                                                 E. Harslen
                                                 J. Heafner
Network Working Group   RANL
Request for Comments: 50   4/30/70

                    Comments on the Meyer Proposal

We find the Meyer proposal (Note #46) to be the most acceptable
to dare, for exactly the reasons that he enumerates; viz., simple,
suffices for most planned uses of the Network, easy to implement,
can be extended. It does not encompass everything that has been
suggested recently, however, we do agree with the items that are
proposed and we feel that the missing features are probably not
worth doing battle over and thus delaying the specification.

We make the following comments on the seven issues rasied in
Note #47.

    1)   We agree with Steve that dynamic reconnection will later
         be required for more sophisticated uses of the Network.
         We also agree with the Project MAC people that it
         unnecessary initially. A better job can be done of dynamic
         reconnection given some Network experience and the specific
         needs of its use.

    2)   INT is easy to implement and serves a useful purpose.

    3)   We favor including a sub-field for instance tag identifier.
         We see the need for both cases; a) where multiple processes
         should appear indistinguishable, and b) where a given
         user owning multiple processes must distinguish among
         them. Those program parts that should not distinguish
         among processes should simply ignore the instance tag.
         Tom's suggestion to use part of the user number sub-field
         merely reduces the combined length of sub-fields from 32
         bits to 24 bits; the problem remains.

    4)   We disagree with both Steve and MAC in that no special
         structure should be imposed on the data transmitted. We
         prefer the "message data type" mentioned by E. I. Ancona,
         Note #42, page 1. An example of its use was cited in
         Note #39, page 2, transmit vs broadcast.

With regard to a standard character set, we strongly
support adopting one in the beginning, and in particular
ASCII. We have observed that most sites have previously
suggested ASCII. Is there anyone who objects?

5) Word boundary alignment is more attractive than double
    padding.

6) Steve's suggestion of short-term queueing of RFCs is
    acceptable as an option.

7) We support the UCC in Note #46 for three principle reasons:

    a) In general the user should not know the remote socket
        code of the process to whom he wishes to communicate.

    b) The additional duplex connection can provide some
        superfisory control over process behavior, possibly
        in conjunction with the interrupt procedure.

    c) Most of the other proposed methods demand queueing.

    We think there must be a standard UCC, yet we encourage
    parallel experimental UCCs.

We make two additional comments on Note #46 that were not reiterated
in Note #47.

BLK and RSM are more straightforward than previous suggestions and
they do not deny multiplexing over a given link. With regard to
the use of links, we refer to an example given by Bob Kahn where
an intermediate IMP goes down and eats some's RFNM. This
should not necessitate reconnection.

In Note #46, page 6, the statement that the UCC has the ability
to close connections to a dead process is installation dependent.
In our particular case the NCP is notified directly of process
failure due to the particular software interface through which all
processea, including NCP, must communicate.

JFH:hs

        [This RFC was put into machine readable form for entry]
          [into the online RFC archives by Gary Okada 7/97]