

PROJECT MAC

May 13, 1974

Computer Systems Research Division

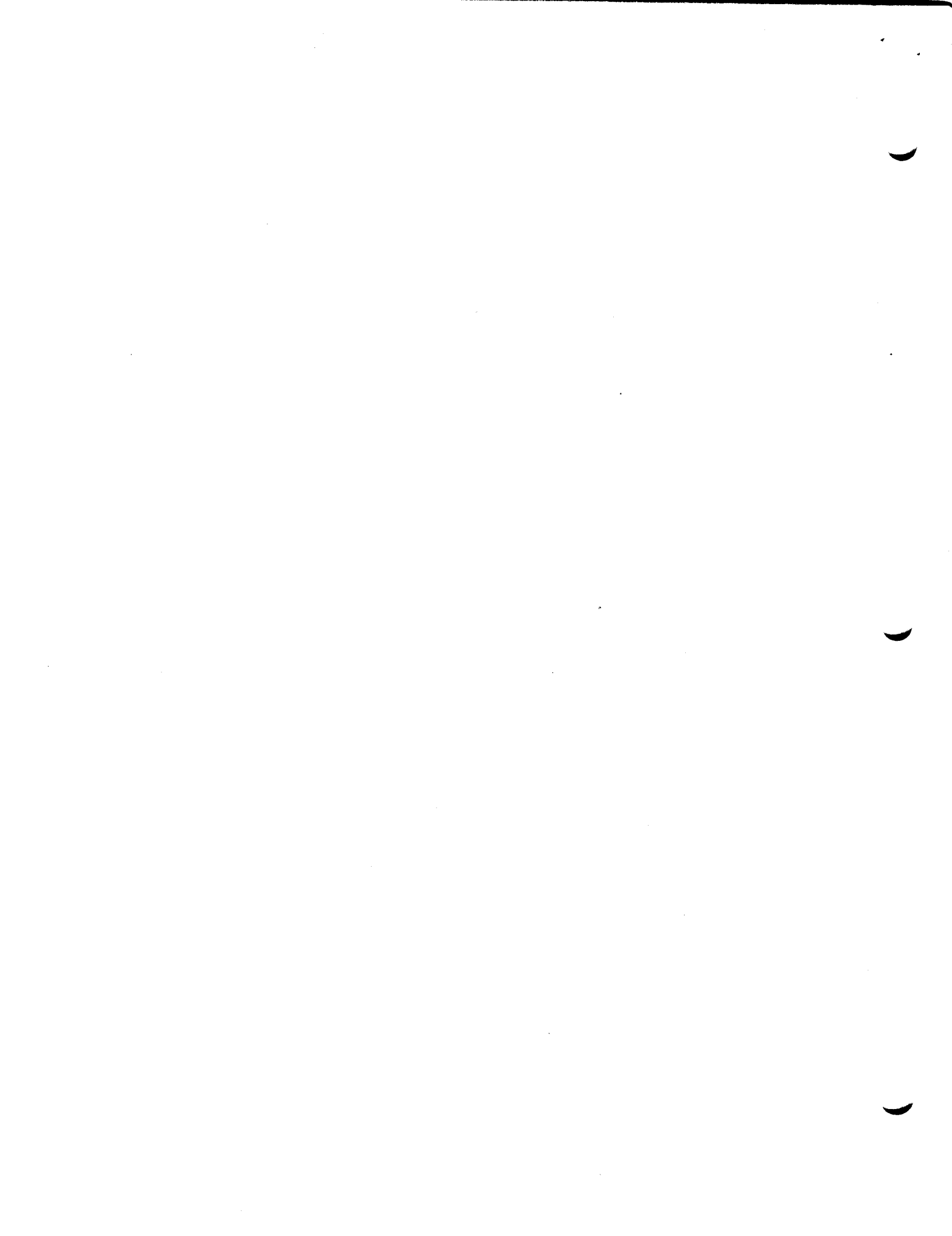
Request for Comments No.52

CSR TASK REPORT

by David D. Clark

The attached task report covers the period from the issue of the previous report, November 1, 1973, to May 1, 1974. In the future, these reports will be issued quarterly, with the next report to describe work progressing through July 1, 1974.

This note is an informal working paper of the Project MAC Computer Systems Research Division. It should not be reproduced without the author's permission, and it should not be referenced in other publications.



A. Certification Tasks.

I. Removal of Linker from Ring 0.

The task of removing the linker from ring 0 is nearing completion. This task is an important first step in pruning unnecessary programs from the portion of the system which must be certified. Coding is completed for all programs needed in order to snap a link in the user ring. Testing of these modules has proceeded to the point where a bootload of this system has been attempted. Additional programs must be coded in order to complete this task, in particular, programs which change the search rules and which unsnap links. E. Thomas will do the coding of these programs, while P. Janson completes the check-out of the linker with the assistance of R. Kanodia.

II. Recovery from Errors.

B. Williams continues to study the error recovery mechanisms of the Burroughs 7700 operating system, in the hope that the knowledge thus gained may be applied to Multics in order to make such tasks as salvaging less ad hoc and easier to certify. He recently received from Burroughs a listing of the operating system for the 7700, which he is studying in detail.

III. Removal of Name Space Management from Ring 0.

R. Bratt has described a scheme for moving certain functions of the KST into the user ring, thus further reducing the bulk of code to be certified. In particular, he proposes that the association between names and segment numbers be maintained in the user ring rather than in ring 0, leaving in the supervisor only the association between segment numbers and unique ID. One of the features of his design is a scheme which eliminates the drawbacks of allowing the user to directly initiate directories.

It is possible that out of this and related research, a broader proposal will emerge to remove essentially all of directory control from ring 0.

IV. Restructuring of Page Control.

Research is continuing on various ways to reorganize page control. Using the language devised by B. Greenberg to describe page control in his thesis, D. Clark constructed a version of page control which handled read-write sequences in a separate process. This approach to reorganization is currently being further pursued by A. Huber, who is attempting to produce a version of page control which uses processes to execute all the page control functions except the act of fetching the missing page. It is felt that by isolating functions in **separate** processes, and constraining them by restricting the inter-process communication paths, it will be easier to understand and certify the overall algorithm. One of the other benefits of structuring page control in this way is that it should be possible for several processors to take and handle a page fault simultaneously, without interfering with each other.

V. Restructuring of Traffic Control.

D. Reed is currently exploring techniques for restructuring the Multics traffic controller in order to speed up the act of switching from one process to another. His intention is to split the traffic controller into two parts, separating out the actual act of switching from one process to another from the more complex act of deciding which processes are eligible to run. This division will hopefully make the algorithm easier to understand. By making the overhead of process switching very small, it will be economically feasible to run various portions of the supervisor in separate processes. An example of the need for this capability is the restructuring of page control into multiple processes, described above. Portions of the traffic controller itself, in particular the loading algorithm, might also be placed in such a process.

VI. Fast Processes in Ring 0.

As the above two sections would suggest, one very promising approach to understanding the structure of ring 0 is to separate portions of the supervisor into separate processes. In order for this to be possible, it is necessary that there be available a class of process which is very inexpensive to run. The modification to the traffic controller discussed above is one of the tasks required to make such an efficient process. There are several other modifications to the system which must be performed in order that such an efficient process be available. R. Mabee is currently designing these modifications.

VII. Restructuring of Network Control Program.

Because of our group's direct involvement in the connection of Multics to the ARPA network, we have developed significant expertise in that portion of the system. For this reason, the network software is a good candidate for investigation of techniques for simplification of the system kernel. D. Wells is currently studying various ways to separate the network software into processes, some of which need not run in the supervisor of the system. In this way it is hoped to reduce the bulk of the supervisor code related to the ARPA network. Notice that the ARPA network is especially interesting in this respect, since, being a multiplexed facility, some protection is required of the demultiplexing and multiplexing of information. Any technique which removes this from the supervisor, without compromising its security, is an especially valuable modification to the system.

VIII. Enhancement to Buffering Strategy for ARPA Network.

In a related modification to the network software, R. Kanodia has been studying a technique to simplify and improve the buffering strategy used for the ARPA network in Multics. He is soon to publish a RFC describing a buffering strategy for the read side of the ARPA

network which is simple, effective for a wide variety of message lengths, and which interacts well with the virtual memory of this system. In fact, the buffering strategy employs the virtual memory of the system for its memory allocation technique.

IX. System Initialization.

V. Voydock is currently studying procedures by which the Multics system is initialized. Clearly, in order to certify the Multics system, it will be necessary to certify the "initial state" of the system; the ad hoc initialization techniques currently used make such a certification very difficult, if not impossible. The intention of this study is to discover new ways of initializing the system which are more amenable to certification. As a first step, a detailed study of current initialization is being produced, and should be available in draft form in the second week of May.

X. Formulation of Criteria for Inclusion of Modules within the Kernel.

R. Feiertag is currently attempting to determine the general rules which specify whether a module should or should not be included within the kernel of the system. He is attempting to identify these rules by studying a number of specific parts of the current system and identifying the tradeoffs related to moving these particular parts out of the kernel. He is currently studying page control toward this end. It is hoped that a document describing what must be in the kernel given the Multics architecture will be available by the end of May.

XI. Application of Formal Program Certification Techniques to the Multics Kernel.

D. Hunt is currently considering whether a simplified version of the Multics kernel is amenable to certification by any of the formal techniques currently being developed, for example structured programming or correctness proofs. He is currently considering how to simplify the kernel for this experiment. A document describing the techniques

he proposes to use, and the kernel on which he intends to apply them, may be available by the middle of May.

XII. Study of Multics Security Holes.

An attempt is made periodically to catalogue all known ways to violate the security of Multics, including ways to crash the system, because study of how bugs arise and what is required to fix them gives insight into the problems of certification. Two RFC's have been issued by J. H. Saltzer, P. Janson and D. Hunt which describe all the security holes known to have been fixed in Multics during the last year, including those which were fixed by the new 6180 hardware. These RFC's contain a discussion of what insight may be gained by studying each of these repaired bugs.

XIII. Multitasking in the User Ring.

Several experiments are currently in progress which are intended to allow the multiplexing of one process amongst several tasks in the user ring. There are several goals for this research. First, one of the complexities found in the system nucleus has to do with the handling of the Multics quit signal. If it were possible to handle this quit within the user ring it would eliminate that portion of the kernel. In order to do this, however, it is necessary to understand what process structure in the user ring is appropriate to deal with quits. Tasking is a technique which can be used to explore this question. Similarly, there are other aspects of process intercommunication which might be removed from the kernel if it were understood what mechanism could best replace them in the user ring. There is also an immediate need for a multitasking capability, in order to multiplex a process for reasons of efficiency without the use of event call channels, which are undesirable because they impose an awkward coding style on the user. D. Anderson, R. Frankston, D. Reed and D. Wells are working on these experiments.

XIV. Implications for Multics of Terminals Containing Processors.

A project is starting to study the question of what role a large operating system should play in the future, given the current trends in hardware, software and networking. In particular, of what use will Multics be if terminals have within them a processor as powerful as a PDP-10 or 6180. This question has implications for network research, since new protocols will be required to communicate between powerful terminals and central machines. The question also has implications for certification, since functions such as human interfacing can be performed in the terminal, where they need not be audited. D. Clark, W. Montgomery, K. Pogran and E. Thomas are tentatively considering this question.

XV. Implementation of "Lucifer" Incryption Algorithm.

G. Benedict, as his Bachelor's thesis, is implementing the Lucifer algorithm on Multics. This algorithm was designed by IBM as a hardware encoder to protect transmitted data. It could be used on Multics to protect files or to transmit information in a secure manner through the ARPANET. The particular structure of the scheme, which employs 128 bits keys to code and decode, could be used as the basis of a two-way authentication scheme between intelligent terminal and central computer, as described in the previous section. The algorithm encrypts blocks of 128 bits; an initial coding in ALM will code 128 bits in 5.5 ms., or 0.4 ms./character.

B. Performance Tasks

I. Network Performance.

As part of the integration of network and typewriter buffering in ring 0, it is necessary to know some of the characteristics of messages sent over the network. There are currently meters in ring 0 which collect this information, but they are inoperative due to a bug. A. Benjamin has taken on the task of attempting to gather this information

II. Hardware Measurements of the 6180 Processor.

D. Gifford is continuing his interest in making hardware measurements on the 6180 processor. The measurements made last November were described in RFC 48 issued in February. The intention is to issue another RFC containing further analysis of the results of these measurements.

III. Model of Disc Traffic.

L. Scheffler is writing a Master's thesis which discusses a probabilistic model of movable head disc subsystems. His optimizing dim is being used as a test case for this model. The thesis should be done this term.

IV. Multics Working Set Estimator.

The Bachelor's thesis by D. Reed which was submitted last June proposed an algorithm for estimation of working set size which could be used by the traffic controller on Multics to set the maximum number of eligible processes. No further research in this area is anticipated at the present time because it is not obvious that a working set estimator is required for the 6180 Multics. However, a report of research to this point will be prepared, perhaps this Summer.

C. Network Tasks

I. Implementation of New Telnet Protocol.

Multics software has been modified to support a revised definition of the network Telnet protocol. The revision is essentially complete at this point, except for final debugging. A version of the new Telnet software has been installed.

II. Installation of User Telnet Program.

There has existed for some time a private program, maintained by D. Wells, which implements the user side of the Telnet protocol. D. Wells has undertaken the task of upgrading this program to installation standards and submitting it to the network library. The work remaining consists of documentation and minor recoding, which should be done within a month.

III. New File Transfer Protocol.

The coding of the programs to implement the new file transfer protocol has been suspended, pending stabilization of the protocol itself. K. Pogran has been participating in this revision of the protocol. It is intended that K. Pogran will undertake the recoding whenever the specification is ready.

IV. Multics Programmers' Manual, Network Users' Supplement.

The status of the network user supplement is as follows. RFC 45, issued January 11, contains an outline of the NUS. The first two chapters identified in that outline, Introduction to the ARPA Network and Overview of the Multics Network Attachment, are also a part of that RFC. Chapter 3 of that outline, Use of Multics From the Network, exists currently in draft form. Chapter 4 of that outline, Use of the Network from Multics, has been partially written. A writeup of the "user Telnet" command is currently being prepared by D. Wells. A writeup of the "user FTP" command is being prepared by K. Pogran. D. Wells is also preparing writeups of the IOSIMS. Material in chapters 5 and 6 of the outline has been moved from the NUS to the network program logic manual (See next section). It is intended that a rough draft of all the NUS be completed by the middle of May. M. Padlipsky is acting as editor.

V. Program Logic Manual for the Network.

K. Pogran has taken on the responsibility of editing the program logic manual describing the network. He has prepared a document containing an outline of the PLM, which will be distributed in the next few days. It is intended that by scrounging material already written, a PLM be assembled within the next two months. It appears that this schedule can be met without a great writing effort on the part of the staff members.

VI. Status of the Asynchronous Bit Serial Interface.

The asynchronous bit serial interface, or ABSI, continues to be a source of annoyance to us. Although its rate of failure has dropped markedly in the last several months, it still requires our attention more often than we would desire. We hope that we will be able to hand off the maintenance of the ABSI to John Ward of ESL in the next few months. We cannot do this, however, until he has familiarized himself with the design of the ABSI, and his ability to do this is hindered by the lack of adequate documentation. The longer range plan is that John Ward should design a version of the ABSI which is mechanically more stable than the one we are now using, and that he should take over the maintenance of this on a permanent basis. He has not yet expressed his willingness to undertake this task, but it is hoped that this plan will ultimately prove successful. For the time being, K. Pogran is continuing to repair the ABSI as necessary.

VII. Common Command Language.

M. Padlipsky has been involved in several subcommittees of the network Using Committee. In particular, he is chairman of the committee to develop a common command language. He has contributed to this committee a document on protocol initially called the unified user level protocol, which describes one such common command language.

The committee is currently attempting to develop a specification based on this and other inputs.

VIII. Neted, A Common Network Editor.

M. Padlipsky, in his attempt to develop a common command language, devised a specification for a network wide text editor called neted. This editor has been accepted enthusiastically by several members of the network community, and has now been implemented on several machines. There exists a subcommittee of the Using committee, chaired by M. Padlipsky, whose function is to stabilize the specification of neted. A standard for the editor has been submitted to the full committee at this point.

IX. RSEXEC Implementation in Multics.

A study continues of the possible implementation of RSEXEC on Multics. So far, there have only been limited implementations; in particular, a program has been written to exercise a RSEXEC server on a foreign host to "link" terminals with a user on that system. W. Montgomery intends to produce a program to implement a RSEXEC server which will support, initially, those functions of RSEXEC not dealing with files. It appears at this point that handling files in the RSEXEC protocol is the most difficult feature for us to implement.

X. A Network Mail Queuer.

W. Montgomery, as part of his investigation into providing network services, has designed and implemented a mail queuer which will hold mail and transmit it later when the receiving machine is up and available. This mail queueing service is being offered to the Multics network community as an experimental service, and it is intended that the process implementing this service be installed as a consoleless daemon so that this service can be available at all

times. It is intended that this process be multiplexed to provide other network services, including support of RSEXEC and receipt of inbound network mail. In order to multiplex this process, advantage will be taken of the implementation of multitasking in the user ring, as described in section A, Item XIII.

XI. Revision to Network Supervisor Software.

The network control program, or NCP, was recently recoded to Version 2 PLI. In the process of this recoding, certain small revisions were made to the code. This Version 2 NCP has been submitted and installed in the system. The IMP dim is currently being revised to fix certain small bugs, and to add certain small features, by R. Kanodia. It is expected that this modified version of the IMP dim will be installed in the next month. This revision does not contain any major changes.

XII. Maintenance of Network Software.

The Computer Systems Research Division has responsibility for the software which runs the network on the service system. Maintenance tasks such as bug fixing continually occur. All members of the network group contribute to this effort. It is hoped that in the future this responsibility can be shifted to the Information Processing Service, under PDO. Harvey Baumel of PDO has been assigned to study the network; however, it does not appear that he will be able to take over these functions in the immediate future, because of his heavy workload.

XIII. ANTS Steering Committee.

K. Pogran has been serving as a member of the ANTS steering committee which is overseeing the development of the ARPANET terminal system at the University of Illinois. This committee has also been given the responsibility of reviewing ELF, another PDP-11 operating system.

See also certification tasks VII and VIII.

D. Miscellaneous Tasks

I. Implementation of LISP on Multics.

A version of LISP identical to that running on the PDP-10 machines at Project MAC has now been installed on Multics. Performance measurements on this version of LISP indicate that performance is comparable, if not better, than performance of the PDP-10 version. The current effort of the LISP maintainers has been the preparation of the manual describing this version of LISP. This manual has been completed and is now being printed.

II. Implementation of MACSYMA on Multics.

Using the version of LISP described in the previous section, a version of MACSYMA has been executed on Multics. Although no extensive performance measurements have been made on this version of MACSYMA, it appears that performance is again comparable to that on the PDP-10 system in terms of processor speed.

III. Up-Grading of CSR Terminals.

It is the intention of the Computer Systems Research Division to replace some of its 2741's with video terminals. A study is being made to determine the appropriate terminal which is available for this purpose.

IV. Use of the Terminet 1200 as a Remote Printer.

With the arrival of the Vadic modem and the anticipated installation of the supporting software on Multics, it appears that it is now possible to consider using the Terminet 1200 for its intended purpose as a line printer. The intention is to provide software so that the Terminet can be driven by the I/O Coordinator as are all the other system printers. So that the Terminet can be operated in this fashion, it is necessary to modify the software currently available for control of remote printers.

V. Maintenance of System Listings.

The conversion from microfiche to hard copy listings of the hard core is virtually complete. Difficulties with the software which maintain the tape of all listings have delayed us in obtaining the last few listings needed to complete this set; however, the listings are now thought to be complete. Microfiche listings of other portions of the system continue to arrive sporadically. It is not known the extent to which these microfiche listings are out of date.

VI. Organization of Historical Documents.

D. Jones has taken on the task of organizing and filing documents related to the development of the Multics system. These are currently being stored in Room 540, where they will be available to expeditions of an archaeological nature. Included in this collection is a xerox of the system book for all systems created since the concept of a system book was created.

VII. Identification of Participants in Multics Project.

D. Jones has been completing the list of all persons known to have contributed to the Multics Project, as part of the organization of historical information. The count currently stands at 445 people.