2021-06-12 ieee 1394 for grilling

To begin with, a reader emailed me an objection to my claim that Smalltalk has never been used for anything. They worked at an investment bank you have heard of where Smalltalk was used for trading and back office systems, apparently at some scale. This stirred a memory in me--in general the financial industry was (and to some extent is) surprisingly interested in "cutting edge" computer science, and I think a lot of the technologies that came out of first-wave artificial intelligence work really did find use in trading especially. I'd be curious to hear more about this from anyone who worked in those environments, as I know little about finance industry technology (despite my interest in their weird phones). Also, I am avoiding naming this reader out of respect for their privacy and because I neglected to ask them if it's okay to do so before going to publish this. So if you email me interesting facts, maybe do me a favor and mention whether or not you mind if I publish them. I'm bad at asking.

And now for something completely different.

Years ago, at a now-shuttered Smith's grocery store in my old home of Socorro, New Mexico, I did a dramatic double-take at a clearance rack full of *Firewire.* This Firewire was basically a steel cable used like a skewer but, well, floppy. The name got a chuckle out of me and this incident somehow still pops into my mind every time I think about one of my "favorite" interconnects: IEEE 1394.

IEEE 1394 was developed as a fast serial bus suitable for use with both storage devices and multimedia devices. It was heavily promoted by Apple (its original creator) and present on most Apple products from around 2000 to the switch to Thunderbolt, although its popularity had decidedly waned by the time Thunderbolt repeated its mistakes. FireWire was never as successful as USB for general-purposes usage. There are various reasons for this, but perhaps the biggest is that FireWire was just plain *weird.*

What's it called?

IEEE 1394 was developed by several groups in collaboration, but it was conceived and championed by Apple. Apple refers to it by the name FireWire, and so do most humans, but Apple held a trademark on that name. While Apple made arrangements to license the trademark to a trade association for use on other implementations in 2002, long after that most PC manufacturers continued to use the term IEEE 1394 instead. I am not clear on whether or not this was simple aversion to using a name which was strongly associated with a competitor or if these implementations were somehow not blessed by the 1394 Trade Association.

In any case, you will probably find the terms FireWire and IEEE 1394 used with roughly equal frequency. For further confusion, Sony uses the term i.LINK to refer to IEEE 1394 on their older products including cameras and laptops. Wikipedia says that TI also refers to it as Lynx, but I haven't seen that name personally and cursory internet research doesn't turn up a whole lot either.

The lack of a single, consistent brand identity for FireWire might be seen as its first major mistake. My recollection from FireWire's heyday is that there were indeed people who did not realize that FireWire devices could be used with non-Apple computers, even though "IEEE 1394" interfaces were ubiquitous on PCs at the time. I think this must have negatively impacted sales of FireWire peripherals, because by the time I was dealing with this stuff the only storage peripherals being sold with FireWire were being marketed exclusively to Apple users by historically Apple-associated brands like Macally and LaCie.

What does it look like?

Further contributing to compatibility anxiety was the variety of physical connectors in use. The major FireWire connectors in use were (most commonly) called Alpha, Sony, and Beta. The difference between Alpha and Beta was one of speed, as Alpha was designed for FireWire 400 (400Mbps) and Beta for FireWire 800 (800Mbps). Even this change, though, required the use of so-called "Bilingual" cables with Alpha on one end and Beta on the other.

The Sony standard, which worked only with FireWire 400, was smaller and so popular on mobile or otherwise low-profile devices. A number of laptops also used this smaller connector for reasons I'm not completely clear on (the Alpha connector is not significantly larger than USB).

The result was that practical use of FireWire frequently required adapters or asymmetric cables, even more so than USB (where the device connector was inconsistent) since both ends had a degree of inconsistency involved. The hassle was minor but surely didn't help.

Just to make things more fun, FireWire could be transported over twisted pair (UTP) and efforts were made towards FireWire over single mode fiber. I'm not aware of any significant use of these, but the idea of running FireWire over UTP will become significant later on.

Is it cooler than USB?

Unlike USB and other contemporary peripheral interconnects, FireWire had complex support for management and configuration of the bus. Unlike USB which was 1:1 computer to device, FireWire supported arbitrary groups of up to 63 devices in a tree. While there is a "root node" with some centralized responsibility in the operation of the bus, any device can send data directly to any other device without a copy operation at the root node.

This meant that FireWire was almost more a *network* protocol than a mere peripheral interconnect. In fact, it was possible to transport Ethernet frames over FireWire and thus use it as an IP network technology, although this wasn't especially common. Further supporting network usage, FireWire supported basic traffic engineering in the

form of dedicated bandwidth for certain data streams. This was referred to as isochronous mode, and its ability to guarantee a portion of the bus to real-time applications is reflective of one of FireWire's major strengths (suitability for multimedia) and reminds me of just how uncommon this is in common computer systems, which makes me sad.

Despite the common perception in the computing industry that opportunistic traffic management is better^wmore fun^weasier to implement, FireWire's allocated bandwidth capability turned out to be one of its most important features, as it fit a particular but important niche: camcorders.

The handheld camcorders of the early 2000s mostly used DV (digital video), which recorded a digital stream onto a magnetic tape (inexpensive random-access storage was not sufficiently durable or compact at the time). In order to transfer a video to a computer, the tape was played back and the contents of the tape sent directly back to the computer, which recorded it. USB proved incapable of meeting the task.

It's not quite as simple as USB being too slow; USB2.0 could meet the data rate requirements. The problem is that USB (until USB 3.0) was polling-based, and so reliable transfer of digital video from a tape relied on the computer polling sufficiently frequently. If it didn't--say because the user was running another program during the transfer--the video would be corrupted. It turns out that, for moving digital media at original quality, allocated bandwidth matters.

Note that FireWire is effectively acting as a packetized video transport in this scenario, just with some extra support for a control channel. This is very similar to later video technologies such as HDMI.

Did the interesting features become a security problem?

The more complicated something is, the more likely it is that someone will use it to steal your credit card information. FireWire is no exception. Part of FireWire's performance advantage was its support for DMA, in which a FireWire device can read or write information directly from a computer's memory. This was a useful performance optimization, especially for high-speed data transfer, because it avoided the need for extra copies out of a buffer.

The problem is that memory is full of all kinds of things that probably shouldn't be shared with every peripheral. FireWire was introduced before DMA was widely seen as a security concern, and well before memory management units that provided security protections on DMA. On many real FireWire devices, access to physical memory was completely unrestricted. Every FireWire device was (potentially) a memory collection device.

What happened to FireWire?

Consumer adoption was always poor outside of certain niche areas such as the DV video transfer use case. I suspect that a good portion of the issue was the higher cost of FireWire controllers (due to their higher complexity), which discouraged FireWire in low-cost peripherals and cemented USB as a more, eh, universal solution. Consumer perceptions of FireWire as being more complex than USB and somewhat Apple specific were likely an additional factor.

That said, the final nail in FireWire's coffin was probably a dispute between Apple and other vendors related to licensing costs. FireWire is protected by a substantial patent portfolio, and in 2002 Apple announced a substantial $1-per-port licensing fee for use of the technology. Although the fee was later reduced, it was a fiasco that took much of the wind out of FireWire's sails, particularly since some major partners on FireWire technology (including Intel) saw it as a betrayal of previous agreements and ended their active promotion of FireWire.

In summation, FireWire seems to have fallen victim to excessive complexity, costly implementation, and licensing issues. Sound familiar? That's right, there's more commonality between FireWire and ThunderBolt than just the name.

While Apple stopped supporting FireWire some years ago, it continues to see a few applications. IEEE 1394 was extended into embedded and industrial buses and is used in the aerospace industry. It also continues to have some use in industrial automation and robotics, where it's used as a combined transport for video and control with machine vision cameras. That said, development of the FireWire technology has basically stopped, and it's likely these uses will fade away in the coming years.

Last of all, I have to mention that US cable boxes used to be required to provide FireWire ports. The reason relates to the conflict of cable providers and cable regulators in the United States, which will be its own post one day.