Apple Address Resolution Protocol — Draft Proposal

May 18, 1987

Apple Network Systems Development

Scope: this document specifies a proposal for mapping between a set of physical node addresses (e.g. Ethernet) and a logical set of (possibly) dynamically assigned node addresses (e.g. AppleTalk). Although the Ethernet/AppleTalk mapping is the one of immediate interest, others should be possible under this proposal. This document does not attempt to address the problem of more than 254 AppleTalk nodes on a physical medium.

Notation: to distinguish between the two classes of addresses, we will call that address which is determined by the physical and link layers of the network (e.g. the Ethernet node address) the *physical address*, and that address which is used by higher level protocols (e.g. the AppleTalk node address, as used by DDP) the *logical address*. Note that for a node on a given physical medium there can be only one physical address, but there could be multiple logical addresses if different *protocol sets* are running within the node (e.g. DDP and TCP/IP).

Functions performed by AARP: The Apple Address Resolution Protocol (AARP) sits between the link access and network layers and performs the following functions:

(1) *Initial determination of a unique logical address for a node using a given protocol set.* This address will be unique among all nodes on the physical medium. In the case of AppleTalk, this logical address will be dynamically determined, although it could be assigned by other means.

(2) *Mapping from logical address to physical address.* Given a logical address, AARP will return the corresponding physical address, or an error indicating that no node on the medium has such a logical address (for the given protocol set).

(3) *Filtering of packets within a given protocol set.* For all packets received by a given node, AARP will verify that the logical destination node address of the packet is equal to the node's logical address or "broadcast" (i.e. an address indicating all nodes). If this is not the case, the packet will be discarded and not delivered to higher levels. This functionality could be considered as part of AARP's client, but is provided here for completeness.

The Address Mapping Table: AARP maintains an Address Mapping Table (AMT). The AMT contains a list of logical addresses and their corresponding physical addresses. The AMT basically serves as a cache of known logical-to-hardware address mappings. Whenever AARP learns of such a mapping (as described below), an entry is made in the AMT. The size of the AMT is implementation dependent — if there is no room for a new entry, entries must be purged using some sort of LRU algorithm. Note that it is concievable that the AMT maintain one entry for each logical address, although this is not necessary (in the case of AppleTalk-to-Ethernet, this would require less than 2K).

Address Resolution: When an AARP client makes a request to determine a physical address, AARP first scans the AMT for the given protocol set. If the given logical address is in the table, AARP simply returns the corresponding physical address. If not, AARP attempts to determine the address by broadcasting an *AARP request packet* on the network. This packet indicates the logical address for which a mapping is desired, as well as the protocol set for that mapping. Nodes receiving an AARP request check to see if their logical address for the given protocol set is the one being requested. If so, they send back, to the requestor, an *AARP reply packet*, indicating their logical-to-hardware address mapping. The requesting AARP then enters this mapping in its table and returns it to its client. If there is no reply within a given timeout period, AARP retransmits the packet as specified below and then returns an error to its client if there is still no response (there is no such node out there).

Choosing an address: At initialization time, AARP needs to determine the logical address for the node under each protocol set it is handling. AARP includes one way of making this assignment; clients are free to use other ways and to inform AARP of the address so determined. The only requirement with these methods is that the addresses are unique for the given medium.

AARP includes the ability, at initialization time, to dynamically pick an unused logical address. Upon being requested to do so by its client, AARP picks a logical address at random. It then sets that address as the node's *tentative* logical address (if there is already a mapping for that logical address in the AMT, AARP picks another random number until a free address is found). AARP then proceeds to broadcast out *probe packets* to determine if anyone else on the network is already using that address. These probe packets contain the tentatively chosen address. Any node receiving a probe packet whose tentative address matches their logical address must respond with an AARP response packet. Upon receiving this packet, the probing node knows the address is in use and proceeds to try another one.

If no response is received, after the number of retries specified below, the logical address is marked *permanent* and returned to the client. To avoid the possibility of two nodes picking the same address at the same time, if a node receives a probe packet for its tentative logical address, it should assume that address is in use and go on to try another one. A node should never respond to AARP probes or requests while it is probing.

Examing packets: in addition to receiving and processing its own packets, AARP must receive and process all packets for any protocol set for which it is currently active (i.e. performing translation). There are two reasons for this. The first is that it must verify that the incoming packet is indeed addressed to its client. At initialization time, AARP's client must tell AARP how to determine this. Details are implementation dependent, but a simple method of doing so would consist of the client passing AARP the offset in the packet at which to look for the destination logical address. The client would also have to pass an indicator as to how to tell if the packet were some type of "broadcast," i.e. a packet intended for a subset of all the nodes on the medium (often actually all nodes on the medium). If the destination neither matches the node's logical address nor is a valid "broadcast" address, AARP must discard the packet and not deliver it to its client; it was sent to this node by mistake.

"Gleaning" information: The second reason for AARP to examine packets for the protocol sets it is dealing with is one of efficiency. At the hardware level, these packets will generally contain the source physical address. At the logical level, these packets will generally contain the source logical address. Thus, once a packet is determined to be valid (i.e. to the correct logical destination), AARP can "glean" the mapping for the source node without ever sending out a packet. AARP can then add this entry to its AMT. Note that if the entry already exists, but specifies a different mapping, the entry should be superseded, as a new node has taken over the logical address (the old node must have "gone down.")

Note that this "gleaning" of source information from client packets is not a requirement of AARP. Certain protocol sets may not even contain the required information. It also may be deemed too inefficient for certain implementations to add an entry to the AMT for each incoming packet.

Source information can also be gleaned from AARP request packets. Since these packets are broadcast, every AARP implementation will receive them. These packets will always contain the source physical and logical addresses. AARP should always add this information to its AMT. Note that AARP should not glean any information from probe packets, as it is tentative.

Aging: An AARP implementation may wish to age AMT entries. With each AMT entry, it would associate a timer. The timer would be restarted every time a packet came in which caused the entry to be confirmed or reset (i.e. a packet which contained that entry's logical address). If the timer reached a certain value, the entry would be deleted from the table. The next time that logical address was asked for, AARP would send out a new request (if it had not gleaned the info since the entry was aged).

Aging is needed to prevent the following situation. A node goes down or takes itself off the net. Another node (with a different physical address) starts up and acquires the logical address of that first node. An AARP implementation in a third node needs to learn about this change in mapping. Unless the new node broadcasts an AARP request, the third node's entry will continue to contain the old (wrong) information. Aging of the entry would solve this problem.

An alternate approach, instead of timed-aging, would be to age an entry whenever a probe packet for that logical address is received. This would guarantee the table would always contain correct information, although it would also result in un-needed aging in the case where a new node probes for an address that's already in use.

Packet details: AARP packets are encased in the standard link access header for the given medium. Following this there is a header which is constant for a given logical/physical mapping. This header consists of a two-byte *physical type*, a two-byte *protocol type*, a one byte *physical address length* and a one byte *logical address length*. The physical type is a pre-defined indicator as to the physical medium type. The protocol type indicates the desired protocol set — again this is a pre-defined constant. The physical address length and logical address length indicate the length, in bytes, of the respective address fields.

Following this standard header is a two-byte *command* field, indicating AARP request, response, or probe. Following this are the physical and logical addresses of the sending node (the length of these fields is indicated by their respective length fields).

Last in the packet are the physical and logical addresses of the node to which the packet is being sent. Note that in the case of an AARP request packet, the physical address of the destination is unknown and should be set to zero. The logical address should be that address for which the physical address if being sought. In the case of a probe packet, both the source and destination logical addresses should be set to the sender's tentative logical address and the destination target address should again be set to zero. Figure 1 diagrams a generic AARP packet.

Retransmission details: AARP must retransmit both probes and requests until either a reply is received or a maximum number of retries is exceeded. The specifics of the retransmit count and interval are dependent on the characteristics of both the physical medium and protocol set used, in addition to the desired thoroughness of the search. In general, probe retransmission must be fixed for a given physical/logical combination, but request retransmission can be left as a client-dependent parameter.

Packet specifics: The following constants are currently defined for AARP:

Protocol Type for Ethernet-like media (in LAP header): $80F3

AARP physical type for Ethernet: $0001
AARP (and Ethernet) protocol type for AppleTalk: $809B
AARP physical address length for Ethernet: 6
AARP logical address length for AppleTalk: 4 [first three bytes of the address **must** be zero -- these are reserved by Apple for future use]

AARP request command = $0001
AARP response command = $0002
AARP probe command = $0003

AARP probe retransmission interval for Ethernet/AppleTalk: 1/30 second
AARP probe retransmission count for Ethernet/AppleTalk: 20

Figure 2 illustrates AARP packets for the specific case of AppleTalk/Ethernet.

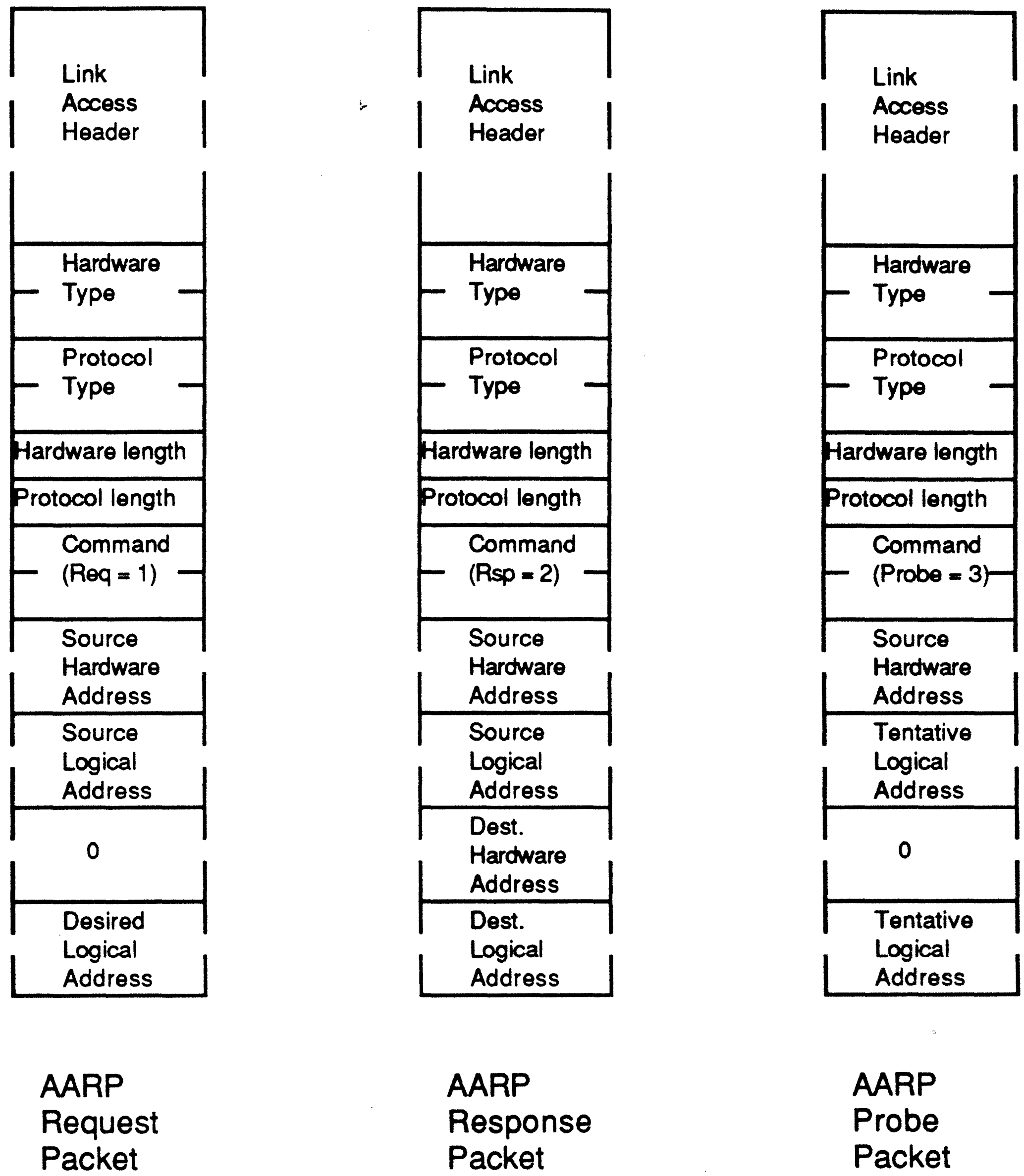

Figure 1 — Generic AARP Packet Formats

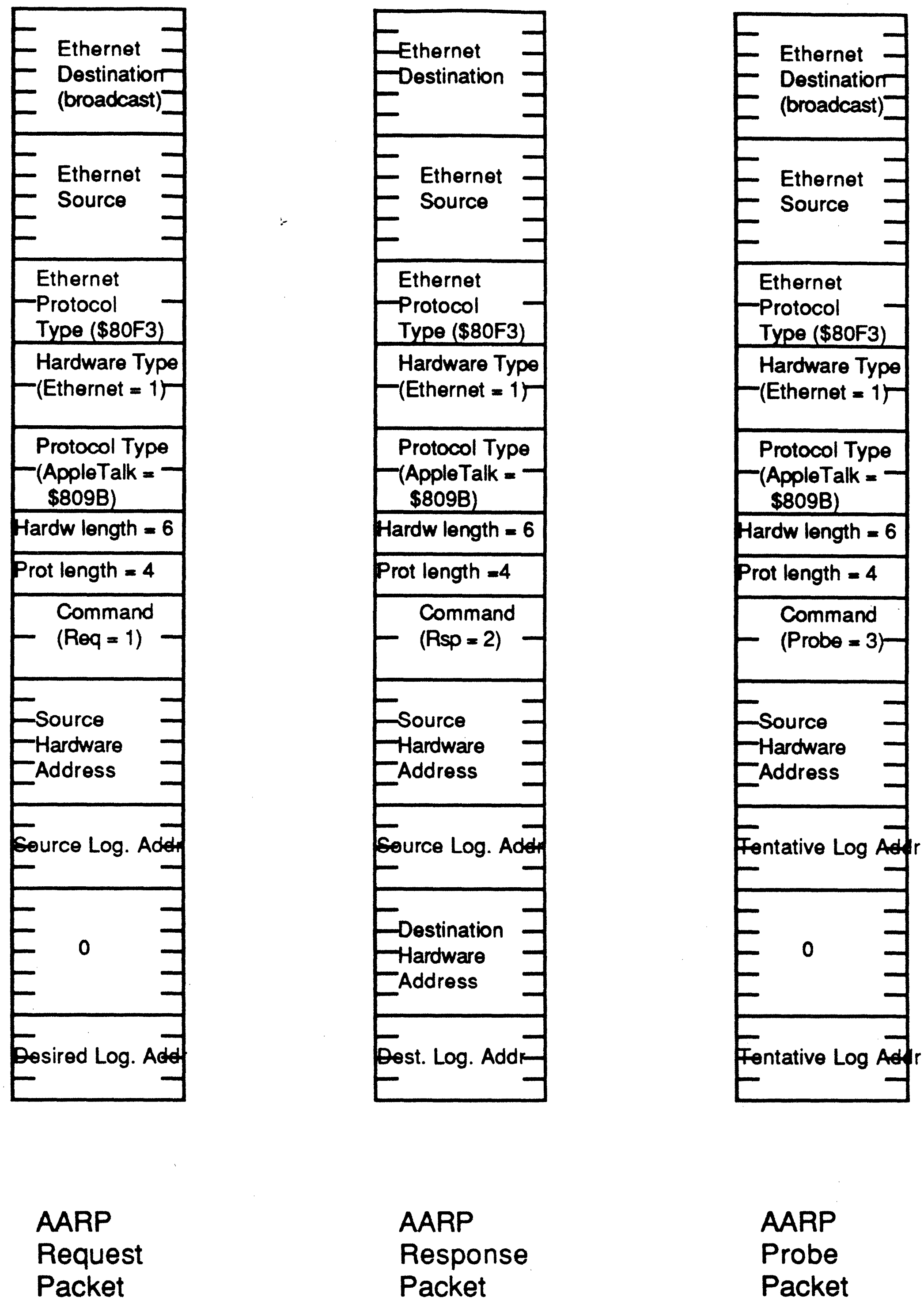

Figure 2 — Ethernet/AppleTalk AARP Packet Formats