user interface which will be more suitable to indiviaual needs. It should also be a useful test case for exploring ideas about distributed file systems and how to implement them.

Miscellaneous Tasks.

I. Creation of the System Load Generator Using the ARPA Network.

We are continuing with an Undergraduate Research (UROP) Project which is creating a system load generator which drives Multics through the ARPA Network. At least four undergraduates are participating in this project. They have spent the fall semester familiarizing themselves with the Multics System; during IAP and the second semester they will commence the design and implementation of the load generator itself.

II. Use of TERMINET 1200 and ODEC as Multics Printers.

Jeff Goldberg has explored in some detail what must be done in order to make the current Multics Daemon Software operate various remote printers. It appears that many of the modifications to this software which he required are already being made as part of the implementation of the access isolation mechanism on Multics. He is therefore monitoring this development with the intention of making sure that the modifications are compatible with our needs, and at the same time is attempting to provide an interim version of the software which can be used until such time as modified standard software is available.