Best Available Copy

Center for Research in Computing Technology

Aiken Computation Laboratory
Harvard University
Cambridge, Massachusetts 02138

Modification to Internal Protocol Model

by

William R. Bush

December, 1978

Sponsored by
Defense Advanced Research Projects Agency (DoD)
ARPA Order No. 3079.3
Monitored by Naval Electronic Systems Command
Under Contract #N00039-78-G-0020

unclassified

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) (6) Modification to Internal Protocol Model.		5. TYPE OF REPORT & PERIOD COVERED (9) technical report
		6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) (10) William R. Bush		8. CONTRACT OR GRANT NUMBER(s) (15) N00039-78-G-0020, ARPA Order-3093
9. PERFORMING ORGANIZATION NAME AND ADDRESS Center for Research in Computing Technology Harvard University Cambridge, Massachusetts 02138		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS Department of the Navy Naval Electronic Systems Command Washington, D.C. 20360		12. REPORT DATE (11) December, 1978
		13. NUMBER OF PAGES five
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)		15. SECURITY CLASS. (of this report) unclassified
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

unlimited (12) 8 p.

DISTRIBUTION STATEMENT A
Approved for public release;
Distribution Unlimited

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Transmission Control Program (TCP)
MSG interprocess communication protocol
flow control

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

An abstract model of the interprocess communication system MSG has been written as a test of software development and maintenance tools. This report evaluates the maintenance effort involved in changing the model to use the Transmission Control Program (TCP) network protocol.

DD FORM 1473 1 JAN 73 EDITION OF 1 NOV 65 IS OBSOLETE
S/N 0102-014-6601

unclassified

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

405 831

Modification to Internal Protocol Model

by

William R. Bush

An abstract model [model] of the interprocess communication system MSG [MSG] has been written as a test of software development and maintenance tools. This report evaluates the maintenance effort involved in changing the model to use the TCP network protocol [TCP].

There are two aspects to converting to TCP. The first is disruption, that is, the minimum amount of work needed to get the old system to work in the new environment. The second is optimization made possible by the new environment, involving enhancements to the functioning system. The abstract model would be disrupted not at all by changing to TCP, and facilitates possible optimizations.

The model's interface to the network is its general paradigm for input-output, the channel. A channel is a data path between two processes. The primitives for manipulating channels are CHOPEN, with which channels are opened, SEND, for sending data, RECEIVE, for receiving data, and CHCLOSE, with which channels are closed. The notion of a channel is that of a TCP connection, and the channel primitives map directly to the TCP primitives OPEN, SEND, RECEIVE and

CLOSE, respectively. A channel identifier, the name given a channel, maps to the TCP socket. The only channel idea not in TCP is that of specifiable data byte size. All TCP connections pass only eight-bit bytes. This restriction does not affect the model, since inter-MSG network connections need only pass eight-bit bytes. In sum, changing to TCP would not change the abstract model at all, and the refinement of the four channel primitives to their TCP counterparts would be straightforward.

Changing to TCP would make possible substantial optimization of the inter-MSG protocol and a corresponding simplification of the abstract model. The fundamental idea is to model inter-MSG transactions in the same way as local transactions, with a separate channel per pending event. This would eliminate the negotiation currently required by the inter-MSG protocol. The optimization is made possible by TCP's reliable transmission feature, which obviates the need for inter-MSG confirmation messages.

In the abstract model, each MSG user primitive that cannot be satisfied immediately (a pending event), has associated with it a channel between the user process and MSG, used for passing the final result of the primitive to the user. In a similar manner each pending event can have associated with it a network channel to the appropriate remote MSG, over which inter-MSG communication passes. In the case of messages and alarms, after the user executes a send primitive MSG executes a non-blocking channel SEND, which automatically opens the channel. When the receiving MSG is prepared to receive a message or alarm, it executes a corresponding RECEIVE. The process class and instance of the sender and receiver are encoded in their respective channel identifiers (socket numbers). When both automatic opens complete the data are transmitted and the channel closed. No confirmation from the receiving MSG is necessary since TCP will see to that. The case of direct connections is similar, the difference being that connection identifiers are exchanged and the channels are then passed to the user processes. The timing of pending events is done through timing the associated network channel primitive. The channel primitive is given the same timeout interval as the pending event. If the primitive fails, the pending event is aborted.

The above discipline simplifies the abstract model considerably, particularly the queue management routines. Twenty three routines are reduced to eleven -- the four message and four alarm routines that initiate output (EnQOutput...), initiate input (EnQReceive...), complete output (Record...Ok), and complete input (EnQInput...), and the three direct connection routines that initiate a connection (EnQOutputOpenConn), complete a connection (EnQInputOpenConn), and close a connection (EnQOutputCloseConn). These routines remain relatively unchanged. The lower level decision routines (exempli gratia, HoldOrRejectMess) become unnecessary. The interface routine DeliverToRemoteHost is changed to reflect the channel strategy. The network server routines are simplified, since formatted MSG protocol items are no longer necessary. The cancelling of failed pending events proceeds as before, but the timer process is no longer needed, since it is subsumed by the timed channel primitives.

References

[model] Holloway, G., Bush, W. and Mealy, G., Abstract Model of MSG, Technical report TR 25-78, Center for Research in Computing Technology, Harvard University, October 1978.

[MSG] MSG design specification, in "Third semi-annual technical report for the National Software Works," Massachusetts Computer Associates, Wakefield, Mass., February 1977

[TCP] Postel, J., Specification of Internetwork Transmission Control Program, Version 4, Information Sciences Institute, University of Southern California, September 1978.