PROJECT MAC

May 27, 1975

Computer Systems Research Division

Request for Comments No.78

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 January 1975 to 31 March 1975.

Certification Tasks.

I. Removal of Linker from Ring 0.

This task is essentially complete, as far as the Computer Systems Research Division is concerned, and little further effort is being expended on it. Honeywell has expressed its intention to incorporate the design into a future version of the Multics standard system, and has incorporated this change into the collection of modifications known as prelinking. The prelinking proposal is described in MTB-169, written by Steve Webber. A projected date for installation is not now known.

II. Removal of Name Space Management from Ring 0.

Dick Bratt completed the testing of his code for this task early in the reporting period. The results of these tests indicated that his ideas were essentially correct, and no further coding or development was undertaken. The only remaining component of this task is the writing of the thesis document itself, which is not yet complete, although a draft has been written. Dick Bratt is now working for CISL, and is attempting to expedite the installation of these mechanisms into the standard system. This installation has been incorporated into the same package as has the user ring linker, and is similarly identified as one of the tasks in Multics Technical Bulletin 169, mentioned above.

III. Separating the Directory Hierarchy from the Segment Catalog.

This task, which is being undertaken by Drew Mason, was described in the last task report as removing the storage hierarchy from ring 0. During the last three months, Drew has determined that it is not appropriate to consider removing the concept of hierarchy from ring 0 as part of his thesis. Rather, he has proposed a partitioning of directory control which would leave both functions inside ring 0. In his proposal, directory control will consist of two components, each with its own data base, as follows. There will be an unstructured segment catalog indexed by unique id, which maintains for each segment such physical attributes as the file map, the bit count, various date-time parameters, and so on. There will also be a directory hierarchy, which maintains the association between character string names and unique ids, and maintains the access control list for each segment. Thus, the principal change between Drew's current proposal and that described in the last report is that the access control list will now be stored in the

directory hierarchy rather than in a separate access hierarchy.

Honeywell is currently coding a new version of the file system, and this file system resembles the structure being proposed by Drew much more than the current file system does. For this reason it will be highly beneficial to Drew to take as his initial system the new rather than the old file system. Drew has therefore spent some time in the last few months attempting to find out at what point a version of the Honeywell new file system will be available for experimentation.

IV. Restructuring of Page Control.

Andy Huber has made significant progress on this task in the last three months. He has produced a fairly detailed description of the modularization which he proposes for all of the page control functions. This description is essentially written in PL/1, and could be evolved into the code itself. It is also written in a form to expedite understanding by people and represents a good snapshot of his current design. Since this representation is fairly lengthy and detailed, it is not being distributed as an RFC, but is available from the author to anyone on the project willing to read it. Andy is currently producing an RFC which describes the current state of his research.

V. Restructuring of Traffic Control.

Dave Reed's redesign of the Multics traffic controller is progressing along three fronts. First, the low level scheduler, which implements virtual processors using real processors, is being implemented. This portion of the design was described in RFC-66 written last December. Second, Dave Reed is designing the high level scheduler which will multiplex these virtual processors among real Multics processes. Third, Bob Mabee and Dave Reed have identified all those portions of the system other than traffic control which must be modified or redesigned in order to run the new traffic controller in the system. Bob Mabee is recoding all these modules. Included in this last category are modifications to interrupt and fault handling, changes to page fault handling in particular, and various other small system changes. As a part of this redesign, Bob Mabee is currently writing a proposal to eliminate from the system the traffic control data base called the ITT. This can be viewed as one step in our attempt to simplify the various interprocess communication mechanisms currently being used in Multics.

VI. Restructuring of the Answering Service.

Warren Montgomery is continuing his design of a version of the answering service in which the code which must be certified as part of the kernel has been isolated and identified. It appears that the restructuring which he is proposing will sharply reduce the bulk of this code. Part of Warren's efforts in the last three months has been devoted to completing a thesis proposal, however some design and coding has taken place.

VII. Fast Processes in Ring 0.

This task is essentially complete at this point. The coding and checkout of the design is finished, a design review has been held, and a Multics Change Request has been submitted to include fast processes within the standard service system. Although the MCR has not officially been voted upon, we currently believe that Honeywell is willing to include this feature in the system. The modification to the typewriter DIM interrupt handler which was provided to test this mechanism will probably not be used, since the new communication package from Honeywell will arrive shortly. Bob Mabee will continue to pursue this task as necessary.

VIII. Study of Multics Security Holes.

As part of our interest in the security of operating systems, we construct annually two documents: a list of all security bugs in the system known at the time, and a list of all those bugs which have fixed in the past year. This latter list, expanded with an explanation or a philosophical discussion of the bug, is distributed as an RFC. Jeff Goldberg has taken on the responsibility for collecting and generating these documents this year. Honeywell FSO has also taken on the responsibility for maintaining a bug list, and it is not now clear whether this should or will supercede the list we normally maintain. We will, however, continue to release the RFC describing all fixed bugs.

IX. Multitasking in the User Ring.

Doug Wells has been prevented from making significant progress on this task by other more pressing responsibilities. A version of the multitask environment exists which allows several tasks to run in the same process, and to call block and wakeup. The principal addition to this mechanism needed to provide a general multitasking facility is some time-slice scheduler. The multitasking environment is needed for

several projects currently underway in the group, for example the implementation of RSEXEC on Multics, so at some point in the future this task will definitely be continued.

X. New I/O Buffer Strategy.

This task is the design of an I/O buffering strategy which uses the virtual memory itself as a buffer management strategy. Due to tasks of higher priority, Raj Kanodia and Dave Clark have made essentially no progress on this task during the quarter. It is strongly hoped that they will be able to return to this at some point. This project may be resumed as part of the restructuring of the Network Control Program, a task which is also currently suspended due to lack of manpower.

XI. Study of System Error Recovery.

This task consists of studying the Burroughs Operating System in order to understand their error recovery mechanisms, which are considered to be very orderly and reliable. Ben Williams, who has been performing this task, has been unable to devote much time to it this quarter due to other commitments. We hope that he will be able to continue this task some time in the future.

XII. A Methodology for Designing
Certifiably Correct Computer Systems

Rich Feiertag is proceeding with his Ph.D. thesis as reported in the last task report. Much of his effort in this last quarter has been devoted to developing his thesis porposal, in which he outlines a methodology in which a system should be structured so as to simplify certification. This proposal will be distributed as an RFC in the future.

XIII. Study of Multics System Initialization.

Allen Luniewski is continuing his project of restructuring Multics initialization. He has further developed proposals outlined in the last quarterly report, which center on the idea of removing much of what is now considered initialization into a phase which rather is viewed as reconfiguration. He has produced RFC-68 which describes in some detail his proposal.

XIV. Restructuring of Network Control Program.

This task has been active in the past, but it is currently not progressing due to lack of manpower. It is hoped that it will be possible to reactivate it in the near future. The central idea behind this task is that the Network Control Program represents an ideal candidate to use in an experiment involving a multiple process implementation of a control algorithm on Multics. The Network Control Program is concerned with the flow of data to and from the ARPA Network. The principal task of the Network Control Program is the management of a multiplexed communication path, which implies the management of multiplexed buffers. It is proposed that the flow of data between these various buffers be implemented by processes, in particular the fast processes which have been implemented by Bob Mabee and which are described in this report as task VII. Several possible restructurings have been informally proposed during the last few months, but no further design work can be expected until more pressing tasks have been eliminated, chief among which are the NSW demonstration described later in this report.

A related aspect of this project concerns the possiblity that common elements can be identified in the software which is required to interface to different multiplexed communication streams. Thus, for example, it is possible that there exists a similar function in the software that interfaces the ARPANET and the 355. A buffer manager is an obvious example of a possible common routine. It is very interesting to try to identify these modules and, if possible, isolate them, since this would markedly reduce the amount of code inside the kernel, and since multiplexed streams, in particular networks, represent one way of bringing all I/O to the computer system. If all multiplexed communication streams could be handled by one set of kernel modules, it would a great simplification of the kernel.

XV. Support of User Defined Object Types.

As the last report mentioned there are two projects in this area. Phil Janson is continuing his consideration of how extended objects can be supported using an appropriate combination of access control list and capability based protection mechanisms. He has not been able to make large progress on this research during the quarter due to other commitments. Doug Hunt continues to investigate the implementation of user defined objects in a pure access control list environment. He is currently considering what policies for protection can be imposed upon these user defined

extended objects. For example he is considering whether it is possible to insure "message confinement" for an extended type manager without complete auditing of the extended type manager code. In general, he is interested in determining what constraints can be imposed upon the extended type manager without complete certification of the code. He has written RFC-73, which discusses his current thinking along the topic.

XVI. Study of Relationships between Security and Reliability.

Harry Forsdick is continuing his study of the relationship between the security of a system and the reliability of that system. His approach has been to study a variety of systems which maintain high reliability as one of their goals, and to investigate mechanisms in the system in the light of their implications for the security of the information stored on the system. He is preparing a series of RFCs which discuss various interesting systems.

XVII. Multics Performance Benchmark on Development Machine.

Drew Mason has published RFC-71 which describes his attempt to develop a stable version of the Multics performance benchmark which can be run on the development machine. The development of this benchmark has been taken over by Vic Voydock. Vic has developed a version of the benchmark which can, for debugging purposes, be run on-line, and he has developed a version of the benchmark which, using some of the suggestions proposed by Drew in his RFC, is extremely stable in the virtual CPU time required for the run. The real time and the number of page faults still vary somewhat, and do not vary proportionally.

Technology Transfer and Network Related Tasks.

I. Production of MPM Network Users Supplement.

On January 24, 1975, we issued a preliminary edition of the Network Users Supplement to interested members of the network community. This is a truly significant event, to be overshadowed only by the release of the official first edition, which should occur sometime within the next six months. At this point, the NUS is in good enough shape that it is in practice useful to those who are attempting to access the network. There are, however, certain sections which are either missing or incomplete, and other sections which require rewriting. We are hoping that comments on this preliminary edition will be received so that they can be incorporated into the official first edition. Ken Pogran has put a significant percentage of his time into overseeing this publication.

II. Implementation of RSEXEC on Multics.

We are currently considering both a server and a user RSEXEC on Multics. Jerry Rudisin has essentially completed the design of server RSEXEC as part of his bachelor's thesis. As of this point, code exists which will support the RSEXEC link mechanism and coding has begun on those modules which support the RSEXEC file system functions. The major remaining problem is to convert from the general RSEXEC protocol to the Tenex RSEXEC protocol, which must be used if file system functions are to be implemented. It does not appear that there are any intrinsic difficulties in this conversion.

A proposed design for a user interface to RSEXEC on Multics is described in RFC-67, A Network-Wide File System for Multics, by David Clark. This implementation of user RSEXEC would allow one to view files on other systems as being part of large hierarchy which included both the local and distant file systems. Further design of this user RSEXEC interface will not resume for some time, since it cannot be implemented until the user ring name space management mechanism designed by Dick Bratt is installed in the standard system.

III. Proposal for Extensions to iox_ Interfaces.

Doug Wells has implemented an experimental version of an I/O system interface which is compatible with iox_, but which provides various capabilities not supported by iox_, including variable delimiter characters, variable byte size, and asynchronous reading, writing

and opening. This interface will be used to support various experimental programs which we are currently developing within the group. As time permits, a document describing this interface and its motivation will be completed and distributed.

IV. Preparation of Network Program Logic Manual.

Ken Pogran, the editor of the network PLM, has been prevented from working on this task by several other projects which have occupied all his time, especially the preparation of the NUS, the development of a new server ftp, and the integration of the new mail protocols into the network mail facilities. We cannot be too sanguine about immediate progress on the PLM, since other high priority tasks, such as the National Software Works, loom on the horizon.

V. New Version of the User FTP Command.

The development of the new user_ftp command, which was described in the previous task report, has been delayed by other tasks of higher priority. We have determined to implement user_ftp using network IOSIMS, which means that the IOSIMS themselves must first be created. We are progressing on the coding on these IOSIMS, as discussed below, and we will be very disappointed if the user_ftp command is not available by the end of the next reporting period.

VI. New Version of Server FTP.

A new version of the server FTP has been coded and installed by Ken Pogran. This version of the server FTP supports the new file transfer protocol, and also provides a number of features not previously supported. The software now supports the so called file access protocol, an experimental protocol for accessing part rather than all of a file. The protocol is used as part of the network version of the interpret_fdump command. The new software supports the FTP NLST command, which premits the listing of a directory. The new software also supports a specialized limited environment used for the receipt of inbound network mail. The mail receiving software has been modified as part of this installation to interface to the new mail facilities.

VII. Specialized File Transfer Commands.

As a result of the mini-service which is being run on the CISL development machine, a need has arisen for a convenient specialized command to transport files between two Multics sites. We have developed commands

for this purpose, which are currently in use on the development machine. These commands are not yet to installation standards, since they do not deal gracefully with all errors, but an improved version of these commands should prove useful to new Multics sites comming on to the network, such as the Rome Multics, discussed below.

VIII. Redesign of ABSI by Electronic Systems Laboratory.

The redesign of the asynchronous bit serial interface (ABSI) by John Ward and his associates at the Electronic Systems Laboratory is proceeding well. They have produced a version of the ABSI using commercially available components, and a different layout involving five boards rather than one large board. A prototype of this design has been assembled and has been used in Multics service for a reasonable period with no errors detected. ESL is now having printed circuit boards etched for several copies of this new ABSI. The first of these will go to the Rome Multics site, with later ones being used at M.I.T. itself.

IX. Bringing the Rome Multics onto the ARPANET.

Plans are progressing for the attachment to the ARPANET of the Multics located at Rome Air Development Center (RADC) in Rome, New York. We have gathered together all of the software which must be installed on the system in order to operate the network, and are preparing memos which describe the installation and activation procedures for the network. It is expected that shortly after the ABSI is delivered, as described above, that Ken Pogran will visit RADC for one or two days to assist in the actual installation and to present a tutorial on network operation and utilization. We expect this will happen approximately at the end of April.

X. Recoding of the Net_Mail Command.

The net_mail command is responsible for sending mail from Multics over the ARPA Network. This command is currently being rewritten by Steve Swernofsky and Ken Pogran in order to bring the command up to installation standards and to add several new features. The command is being rewritten to reference the network through the I/O system rather than by direct ring 0 system calls. Among the new features which it will support is the sending of mail to a mailing list maintained in a seperate file. Since most network development is done

by the exchange of memos through the network itself, sending of messages to a large mailing list is a common activity which this modification will significantly expedite.

XI. Analysis of Bugs in Network Software.

Several annoying bugs have been identified during the reporting period. Several of these were flushed out by the installation of the new IMP DIM, which contained a couple of subtle bugs itself. We also seem to have problems with the network control program under certain circumstances. In our attempt to locate some of these bugs, we have developed or improved several analysis tools, which enable us to dissect error conditions on the network. A significant amount of time was devoted to the development of these tools.

XII. Development of Network Host Information Data Base.

There currently exists a table on Multics, called the Net Host Table, which maintains an association between host number and the character string name of the host. It is currently rather difficult to maintain this table, since it is installed in the libraries and thus can only be updated by an on-line installation. Since host names change fairly often, and with little advanced warning, the long delay associated with on-line installations means that this table is seldom completely up to date. For this reason, Nancy Federman has been developing a new version of the Net Host Table, which can be updated dynamically through a ring one date. This will provide an ability to respond quickly to changes in the network configuration. It will also provide several new capabilities, especially the ability to associate various attributes with network hosts. Thus, for example, it will be possible to determine whether a site on the network is a Multics, which will allow certain specialized tools to operate only between Multics sites on the network. The design and coding of this new table is essentially complete.

XIII. Development of National Software Works on Multics

During this last quarter, the Network group has started to participate in the development of the National Software Works (NSW) in the ARPA Network. The National Software Works is a scheme in which a powerful environment for program development is created using the best tools from various systems on the ARPA network, tying these systems together by a uniform protocol by which these tools may be accessed. Our specific contribution to this project will be to

for this purpose, which are currently in use on the development machine. These commands are not yet to installation standards, since they do not deal gracefully with all errors, but an improved version of these commands should prove useful to new Multics sites comming on to the network, such as the Rome Multics, discussed below.

VIII. Redesign of ABSI by Electronic Systems Laboratory.

The redesign of the asynchronous bit serial interface (ABSI) by John Ward and his associates at the Electronic Systems Laboratory is proceeding well. They have produced a version of the ABSI using commercially available components, and a different layout involving five boards rather than one large board. A prototype of this design has been assembled and has been used in Multics service for a reasonable period with no errors detected. ESL is now having printed circuit boards etched for several copies of this new ABSI. The first of these will go to the Rome Multics site, with later ones being used at M.I.T. itself.

IX. Bringing the Rome Multics onto the ARPANET.

Plans are progressing for the design of a file cataloging mechanism, and will look at other file system features that are relevant to the operation of a backup facility.

Miscellaneous Tasks

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

We are continuing with the undergraduate research project which is creating a system load generator which drives Multics through the ARPA Network. During this quarter, the most significant progress has been made on the sticky question of what constitutes an appropriate test load. It is expected that progress will continue on this project through the spring and perhaps through the summer.

II. Use of Terminet 1200 as Multics Printer.

During this quarter we have brought up the Terminet 1200 a as a device driven by the standard Multics I/O coordinator. It appears that this service, which provides speedy printing of short files, is filling a significant need, since the Terminet is printing a significant percentage of the time during the day.

The only difficulties with the service are the rather surprising cost, approximately four times the cost of printing a file through the standard system printer, and the persistant unreliability of the Vadic modems. We are attempting to cope with both these problems as best we can, without investing time in the project.