Data Packet (Frame) Standards

Over the years, several frame types have been defined for Ethernet:

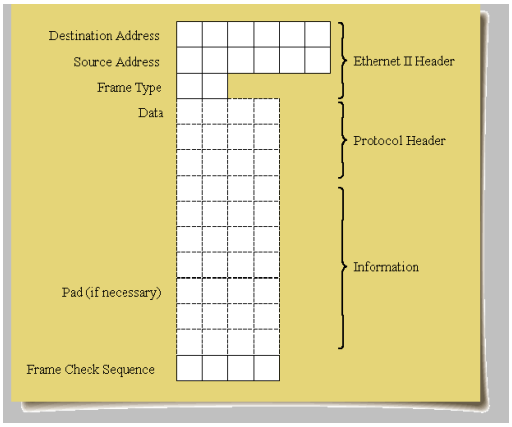
- Ethernet II
- IEEE 802.3 "raw"
- IEEE 802.3 with 802.2
- IEEE 802.3 with 802.2 SNAP

These frame types are discussed in more detail below, along with a little of the history of how they were developed.

Ethernet II. In the early days of computer networks, Digital, Intel, and Xerox got together and specified a networking standard that they called Ethernet. This standard included the definition of a data-link-level access method and a packet format which shared the Ethernet name.

The Ethernet II standard specified that a header be added to the data before sending it on the network medium. The frame format follows the rules to access a network using the CSMA/CD access method. Figure 1 illustrates the fields defined for Ethernet II data packets.

Figure 1: Ethernet II frame definition.



A *preamble* (not shown) is used to synchronize the receiving stations and to indicate that data is about to begin.

The *destination address* and *source address* fields contain the local node addresses of the sender and intended receiver, respectively.

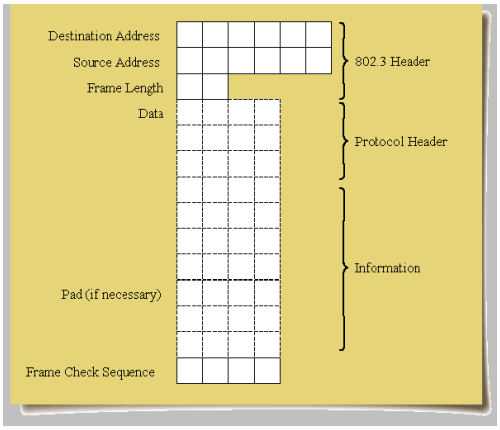
The *frame type* field indicates which upper-layer protocol (such as NetWare's IPX/SPX or the Internet's IP) should be used to interpret the information in the data portion of the packet. The values for this type field were defined and managed by Xerox. For example, Novell was assigned the hexadecimal value 8137 for IPX/SPX.

The *data* part of the packet contains the protocol header and the actual information being transmitted. The minimum size of an Ethernet packet is 64 bytes. If there is not enough data to fill the packet, a *pad* field is added to make the packet size equal to the minimum.

The *frame check sequence* (FCS) field is used to check the integrity of the packet. Before placing a packet out on the wire, the sending node takes all the bytes within the packet (excluding the preamble and the FCS field itself), performs a mathematical calculation called a cyclical redundancy check, and places the result at the end of the packet. When the packet arrives at the destination, the receiving station performs the same mathematical calculation and should receive the same result. If not, it assumes something has been corrupted and discards the packet. This is known as bit-level error checking.

IEEE 802.3. Eventually, both the Ethernet media and packet format were pursued by the standards committees of the IEEE. Working from the original Digital-Intel-Xerox specification, IEEE proposed its own Ethernet standard which they called 802.3 (named after the committee that worked on it). The IEEE 802.3 frame format is almost identical to the Ethernet II format. The only difference is that a *length* field is used in place of the type field, as shown in Figure 2. This field indicates the length of the data portion of the 802.3 packet. (The maximum length value for Ethernet/802.3 frames is 1518 decimal.)

Figure 2: IEEE 802.3 frame definition.



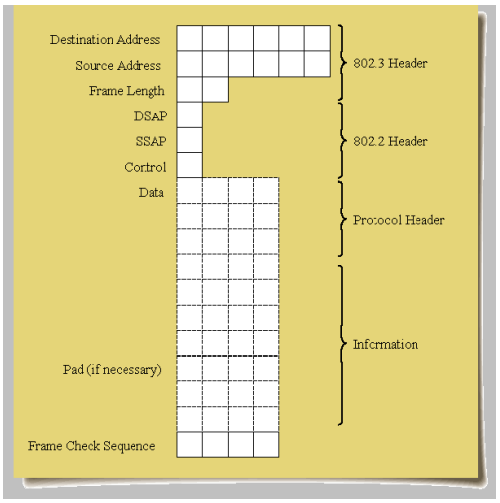
You might wonder how a router can tell if it's receiving an Ethernet II frame or an 802.3 frame. As it turns out, the assigned values for the Ethernet II *type* field are always greater than 1500 decimal. Since the maximum frame size for Ethernet is 1518 bytes, the length field will always contain a value less than that. This is the only way for routers to tell the difference between the two frame types.

Without a protocol type field, however, it is impossible to determine what protocol to use for interpreting the encapsulated data in an 802.3 frame. If more than one upper-layer protocol exists on the network, the packet may be incorrectly routed.

IEEE 802.3 with 802.2. The information necessary to properly route 802.3 packets is provided in the IEEE 802.2 standard, which wasn't fully developed until quite some time after the 802.3 standard. The 802.2 header envelopes the data before it is encapsulated within an IEEE 802.3 header. Because the 802.2 fields comprise the Logical Link Control (LLC) layer, the framed data is sometimes referred to as the Logical Link Control Protocol Data Unit (L-PDU). For brevity, we'll refer to the standard IEEE 802.2 data-link layer header enclosed by an IEEE 802.3 physical layer frame as the "IEEE 802.2" frame type.

The IEEE 802.2 frame format adds several fields to the header: a destination service access point (DSAP), a source service access point (SSAP), and a control field, as shown in Figure 3.

Figure 3: IEEE 802.2 frame definition.



A *service access point* denotes the point of service the packet is intended for, or what upper-layer protocol is supposed to use the data. Both the DSAP and the SSAP fields contain values that identify the upper-layer protocol type of the packet. For example, NetWare IPX/SPX packets will contain the hexadecimal value E0 in the DSAP and SSAP fields.

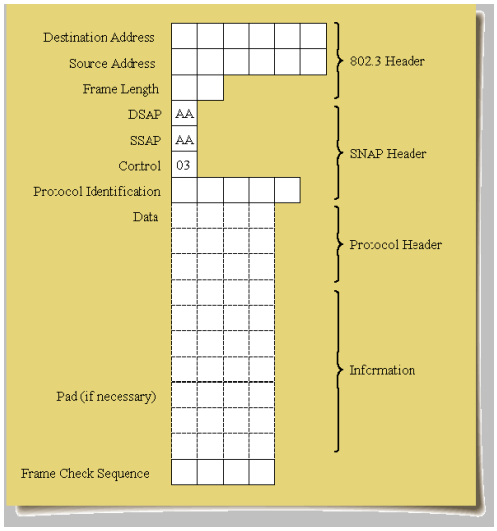
The *control field* is used by certain protocols for administrative purposes. Currently, NetWare's IPX/SPX protocols do not use this field other than to set its value to 03, which denotes 802.2 unnumbered format.

When a receiving station receives an IEEE 802.2 packet, it can determine that it is an 802.3 frame with an 802.2 header inside of it. It can then determine the protocol type from the information in the DSAP and SSAP fields.

IEEE 802.2 with SNAP. After the 802.2 frame was defined, there was some concern that the one-byte DSAP and SSAP fields were not adequate for the number of protocols that might eventually need to be identified. One might think that 256 is more than enough, but a lot of the possible values were reserved from the beginning.

In response to pressure from Apple Computer and the TCP/IP community, another frame standard was defined for both Ethernet and Token-Ring. It was called the subnetwork access protocol (SNAP). This frame type adds a five-byte *protocol identification* field at the end of the 802.2 header, as shown in Figure 4. This is where the protocol is identified.

Figure 4: IEEE 802.2 SNAP frame definition.

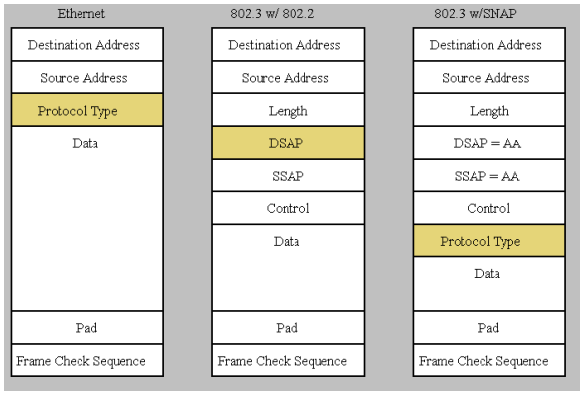


To distinguish an IEEE 802.2 SNAP packet, the value of the DSAP and SSAP fields in the 802.2 header are both set to AA. If a router finds AA in the DSAP and SSAP fields, it knows this is a SNAP-based packet and it should look for the protocol type in the *protocol identification* field.

Identifying the Incoming Protocol

In computer networks that incorporate multiple protocol stacks, a field for specifying upper-layer protocols must exist in the header information. Figure 5 indicates which field is used for this purpose in Ethernet II, IEEE 802.2, and IEEE 802.2 SNAP frames.

Figure 5: Fields used to identify the incoming protocol.

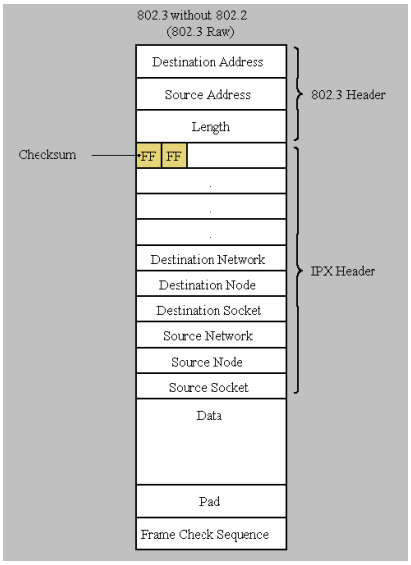


As mentioned previously, Digital's Ethernet II frame type uses a frame type field in place of the length field in the 802.3 header to specify the protocol type. The other two frame types contain fields set aside specifically for identifying the protocol.

When Novell initially implemented its IPX/SPX protocols on Ethernet, IEEE had not fully developed the 802.2 standard. For this reason, we used the 802.3 frame specification without the 802.2 header. But the 802.3 standard wasn't meant to be used all by itself; it was meant to be combined with the 802.2 Logical Link Control information. Thus the IEEE 802.3 frame used *without* 802.2 was nicknamed the 802.3 "raw" frame type. Novell's IPX is the only protocol that uses the 802.3 raw frame type.

To accommodate additional protocols on a network, Novell decided to use the first two bytes in the data portion of the packet, the IPX *checksum* field, to identify an 802.3 raw frame using the IPX/SPX protocol. All LAN drivers would use the value 0xFFFF in these two bytes to designate the packet as 802.3 raw, as shown in Figure 6.

Figure 6: Novell's Ethernet_802.3 "raw" packet always has the value 0xFFFF in the checksum field of the IPX header.



Notice that NetWare's IPX header is located at the beginning of the data portion of the 802.3 raw packet. Thus the IPX header always begins with FF-FF. The IPX header also contains information such as the destination and source network, node, and socket addresses.

To identify a packet as 802.3 raw in a mixed environment, receiving stations must first determine whether the value in the length field is less than 1518 bytes (to distinguish it from an Ethernet II frame type). If so, they check the value of the next two bytes. If it is FF-FF, they know it is an IPX packet because FF-FF isn't currently used in 802.2 DSAP and SSAP fields. This process is illustrated in Figure 7.

Figure 7: Identifying an 802.3 raw (IPX) packet.