telecomkh.com

http://www.telecomkh.com/en/business-communications/news/bob-metcalfe/ethernet-innovation-summit/5389

From ALOHA to Ethernety?

05/28/2013

Bob Metcalfe at Ethernet Innovation Summit

Bob Metcalfe, Co-Inventor of Ethernet & UT Austin Professor of Innovation, sets the scene with reflections on how CSMA emerged from Hawai, the early choices and challenges, and some ongoing implications of Metcalf´s Law

Paul Saffo - Futurist
So I have the pleasure of introducing our prime suspect in all of this. Where are you, Bob? Bob Metcalfe? He's good, okay.
Bob is going to come up, and I just want to say a word or two about him. He's got two bachelor's degrees from MIT, a PhD in '73 from Harvard, and one important detail, his first run at the PhD, Harvard, his committee turned him down. So he got his PhD tossed out, which was very fortunate, because then as he was working, the came out to PARC, and Bob Taylor said, "Forget that university on the east coast. We don't care what degrees you have. Come on out, and when you get around to it, you can finish your degree." It turns out that's part of the innovation process.
But just a few other things. He has had an extraordinary career. He has more awards than one could shake a stick at. He won the National Medal of Technology. He's had the IEEE Medal of Honor, the IEEE Alexander Graham Bell Medal and the ACM Grace Hopper Award.
As you will see with our panellists to come that we're going to introduce panellists with a single word. In the case of Bob, a single word will not do, so the words that I associate with Bob are engineer, innovator, entrepreneur, publisher, professor and, above all, tremendously innovative troublemaker.
Bob, the stage is yours.

Bob Metcalfe
Did you take the clicker with you? So we've decided to put my view of 40 years into 20 minutes.

Unfortunately, I've got 40 years' worth of slides here, so they're going to go by quickly. Here's what I think we're doing today, and I'm basically just going to repeat what Paul said.
We want to gather innovation lessons, and I've begun writing them down. I have four of them so far. We're going to hopefully sing Ethernet's unsung heroes, and it's true, I've gotten much more credit for Ethernet than I deserve, and the other people who are a little annoyed about that, are here in the audience today, and we hope to bring them forward. And then we're going to have a party, and then tomorrow we're going to catch up on what I believe is about a $100b industry that calls itself Ethernet.
So, depending on the meaning of the words I and invent and Ethernet, there are some values of those three values for which I can say I invented Ethernet, and I'm going to -- in the midst of this talk, since there's a lot of bogging down in what the word Ethernet means, I'm going to explain exactly what it was, a very small thing that I and Dave Boggs and David Liddle and Tat Lam and John Shoch actually invented, at Xerox PARC, which is one of the many meanings of the word Ethernet. Of course, Ethernet is now a brand that has escaped that, so I hope to get through that, and there will be a lot of technical detail during that portion of the talk.
So here's the bigger context for Ethernet's invention. This is the history, going back to ARPAnet, all the way through today. As you can see, I wrote this by hand and then I photographed it and then I uploaded it into PowerPoint. And I went through those steps because, when Ethernet was being invented, PowerPoint did not exist. We would photograph them with a 35-millimeter camera.
Key events here, you see the ARPAnet, Internet 1.0. You see the invention in '73, the LAN wars from '81 to about '84 we'll talk a little bit about. Fast Ethernet coming out in '95. We called the 100 megabits per second, we called that Fast, remember. And then what followed. So we're talking about a small part of a very long history.
You can't see that, but that's the cover of my PhD dissertation, but what's noticed is the map of the Internet. Can you make that out at all? You see those dots? Those dots are not cities. Those dots are not buildings. Those dots are computers. That's how many computers there were on the Internet in 1973. You can count them. I'm now a professor at the University of Texas at Austin, and I love to show them this, because you'll notice the Internet goes through Texas without actually stopping there.
And what we were involved in at Xerox PARC was called distributed computing. That was the term. For example, the course I taught at Stanford had that term in it, distributed computing, and basically we were going from a box-centric view of the world, in which there were lots of little terminals connected to a big box in the centre to a net-centric world, in which the network was in the centre.
I'm not sure when we first started calling the network the cloud. Of course, now, the cloud is used everywhere, but the network was in the middle. That's the picture. What I wanted to point out in this picture that you just saw a moment ago, so on the right is the Texas Instruments Silent 700, which we all had one of at Xerox PARC. And its modem, acoustically coupled, ran at 300 bits per second, 30 characters per second.
In the upper middle at the top was a box of 35-millimeter slides. That was the method we used for giving presentations, overheads and 35-millimeter slides. And you'll see, right in front of me there, is what's called a rolodex. So people had rolodexes. I had a rolodex for a very long time. I still have it, by the way. It became really big after a while. And then you'll see there's a telephone there.
Over here are some pads on the lower left, but those aren't iPads. Those are just regular pads. So let me describe the innovation space in which that aforementioned group of people were operating. Try to recall, this is the space in which we were attempting our innovation. We were going to put personal computers in a building, the Xerox Alto. The notion of having a computer on the desk was at that time controversial, having a building full of PCs was. That may have been the first time it ever occurred, and our great fortune was to be given this problem that had not previously existed. What do you do with a building full of personal computers?
So hundreds of machines, spread out over a kilometre, one per desk. They had to be small and cheap. Cheap meant the computer cost $30,000, and we were going to be upgrading, going from the Silent 700 to the PC, and the networking ended up going up a factor of 10,000 in that one step.
Second, the applications we had in mind. We were building a network -- well, first of all, these PCs were replacing Silent 700s, so they should do at least what the Silent 700 did. So I wrote Telnet for

the Alto. You all know what Telnet is? So we needed Telnet on the Alto so we could go log into machines around the Internet. And then we were building a laser printer called EARS, 500 dots per inch, a page per second. You multiply 8.5 times 11 times 500 times 500 per second, then you get about 20 megabits per second, and we wanted to keep the printer busy. So the network had to be a significant fraction of 20 megabits per second.

We also had this general notion, which was associated with the overall effort of producing PCs, that if we built it they would come. That is, we were not really good at listing what the applications were going to be for all this, but there was a build it and they would come mentality, and that has happened. So in the history of Ethernet, every time we've built it, they have come, and by they, I mean the new applications.

Now, we also had some digital technology to play with, and this I sort of put up defensively, because I'm frequently approached by young engineers who say, "Why did you do Ethernet that stupid way?" And the answer is, we didn't have LSI, we had MSI, Medium Scale Integrated Texas Instruments 7400 series semiconductors in which, for example, you could get two flip-flops on a chip.

We had FIFOs, which we needed to get the data in and out of the computer. Those were pretty integrated. David Boggs found a CRC chip to check whether the bits had been damaged in transmission. The MOS memory is what enabled the Alto, and that was a part called the 1103 from Intel that had 1,000 bits on one chip at a penny a bit, roughly, and that was a big enabling technology. And then we had this microtasking microprocessor, the Alto itself, to play with. And we wrote our programs and protocols in BCPL and Assembly Language.

So that was the enabling digital technology, and I put that up mainly as an excuse for some of the stupidities for the early Ethernet, because that's all we could do. We didn't have LSI. And then there was the wire. So I had just finished putting part of MIT and Xerox on the Internet with a cable about this thick called the IMP cable that went from the minicomputer into the packet switch. The cable was this thick, and when I put it on my board, I had to bolt it, because it was very heavy and could easily break and strain weave it with cable ties and so on.

And then so cabling was a problem, and our building had a room, a rat's nest, full of cables in it, which was a problem. So it was very much on our mind to solve the cabling problem, and then we found the ALOHAnet, and I notice Norm Abramson is here, the inventor of the ALOHAnet, and it had two radio channels. So I went on an expedition, a field trip. I spent a month in Oahu at the University of Hawaii, studying the ALOHA Network. I recommend that field trip.

So we wanted to solve the rat's nest problem. We had a large number of cables that would be very long, and they could end up being thick, with big, unreliable connectors. And the speed had to be faster than 300 bits per second, faster than the ALOHA Network, which was about 9,600 bits per second, faster than the ARPAnet. The ARPAnet, we were really only getting 15 kilobits per second in and out of our time-sharing system.

But we had this damn printer, and it was running at 20 megabits per second, so that was driving the speed. So we decided to choose a medium that had one wire. We looked at zero, by the way. ALOHA Network had zero wires. The trouble was the modems were as big as this podium, right, Norm? And we needed to put one on every desk, so that wasn't going to work for us, and plus, we needed to run at megabits per second, so hope you don't mind, we switched back onto wires, but only one of them, which we called the Ether.

And then there was the sharing part, so imagine 256 computers spread around the building. As the data is going back and forth, how would that be clocked, tick, tick, tick, data, data, data? Would there be one clock, or would there be a distributed clock? And we decided we weren't going to have a big central clock. The network was going to be completely passive, which meant the clocking of the data had to be in the data.

And then we needed to arrange for the stations to take turns. They had this one wire, but they had to take turns using it. How were we going to do that? And the it had to do that efficiently with low latency and fairness, and it was in the context of that that I ran across a paper by Norm Abramson at the AFIPS Conference Proceedings at the Fall Joint Computer Conference of 1970, and it had math in it, this paper. And that was important, first, because it was math that I could understand, but second of

all, Harvard wanted my thesis to be more theoretical, and you had math in the paper. So I said, "Aha, theory." And in that was the math of randomized retransmissions, which I'll return to in a second.
Then, we needed reliability. We were going to have a big cable running through the building that everything was going to be connected to. Wouldn't it be terrible if somewhere in the building that cable got grounded or shorted or broken? So a lot of energy went into making the reliability of the connection there very high, and a man named Tat Lam, who I haven't seen for 20 or 30 years, he engineered that.
He kept telling us it had to be under picofarads. That's all I remember, four picofarads. And so he achieved that with a transceiver, and so the transceiver could be on, but it had to mostly be off, and it had to be safely off. Its default mode had to be off, letting go of the cable, so the packets from other stations could zip by.
And then in hardware, Dave found the CRC chip, which was one of the last things to squeeze on the board. We also had a software checksum because we weren't sure that the Alto was that reliable, so we wanted to be sure that the bits got all the way into memory and not dropped by one of the MOS memories we were using, so we had a software checksum, and then there were the protocols.
And this was key, because we realized that the Ethernet, this network, would exist in a hierarchy of protocols with seven layers, and then we came to rely on those layers, simplifying our design. So, for example, the Ethernet had no acknowledgements in it. There were no acknowledgement packets. They were just packets. The higher-level protocol would create the acknowledgements so we didn't have to, which made Ethernet simpler.
Now I'm coming to the exact thing that we invented. We came across a method of encoding called Manchester encoding, which had the feature that in each bit cell, the first half of the bit cell would be the bit and the second half of the bit would be the complement of the bit, guaranteeing that in the middle of the bit cell, there was a transition from bit to bit [bar], and that transition would be the clock, so the clock was encoded in the data and it came in the middle of every bit cell. And another cool thing about this encoding was that the cable was either on or off. See, you could drive the cable to, say, five volts. David, what was the voltage?

Dave Boggs
Three and a half.

Bob Metcalfe
Three and a half? Thanks -- 3.5 volts, or could ground it or you could let it go. And so in the case of Manchester, there were only two states, yanking it up to 3.5 volts or letting it go. So half the time, while you were transmitting, you were letting the cable go, which had very important consequences. It gave us huge advantages over the ALOHA Network. Of course, the ALOHA Network couldn't listen to what it was doing. It could only deduce what had happened after the fact, and this gave us great advantages.
So with Manchester encoding, we then go to the ALOHA paper. So the idea was the stations were going to put packets on the Ether, and if they got through, fine. If they didn't, they would retransmit, like the ALOHA Network taught us. So I looked at Norm's math, and two things about it were annoying, Norm, as you know after my complaining for 40 years. One was he assumed the network had an infinite number of users and that they kept on typing even if the packets got lost.
I knew that the ALOHA Network had six users, which is a lot less than infinity, and people would stop typing if the packets didn't get through. So in the process of simulating this to see what the numbers really were, I wrote a computer simulation of the ALOHA Network, and there came this awkward moment in writing the code where I had to specify the mean of the retransmission interval, and Norm's math did not have any such thing in it.
So that was the first departure, is I actually had to specify what the mean of the retransmission interval was. And then I started running the simulations, and I started noticing that every time I ran the simulation, even with all the parameters constant, I got a different answer each time, which meant that the state space of the channel was bimodal, the right-hand diagram. That is, Norm's math dealt with the average, but as you can see, the average is very unlikely. It's in the middle of this big valley. So

then we applied Manchester -- and this is it. This is the thing. This is the invention right here -- Manchester encoding to make that channel stable, and we did it like this.
Because we had Manchester encoding, every time a packet was going by on the network, you could tell within a bit time by just listening. Just listen to the cable and you could hear the Manchester. And if there was somebody already transmitting, that was not a good time to start. So we could avoid collisions by just listening first, and that was called carrier sense, and then, because the network was off half the time when you were sending, while you were sending your own packet, you could be examining whether anyone else was sending in the half-bit cells of each bit.
You could listen and tell if there was a collision. So we had collision detection and the radios of ALOHA Network couldn't do that. And then, our packets weren't just card images, 80-column card images. They could be really big packets, so the ratio of the packet length to this collision avoidance mechanism was very large, so we could amortize one collision interval over a very long packet, so we were therefore able to get high efficiencies. And then one more thing, how do you get the channel stable? And what we discovered is, if you hit too much traffic, the proper instinct was not to try harder. You send a packet and it gets collided, you don't try harder. What do you do?
You back off. You try less hard. You back off, and so back off -- so do you all see? This was the invention that that group that I mentioned before, that's what we came up with, using Manchester encoding with the constraints that I enumerated prior, and this is where the ALOHA Network came in, taught us how to do randomized retransmission, and what we added was the fact that we were on a cable and this backoff thing.
So that led to this memo, written on May 22nd, and I like showing this because I really like annoying the people who invented wi-fi, because there's four or five of them arguing about who invented it, and I've met them, all of them, and they're always kvetching about who invented it, sort of like the Ethernet people, who invented Ethernet? And I've explained exactly what we invented. We didn't invent everything.
So what I do is I generally claim to have invented wi-fi, too, and you see on this diagram, on the far right-hand side, what does that say over there? Radio Ether? Well, what do you think that is? Well, it's either ALOHA Network, which is probably what I had in mind, or it's a very long-term anticipation of the arrival of wi-fi, and therefore I invented it, for those of you who are -- so this is where we decided to call it the Ether.
We chose thick coaxial cable because it could be tapped passively, but we anticipated other media would be used, so we didn't call it coax-net. We called it Ethernet, because the ether could be coax, twisted pair, radio, optical fibres, power line, whatever you wanted. And that was the diagram in 1976 of what we built.
That big yellow thing is the cable. The official colour of Ethernet is yellow, and of course, this bears no resemblance to the Ethernets that you're using today, in which that big yellow cable has been collapsed to a box and the rat's nest has been recreated. So that's the original transceiver by Tat Lam, the Jerrold cable tap suggested by David Liddle, this thick coax. Was it 50 or 75 ohms, David?

Dave Boggs
Seventy-five ohms.

Bob Metcalfe
Seventy-five ohms, and then we went to 50 ohms later?

Dave Boggs
Ten megabits [inaudible].

Bob Metcalfe
That's a very important technical fact there, the ohms of the cable. It's also a half-inch thick, which was later a problem.
I saw this -- this exists -- yesterday. This is the Ethernet at Xerox PARC, at PARC, which was near where the EARS printer was. And this is Dave Boggs on the left and Ron Crane, so Dave Boggs and I

invented, with the other people I mentioned -- did the work of building the first Ethernet. Ron Crane there in the back, he's the one who picked it up at Xerox later and joined me at 3Com Corporation to make it an IEEE standard.

And then followed the LAN wars, and these are some of the weapons of the LAN wars. Promotion, which is standard were a good thing, get on the Ethernet bus, I'm plugging Ethernet, etc. That lasted two or three years, and then two things happened that helped. One was the arrival of the IBM PC in August of '81. Finally, a PC worthy of Ethernet. That is the 8-bit Apple II, the Ethernet was gross overkill for an Apple II, but it worked for the IBM PC.

And then the other breakthrough, happened in the late '80s, was the switch from coax to twisted pair. We were fighting in the market with the IBM token ring, which used twisted pair, and the customers kind of liked that, so Ethernet decided to become twisted pair, too, and that sealed the doom of the token ring.

So now where is Ethernet going -- have I run out of time yet? Keep going? Ethernet is not done yet, so here are the five directions in which -- and probably others, but these are the five I thought of, each with its own preposition, up, into, over, across and down. So Ethernet as a LAN continues up, and the IEEE has recently started a project to standardize 400-gigabit-per-second Ethernet, on its way to terabit.

Ethernet has also left the LAN and entered the WAN, where it is slowly wiping out SONET, which is the previous WAN infrastructure. That is, in the early days of the Internet, we relied on the telephone company. Now we don't so much anymore.

It's gone over the airwaves, hence wi-fi across the Telechasm. Thanks to the MEF, the Metropolitan Ethernet Forum, Ethernet has now become a service offering of the carriers, going across the Telechasm between the LAN and the WAN, and Ethernet, under many names is now going down. We're now going to network-embedded microcontrollers, about 10b of which are shipped every year, and most of them are not networked yet, and that's being taken care of now. So that's where Ethernet's still going.

And then let me just close with the innovation -- my view of the innovation horizon ahead. As you know, the Internet, with Ethernet as its plumbing, has disrupted a series of industries, like music, books, telecom, television, you name it. There's a bunch of them.

Well, there's three more industries about to be disrupted, and they'll be disrupted by these new kinds of traffic. So down the left, you have the new kinds of traffic, video, mobile and embedded. And then, across the top, you have the three new industries to be disrupted. These industries are begging to be disrupted, can't wait -- energy, health care and education. And so what I find it useful to do is to look at each of the boxes there to see how will that traffic disrupt that industry? And in it's in those boxes that we find the needs for ever-faster, better Ethernet.

So that is the end of my remarks -- with six seconds to spare. So, Paul, are you going to come back up? So that was a warm-up for the day's festivities? Are we going to stand here, or do you want to sit down?

Paul Saffo
Let's stand. Here, give Bob a hand first, here. So I'm going to quiz him for about 15 minutes, but I would be especially -- there's feedback in this room.

Bob Metcalfe
That was me, breathing.

Paul Saffo
That was you, breathing. Still haven't gotten error correction for analog yet. But I also would love to bring in comments or questions from the audience. And here's how we're going to do questions and comments. I'm particularly interested in the folks who were part of this revolution. So if you have a question -- everyone just practice with me, do this. Come on, everyone in the room, raise your hand. If you have something that you just absolutely need to toss into this conversation at that exact

moment in time, do this. Put up two hands. We'll bring it to a screeching stop and we'll get a microphone to you. That sound cool?

Bob Metcalfe
Cool.

Paul Saffo
So go back a slide. Let's see if they can do it. The other aspect about Bob is, he is a fearless prognosticator, and he prognosticates often and has been teased terribly for some statements where people decide he wasn't quite right. There was that period in the '90s when you were kind of dissing wireless.

Bob Metcalfe
I was right about that.

Paul Saffo
We're coming to that. You're finishing my thoughts. As someone who prognosticates for a living, I count on people like Bob, because I get some of my best ideas from listening to him. So I love starting with this very tantalizing slide. Looking ahead, what's the biggest surprise you think we're going to see in the next three to five years, coming out of the telecom space?

Bob Metcalfe
Hard question. The most exciting surprise, I think, is going to be MOOCs.

Paul Saffo
Everybody know what a MOOC is? Massively open online courses.

Bob Metcalfe
So education is about to be disrupted. Most of us in this room probably got educated along the way, and that whole thing is about to get -- you know like iTunes did to music?

Paul Saffo
You didn't do bad for a guy who never made it to Stanford.

Bob Metcalfe
I took management of high technology at Stanford University.

Paul Saffo
I take it back. That was the maraschino cherry. Anyway, you're right, Stanford is ground zero.

Bob Metcalfe
So the MOOCs are happening. You all know what a MOOC is and are generally in favour? Or do you have serious questions about the loss of interpersonal contact between the teacher and the student? Do you have those kind of worries? So here's how I handle those. It goes back to the invention of another bad idea, the BOOC, the B-O-O-C, which is today spelled B-O-O-K. It was obviously a very bad idea, because before books, we would sit around the campfire and we would hear the story directly from the storyteller, but now we have these damn book things.
You've read the Great Gatsby, but you've never met F. Scott Fitzgerald. That's a problem. So the MOOC is really a bad idea.

Paul Saffo
Okay.

Bob Metcalfe
You weren't expecting that kind of answer.

Paul Saffo
You're not getting off that easy. You're not getting off that easy. At Stanford, on MOOCs, we're...

Bob Metcalfe
Stanford keeps coming up here.

Paul Saffo
It's just up the road. Udacity launched, Coursera has launched. We've got two big operations. It's going to revolutionize things. What do you see? Just one or two tantalizing specifics of how MOOCs change things with education.

Bob Metcalfe
Well, for one thing, education is not going to occur between the ages of five and 22 anymore. Education becomes a lifelong learning thing. So I'm taking 6.00x through EdX. That's Introduction to Computer Science and Computer Programming. I'm learning Python from my own desk in Austin, Texas. My son is at his desk in San Francisco. We're both taking the course, not that there's anything competitive going on there, and I'm older than 22, and I'm learning Python.
So there's elimination of this age discrimination. You know, there's age discrimination going on at universities. I'd have a hell of a time getting admitted to the University of Texas, or Stanford -- excuse me. Just mentioning Stanford.

Paul Saffo
Back in 2005, I was having lunch with a friend and I told him -- we were both approaching 50, and I said, "I know the definition of middle age. I'm a professor at a university that would not admit me again as a student, and I own a house in Hillsboro that I could not afford to buy again if I had to buy it." So when the 2008 crash happened, I felt so much younger, because at least then I could own my house.

Bob Metcalfe
So age and wealth. It costs a lot of money go to Stanford, 50 grand a year. So with EdX or the other kinds of MOOCs -- they're only called MOOCs for now. They'll be called something else as they evolve. But now, anyone, anywhere in the world can take Stanford or MIT -- anywhere. So MIT recently reported on the people who were currently taking 6.00x with me, and they believe that there was a person in that course in every country in the world. There was some debate about how many countries there were and whether they counted Texas or not. But there was somebody from every country in the world, some of whom had these really -- they were taking MIT's freshman programming course from a hut in Bangladesh.
So second is getting to the other seven -- basically, like we solved bandwidth with the Internet, we're now going to solve ignorance with the Internet. That was grandiose, wasn't it?

Paul Saffo
I actually think it is. Bob, one of his many qualifications is he's got a law named after him. How many people know what Metcalfe's law is? Show of hands. The laws -- one never names a law after oneself. That would be unbelievably gauche. This was conferred or inflicted on Bob by George Gilder back in the mid '90s, and Metcalfe's law say the value of a network goes up as a function of the square of the number of participants. So the more people you put together, it doesn't just become arithmetically better. It's nonlinearly better.
And I think MOOCs are a wonderful instantiation of Metcalfe's law, and I agree with you. The change

is, this isn't about everybody listening to the professor. This is about here interaction using the professor as the...

Bob Metcalfe
The MOOCs don't do that very well.

Paul Saffo
Yet.

Bob Metcalfe
Yet, but that's a problem that the MOOCs need to solve, is getting a scaling method of peer-to-peer instruction going.

Paul Saffo
Is Dave Liddle in the audience?

Bob Metcalfe
No.

Paul Saffo
He's not. Well, Dave likes to say that whenever we have a new technology, we tend to pave the cow paths. We always do an old thing with the new technology, and then it takes us time to get around to doing the big innovations, which, question for you, when did you realize that this thing you've started with a memo on May 22nd was going to be a really big deal? How long before you realized, other than getting your Harvard degree, which I realize is a big deal?

Bob Metcalfe
Well, there's 20 times, not just one time. So one of the times was, and Dave Boggs will remember this. For a while there -- where are you, David? David Boggs. So we built the first two Ethernet controllers for Alto's Michelson and Morley, and then there was our cubicle area, and then all of the Altos in that cubicle area were all connected. And each of the research scientists got to buy an Alto, and there was a little box, did you want Ethernet or not? Ethernet was an option.

Paul Saffo
Let's run the mic over to David there, get him, in case he has a comment. So keep going.

Bob Metcalfe
So Ethernet was an option for a short period of time, so here's the first answer to your question. One day, somebody removed the terminator from the end of the coax for some reason. They were beautiful things, the little terminators. They feel good in your hand. So somebody stole it, which meant that the cable was no longer functioning. An what happened was 10 people stood up, looking over the cubicles, wondering what had happened, and in that moment, we realized that Ethernet was no longer an option, and thereafter that box was removed and people got Ethernet every time.

Paul Saffo
And I would say to that, today, communications, networking, has become like oxygen. If you lose it for a couple of seconds, you notice it's missing. You're starting to get a headache at 30 seconds and you're unconscious at two minutes.

Bob Metcalfe
Well, I left my iPhone in the hotel this morning. Can you imagine how I feel now? Excuse me, I just need to hold onto something.

Paul Saffo
You can borrow my iPhone. I've got to get a new one here. Dave, did you have a comment?

Dave Boggs
No. He told the story correctly.

Paul Saffo
That may be a first, Bob. We're pretty good here. So we have got two minutes left. If there was somebody with a question, we'd take the question. Right down in front. Also, the way we do questions is look where the person holding the mic is, and then the person in the room farthest from them, raise your hand. Tell us who you are and your question.

Thierry Outrebon, 01 Informatique
I'm from 01 Informatique in France, so in 1988, were you afraid of token ring?

Bob Metcalfe
Afraid?

Paul Saffo
Were you afraid of token ring?

Bob Metcalfe
Yes, imagine for about 10 years, every day, your wife, your Board of Directors, all of your employees, every market analyst around, would tell you that you were doomed, and that the IBM token ring -- because IBM made standards. It was the dominant computer company. They foisted what I like to call SNA on us, and token ring was coming, had come, and was -- eventually. There were a few people enthusiastic about it, but basically IBM was going to kill it.
Imagine living like that for 10 years, and then Ethernet beat token ring, so that has scarred me. So, for example, I don't take advice very well, because for 10 years, I ignored all this advice and it worked out, so I'm hoping to recreate that. So yes, terrified because of the power of IBM and their dominance.
What they underestimated was the power of an open standard, and I'm referring to the old IBM now. There's a new company called IBM which is different, but in that old IBM, in its dark little heart, was not committed to open standards, and its products were not interoperable. 3Com Corporation shipped IBM token ring before IBM did, and we had trouble selling it because our version of the token ring didn't have SNA dust sprinkled all over it.

Paul Saffo
We've got a two-hander over here. Say who you are.

Geoff Thompson
Hi. My name's Geoff Thompson, and Bob knows me from way back. Were you ever afraid of AT&T? I mean, AT&T had ISDN that was supposed to be their answer to data.

Bob Metcalfe
ISDN ran at 56 kilobytes per second.

Geoff Thompson
Yeah, that was fast.

Bob Metcalfe
That was kilobits per second, here. Ethernet started at the M level.

Geoff Thompson
That's faster than 300.

Bob Metcalfe
I refuse to answer that question.

Paul Saffo
I was real afraid of AT&T 40 years ago today, but that was only because of AT&T security.

Bob Metcalfe
By the way, there's a new AT&T, and it's headquartered in Dallas, Texas.

Paul Saffo
Yes, please.

Robyn Metcalfe
Back here. Your wife did not say you were doomed for 10 years.

Paul Saffo
And that was someone with deep inside knowledge on what his wife said. So last question before we push on, failure. I kidded you a little bit about your PhD, which is the normal nature of things. You've had an extraordinary arc of a career. You've crammed more careers into the last couple decades than most people cram into their entire lives. One of my favourites was you were a publisher at InfoWorld, a magazine that, for what, five years, refused to let you be a columnist? Stewart Alsop, as I remember, wouldn't let you be a columnist, and then you ended up as publisher.
Talk about this -- Stewart told me. Okay, well, maybe I'm wrong.

Bob Metcalfe
Robyn, did Stewart? Oh, no.

Paul Saffo
So talk about -- we think of entrepreneurship as all about success, and yet we know most entrepreneurial ideas fail, most entrepreneurs fail. Most Silicon Valley companies fail. This is why, as we say, venture capitalists sleep like babies. They sleep for two hours, wake up and cry, sleep for two hours, and you've been a venture capitalist. So you've been on both sides of the risk of failure. What insights do you have for the rest of us about how do you deal with the prospect of failure? How do you make failure into success?

Bob Metcalfe
You don't fail on purpose. People say failure is really good and we do it a lot, but that doesn't mean you should do it on purpose. So failure is a frequent consequence of attempting something hard, and then I'll share the conventional wisdom, which is you need to fail fast. This is a problem. Once you start doing something, you become committed to it, and then long after you should have abandoned it, you stick with it. So failing fast is an important thing. I'm not sure I was ever very good at that, failing fast.

Paul Saffo
Also, if you have -- for those of you from outside the country, this is Bob Metcalfe -- I'm not real good at pool, but I like to play for money, never been on this riverboat before.

Bob Metcalfe
You suggesting that I'm hustling you?

Paul Saffo
No. I'm saying that you're very modest.

Bob Metcalfe
Modest, yes. Well, some people have so much to be modest about.

Paul Saffo
There you go again. Good, well, I think we're right at the end of the time.

Bob Metcalfe
No, we've got three minutes and 30 seconds.

Paul Saffo
No, that's going the other way.

Bob Metcalfe
Oh, it is.

Paul Saffo
This is a software developer's clock that keeps counting past the deadline.

Bob Metcalfe
Oh, I see.

Paul Saffo
So let's give a hand for Bob. He'll be back.