The ISP Column

A monthly column on all things Internet

Happy Birthday Ethernet!

May 2003

Geoff Huston

Another day, another birthday. This time its Ethernet's turn for a few minutes in the spotlight. On the 22nd May 1973 Bob Metcalf authored a memo that described "X-Wire", a 3Mbps common bus office network system developed at Xerox's Palo Alto Research Center (PARC). Hang on, was that 3Mbps? True, the initial Ethernet specification passed data at a rate of 3Mbps, not 10Mbps. That came later, as did the subsequent refinements to the Ethernet technology that allowed data transfer rates of 100Mbps, 1Gbps, and most recently 10Gbps. Doubtless 100Gbps Ethernet rates are on their way.

There are very few networking technologies from the early 70's that have proved to be so resilient (TCP/IP is the only other major technology from that era that I can recall), so its worth looking at Ethernet a little closer to see whats behind it. What was the difference between Ethernet and a variety of other emerging technologies for local area networking, such as Token Ring or DQDB?

Perhaps one clue is that there is an intriguing mix of simplicity and ingenuity within the original Ethernet design. On the one hand its surprising what Ethernet does not do: there is no explicit acknowledgement of receipt of a packet, nor any indication of transmission failure or nor indication of any form of network failure, such as network segmentation. There is no consistent network 'clock' to time the data pulses, nor any form of imposition of fair sharing of the network to competing demands. Nor is there any level of predictability within the system. Individual transmissions can be blocked for up to half a second, using a randomly generated wait interval, which means that there is the potential for almost unconstrained jitter. There's no flow control, or any form of prioritization. The system was not originally designed to be full duplex, so bidirectional communication could only be supported through the rapid exchange of Ethernet frames. On the other hand, even through there is no overarching network 'control' within an Ethernet architecture the common bus Ethernet architecture is surprisingly effective at congestion management, and its possible to achieve sustained Ethernet loads at 95% or the total rated capacity.

So whats the magic here? The best place to start is by looking at the second generation of the technology, the 1980 Ethernet specification, published by Digital Equipment Corporation, Intel and Xerox (DIX).

No Clock!

In looking at an Ethernet frame, firstly, and perhaps surprisingly for a high speed system, Ethernet is asynchronous. The wire does not provide a constant clocking signal that serves as the substrate for clocked data. The Ethernet frame starts with an enforced idle time of 96 bit times (9.6 μseconds), followed by a 64 bit preamble. The preamble is an alternating pattern of 1 0 1 0,... terminating with two 1's in bits 63 and 64. The purpose of the preamble is simple: it sets the clocking rate for the subsequent data packet. The receiver's task is to look for activity on the wire and synchronize its clock against the regular signal being received as a preamble. Once the receiver's clock is in sync with the data, it only has to stay in sync for a further 1518 octets, or 12,144 bits, or just a little over one millisecond at 10Mbps. There's no doubt that making a 20Mhz clock stable for 1 millisecond is a somewhat cheaper task than making it stable for some years, and part of the reason why Ethernet took off was that it was possible to use simple circuitry to construct transmitters and receivers, and the clock is an important consideration.

Any size you want!

Well not really, but close. The DIX specification allowed for individual Ethernet frame payloads to be between 46 and 1500 octets. A minimal TCP/IP ACKnowledgement packet is 40 octets, which fits comfortably into an Ethernet frame, while a 1500 octet frame carries a 24 octet overhead, or 1.57% media overhead. Even when you allow for the 9.6usecond interframe gap, the media overhead for maximum-sized Ethernet packets is still a very enviable 2.3%

The outcome of this variable-sized packet encoding is the ability to maximize efficiency of the media layer, allowing both small and large packets to be carried without excessive overhead and without excessive payload padding.

The cost of variable sized packets is increased potential for network- induced jitter, where a clocked real-time stream of data may have its timing altered as it passes through an Ethernet network. On the other hand there is a trade-off between data timing and network utilization, and, like TCP itself, Ether opted to head down the path of producing maximal efficiency rather than sacrificing speed and capacity for the sake of preserving implicit data timing integrity. In retrospect it was an astute decision.

There is also an ingenious relationship between the minimum packet size and this CSMA/CD algorithm. The one thing Ethernet attempted to maintain was the property that a transmitter is always aware if a collision occurs. Hence a packet must be "long' enough that the leading bit of the packet must be able to propagate to the other end of the Ethernet LAN, and the collision with the leading edge of another transmitter must propagate back to the original transmitter before the transmission ceases. i.e. the total end to end length of the LAN must be one half the minimum frame size. You can make the minimum frame size smaller, but the LAN itself shrinks, or you can support longer LANs, but at the expense of less efficient payloads because of a larger minimum frame size.

All this relates to the speed of electromagnetic propagation over a copper conductor, which in turn relates to the speed to light in a vacuum.

The Speed of Light

The speed of light in a vacuum, or the physical sciences constant c, is probably the most researched constant in all of science. According to electromagnetic theory, its value, when measured in a vacuum, should not depend on the wavelength of the radiation. According to Einstein's prediction about the speed of propagation of light within the general theory of relativity, the measured speed of light does not depend on the observer's frame of reference; the speed of light in a vacuum is a universal constant.

Estimates of the value of c have been undergoing refinement since 1638, when Galileo's estimate of If not instantaneous, it is extraordinarily rapid was published in "Two New Sciences". The currently accepted value is 299,792.458 kilometers per second.

The speed of light in glass or fiber-optic cable is significantly slower, at approximately 194,865 kilometers per second.

The speed of propagation of electrical charge through a conductor is a related value; it, too, has been the subject of intense experimentation. Perhaps the most bizarre experiment was conducted in Paris, in April 1746, by Jean-Antoine Nollet. Using a snaking line of some 200 monks, connected by a mile-long iron wire, Nollet observed their reactions when he administered a powerful electric current through the wire. The simultaneous screams of the monks demonstrated that, as far as Nollet could tell, voltage was transmitted through a conductor “instantaneously”. Further experimentation has managed to refine this estimate, and the current value of the speed of voltage propagation in copper is 224,844 kilometers per second, slightly faster than the speed of light through fiber-optic cable.

Ethernet CSMA/CD Design

Relating this value back to the design of Ethernet, a 10Mbps system running over copper wire will carry bits at 0.75c, or at 224,844 kilometers per second. This means that 256bits at 10Mbps will be contained in 5,756 m of copper cable. The original DIX Ethernet design specifications allowed for a total of three 500m runs of copper cable, plus allowance for 2 repeaters, and a generous allowance for error!

No Scheduler!

The next piece of the DIX Ethernet puzzle is the ingenious CSMA/CD algorithm. A transmitter first waits for any current activity to stop ("carrier sense"), and then it will wait a further 9.6 µseconds and then commence transmission of the frame. While it is transmitting its frame it monitors the medium to ensure that no other transmission is taking place.

If it detects another transmission (a "collision") then the transmitter sends a "jam" signal for another 32 bit times and then aborts the transmission and "backs off" for an interval, before trying again with the initial carrier sense step. The backoff interval is a multiple of a "slot time" (where a "slot" is a minimum sized Ethernet packet, 512 bits, and a slot time is the time to transmit 512 bits, 51.2 µseconds). The backoff interval is a calculated as a random number r where 0 <= r < 2**k, and where k = MIN(n,10), where n is the frame's collision counter. Thus, if a transmitter encounters a collision for the first time if will back off between 0 and 1 slot time (0 to 51.2 µseconds). If this results in a second collision then it will back off for a time of between 0 to 3 slot times (0 to 153.6 µseconds), and so on until the 10th collision gives a random interval of between 0 and 1023 slot times. After the 16th blocked transmission attempt the frame is discarded and the MAC layer reports an error. Any resemblance to a hash array using exponentially spaced collision side chains is not a coincidence, and the packing outcomes for such hash tables and Ethernet are remarkably similar. So how long could a frame wait before being sent on the media? The worst case is just under half a second.

Of course the real issue here is that the algorithm is fair, in that over time all transmitters will have equal probability of being able to transmit on the channel. There is no controller, and no critical point of failure in a LAN, nor is any connected station reliant on the correct operation of any other connected station. Ethernet is indeed a peer network technology that operates in a "plug and play" fashion.

Unique MAC Addresses!

The next Ethernet innovation was in addressing. A common technique for LANs was to use short (8 bit) address fields and instruct the LAN administrator to configure each connected device with the next available address. Ethernet took an entirely different approach, and Ethernet uses a 48 bit address field. Each manufacturer is assigned a block of numbers and burns a globally unique mac address into each Ethernet device. What comes shipped is an Ethernet device with a globally unique address. This allows the end user to simply plug the device into any LAN in the knowledge that there will be no local address clash. Not only did this approach of using unique MAC addresses make wired networks easier to set up, it also proved remarkably useful in wireless 801.11 WiFi networks, where devices can associate within a wireless service realm without causing havoc over clashing wireless addresses. It remains to be seen how long the 48 bit field will last, but it has been pointed out that if you manufacture 10,000,000 devices a day, it will take 40,000 years to run through the 48 bit address space.

Ethernet = LAN

So, from a technical perspective, Ethernet achieved an elegance in simplicity by avoiding over-designing the technology. But it took more than that to achieve the level of ubiquity that Ethernet has enjoyed in the face of well-funded competition. The decision by Digital Xerox and Intel to create an open standard for the 10Mbps Ethernet technology was also a significant factor, enabling a large collection of vendors to build interoperable products. The consequent adoption of this standard by the IEEE 802 committee and the release of the IEEE 802.3 Ethernet LAN Standard took this industry alliance and pushed it firmly into the realm of an International standard. The consequent market competition ensured that products were keenly priced and the wide range of vendors forced each vendor to strictly adhere to a common standard in order to ensure that their products inter-operated with others. The increasing volume of deployment also allowed manufacturers to achieve economies of scale, and at that point Ethernet had it made - it was faster, simpler, cheaper than anything else around for local networks.

Ethernet Evolution

Nothing stays stable for very long, and Ethernet has moved on as well. The DIX Ethernet specification used thick coaxial cable and transceivers that were pretty chunky to say the least. By 1985 the IEEE standardized 10Base2, a wiring scheme that used thinner, more flexible coaxial wire. From there Ethernet moved to standard office cabling systems, and the IEEE standardized Ethernet over twisted pair (10BASET) in September 1990.

The twisted pair standard was also an outcome of a changing topology of Ethernet deployments. The original concept of Ethernet was a 'snake' coax cable, passing close to each workstation in a false ceiling or under a raised floor. Each station would use a drop cable to attach to this common cable. But it was increasingly impossible to squeeze everyone onto a single cable segment, and there was interest in both 'extension' devices that allowed for larger Ethernet systems, and interest in altering the underlying local topology to use the emerging office structured cabling systems that used a star hub twisted pair wiring approach.

But lets start with repeaters. Repeaters were first thought to be simple devices that acted in a similar fashion to line amplifiers. A repeater was a two port device that picked up bits from one segment and retransmitted them on the other. The repeater faithfully reproduces collision events and jam signals, and, from the perspective of the signaling protocol is entirely invisible. The modification to the basic model came with multi- port repeaters, where signals received on one segment were faithfully reproduced on all other segments. But as networks expanded with various forms of repeaters, the probability of collisions increased, and the seemingly infinite capacity of a shared 10Mbps channel started looking very finite indeed.

The next step was to segment the LAN into a number of distinct collision realms. Bridges were perhaps the first real change to the original Ethernet concept, as a bridge does not pass on collisions. A bridge picks up Ethernet frames from one collision domain LAN, inspects the destination MAC address, and retransmits the frame on the other collision domain if the Mac address is known to be on the other domain. Bridges are essentially 'transparent' devices, in that when two stations communicate across a bridge there is no way that either station can discover that there is one or more intervening bridges. The packet format is unaltered by the bridge, and the bridge passes on all broadcast packets as well as all unicast packets.

Of course the real giveaway sign of a bridge is increased latency. As a bridge reassembles the entire packet before switching it onto another LAN, there will always be a minimum latency of the packet size between any two stations. Concerns about the extended latencies encountered in large LANS lead to the interesting concept of early-switching, where the switching decision was taken as soon as the destination MAC address was received, and the remainder of the packet was switched through at the wire rate. If the source packet encounters a collision, the collision will need to be reproduced on the destination LAN where the packet was already being transmitted.

From this 2 port model comes the notion of a multi-port Bridge, and in a multi-port bridge its possible to use an internal switching fabric that allows multiple packets to be switched between LAN interfaces in one switching cycle. This is the core of the LAN switch, where a number of individual LAN ports are interconnected via a switching fabric.

The other development was the introduction of the full-duplex Ethernet architecture. This is still an Ethernet LAN, but in this case there are only two stations. There are also two distinct communications channels, one allowing the first station to send frames to the second, and the other to allow frames to be sent in the opposite direction. Now one station's transmissions do not interfere with the other, and there is now no LAN length restriction due to the need to enforce a collision restriction. Using this approach it is possible to interconnect two LANs using a wide area serial link between two bridges, and arbitrarily complex topologies of LAN interconnections can be constructed. This collision-free full duplex architecture has been a cornerstone in extending the speed of Ethernet.

But Ethernet is a very simple framing architecture, and the problem with setting up complex topologies is that its way too easy to set up various forms of loops. The Ethernet Spanning Tree protocol was a way to extend the "plug-and-play" approach to complex bridged Ethernet topologies that allowed various forms of automated detection of link redundancy and fail- over from primary to secondary paths.

Faster and Faster

The next evolution of Ethernet occurred in the early 1990's with the introduction of 100Mbps Ethernet. But with the speed change came a change to the basic concept of how KANS are constructed. Remember that in the basic CSMA/CD common bus architecture of 10Mbps Ethernet, the maximum diameter of the LAN is half the size of the minimum packet. That is, if a transmitter is sending a minimum-sized packet, and at the other end of the LAN a transmitter also commences to send the instant before the leading bit of the original transmission reaches the new transmitter, then the collision response must reach the original sender before the entire packet is placed onto the wire. But if we up the speed by a factor of 10 and leave everything else the same, then LANS will shrink from a maximum diameter of 1,500m for a collision domain to around 150m. Of course the other option is to increase the minimum and maximum packet sizes by a factor of 10, but this would represent a relatively inefficient trade-off with many transport protocols, as well as having to implement some pretty tricky features to allow a large packet to be fragmented into a set of smaller packets if you want to interconnect a 100Mbps system to a 10Mbps system. The Ethernet design for 100Mbps started off with the objective of keeping a consistent packet format and packet size range, and allowing all other parameters to adjust. While, in theory, this forces a common bus 100Mbps Ethernet into a relatively small maximum diameter, 100Mbps has been used with various forms of switching hubs, allowing a twisted pair run of 100m from the hub point to the station,

The next step was to 1 Gigabit, and again the frame size range of 46 to 1500 bytes of data was preserved. Like the 10 and 100Mbps Ethernet specifications a point-to-point connection can operate in either half- duplex or full-duplex mode. The half-duplex mode of operation was problematic at this speed, as the network extent is reduced to some 15m at this speed, so the half-duplex version of Gigabit Ethernet supports "carrier extension", where the slot time is extended by a factor of 8 to 4096 bit times. Coupled with this is the extension of 'frame bursting' allowing multiple short frames to be packed into a single contention interval in half-duplex mode. Full duplex gigabit Ethernet does not need such modifications, and operates with the same 96 bit inter-frame spacing, the same 64 byte - 1518 byte frame size range and the same frame format.

And then we've seen 10 Gigabit Ethernet. With this standard the entire concept of half duplex operation has been dropped, and with any remnant of CSMA/CD. 10G Ethernet operates only in full-duplex mode, over fibre optic cables. The interesting aspect of this development is dual carriage standards, where the carrier rate is 10Gbps over a fiber connection, and 9.29Mbps if a wide area OC192 SONET carriage system is used.

As well as raw speed Ethernet has been heading an a number of other directions, including adding in flow control primitives to the MAC layer, packet prioritization, Virtual LAN support to allow trunking of multiple LANS in a single LAN link, and of course into other media. The current hotspot of wireless access is not any of the various incarnations of 3G, but a method of passing Ethernet frames over wireless, the 802.11 family of standards, or WiFi. WiFi has taken the mobile computing world by storm, and not only are the number of access points growing daily all over the planet, the speed of these systems is also increasing. The latest outcomes pace the wireless systems at speeds up to 55Mbps, a speed undreamt of in the 3G world.

So what's left of Ethernet in this latest 10 Gigabit incarnation? Now that we've reintroduced constant clocking, scheduling control via 'smart' switches, flow control and prioritization, and dispensed with carrier sense multiple access, collision detection, and returned to a full-duplex mode of operation perhaps we're now down to the essential fundamentals of what makes a network design an instance of Ethernet. Despite what we may have thought at the time, Ethernet is not a CSMA/CD common bus Local Area Network. Ethernet has turned out to be no more and no less than a global addressing scheme for devices that share a common framing format to transmit data.

Disclaimer

The author is a member of the Internet Architecture Board (IAB). The opinions expressed in this article are entirely those of the author, and are not necessarily shared by the IAB as a whole.

The above views do not represent the views of the Internet Society, nor do they represent the views of the author's employer, the Telstra Corporation. They were possibly the opinions of the author at the time of writing this article, but things always change, including the author's opinions!

About the Author

GEOFF HUSTON holds a B.Sc. and a M.Sc. from the Australian National University. He has been closely involved with the development of the Internet for the past decade, particularly within Australia, where he was responsible for the initial build of the Internet within the Australian academic and research sector. Huston is currently the Chief Scientist in the Internet area for Telstra. He is also a member of the Internet Architecture Board, and is the Secretary of the APNIC Executive Committee. He was an inaugural Trustee of the Internet Society, and served as Secretary of the Board of Trustees from 1993 until 2001, with a term of service as chair of the Board of Trustees in 1999 – 2000. He is author of *The ISP Survival Guide*, ISBN 0-471-31499-4, *Internet Performance Survival Guide: QoS Strategies for Multiservice Networks*, ISBN 0471-378089, and coauthor of *Quality of Service: Delivering QoS on the Internet and in Corporate Networks*, ISBN 0-471-24358-2, a collaboration with Paul Ferguson. All three books are published by John Wiley & Sons.

E-mail: gih@telstra.net