Network Working Group
Request for Comments: 979

Andrew G. Malis
BBN Communications Corp.
March 1986

PSN END-TO-END FUNCTIONAL SPECIFICATION

Status of this Memo

This memo is an updated version of BBN Report 5775, "End-to-End Functional Specification". It has been updated to reflect changes since that report was written, and is being distributed in this form to provide information to the ARPA-Internet community about this work. The changes described in this memo will affect AHIP (1822 LH/DH/HDH) and X.25 hosts directly connected to BBNCC PSNs. Information concerning the schedule for deployment of this version of the PSN software (Release 7.0) in the ARPANET and the MILNET can be obtained from DCA. Distribution of this memo is unlimited.

1 Introduction

This memo contains the functional specification for the new BBNCC PSN End-to-End (EE) protocol and module (PSN stands for Packet Switch node, and has previously been known as the IMP). The EE module is that portion of the PSN code which is responsible for maintaining EE connections that reliably deliver data across the network, and for handling the packet level (level 3) interactions with the hosts. The EE protocol is the peer protocol used between EE modules to create, maintain, and close connections. The new EE is being developed in order to correct a number of deficiencies in the old EE, to improve its performance and overall throughput, and to better equip the PSN to support its current and anticipated host population.

The initial version of the new EE is being fielded in PSN Release 7.0. Both the old and new EEs are resident in the PSN code, and each PSN may run either the old or the new EE (but not both) at any time, under the control of the Network Operations Center (NOC). The NOC has facilities for switching individual PSNs or the entire network between the old and new EEs. When the old EE is running, PSN 7.0’s functionality is equivalent to that provided by PSN 6.0, and the differences listed in this memo do not apply. Hosts on PSNs running the old EE cannot interoperate with hosts on PSNs running the new EE.

There are two additional sections following this introduction. Section two describes the motivation and goals driving the new EE project.

Section three contains the new EE’s functional specification. It describes the services provided to the various types of hosts that

are supported by the PSN, the addressing capabilities that it makes available, the functionality required for the peer protocol, and the performance goals for the new EE.

Two notes concerning terminology are required. Throughout this document, the units of information sent from one host to another are referred to as "messages", and the units into which these messages are fragmented for transmission through the subnetwork are referred to as "subnet packets" or just "packets". This differs from X.25's terminology; X.25 "packets" are actually messages. Also, in this report the term "AHIP" is used to refer to the ARPANET Host-IMP Protocol described in BBN Report 1822, "Specifications for the Interconnection of a Host and an IMP".

2 Motivation

The old EE was developed almost a decade ago, in the early days of packet-switching technology. This part of the PSN has remained stable for eight years, while the environment within which the technology operates has changed dramatically. At the time the old EE was developed, it was used in only one network, the ARPANET. There are now many PSN-based networks, some of which are grouped into internets. Originally, AHIP was the only host interface protocol, with NCP above it. The use of X.25 is now rapidly increasing, and TCP/IP has replaced NCP.

This section describes the needs for more flexibility and increases in some of the limits of the old EE, and lists the goals which this new design should meet.

2.1 Benefits of a New EE

Network growth and the changing network environment make improved performance, in terms of increasing the PSN's throughput, an important goal for the new EE. The new EE reduces protocol traffic overhead, thereby making more efficient use of network line bandwidth and transit PSN processing power.

The new EE provides a set of network transport services which are appropriate for both the AHIP and X.25 host interfaces, unlike the old EE, which is highly optimized for and tightly tied to the AHIP host interface.

The new EE has an adjustable window facility instead of the old EE's fixed window of eight outstanding messages between any host pair. The old EE applies this limit to all traffic between a pair of hosts; it has no notion of multiple independent channels or

connections between two hosts, which the new EE allows. A network with satellite trunking, and consequently long delays, is an example of where the new window facility increases the EE throughput that can be attained. TACs and gateways provide another example where the old EE's fixed window limits throughput; all of the traffic between a host and a TAC or a gateway currently uses the same EE connection and is subject to the limit of eight outstanding messages, even if more than one user's traffic flows are involved. With the new EE, this restriction no longer applies.

Supportability also motivates rewriting the EE software. The new EE can be written using more modern techniques of programming practice, such as layering and modularity, which were not as well understood when the old EE was first designed, and which will make the EE easier to support and to enhance.

Finally, the new EE includes a number of new features that improve the PSN's ability to provide services which are more closely optimized to what our customers need for their applications. These include new addressing capabilities, precedence levels, end-to-end data integrity checks, and monitoring and control capabilities.

2.2 Goals for the New EE

The new EE's X.25 support is greatly improved over that provided by the old EE. One element of this improvement is at least halving the amount of per-message EE protocol overhead. Another element is the unification of the different storage allocation mechanisms used by the old EE and X.25 modules, where data transferred between the old EE and X.25 must be copied from one type of structure to the other.

The new EE presents, as much as possible, a non-blocking interface to the hosts. If a host overwhelms the PSN with traffic, the PSN ultimately has to block it, but this should happen less frequently than at present.

In the old EE, all of the hosts contend for the same pool of resources. In the new EE, fairness is enforced in resource allocation among different hosts through per-host minimum allocations for buffers and connection blocks as part of a general buffer management system. This insures that no host can be completely "shut out" of service by the actions of another host at its PSN.

The EE supports four precedence levels and optional (on a per-network basis) preemption features.

Addressing capabilities have been extended to include hunt groups.

Instead of a fixed window of eight outstanding messages between any host pair, the maximum window size on an EE connection is configurable to a maximum of 127. The EE allows host pairs to set up multiple connections, each with an independent window.

A result of the old EE's reliance on destination buffer reservation is that subnet packets can be lost if an intermediate node goes down. The new EE uses source buffering with retransmission in order to provide more reliable service.

The new EE has a duplex peer protocol, allowing acknowledgments to be piggybacked on reverse traffic to reduce protocol overhead. When reverse traffic is not available, acknowledgments are aggregated and sent together.

The result of this development will be end-to-end software with greater performance, supportability, and functionality.

3 End-to-End Functionality

This section contains the new EE's functional specification. It describes the services provided to the various types of hosts that are supported by the new EE, the addressing capabilities that it makes available, the functionality required for the peer protocol, the performance goals for the new EE, the EE's network management specification, and provisions for testing and debugging.

3.1 Network Layer Services

The most important part of designing any new system is determining its external functionality. In the case of the new EE, this is the network layer services and interfaces presented to the hosts.

3.1.1 Common Functionality

The following three sections list details concerning the new EE's support for the X.25, AHIP and Interoperable network layer services. In the interest of brevity, however, additional functionality available to all three services is listed herein:

- In order to check data integrity as packets cross through the network, the old EE relies on a trunk-level,

hardware/ firmware-generated, per-packet CRC code (which is either 16 or 24 bits in size, depending on the PSN-PSN trunk protocol in use) and a software-generated per-packet 16-bit checksum. Neither of these are end-to-end checks, only PSN-to-PSN checks. For the new EE, the software checksum has been extended to be an optional 32-bit end-to-end checksum, and the per-packet software checksum has been reduced to a parity bit.

The network administration now has a choice as to which is most important, efficient utilization of network trunks (due to the reduced size of the per-packet headers), or strong checks on data integrity.

Those hosts that require strong data integrity checking can request, in their configuration, that all messages originating from this host include a 32-bit per-message end-to-end checksum. This checksum is computed in the source PSN, is ignored by tandem PSNs along the path, and is checked in the destination PSN. If the checksum does not check, the EE's regular source retransmission facilities are used to have the message resent.

- The old EE's access control mechanism allows 15 separate communities of interest to be defined, and uses an unnecessarily complicated algorithm to define which communities can intercommunicate. This mechanism is being expanded to allow 32 communities of interest, rather than the previous limit of 15. The feature that allowed hosts to communicate with a community without actually being a member of that community has been removed because it was never utilized.

- The addressing capabilities of the PSN have been improved by the new EE. In addition to continuing to support the old EE's logical addressing facility, hunt groups (for both AHIP and X.25 hosts) have been added. These are described further in Section 3.2.

- Connection block preemption is supported on a configurable per-network basis. If a network is configured to use connection block preemption, then lower-precedence connections can be closed by the PSN, if necessary, in order to maintain configured reserves of PSN resources for higher-precedence connections.

- The new EE supports congestion control and improved resource allocation policies which ensure fairness and graceful degradation of service under extreme load. Certain resources can be prereserved to each host port, and each port can also be limited in its use of shared resources. This ensures that no host can be totally shut out from PSN resources by the actions of other hosts at the same PSN. In addition, each PSN is sensitive to congestion in both of the PSNs at the endpoints of each connection, and it can exert backpressure (flow control) on hosts, as necessary, to prevent congestion.

3.1.2 X.25

The new EE's X.25 service represents an improvement over the X.25 service available from the old EE. The following paragraphs summarize the X.25 support in the new EE:

- The new EE provides both DDN Standard and Basic X.25 service, as described in BBN Reports 5476, "DDN X.25 Host Interface Specification," and 5500, "C/30 PSN X.25 Interface Specification," respectively. In addition, the description of DDN Standard Service, Version 2, is found in Section 3.1.4 of this document.

- All data packets and call requests are source-buffered in the source PSN to provide a better level of reliability for network traffic. This should keep the network from issuing a reset on an open connection as a result of a lost packet in the subnet or any other occasional subnetwork failure. Except in cases of extreme network or node congestion, recovery from lost subnet packets is automatic and transparent to the end user or host.

- Both local and end-to-end significance for host window advancement (based upon the D bit from the host) are planned, but only end-to-end significance is included in the initial release (the old EE did not include local significance). The D bit is passed through the network transparently.

3.1.3 AHIP

Another service provided by the new EE is defined in BBN Report 1822, "Specifications for the Interconnection of a Host and an IMP", as amended by Report 5506, "The ARPANET 1822L Host Access Protocol". This ARPANET Host-IMP Protocol (AHIP) service is

supported in a backwards-compatible manner by the new EE; since this is a BBNCC-private protocol, the new EE can improve the service to better match its current uses (the AHIP protocol was first designed over twelve years ago). The main changes to AHIP are to remove the absolute eight-message-in-flight restriction for connection-based traffic, and to improve the PSN's "datagram" support for non-connection-based traffic.

For this new support, datagram service is planned (for PSN Release 8.0) to include fragmentation and reassembly by the network, but without requiring the network overhead used by connections, and without the reliability, message sequencing, and duplicate detection that connections provide. However, "destination dead" indications will be provided to the source host where possible and appropriate.

With the new EE, hosts are also able to create multiple connections between host pairs by using the 8-bit "handling type" field to specify up to 256 different connections. The field is divided into high-order bits that specify the connection's precedence, and low-order bits that distinguish between multiple connections at the same precedence level. Since the new EE is using four precedence levels, the handling type field is used to specify 64 different connections at each of the four precedence levels.

AHIP connections will continue to be implicitly created and automatically torn down after a configurable period (nominally three minutes) of inactivity, or because of connection block contention.

To summarize the new end-to-end's AHIP support:

- The old EE's AHIP services are supported in a backwards-compatible manner (except where listed below).

- The old EE's uncontrolled (subtype 3) message service will be replaced, in PSN Release 8.0, by the datagram service mentioned above. This service will provide fragmentation and reassembly, so that there is no special restriction on the size of datagrams; will not insure that messages are delivered in order or unduplicated, or provide a delivery confirmation; will notify the source host if the destination host or PSN is dead; will not require the connection block overhead associated with connections; and may lose messages in the subnet, without notification to the source host, in the event of subnet

congestion or component failures. This service could be useful for applications that do not need the absolute reliability or sequentiality of connections and therefore wish to avoid their associated overhead.

Datagrams are not supported by the new EE in PSN Release 7.0.

- Connections no longer have the old EE's "eight messages in flight" restriction, and a pair of hosts can be connected with up to 256 simultaneous implicit connections. In addition, multiple precedence levels are supported.

- The new EE supports interoperability between AHIP and X.25 hosts (see Section 3.1.4 for further details).

- AHIP local, distant, and HDH (both message and packet mode) hosts are supported. The new EE does not support VDH hosts. VHA and 32-bit leaders are supported.

- Packet-mode HDH has been extended to allow longer packet data frames (see BBN Report 1822, Appendix J, for a description of the HDH protocol). Middle packet frames can now contain up to 128 octets of data, rather than the previous 126 (although there must still be an even number of octets per frame). Last packet frames can now contain up to 127 octets of data, rather than the previous 125, and the number of octets need not be even. However, the maximum total message size is still 1007 data octets. The PSN uses these new packet frame size limits when sending packet frames to packet-mode HDH hosts unless the host is configured to allow only 126-octet frames. In addition, there are restrictions on packet-mode HDH when interoperating with DDN Standard X.25 hosts; these restrictions are discussed in Section 3.1.4.

3.1.4 Interoperability (DDN Standard X.25)

One of the main goals of the new EE is to provide interoperability between AHIP and X.25 hosts. On the surface, this may appear difficult, since the two host access protocols have little in common: X.25 presents a connection-oriented interface with explicit windowing, while AHIP presents a reliable datagram-oriented interface with implicit flow control. However, they both have the same underlying

functionality: they allow the hosts to submit and receive messages, and they both provide a reliable and sequenced delivery service.

The key to interoperability is the fact that in the new EE, both X.25 and AHIP connections use the same underlying protocols and constructs. The new EE has AHIP and X.25 Level 3 modules that translate between the specific host protocols and the EE mechanisms. Since these Level 3 host modules share a common interface with the EE, the fact that the two hosts on either side of an EE connection are not using the same access protocol is largely hidden.

As a result, the new EE supports basic interoperability. However, there are some special cases that need to be mapped from one protocol to the other, or just not supported because no mapping exists. For example, AHIP has no analogue of X.25's Interrupt packet, while X.25 does not support an unreliable datagram service such as AHIP's subtype 3 messages. For each of these cases, the recommendations of BBN Report 5476, "DDN X.25 Host Interface Specification," have been followed.

The interoperable service provided by the new EE is called DDN Standard Service, Version 2. Standard Service, Version 1, is defined in BBN Reports 5760, "Preliminary Interoperable Software Design," and 5900 Revision 1, "Supplement to BBN Report Nos. 5476 and 5760".

The major differences between Versions 1 and 2 are:

- Version 2 offers improved performance over Version 1.
- The EE now provides four precedence levels. Therefore, the four precedence levels allowed in the DDN-private Call Precedence Negotiation are mapped directly to subnet precedence levels, instead of being collapsed into two subnet precedence levels as in Version 1.
- On an interoperable connection, the X.25 protocol ID in an X.25-originated message is translated to an AHIP link number (the upper eight bits of the message-ID field) using a lookup table. Version 1 supports only the IP protocol ID and corresponding link number of 155 (decimal). Version 2 allows new values to be added to the lookup table. At present, IP is the only protocol supported. In addition, the AHIP link number is also used to distinguish one connection from another. This

guarantees that when an AHIP host is sending messages to an X.25 host, messages using different link numbers come into the X.25 host on different X.25 connections.

- Since a "translation module" is no longer necessary in the PSN, interoperable connections now have end-to-end significance, with a direct correspondence between X.25 RRs and AHIP RFNMs. This preserves the meaning of the RFNM as defined in Report 1822. Although Release 7.0 only offers end-to-end significance, the D bit is passed transparently on Standard Service connections between two X.25 hosts.

- Up to 256 simultaneous connections are supported between host pairs that are using the same addresses and precedence levels. Version 1 only supported one such connection.

The following Version 1 services are not offered by Version 2:

- Permanent Virtual Circuits.

- X.25 protocol bypass (a BBN-private service).

A number of items in Report 5760 were the subject of some discussion, and three of them need to be specifically mentioned here. First, for DDN Standard Service, Version 1, acknowledgments have local significance only, and the D bit must be set to 0 in the call request. In DDN Standard Service, Version 2, only end-to-end significance is being provided, as was mentioned above. For backwards compatibility with Version 1, the D bit can be set to 0 or 1 in a call, but hosts are advised that only end-to-end significance is provided in Version 2.

Second, non-standard Default Precedence is not supported by either Standard Service Version 1 or Version 2. Support for this facility in Version 1 was withdrawn at the request of DCA.

Third, although DTEs are allowed to request maximum packet sizes of 16, 32, and 64 octets, the DCE always negotiates up to 128 octets, as per Section 6.12 ("Flow Control Parameter Negotiation") of the CCITT 1984 X.25 Recommendation. This is true of both Version 1 and Version 2. Since IP and TCP are required when Standard Service is in use, this is a reasonable restriction (due to the length of IP and TCP headers).

One issue must be raised concerning interoperability between X.25 and packet-mode HDH hosts. In order to efficiently interoperate, packet-mode HDH hosts should completely fill their middle packet frames with 128 octets of data. Packet-mode HDH hosts that send or require receiving middle packet frames with less than 128 octets of data can still interoperate with X.25 hosts, but at a greater expense of PSN CPU resources per message.

3.2 Addressing

The old EE supports, for both AHIP and X.25 hosts, two forms of host addressing, physical and logical.

Physical addressing consists of identifying a host port by the combination of its PSN number and the port number on that PSN. Logical addressing allows an arbitrary 16-bit "name" to refer to a list of one or more host ports. The EE tries to open a connection to one of the ports in the list according to the criterion chosen for that name: first reachable in the ordered list, closest port (in terms of routing delay), or round-robin load sharing.

For the new EE, logical addressing is supported on an explicit per-connection basis: all logical-to-physical address translations take place in the source PSN when a connection is established. Once this translation has occurred, all data messages on the connection are sent to the same physical address.

In addition, hunt groups are also now supported for both X.25 and AHIP hosts. This new capability allows host ports on a destination PSN to be combined into a "hunt group". The ports share the same group identifier, and incoming connections are evenly spread over the ports in the group. This differs from logical addressing’s load sharing, where all name translations take place in the source PSN, the different ports can be on any number of PSNs, and the load sharing is on a per-source-PSN basis. By contrast, all of the host ports in a hunt group are on the same PSN, the group-to-port resolution takes place in the destination PSN, and the load sharing of incoming connections can be guaranteed over the ports by the destination PSN. For X.25, hunt groups comply with Section 6.24 of the 1984 X.25 Recommendation. Note that Called Line Address Modification is not supported.

3.3 Protocol Functionality

The EE peer protocol runs between EE modules in PSNs on either end of an EE connection. This protocol and its mechanisms have to perform the following functions:

- Provide full duplex connections (the old EE provides simplex connections, and any two-way traffic, such as that generated by TCP, requires two subnet connections).
- Open a connection and optionally send a full message's worth of data as a part of the open request (the old EE requires a separate opening sequence in each direction before data can flow).
- Reliably send connection-oriented messages, properly fragmented/reassembled and sequenced.
- Close (clear) a connection (normally, or in a "clean-up" mode after a host or PSN dies).
- Reset a connection (like the X.25 reset procedure).
- Be able to send a limited amount of out-of-band traffic associated with a connection (like the X.25 interrupt).
- Use source buffering with message retransmission (after a timeout) to insure delivery (the old EE depends on destination buffer preallocation, which adds protocol overhead and cannot recover from lost packets in the subnet).
- Use an internal connection window of up to 127 messages.
- Support two types of ACKs, Internal ACKs (IACKs) and External ACKs (EACKs), which are further described following this list
- Have an inactivity timer for each connection. For AHIP and Standard X.25, the connection is closed if the timer fires. For Basic X.25, the EE uses an internal Hello/I-Heard-You sequence with the PSN on the other end of the connection to check if the other end's host or PSN is still alive. If not, then the connection is closed.
- Be able to gracefully handle resource shortages and avoid reassembly lockup problems.

As mentioned above, the protocol supports two types of acknowledgments, IACKs and EACKs. Both types of ACKs apply to messages only; individual packets are not acknowledged. Since windowing is being used, an individual ACK can be used to acknowledge more than one message.

IACKs are used to cancel the retransmission timer and free source buffering, and are sent when a message has been completely reassembled and delivered from the EE to either the AHIP or X.25 level 3 module. This allows the EE to avoid unnecessary message retransmissions, and speeds up the process of freeing source buffering when destination hosts are slow to accept messages or, in the case of X.25, slow to advance the PSN's window to the destination (X.25 does not specify any time limit for a host to acknowledge that it received a message).

EACKs are used to advance the end-to-end window and to cause one or more end-to-end X.25 RRs or AHIP RFNMs to be sent to the source host. An EACK is sent when an X.25 host acknowledges a message or when an AHIP host actually receives it.

Both types of ACKs are piggybacked, if possible, on reverse traffic to the source PSN (for any connection). Whenever a packet is sent to another PSN, it is filled to the maximum allowed subnetwork packet size with any outstanding ACKs that may be waiting to be sent to that PSN. After a configurable period, all outstanding ACKs for the same PSN are aggregated together and sent. In addition, succeeding ACKs for the same connection can be combined into one, and EACKs can be used to imply that a message is being IACKed as well (if the destination host is speedy enough when receiving or acknowledging messages to allow IACKs and EACKs to be combined).

This ACK aggregation timer interacts with the source buffering retransmission timer in the following manner: whenever a message is sent from a host on one PSN to a host on a second PSN, an IACK is sent back to the first PSN when the message has been completely reassembled by the destination EE, and an EACK is sent when it has been delivered (and perhaps ACKed) by the destination host. The IACK must make it back to the source PSN within the limits of the retransmission timer, or unnecessary retransmissions could be sent across the network. This limits the ACK aggregation timer to being shorter than the source buffering retransmission timer.

If the destination host is quick enough when accepting traffic from its PSN (with respect to the ACK aggregation timer), then the EACK can be combined with the IACK, and only the EACK would be

sent. If the destination host is even quicker, multiple IACKs and EACKs could be combined into one EACK. In the best case, if there is a steady stream of traffic going between the two PSNs in both directions (but not necessarily over the same connection or even between the same pairs of hosts in each direction), then all of the IACKs and EACKs could be piggybacked on data packets and cause no additional network packets other than the data packets already required to send the data messages across the network. In the worst case, however, such as when there is only a one-way flow from a source PSN to a destination PSN and the destination host is very slow to accept the messages from the network, then each data message could result in separate IACKs and EACKs being sent back to the source PSN in individual packets. However, even though the IACKs may cause additional packets to cross the network, they are still less expensive than the source retransmissions that they are used to prevent, and they also serve to free up valuable source buffering space.

3.4 Performance and Capacity Goals

Performance and capacity goals for the new EE include:

- Throughput: The AHIP host-host and host-trunk maximum throughput (in packets/second) will be at least as good as at present, and should improve for those situations that currently entail traffic limitations based upon the old EE's underlying protocol. The current X.25 intrasite host-host and host-trunk throughput will each improve by at least 50%. The store-and-forward throughput for the new EE's X.25-based traffic will improve by at least 100%.

- Connections: The new EE will support at least 500 simultaneous connections per PSN, and will be able to handle at least 50% more call setups per second than at present.

- Buffering: The EE will have at least 400 packet buffers available to source-buffer and/or reassemble messages.

- Network size: The EE protocol and module will use data structure and message field sizes sufficient to support at least up to 255 hosts per PSN and 1023 PSNs per network (however, other PSN protocols and modules presently constrain these figures to 63 hosts per PSN and 253 PSNs per network).

- Other: The EE will support four message precedence levels

and a maximum message length of 1024 bytes. For logical addressing, the EE will support at least 1024 logical names and at least 2048 address mappings per network.