M.I.T. Laboratory for Computer Science November 18, 1977

Computer Systems Research Division Request for Comments No. 151

CSR TASK REPORT

from David D. Clark

The attached report covers progress of the Computer Systems Research Division of the Laboratory for Computer Science in the period 1 July to 30 September 1977.

This note is an informal working paper of the M.I.T. Laboratory for Computer Science, Computer Systems Research Division. It should not be reproduced without the author's permission and it should not be cited in other publications.

ARPANET Related Tasks

I. Improvements to Network Mail Facilities

During this period, the M.I.T. Information Processing Services assumed operational control of the ARPANET mail facilities. The Network_Server and NETML processes were moved to the Daemon project and IPC personnel are now providing operational maintenance. With this change, CSR is no longer involved in the day to day operation of the M.I.T.-Multics attachment to the ARPA Network.

Charlie Davis worked on the Multics ARPANET mail software during the past quarter. A new net mail queuing scheme has been debugged; it will be installed in the network library after further testing and refinement.

II. ARPANET Software Support

An agreement was reached whereby beginning with Multics Release 6.0, the Multics ARPANET support programs will be distributed with the normal Multics system release package. In preparation for this event, the Answering Service and Hardcore portions were modified to adhere to the latest system language conventions and to include source-module message documentation. This documentation (along with the documentation for other portions of Multics) will be extracted and compiled for use by operators and System maintenance personnel at Multics sites.

III. Multics TCP

The Multics implementation of the ARPANET Transmission Control Protocol (TCP) was begun during this period. The Hardcore portion of the IMP DIM was modified to allow transmission of uninterpreted IMP messages. Drew Mason began work on a first version of the outer-ring portions of the implementations. The implementation will not strike for efficiency. The goal is to provide a working program with which we can experiment to learn more about TCP and network protocols. A working version of TCP should be available for experimentation by the end of November.

IV. RSEXEC

At the request of ARPA, there was some activity with the Multics RSEXEC implementation. Some minor bugs were fixed in the Multics server, and the server is now running as a daemon process. Harry Forsdick of BBN modified the TENEX RSEXEC implementation to better interface to Multics. Colonel Russel of the ARPA office is now using RSEXEC to access the Consistent System during demonstrations of the ARPANET.

Research in Distributed Systems

I. LCS Local Network

Progress on the local net was somewhat impeded during August by the almost total absence of personnel. Those who were here wwere substantially distracted by the random interruptions which inevitably arise.

Hardware

As a part of our development of the LCS Network, we have been planning a comparative study between the ringnet and the ethernet technology for the purpose of determining any possible operational advantages of one over the other. Liba Svobodova started working on specifications of measurement facilities needed to study performance and reliability of the network. She also studied related analytical work in order to assess usefulness of analytical models as tools for comparative performance evaluation. Finally, she addressed some reliability issues in connection with ethernet protocols. The plan and the results of these preliminary studies were reported in LNN 13, 14, 15 and 16.

During the summer, Steve Kaplan continued the development of the PDP-11 interface to the ARPANET, designing and constructing the DMA interface. Tim McMahon returned in September and continued with the construction and debugging of this interface. We anticipate that it will be up and running in the next quarter.

The arrival date of the local network interface continues to slip. During August we took delivery of the prototype version of the LNI, and Mike Lyle and Paul Mockapetris spent some time at the Laboratory. The prototype version, however, is sufficiently out of date that they concluded it was not worth the trouble of upgrading the prototype to the current design. Therefore, any substantial work on the LNI here must await the arrival of the next version. Ken Pogran has done as much advanced planning as possible, including parts procurement, so that we should be prepared to prodeed at full speed when the LNI does finally arrive.

In the early part of the quarter, Charlie Davis made some progress toward getting the CSR Division's Motorola 6800 microprocessor ready to act as a terminal concentrator on the LCS Network. Several tools were developed which enable programs to be downloaded from the PDP-11/70 UNIX system into the microprocessor. These tools were used extensively in exploring the capabilities of the 6800, and in gaining the experience needed to complete and debug an implementation of DSP. Further progress in this area is awaiting the development of hardware to interface the microprocessor to the Local Network.

Dave Reed and Ken Pogran attended the National Bureau of Standards Workshop on Local Area Networking, where they met a number of people with problems similar to our own. There appears to be widespread interest in the construction of local nets.

Protocols

During July, our group hosted a meeting of the committee developing TCP. Some progress was made at this meeting reconciling some of the substantial differences between TCP and DSP. We discussed in some detail our interests in incorporating our ideas about unique socket number connection initiation into the next version of TCP. Dave Reed was principal organizer at this meeting.

Dave Reed also worked on a concept for packet routing in an internetwork environment, called logical path routing. This last idea has been tabled for a while, because of lack of time.

Doug Wells began the task of developing the Gateway machine between the LCS Network and the ARPA Network. We negotiated arrangements to get implementations of TCP and NCP for the machine. We also started the task of trying to keep the UNIX operating system up-to-date with the DSSR version running on the 11/70.

Steve Kent has constructed a list of tasks to be performed as part of our experimentation with encryption in the local network. We will attempt to find a variety of UROP participants who can carry out some of these tasks during the fall.

II. Study of Objects in Current Systems

Karen Sollins is continuing her study of object-oriented distributed systems by moving into the area of type managers and how they might be distributed. The current short term project is to design a small relational database in CLU, in order to work on such a database on top of a substrate of extended types in a centralized situation, as a precursor to distributing such information. The other puspose of this project is to obtain experience with procedure abstractions with the idea that they may be part of the solution to the problem of how to construct distributed type managers.

III. Adaptive File Allocation in a Distributed System

During this quarter Allen Luniewski began a study of algorithms to dynamically allocate files in a distributed system in response to slowly changing access patterns for those files.

IV. Semantic Model for Distributed Computing

Warren Montgomery was absent from LCS during the months of July and August working for PRIME Computer Company. During September, he has been investigating a semantic model for distributed computing based on the notion that programs can be modeled as collections of processes which communicate through message passing and not through shared objects. Work so far has concentrated on the justifications for such a model and its relationship to others which have been proposed.

V. User Level Naming in Distributed Systems

Throughout the quarter, Dave Reed continued to work on his Ph.D. thesis in naming of distributed objects. The major results of this work are a thesis proposal, to be submitted early in October, and a number of new ideas involving the use of a notion of "distributed time" similar to that of Lamport's "Time, Clocks and the Ordering of Events in a Distributed System". An important idea seems to be associating a "clock" with each computational environment to mediate the view of the system's data through that environment. This idea seems to lend great insight to several distinct kinds of problems, such as the problem of getting a transaction to update either all or none of its outputs in the face of unreliable communications and distributed processors, or the problem of getting a consistent set of input values from distributed cells that are constantly being updated where locking is prohibitively costly.

VI. Communication Security

Steve Kent was at LCS only during September. During that month he prepared for and attended the 5th Data Communications Symposium at Snowbird, Utah where he presented a paper on communication security. Also during that month he began to explore techniques for permitting mutually suspicious programs to interact in a distributed computing environment while providing for confinement of these programs. This is relevant to the use of proprietary programs with sensitive data in such an environment. One technique, which appears to be very flexible and efficient in terms of processor and communication utilization, employs a module consisting of a processor, cache memory, and an encryption device in a module sealed from tampering and electromagnetic emissions. This approach will be investigated in greater detail in subsequent quarters.

Miscellaneous Tasks

I. Secure Input/Output

Gene Ciccarelli has spent the last quarter researching and writing his thesis: on the implementation of multiplexed I/O streams within the security of an operating system. He is expecting completion by December.

II. Improved Text Editor

Gene Ciccarelli demonstrated that a real-time display editor can be implemented on Multics, using two processes to distribute the editing functions and response necessary. One process wakes every character and edits within a line; the other process wakes less frequently, when the first finds a command it cannot interpret, generally line-oriented commands. The two processes interact via the ARPANET.