1 GSW… Project 802 and the LLC

If you've read the chapter on Ethernet, you might have noticed that the IEEE standard for the Ethernet network is called 802.3. Why .3? What, you might wonder, are 802.1 and 802.2 all about? Come to think of it, we're engineers so we count from zero: so what does 802.0 do?

In brief: 802.0 organises the meetings themselves and writes the rules the other groups have to follow. 802.1 is concerned with the architecture of the networks and protocols, including the interfaces to the higher layers of the protocol stack, virtual LANs and bridges[1]. 802.2 is concerned with the logical link control (LLC) layer: a layer of the protocol stack in-between the multiple-access layer and network layer protocols that provides multiplexing and allows low-level flow control and error control. 802.3 is concerned with the multiple access and physical layers of the IEEE 802 version of Ethernet. Of course, since then a lot of other LAN standards have come along, and other groups have been set up to standardise them, most notably 802.11 (Wireless LANs), 802.15 (Bluetooth and others) and 802.16 (WiMAX).

That's the one paragraph version: for more details, read on…

1.1 Introduction to Project 802

In February 1980, the IEEE project 802[2] had its first meeting. The declared aim of this standards project was to write a standard for a local area network (LAN) running between 1 MBit/s and 20 MBit/s, with the key definition of a LAN being a network in which any station can talk directly to any other station: in other words, no routeing is required.

Over the years since, the limitations on speed and distance were removed, and the project now includes standardisations efforts for networks running at speeds up to 10 GBit/s (and rising), and over metropolitan areas as well. It's now officially known as the IEEE 802 LMSC (LAN / MAN Standards Committee).

Originally, there were four committees formed: the *executive committee* (802.0) which managed the whole process, including organising meetings, approving new projects and resolving disputes between the other committees, and the three original *working groups*: 802.1, 802.2 and 802.3.

The 802.1 working group is responsible for the architecture of the network, and the interface to higher layers of the protocol stack. This includes devices like bridges, virtual LANs and security considerations. The architecture they designed is illustrated in the figure below: there are three or four layers to the protocols, the logical link control layer (LLC) specified by the 802.2 working group, and the multiple-access control (MAC) layer, physical (PHY) and possibly physical media dependent (PMD) layers, specified by the other working groups for each new system.

---

[1] See the chapter on Bridging and Switching in LANs for more about bridges.

[2] Why is called project 802? I've heard some reports that it's because the first meeting took place in February 1980, (hence 80-2), however the project had its name before then. There's another rumour that the original meeting that proposed the whole idea of a LAN standards committee to the IEEE took place in room 802, but the official reason is that 802 was simply the next available number the IEEE could assign to a standards committee. (The IEEE write a lot of standards.)

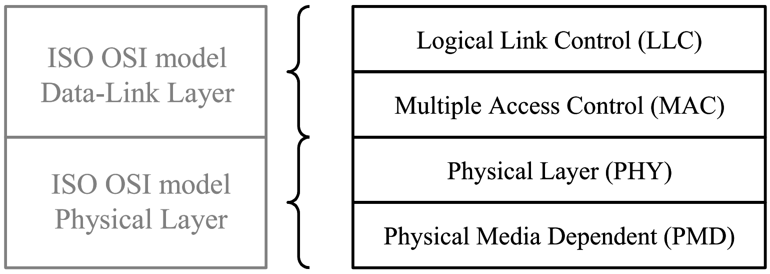

Figure 1-1 Protocol Stack of Project 802 LANs

The logical link control layer provides two main functions: multiplexing (allowing multiple different network layer protocols to run over the same LAN), and error/flow control (requesting re-transmissions of frames that are corrupted, and requesting the transmitter to stop transmitting frames at times when the receiver is not ready to receive them).

The multiple access control layer is responsible for determining when each station is allowed to transmit. 802.3 was set up to use a carrier-sense multiple access with collision detection (CSMA/CD) protocol.

The earlier LAN standards had only a single physical (PHY) layer that covered everything from modulation and coding formats to the design of the connectors. More recent standards sometimes have both a PHY layer (which specifies the modulation and coding formats), and one or more lower PMD layers which specify how the network should work with different types of media, for example copper, single-mode fibre, multimode fibre, etc.

Back to 1980, and it's important to note that the IEEE has a very sensible rule for standards: they won't standardise two solutions to the same problem; this is known as the *distinct identity* criterion[3]. Project 802 was originally set up to write one LAN standard, and they chose the CSMA/CD access method already being used successfully by Ethernet. Unfortunately, that really didn't please General Motors or IBM, both of whom were in the process of developing their own local area networks, and both of whom didn't want a competing system being given a seal of approval by the IEEE.

General Motors and IBM both promptly turned up the to project 802 meetings, and proposed that their systems should be considered for standardisation. Result: deadlock. No one standard could get the support of everyone in the meetings, and the IEEE would not standardise more than one solution to the same problem: connecting up computers in an office environment.

Eventually, the deadlock was broken by some crafty 'interpretations' of the distinct identity criterion, so that all three systems could claim they were addressing slightly different problems. That allowed the 802.0 executive committee to approve two new working groups: 802.4 (Token Bus, the General Motors proposal) and 802.5 (Token Ring, from IBM)[4].

---

[3] This is one of five criteria that all new proposals for standards must satisfy before work can begin writing them. (The other four are broad market potential, compatibility with other IEEE standards, technical feasibility and economic feasibility.)

[4] For more about how Token Ring and Token Bus works (or rather worked, since both standards are effectively obsolete now), see the chapter on Contention-Free Multiple Access.

Since then proposals for new LAN standards have proliferated, and at the last count there were twenty-one working groups (not including the executive committee 802.0), although only eleven of them are actively meeting at the moment[5].

- 802.1 Higher Level LAN Protocols Working Group
- *802.2 Logical Link Control Working Group (Inactive)*
- 802.3 Ethernet Working Group
- *802.4 Token Bus Working Group (Disbanded)*
- *802.5 Token Ring Working Group (Inactive)*
- *802.6 Metropolitan Area Network Working Group (Disbanded)*
- *802.7 Broadband TAG (Disbanded)*
- *802.8 Fibre Optic TAG (Disbanded)*
- *802.9 Integrated Services LAN Working Group (Disbanded)*
- *802.10 Security Working Group (Disbanded)*
- 802.11 Wireless LAN Working Group
- *802.12 Demand Priority Working Group (Disbanded)*
- *(802.13 never existed for marketing reasons)*
- *802.14 Cable Modem Working Group (Disbanded)*
- 802.15 Wireless Personal Area Network Working Group
- 802.16 Broadband Wireless Access Working Group
- 802.17 Resilient Packet Ring Working Group
- 802.18 Radio Regulatory Technical Advisory Group
- 802.19 Coexistence Technical Advisory Group
- 802.20 Mobile Wireless Access Working Group
- 802.21 Media Independent Handoff Working Group
- 802.22 Wireless Regional Area Networks

IEEE Project 802 is without doubt the most important local area networks standardisation project in the world, having written the standards for just about everything except HiperLAN[6], FDDI[7] and ATM[8]: and these other LANs (now mostly obsolete) shared many features (including in most cases the LLC and the addressing scheme) with project 802 LANs.

The remainder of this chapter focuses on the work of the 802.0, 802.1 and 802.2 working groups, and in particular working group 802.2 which deals with the logical link control (LLC) layer that sits directly above all the media access control layers[9].

---

[5] Information as at August 2007: for more details see http://grouper.ieee.org/groups/802/index.html.

[6] HiperLAN is a European wireless LAN standard. The idea of a separate European standard has fallen apart now, with everyone in the industry moving towards 802.11.

[7] Fibre Distributed Data Interface. This was the first 100 MBit/s LAN, now mostly overtaken by 100 MBit/s Ethernet (although FDDI does have some advantages in terms of real-time capabilities).

[8] Asynchronous Transfer Mode (ATM) is much more than just a LAN, but there were a series of LANs based on the idea; these however are almost extinct now.

[9] Note that 802.2 is now hibernating: the LLC is pretty much fixed in stone. If anyone finds a bug, or wants a new variety of LLC written, it'll be woken up again.

1.2 802 LAN Addresses

The IEEE project 802 controls the assignment of LAN card addresses[10] throughout the world. If you want to give a network card a unique address by which it can be recognised, you need to ask the IEEE first, and they will sell you a range of addresses that you can use. This ensures that no two network cards have the same address[11].

802 addresses are 6 bytes (48 bits) long[12], and you have to pay for them, although only a nominal amount, to cover the costs of maintaining the register of addresses[13]. They are conventionally written in the form XX-XX-XX-XX-XX-XX where the 'X's are hexadecimal digits, and each byte is separated by a dash.

If the IEEE approves your request for addresses, they will assign you a set of 2^{24} addresses (known as an *Organisational Unique Identifier* or OUI). They do this by telling you what the top three bytes must be: you are then free to assign the bottom three bytes in any way you like; provided, of course, that no two stations in any one network have the same address. You might be wondering why there are so many (2^{46}) possible station addresses to assign, when there is no chance that anyone will ever have that many stations on a single network? The reason is that having so many possible addresses allows every network card to be given a unique identifying number when it is manufactured, and this means network administrators do not need to keep records and assign unique addresses themselves. No two network cards in the world will have the same address.

Why did I say there were 2^{46} addresses and not 2^{48} when the addresses are 48 bits long? Because two of the bits have special meanings: the first bit is called the ‘group/individual’ bit, and the second bit is called the ‘local/global’ bit. (These are the first bit and the second bits to be transmitted respectively.)

Group addresses allow a set of stations to be defined to receive a frame, so that multiple stations can receive a single transmission. For example, there is a group address for network management to which error and warning messages are sent: any station that is monitoring the health of the network can be set to receive frames sent to that address; there’s another group address for all bridges. There is also a special group address (FF-FF-FF-FF-FF-FF) that is a broadcast address: every station on the network will receive frames sent to this address[14].

---

[10] Everyone in the industry refers to the unique numbers identifying LAN stations as ‘addresses’ (sometimes as *MAC addresses* or *hardware addresses*), even though they don’t contain any information about where the station is, so they should perhaps be called ‘names’ instead, and in any case the IEEE now consider the terms ‘MAC address’ and ‘hardware address’ to be obsolete.

The IEEE now recommends the name ‘EUI-48’ for this address (Extended Unique Identifier – 48). An EUI-48 looks exactly the same as a MAC address, but technically, a MAC or hardware address was assigned to a particular piece of hardware, whereas an EUI-48 can be used by a piece of hardware, or a software program, or any other entity that might want to use a globally unique address on a network.

[11] Provided they use globally assigned addresses, of course. More about this later.

[12] Historically they could be 16-bits long as well, but these days all 802 LANs use 48-bit long addresses.

[13] At the time of writing (August 2007) they cost $1650 for 2^{24} = 16.7 million addresses. If you don’t want this many, you can apply for an Individual Address Block (IAB) that costs $550, and provides 4096 addresses. There’s a list of which companies have which addresses at http://standards.ieee.org/regauth/oui/index.shtml.

[14] This is part of a convention in network addressing that all '1's means ‘everybody’ and all '0's means ‘this’.

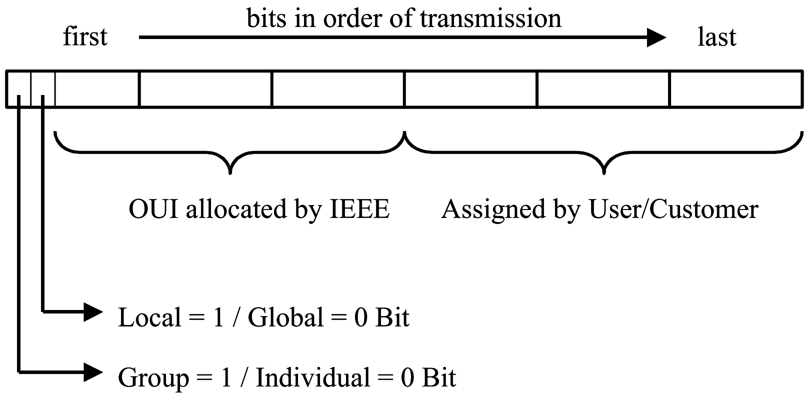

Figure 1-2 Structure of 48-bit MAC (Physical or Hardware) Addresses

The global/local bit distinguishes between addresses that have been globally assigned by the IEEE (and are therefore guaranteed to be unique) and those that have been locally assigned by the network manager[15].

1.2.1 Bit Ordering

There are two ways to send a byte across a serial link. You can either send the least significant bit first and the most significant bit last, or vice versa. The convention in serial links that Ethernet followed was to send the least significant bit first (as opposed to the standard written by the 802.5 working group for a token ring network, which sends the most significant bit first). That wouldn't be a problem in itself, except that someone once decided that it would be a good idea if the IEEE addresses were defined not in terms of a most significant bit, but in terms of which bit was transmitted along the wire first.

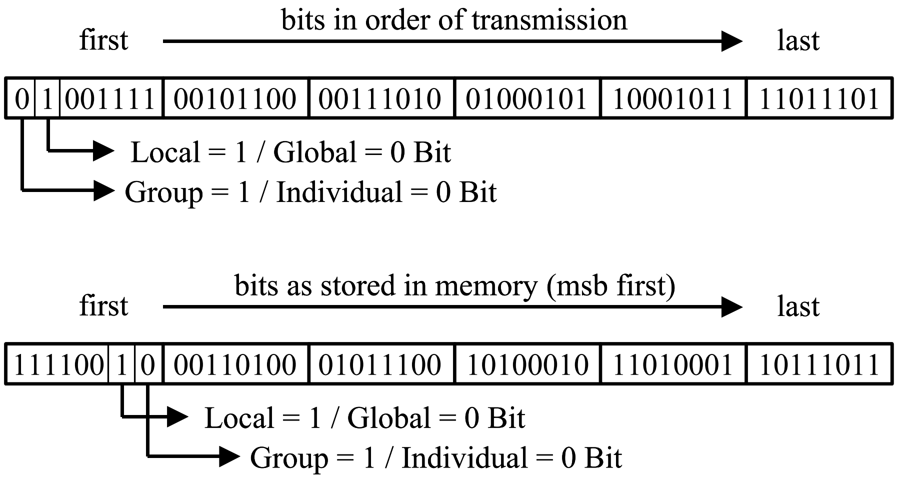

Figure 1-3 The Ethernet Address f2-34-5c-a2-d1-bb

[15] Some network managers like to assign network addresses themselves. This requires great care: two cards on the network with the same address can cause the entire network to fall over in strange and unpredictable ways.

So, the group/individual bit is defined as the first bit to be transmitted onto the wire, which as Ethernet stores addresses means that it's the least significant bit in the first byte of the address. Confusing, isn't it? It's caused a lot of people an amazing amount of grief, and is now widely regarded as a bad idea. The diagram above shows the bit ordering as stored in a token ring station (token ring networks send bytes out most significant bit first), with the group/individual bit in the most significant location.

The usual way of writing 802 addresses (the *canonical form*) is in terms of how an Ethernet node would store the address, with the most significant bit (msb) written first. So, for example, f2-34-5c-a2-d1-bb is not a group address, but it is a locally-assigned address.

1.3 Protocol Identifiers: DSAP and SSAP

The contents of the MAC header and footer are described in the chapter on Ethernet, here, I'll move up the protocol stack and have a look inside the data fields in the MAC frames that are being transmitted around these LANs. The first thing to note about these frames is that the first few bytes identify the language used by the rest of the frame[16].

Remember that difference between 802.3 and DIX Ethernet? This is what DIX Ethernet's protocol field is for: everyone who wanted to use a language asked the DIX companies for an identifier for that language, and used the one assigned to them. The IEEE decided to do things a different way, and included two single bytes, known as the DSAP and SSAP just after the length field in the MAC header. (DSAP/SSAP stands for Destination/Source Service Access Point, just in case you were wondering.)

These fields are used for *multiplexing*: allowing more than one higher-layer protocol to use the same network card, by assigning each higher-layer protocol a different SAP value, and the LLC data be passed up to the relevant higher-level program. For example, a computer might have some programs using IPv4 and some using IPv6 at the same time, and the contents of the DSAP field would tell the destination node which Internet layer protocol to pass the newly arrived frame on to.

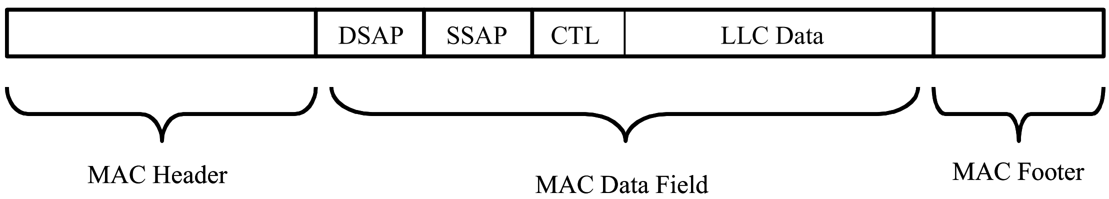

Figure 1-4 The DSAP and SSAP Fields in an 802.2 Frame

The idea of having two one-byte fields here is presumably that the destination and source could each speak a different language[17], or more likely that the same language could be given a different identifying number in each station[18].

---

[16] There is no equivalent in human speech, which is a shame. To avoid the complications of saying something that is unintentionally rude in a foreign language, computers always first announce which language they are about to speak.

[17] No, it's not just you, it is a stupid idea.

Just like station's MAC-layer addresses (which I should now call EUI-48 identifiers), the DSAPs has a group/individual bit[19] as the first bit (to be transmitted on the wire, in other words the least significant bit), and the second bit (transmitted on the wire) is always zero. That leaves a total of 64 different numbers to identify different languages/sessions that can be used over a project 802 LAN. Again, there are special cases: for example a DSAP of FF means all protocols[20].

Sixty-four different globally-assigned protocol numbers is pitifully few. Not even close to enough. So:

1.3.1 The SNAP SAP Bodge

In an effort to allow more protocols, special values for the SSAP and DSAP were defined: AA in hex, 170 in decimal. This particular value has the special meaning: "the first five bytes after the LLC type field are the real SAP values to use". In this way, the number of different unique numbers assigned to different protocols (languages) using 802 LANs is increased[21] to 2^{38}. The assignment of these protocol identifiers is done hand-in-hand with the assignment of OUIs: when the IEEE assigns a three-byte OUI, the same top three bytes used for the EUI-48 identifiers (formerly known as MAC addresses) can form the top three bytes of the SAP field: the bottom two bytes are up to the individual vendors to assign.

As a result, when you get a block of addresses from the IEEE, you actually get 2^{24} EUI-48 identifiers and 2^{16} protocol types.

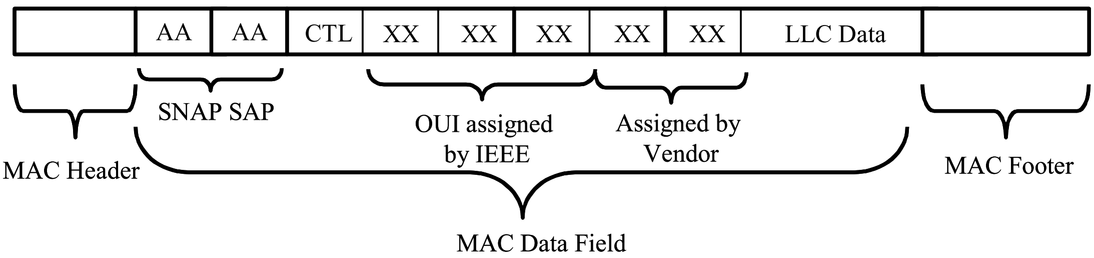

Figure 1-5 The SNAP SAP Bodge

[18] Almost as stupid an idea as the last one.

[19] Why is there a group/individual bit in the DSAP? This means that in theory you could send a message that would be received by a group of different protocols, all speaking different languages. Which would presumably interpret the rest of the data differently. Mad, isn't it?

At least they didn't put a group/individual bit in the SSAP as well… that would have been really crazy. This bit in the SSAP is used as a control bit for the LLC (known as the command/response bit).

[20] If you thought the last idea was crazy… what's the point in delivering a message to all available higher-level protocols, all of which speak different languages? Actually, in real life it's used to mean something completely different: since no sensible 802 network would ever do this, it can be used as a flag to indicate that another protocol (IPX, which doesn't use the LLC at all) is using Ethernet. But that's another story.

[21] 2^{38} and not 2^{40} since just like the EUI-48 identifiers, the first two bits transmitted on the wire have special meanings. The first bit must be zero, the second bit is a local/global bit, so that organisations can specify their own protocol numbers if they chose (although there is no guarantee that they will be globally unique: someone else in the world could decide to use the same protocol number).

1.4 The LLC

The logical link control (LLC) is the upper half of the data link layer in 802 LANs. Other than the DSAP and SSAP fields described above, there can be two other fields in the LLC header: the control field and the information field (although the information field isn't usually there). There are three types of LLC, but type 1 is by far the most common, so I'll concentrate on this one, and only briefly describe the other two types here.

1.4.1 LLC Type 1

LLC type 1 provides an unreliable, connectionless service (in other words a datagram service). It's by far the simplest of the three types of LLC, and by far the most commonly used. In LLC type 1 the information field is not used, and the control field is only one byte long, and indicates one of only three possible types of frame:

- UI (Unumbered Information, CTL = 3), in other words the enclosed data is a datagram, and should be passed up to the network layer identified by the SAP value with no further action by the receiving LLC.

- XID (eXchange IDentification, CTL = 191 or 175) indicates that this frame contains information for the other LLC, although for LLC type 1 there isn't much useful information you can send. Sending XID frames to the broadcast MAC address is a quick way of finding out how many stations are active at any time.

- TEST (CTL = 243 or 227) is used to test whether a suitable link exists, any data in the frame should be copied and returned to the originator. Amongst other things, it can be used to work out what the maximum frame size between the two LLCs is by a process of trial and error.

Any frame arriving at an LLC type-1 layer interface from the layer above has the CTL and SAP fields added, and the result is handed to the MAC layer for transmission. XID and TEST frames are used by management and testing functions only, and don't really form part of the protocol stack; they are rarely used, and are in some senses optional: while stations must be able to respond to any XID and TEST frames they receive, they don't have to provide a means for the higher layers to transmit them in the first place.

1.4.2 LLC Type 2

LLC type-2 provides a connection-oriented reliable service. There is sliding window flow control built in, which uses a window size of 127, requiring seven bits to transmit the frame number. The CTL field can now be either one or two bytes long, depending on whether it contains a frame sequence number or not. There are nine different types of frame, reflecting the greater complexity of this LLC type:

- I (Information) frames contain data, and a seven-bit sequence number for flow/error control.

- RR (Receive Ready) is an acknowledgement, and contains the sequence number of the next frame that the receiver will accept.

- RNR (Receive Not Ready) is an acknowledgement that all frames with numbers less than the current sequence number have been received, but also tells the transmitter not to send any more for a while: the receivers buffers may be full.

- REJ (Reject) is a go-back-N error control frame: the receiver is requesting re-transmission of all frames starting with the frame whose sequence number follows.

- SABME (Set Asynchronous Balanced Mode Extended – don't ask) is the first frame transmitted, this one is used to request that a connection be established. SABME frames do not contain a sequence number, so the CTL field is only one byte long.

- DISC (DISConnect) requests that a connection be discontinued.

- DM (Disconnected Mode) is transmitted in response to a DISC frame, acknowledging the closure of the connection.

- FRMR (FRaMe Reject) indicates receipt of an invalid frame; and forces the transmitter to stop sending more frames until proper operation has been re-established.

- UA (Unnumbered Acknowledgement) is used to acknowledge the receipt of any unnumbered frame, for example a FRMR or SABME frame.

I won't go into the details of how this lot works, for full details read the 802.2 standard[22]. In summary, connections are established by one LLC sending a SABME frame to another, and getting a UA frame in response. Then frames are sent with 7-bit sequence numbers, with error control performed using REJ frames, and flow control by RR and RNR frames.

1.4.3 LLC Type 3

LLC type 3 provides some of the advantages of LLC type 2, with much less complexity. LLC type 3 is a reliable, but connectionless protocol. There is no need to set up a connection before starting to transmit (and therefore no SABME frames). It uses stop-and-wait ARQ, so only a single-bit sequence number is required, and hence the LLC control field can be one byte long for all frames.

In the past, this was fine: LANs were so short and fast that requiring the transmitter to wait for an acknowledgment after transmitting every packet didn't slow it down very much. With 10 GBit/s LANs this is no longer the case, although since LLC type 3 is very rare in any case, it's not much of a problem in practice.

1.5 Virtual LANs

Originally, the definition of a local area network was simple: Ethernet had a bus structure, and all stations on the same LAN were connected to the same bus. Every station could listen in to every frame being transmitted by every other station. It wasn't very secure, but it was very simple.

These days, with bridges, switches and other such devices, the situation is a lot more complicated, and a more precise definition of a local area network is required. Perhaps the most useful definition of a LAN is a set of stations in the same broadcast domain. In other words, if a station transmits a frame to the broadcast MAC address (FF-FF-FF-FF-FF-FF), then

[22] Available free from the GET IEEE 802 website at http://standards.ieee.org/getieee802/

the stations that receive this frame, and only these stations, are on the same LAN as the transmitter.

Virtual LANs allow a single physical LAN to be divided up into a number of logical LANs, so that there are multiple broadcast domains attached to the same switch. All you need to do is tell a station which virtual LAN it belongs to, and it will only receive broadcast frames sent by another station on the same virtual LAN. This provides greater security, and also reduces the amount of processing every station has to do: they only need to process broadcast frames from the other stations on the same virtual LAN, and not every station attached to the same switch.

An interesting variation on this idea is that all stations on the same virtual LAN don't have to be connected to same physical switch or hub. Someone could move office, connect to a different switch, and yet remain logically in the same LAN, receiving the same broadcast frames, and using the same IP address.

For this to work, the broadcast frames being sent from this user's original switch to the switch to which this user is now connected must have some additional information in them: they need to know which virtual LAN they should be transmitted on. There is an extension to the MAC header field to provide this information (defined by the 802.1Q standard) that inserts an additional four bytes into the MAC header field:

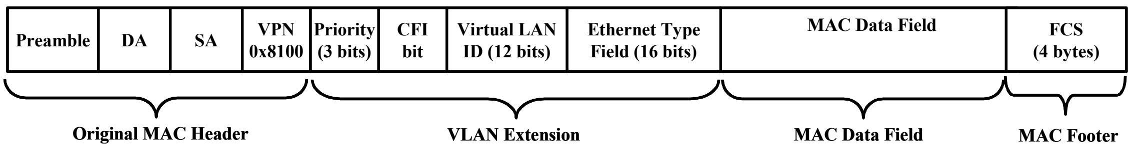

Figure 1-6 Virtual LAN Extension Headers

The original Ethernet type field is replaced by 0x8100, which indicates that the four bytes following the MAC header are an extension header[23]. This extension header then contains four fields:

- A user priority. This allows switches with several packets waiting in their output queues to prioritise packets: for example delay-sensitive information such as voice can be sent before non-delay sensitive information such as email.
- The Canonical Format Indicator (CFI) bit. Set if the frame is a Token Ring frame, and therefore the MAC DA and SA fields need to be bit-reversed when the frame is transmitted onto an Ethernet network[24].
- A 12-bit virtual LAN identifier to determine which virtual LAN this frame belongs to. There can be up to 4094 virtual LANs[25].

---

[23] If the frame is in 802.3 format (with a length field after the source MAC address) then the protocol type number in the SAP/SNAP header is set to 00-00-00-81-00, and the four bytes are included immediately after the LLC header.

[24] Some stations on a VLAN may be Token Ring stations (most significant bit sent first), others Ethernet stations (least significant bit sent first). The switch receiving the VPN frame along a trunk line from another switch needs to know which so it can order the MAC addresses correctly. I did mention that this 'first bit on the wire' convention was really awkward, didn't I?

[25] And not $2^{12} = 4096$, because once again, all zeros and all ones have special meanings and cannot be used to identify a specific virtual LAN. All zeros means that the transmitting station is not part of a VLAN at all.

- The original Ethernet type field that has been effectively "moved along" by four bytes from the position it would have in a non-VLAN frame header.

The user might never see a frame like this, since many Ethernet cards do not recognise these frames, and users do not need to know which VLANs they are on. The additional four bytes can be added and removed by switches when the frames are sent from one switch to another, with details of which virtual LAN the transmitting station belongs to.

If you want to send a packet from a VLAN to another VLAN, even if the two nodes are connected to the same switch, you need to go through a router.

1.6 Key Points

- Understand the structure of the 48-bit EUI-48 identifier (formerly known as the MAC address), and the function of the two special purpose bits.
- Know what the LLC is, and what services are offered to higher layers by LLC type 1, type 2 and type 3, however there is no need to know the details of the LLC protocols, or any of the 'magic numbers'.
- Understand how the LLC formats frames, and how it knows which higher-layer protocol to pass the message on to.
- Know what the SNAP/SAP bodge is, and why it was necessary.
- Know what a virtual LAN is, what they are used for, and how they work.

1.7 Tutorial Questions

1) A 48-bit IEEE MAC address is stored in an Ethernet card as BA-98-76-54-32-10.

a) What type of address is this?
b) What 48-bits would be transmitted onto the wire when this address was in use, and in what order?
c) How would this MAC address be stored in a Token Ring card?

**2) The Source Address field in the MAC address header has a group/individual bit. What's the point of this? Is it useful?

*3) The IEEE have assigned a meaning to the MAC address FF-FF-FF-FF-FF-FF: it is a broadcast address which all nodes on the network should receive. Why shouldn't the IEEE have done this? (Not entirely a serious question.)

**4) SNAP SAP values for the protocol type are 5 bytes long. Why 5, and not 4, 6 or 7 bytes long? Surely 6 (the same length as the addresses) would be more consistent?

5) LLC type-2 uses seven-bit sequence numbers, and uses stop-and-wait flow-control and go-back-N error control. What is the maximum window size that can be used?

A British company connects up its Ethernet network in London to its new office 3000 miles away in New York using a remote bridge, and running LLC type-2. What is the maximum possible data rate achievable over this link? How could the two networks be joined together to give a faster link?

(A remote bridge makes the network appear to be one long Ethernet segment. Assume the link operates at 2/3 the speed of light.)

*6) Suppose a new LLC type was designed that used a selective repeat ARQ error control system. The same company hears about the advantages of selective repeat ARQ, and decides to use this new type of LLC to improve performance. What's the maximum possible data rate over this link now?

**In fact, this long link is noisy, and X% of the frames are lost. What value of X would make the efficiency of selective repeat and go-back-N error control equal?

*7) Broadcast frames on a VLAN where not all the nodes on the VLAN are connected to the same switch use a VLAN header extension to indicate which VLAN the frame should be sent to. Why? Why not just use the source MAC address field to work out which other nodes the frame should be forwarded to?

**8) The minimum and maximum lengths of an Ethernet frame without a VLAN tag are 64 bytes and 1518 bytes respectively (including the MAC addresses and CRC, but excluding the preamble and starting delimiter). What happens to these numbers if a VLAN tag is used?