



Oral History of Bruce Horn

Interviewed by:
Hansen Hsu

Recorded August 17, 2022
Mountain View, CA

CHM Reference number: 2022.0109

© 2022 Computer History Museum

Hsu: It is August 17th, 2022 and I am here with Bruce Horn. I am Hansen Hsu. So to begin with, let's start with where and when you were born.

Horn: So I was born in Torrance, California. That was 1959, about 63 years ago tomorrow. So I'm getting up there. I kind of feel like a bit of the old guard now.

Hsu: And did you grow up there?

Horn: No, I grew up in Palo Alto. And my parents had moved to-- they had met at Stanford and I grew up in and around Palo Alto, moving to Ladera and then Palo Alto proper for basically junior high school and high school. So I went to Terman Junior High School and Gunn High School.

Hsu: Oh, okay. So were your parents graduate students at Stanford?

Horn: They were undergraduates. My mom went to Stanford when she was 16. She had grown up as a teenager in Hawaii and left to go to Stanford, and she met my dad at Stanford and he was a football player. And she was just one of these really smart young people. And they met at Stanford as a-- I guess they worked together in the cafeteria, I guess. That's how they met.

Hsu: Oh, well that's a fun story.

Horn: Yeah.

Hsu: So what did your parents do?

Horn: I think my mom had an IQ of about 200, but she had kids. And at that time, people had limited options, and so she raised the four of us and my dad was a pediatrician at the Menlo Medical Clinic.

Hsu: Oh, okay. Do you have any siblings?

Horn: Yeah, I have three siblings. I've got an older brother and an older sister and a younger sister.

Hsu: Oh, cool. What were your interests or hobbies or favorite subjects as a child?

Horn: Oh, as a child? Well, I've always been kind of an environmentalist and outdoor person, hiking, cycling and so on. I have always been interested in airplanes and flying, and math of course. And one of the things that was lucky for me, of course, growing up in Palo Alto is being in proximity to Hewlett-Packard. And we happened to know somebody who was at Hewlett-Packard and could buy me HP calculators as they came out at HP prices. I remember I bought my very first one after working for an entire summer to save 395 dollars to buy an HP-45 calculator at the Stanford Bookstore. And I remember going in with my pile of money to the Stanford bookstore and buying this calculator. And that was the most amazing event, and I would just sit and play with it for hours and hours and hours. And I still

remember the feel of it and what it smelled like, because it was such an event to have your own little personal computing device in your hand. So that was kind of-- I was a math kid and an outdoor kid at the same time.

Hsu: Interesting. So your interest in the calculator was more from a math perspective rather than from an electronics perspective?

Horn: Yeah. It was just what it could do. The fact that you could actually do all of these mathematical functions in this little thing in your hand. I wasn't drawn to the electronics part of it or even the hardware design. I was drawn to the software of it, the thing that it could do, the features and the functionality. Which I thought was pretty miraculous, actually.

Hsu: Yeah. So was that your first introduction to computing? Or were there others?

Horn: Well, before that actually, when I was in sixth grade, Nils Nilsson, who was the head of the Stanford Computer Science department, brought a terminal, a teletype terminal, to Ladera Elementary School to show the kids what computing was. And so he would dial up to SRI and show us Basic and other programming languages and things that you could do. And that was really the thing that kicked me off into computing was that Nils showed us what was possible. And later on, when I was a teenager and I had money and I could make money and bought the calculator, that's when I did that. But earlier on, I was able to get influenced by Nils, who was an amazing person, by the way. Just a tremendous human. And that kind of got me aware of the computing world and what was possible. A little bit later, I think it must've been in junior high school, I met somebody who-- another kid, and I've forgotten his name, Greg. Anyway, Greg had connections at Stanford, and we would go over to Ventura Hall, where they had IMLACs which were little vector terminals, and you could play with those. And they had Plato terminals for learning all sorts of subjects. And it was kind of a play space that they just allowed kids to wander in and use these terminals if they weren't being used. So that gave me another opportunity to get into that. And that was a little bit before I was invited to come up to Xerox PARC.

Hsu: So you mentioned Nils Nilsson. Did you have any other mentors or teachers that were influential?

Horn: Oh, absolutely. At Gunn I had teachers all along who kind of supported me and recognized that I was kind of into math. And I remember when we moved from Ladera to Palo Alto, and I went in at Terman, they did some sort of math assessment. And I had been doing some algebra in sixth grade because the teacher recognized that I needed more stuff to do. So I kind of placed out of what was going to actually happen in seventh grade. And so I got to do a lot more interesting things. And then, of course, I had a math professor-- I call him a professor, but he was a high school teacher, Hawkinson, who was just tremendous, and also opened my eyes to a lot of what was possible in the math world. Yeah.

Hsu: Wow, thank you. It sounds like you had a lot of freedom to explore all of these interests and go to Stanford and do all these things. Were your parents really supportive of all your interests?

Horn: They were supportive. I think mostly they didn't really know what was going on. I mean, they just saw that I would leave and come back and I'd tell them what would happen. They'd say, "That sounds great." And they didn't really-- that wasn't their interest, right? They were not interested in computers or any of those things, my dad being a pediatrician, and he worked really hard and my mom was busy doing her thing. So it was-- more or less, I was the third child, so I kind of was-- I had the freedom just because there was too much other stuff going on managing the other kids. I just could do whatever I wanted. It was very nice.

Hsu: That sounds great.

Horn: Yeah, it was great.

Hsu: So let's get to PARC. So how did that whole thing happen? How were you first invited to go to PARC?

Horn: So I don't exactly remember, but I think it was in junior high school. Maybe it was in high school. I think I was 13 or 14. And I think it was Hawkinson-- maybe it was another teacher-- who introduced me to Ted Kaehler, who was in the Learning Research Group at PARC. And Ted had come down and said, "I need some kids to work on some things at PARC." And so I ended up being able to go, along with Steve Putz, who was my best friend. And we just got integrated into the Learning Research Group. They'd had a bunch of kids working on Smalltalk as part of this project. But I guess Steve was kind of an electronics genius, and I was kind of into computer software. So we just got to do whatever we wanted. It was apart from the kids' interaction from Smalltalk. So that was really interesting for me.

Hsu: Yeah. I do find that really interesting, that you and Steve, were separate from the other stuff that was going on, with the other kids. Was that just because you were already so advanced?

Horn: Yeah, maybe. That we were a little bit advanced, but we also just showed different interests. So I was really interested in how Smalltalk worked, a little bit less than the Smalltalk system itself, which of course I loved. It's like, "Well, okay. I love this thing. How does it work?" And so I was interested in the virtual machine and how to work on that, and kind of lower-level stuff. Plus, I was there all the time because PARC was an easy walk from my house. And so I was there all the time. And so they realized that I could do stuff for them. And so at one point, the CSL (Computer Science Lab) people decided that I should do the backups for the big-- the timesharing machine we had called MAXC. So that was just a task I had, and I went and did the backups and did the tapes and ran the system there. And it was nice to have a responsibility and that, as a young teenager, these genius adults would allow me to do. It was just a tremendous experience. I can't imagine how it changed my life. Just amazing.

Hsu: Yeah, that sounds incredible.

Horn: Yeah, it was incredible.

Hsu: So did you have any interactions at the time with any of those other kids? Or were you just completely separate from <overlapping conversation>?

Horn: I would say completely separate. I mean, I don't think I ever talked to any of the other kids. I mean, I knew them kind of vaguely, because they came from Jordan and other schools, but Steve and I were from Terman and Gunn, and there was nobody else from Terman and Gunn at PARC at the time. So I didn't really interact with them.

Hsu: Yeah, okay. So then it sounds like-- well, Adele [Goldberg] referred to you as an intern. What was your-- did you have any formal anything, or you were just sort of around and people would give you stuff to do?

Horn: I was just sort of around. I had a badge. I think I had a badge. I'd sign in like an employee, but they didn't pay me. So I have no idea what my actual arrangement was with Xerox Corporation, although at one point, I was working really hard on-- I think it was on the NoteTaker or maybe I was working on Dorado or something. And a graduate student had come in, and I got to know this graduate student. And I found out that he was getting paid. And I remember thinking, "Man, I should get paid, too." So I went to Adele and I said, "Hey, Adele, I'm doing a bunch of work. As much as that guy, the graduate student, and I'm 16. I'd like to get paid." She goes, "Oh, okay. We'll make that happen." And so basically the next week, I started getting a paycheck. So that was pretty amazing. That was my first salaried paycheck was from Xerox PARC. And Adele made that happen, because I just asked. I said, "Hey, this grad student is getting paid. Can I get paid?" And she said, "Sure."

Hsu: Wow, okay. So that must've initiated some sort of formal-- I mean, you would have had to submit a W-2 or something, right?

Horn: I guess. Boy that was-- I mean, I was just a kid. I don't remember. I just remember getting paid and I must've filled out tax forms or something, but it was minimum wage or whatever. But still, as a 16-year-old, getting a salary or whatever it was, just amazing. I should say, by the way, when I was 12 years old, I decided I wanted to go work at Hewlett-Packard. And I wrote a letter to Hewlett-Packard asking for a job when I was 12. And I got a beautiful letter back. I wish I still had it. It said, "Thank you for your interest in Hewlett-Packard. It's our policy not to hire anybody under the age of 18. We appreciate your interest. Please let us know when you've turned 18 and we'll talk about employment at that point." And so I thought, "What a great letter." And the joke is, well, Xerox PARC didn't have any problem with child labor, whereas maybe HP did. But I just thought, A, first of all, I loved Hewlett-Packard for what they were doing and how they treated me as a young person, and then of course PARC and Xerox and all the people at PARC treated me like an actual asset and gave me things to do and even paid me. Amazing.

Hsu: Yeah. So you don't remember what your formal title was, even, at that point?

Horn: No idea. I don't think I had a title.

Hsu: Yeah. Okay, so you started at PARC in 1974?

Horn: I think that's about right. I think I might have been 14.

Hsu: Okay. So you were 14 at that age. Can you remember your first meeting with Alan Kay?

Horn: Well, the first person I actually met when I came to PARC was Steve Weyer.

Hsu: Oh, really?

Horn: And see, even though Ted Kaehler had come to Terman or Gunn, I can't remember at the time, to recruit some kids, he was actually still off at Carnegie-Mellon, I believe. And so Steve Weyer happened to be around. And so I met Steve, and he was the first person I met. And as a young person, you just have a very narrow view of the world, and you go, "This is the guy. This is Steve." And he was super supportive and helpful, and he's still a friend to this day. And I remember meeting Alan kind of peripherally and getting the idea that this is the guy who's the head of LRG (Learning Research Group), and that's pretty interesting. And I don't really remember talking to him much, other than he was super supportive. And we didn't talk very deeply about anything, because I was being managed by Steve Weyer. But of course, later I got to know Alan pretty well, and we would play tennis. And I consider him a huge influence on my future career, of course, as well as Steve and Ted and many other people in the Learning Research Group. Dave Robson, Dan Ingalls, of course.

Hsu: Yeah, could you talk more about working with them and your relationships with the other members?

Horn: I'm pretty sure that they kind of just let me do what I wanted to do. So I would come in and maybe I was working on MAXC, I was doing backups or something. And Ted, every once in a while, would say, "Hey, maybe you should look into this," or Steve would do that, and I would just do whatever I wanted. What was interesting was at that age, at 16 or 15, I had an office with a door and a window and 12 disk packs on the floor. That was my office, even though maybe I wasn't even getting paid at that point. But they gave me an office. And my ongoing joke was that until two months ago, my office was my backpack and I would work off my laptop. It was downhill ever since, right? But PARC treated me like a real member of the group, and I would have things to do every once in a while. I did some small projects in Smalltalk. I did a little flight simulator kind of concept that Ted helped me think about, and how would I do that. And I got a tape, or Ted had gotten a tape of a topographical map of western Nevada. Basically points in a grid, which were heights, and then I would-- Ted explained how I might do some sort of perspective, and I built a little thing that would allow you to kind of fly around over this perspective space. It was really fun. But Ted would give me ideas, Steve would give me ideas. And mostly they just kind of let me do whatever I wanted. And then when it looked like they needed some kind of work done, like if you think about the Dorado and the NoteTaker, kind of low-level stuff, they would just recommend me to do it. And then I would go and that would be my project for a while. So for example, I worked on the Dorado microcode for Smalltalk, which was a huge project, by the way. And Willie-Sue Haugeland was my microcode mentor. We'll call her that. So she kind of helped teach me how to do that, and I spent many, many, many days and nights working on the microcode. Actually, in a very small room with shooting headphones on, because the Dorado was so noisy. I would sit there with the Dorado and the Alto next to it that I'd use to debug it. And I'd work away and write the microcode for Smalltalk. And

eventually, that started functioning and working, and that was the system that was demoed to Steve Jobs later.

Hsu: Oh, really? Oh, okay.

Horn: Yeah.

Hsu: Wow. Let me go back to some of these other programs. So you mentioned the flight simulator program. Was that just something that you just came up with on your own completely?

Horn: I think Ted came up with it. I would think. Ted knew that I was interested in flying. He had somehow gotten this data-- this topographical map data. And I think he just said, "Hey, this would be a good thing to do." And we worked on it together and I learned a lot from Ted about how to think about systems. How would you build such a thing and how would you do the controls and so on, and how you could do the display. And it was very rudimentary, not what I would do now, of course, or even a year after that. I would have learned a lot more. But that's kind of the type of thing I did, and that was actually in the Scientific American article, along with Steve Putz's circuit diagram, which I thought was very cool.

Hsu: Oh, right. Yeah, the circuit diag-. So that was Steve Putz's <overlapping conversation>.

Horn: Yeah, that was Steve Putz. Yeah. Steve actually knew how to build electronics. And so he had a motivation to build a tool to help him make it easier. And he did that in Smalltalk. It was super great.

Hsu: So the two of you worked on separate things. You never collaborated?

Horn: Separate things. No, no, but we hung out all the time. And he was also a math guy and we just had a great time together. Yeah.

Hsu: And then there was also a musical note capture program?

Horn: Yeah, that's right. I did that, too. Again, rudimentary. It was just one of these things to kind of get to understand the real time aspects of I/O and so on. So I did that. Later, I think, one of the other PARC people (Chris Jeffers) did a much more sophisticated one, one that actually understood music. Because I wasn't a musician. But it did function kind of vaguely. It could figure out what the note was and figure out the duration of the note and things like that. But it didn't know how to do the notation the way a real musician would layout notation. That happened later by an actual musician.

Hsu: Yeah. And so both of these programs were demoed and captured on video tape?

Horn: They were. Yeah. Yeah. I don't remember that, but I know I've seen it. I've seen those videos.

Hsu: Right. So you don't remember them actually recording it or being part of <overlapping conversation>?

Horn: No, actually, because I wasn't the musician. When the music capture program was videoed, somebody else was playing a piano-- playing the keyboard. I knew that it was going on, but I wasn't involved in that.

Hsu: Oh, okay. Let's see here. We already discussed the Dorado. Oh, okay, let's talk about the NoteTaker. So how did you get onto that?

Horn: Good question. I think, again, it's just like there was stuff to do, and not enough people. And Doug Fairbairn was kind of a friend of the group, obviously, and we were going to put Smalltalk in the NoteTaker. And I ended up getting kind of lent to Doug to help bring the NoteTakers up and make sure they continued to work. And there were maybe a half a dozen of them, and they'd work for a while and then I'd have to figure out what was wrong with them and they'd get fixed and so on. But I mostly wrote the BIOS, the little operating system underneath that supported Smalltalk. So it supported the Smalltalk system. And it had two processors. It had an emulation processor that ran Smalltalk and an I/O processor which did all of the-- it did BitBLT, serving the display, and it moved the mouse and it watched the keyboard and did sound and so on. We had stereo sound on that, which was really cool. And then these two processors would have to communicate. And so I wrote the little interprocess communication package for that and the support software for the system. And I remember I kind of felt like it was my first portable personal computer. It weighed, like, 50 pounds. But it was really fun.

Hsu: Cool. And so you also worked on a similar Smalltalk '78 environment for a Norwegian computer?

Horn: I did. So that's a slightly longer story. So I had gotten interested a little bit in my heritage. And I talked to my dad, and he said his grandparents were Norwegian and German. And so I thought, "Oh, Norwegian. That sounds good. That kind of fits with my outdoor interests." And so I started thinking, "Well, maybe I should learn Norwegian. It seems like an interesting language." So I got the Spoken Norwegian book by Haugen and I started reading through it. And then I sent a note out to everybody at PARC. Back in the day, we were the only people who had email. So I sent an email out to everybody, and I said, "Does anybody know anybody who speaks Norwegian, so I can go talk to them, or practice?" And Adele sent me a note back and said, "Oh, we have a professor from the University of Oslo coming with his family to spend a year with us at LRG. So that should satisfy your Norwegian speaker request." So Trygve Reenskaug came. Trygve-- the inventor of Model-View-Controller, which is still in use today, that framework and paradigm-- came with Oddbjørg and Gina, one of his daughters. And they were at PARC for a year. And so I got to know them, of course, and become good friends with them. And I taught Gina how to play tennis. And then at the end of that time, Trygve had to go back to Norway, but he wanted to keep working on Smalltalk. So he made a deal, the University of Oslo -- actually at the time, it wasn't the university, but it was the Central Institute for Industrial Research -- in Norway made a deal with PARC to have a license. Get a copy of the Smalltalk image. But they didn't have Altos or Dorados, right? So they wanted to get it to run on their machine. Their machine was called a Micron-2000 and it was a prototype machine. I think there was only one. So I don't know who, maybe Adele again, or maybe it was Alan, said, "Oh, we'll just send Bruce over there. He wants to go to Norway anyway." So they made a deal. I had to go through, and I guess I was 19, had to go through a background check and all these things, right? Because we were sending this technology out of the country. And I went to Norway and I implemented

Smalltalk, the system, on this Micron-2000. And it was the-- it had the image, which was all the Smalltalk code. But I had to write the virtual machine, which was going to be based on the NoteTaker virtual machine, which was in 8086 assembly code. Of course, this being a different machine, there was a lot of work to do to make it all work. So I ended up writing kind of a support system in UCSD Pascal for debugging and bringing up Smalltalk. And so I could actually introspect into Smalltalk and look at the objects with this UCSD Pascal debugger that I had written. And I eventually got it working. And I think that that became a system they used for many years after that. One of the other Norwegian guys, basically before I left, I kind of handed off the whole development to him, and he kept working on it and moved on. But that was a great experience. It was only six months, actually. But I ended up going back to Norway a couple of times after that. It was a great experience. But I lived in the basement of the building where I worked. I lived in-- it's called a "hybel". And the place was called "SI", Sentralinstitutt for Industriell Forskning. And I would just go upstairs to work and I'd come down. And they had a cafeteria, so I hardly ever had to leave. But once a week, I would go to Trygve and Oddbjørg's house and do laundry and have dinner. So that was also a great experience, but it was kind of a lonely experience. All I did was work and visit with Trygve. But it was super great. Super great. I do remember buying-- this is kind of a funny story. I was just barely speaking Norwegian, and buying a big thing of yogurt at the local store. And the yogurt in Norway at the time was just tremendous. I thought, "This is yogurt?" And walking back to the front of the store and tripping dropping and seeing the yogurt go all over the floor. And that made me speechless. I couldn't say a word in Norwegian. But they ran out and they cleaned it up and they handed me a new one. And I thought, "Wow, what a country."

Hsu: That sounds great.

Horn: An amazing place. I just love Norway.

Hsu: Yeah. So this all predated Smalltalk '80, right?

Horn: Yeah.

Hsu: So was any of that work-- did that have any impact on the later Smalltalk '80 work, when it was being ported to other environments?

Horn: Oh, I'm sure it did. By then, things were-- I was finishing up college, right? And I ended up, at that point, I wasn't very involved in the Smalltalk '80 system at all. And I was finishing up being at Stanford and trying to decide what to do next. And I'd been at PARC for a bunch of years, and I remember talking to Adele and everybody and, "What should I do next?" And Adele suggested maybe I should go to Apple or somewhere. Of course, before that, a number of people had started to move on. Larry Tesler. I was working on the NoteTaker with Larry, and I remember going out to pizza with Larry one evening, and he kind of said, "I'd really like to see this kind of stuff out in the world. I'm thinking about leaving PARC." And I said, "Well, maybe you should go to Apple." And it turns out, he'd already been thinking about Apple, and he already was going to go to Apple. And so when that opportunity came to me, it was Apple or VLSI Technology, which was founded by Doug Fairbairn. And Doug was one of my favorite people of all time, so basically I accepted the job from Doug just before I was brought into the reality distortion field by Steve

Jobs, and he changed my mind over a weekend. And I had to call Doug and say, "I changed my mind and I'm going to Apple." But that was in late '81. So I didn't have a whole lot to do with the Smalltalk '80 at that point.

Hsu: Okay. Of course, the Apple stuff I want to talk more about, but let's go back a little bit. So you mentioned you had this close relationship with Trygve. So did you understand what was happening with him developing Model-View-Controller at that point in time? Or did you witness any of that, or were you sort of aware of what he was working on?

Horn: I think I was enough in my own world. I think I was working on the Dorado probably at that point. I really didn't-- we didn't talk about work so much. Mostly, he was just an older friend and mentor kind of, especially when I went to Norway. But he was a friend of the family. I was a friend of his family. So we didn't really talk about work. I think he was very engaged with the rest of the group doing his work. Yeah.

Hsu: Yeah, okay. So you talked about being at Stanford. So how did you manage-- well, first of all, when did you start college at Stanford, and then how did you manage..

Horn: That's funny.

Hsu: ... Being an undergrad and also working at PARC at the same time?

Horn: It was very tough. Okay, so I started in Fall of '77, and the way it worked is I went a year and then I took a year off and I spent it at PARC. And then I went a year and then I took another year off and I went to Norway, I guess, and then I went a year and finished. So I got done in three and a half years or something, but I never went more than one year at a time at Stanford. Mostly because I was drawn to working on Smalltalk and computing in general, and just that was the time. Things were happening, and so I took advantage of that. You could go to college anytime, right? But being able to work at PARC when things are really hopping, or being able to help bring Smalltalk up in Norway, I mean, man, that was really what was going on. And I remember I turned 21 in Norway. So that must've been 1980, summer of '80.

Hsu: Wow. And so did you major in computer science, or did you do something else?

Horn: At Stanford, I majored in mathematical sciences, which is kind of a concrete mathematics. When I went to Carnegie-Mellon for grad school, I did computer science master's and PhD.

Hsu: Okay. And that was much later.

Horn: That was much later. That was after Apple.

Hsu: Okay, we'll get there. Yeah. Okay. So you mentioned that the Smalltalk '76 on the Dorado was the one used in the demos when Steve Jobs visited. So where were you when that happened? And could you maybe discuss what your perspective of that situation was?

Horn: So I was in the building. I was about four offices down from where the demo was occurring. And basically it was Dan [Ingalls] and, I think Larry [Tesler], were giving the demo of various things. And it went on for a number of hours and there were a lot of people around. And mostly I was just kind of hanging out and looking, and I did see, every once in a while, Steve [Jobs] would kind of leave the group. Steve Jobs would leave the group and kind of pace around and think about things. And at one point, and I think this was something that Larry mentioned, too, he started waving his hands and saying, "What are you guys doing? This should be out in the world." Something to that effect. So it obviously impacted Steve tremendously, just kind of what was possible with Smalltalk. And there's the story about where Steve was looking at the screen and said, "I don't like the way the text scrolls. It's jumping by whole line and it should smooth scroll." And Dan basically went into the Smalltalk system, spent a couple of minutes, made the change, and all of a sudden he had smooth scrolling. And that's something you can't do in any kind of production system today. So that blew away Steve. And Steve even said [later] that he didn't recognize the importance of object-oriented programming and live programming, being able to make those types of changes, dynamic programming, at runtime while you're just sitting in the environment. That's another thing we're kind of missing today. The environment-based programming, which makes so much possible and really makes it easy to turn around changes and make progress. I mean, that's the reason that all of the amazing things were developed in Smalltalk, because Smalltalk was so easy to experiment in and try things out and immediately get feedback. It would have taken much, much longer in a standard edit, compile, link, run, debug environment. Just probably not possible.

Hsu: Yeah. So just to clarify, you were still a member of PARC at that time?

Horn: I was still a member of PARC at that time. I know that there's been various stories told, and things interleave a little bit differently, but I was still a member of PARC, and I went to Apple somewhat after that. Because Steve, of course, knew that I was at PARC or had been at PARC. It was still PARC at the time, actually, when he kind of-- well, so the way it worked was I actually called Larry Tesler when I was looking for a job. And I said, "Hey, Larry, I'm thinking about coming to Apple." And so Larry was the person who kind of brought me in and showed me the Lisa and figured out the kind of person I was and realized I'd be better in the Mac group, so introduced me to Steve. So even though I had seen the Lisa first and the Mac was not as far along, it was more kind of my thing. And so the Mac group being much, much smaller, I could get in earlier and do more. But that was thanks to Larry realizing the kind of person I was and where I could really apply my passions.

Hsu: Yeah, okay. Well, so going back, were you actually able to meet any of the delegation from Apple during that visit?

Horn: I don't think I went and talked to them. I thought that was above my pay scale. But I did see them. I was there kind of checking things out. And I talked to our guys afterwards, right? Just to see, "What was that about? How did that go?" And so on. But no, I never met any of the Apple people. And the Apple people were Steve and, I think, of course Bill Atkinson was there. I can't remember the other people, but there were a handful. A handful of people who ended up being in the Lisa team.

Hsu: Okay. So then you graduated from Stanford in 1980..

Horn: One.

Hsu: '81. Okay, so obviously right after that is when you decided-- you're deciding between VLSI and Apple?

Horn: Right. And so that's when I went to Apple.

Hsu: Right. So basically both people you'd worked with on the NoteTaker-- Doug on VLSI and then Larry at Apple.

Horn: Right, exactly. It was funny, because I've never really kind of interviewed for jobs or never had to look for a job before, because I just knew the people and I'd go talk to them. And so of course, it was a very tough decision, because I had huge respect for Doug Fairbairn and huge respect for Larry, of course. And I guess I felt, after the reality distortion field kicked in, that I could do something bigger with the Mac. And Steve was right. I mean, I think that Mac had a much bigger impact on the world and I was able to do more and have a more fundamental effect on where computing was going. Steve really believed that we were building something for the creative class, and for every person to be a better person, be able to do more, be able to express themselves more. And I believe that, and Steve was right, and that's exactly how the Mac kind of impacted the world. It just made computing available to everybody. And I'm very, very lucky to have been part of that.

Hsu: Yeah. So you mentioned that-- so it was Larry that sort of directed you to talk to Steve, to join the Mac team?

Horn: Right.

Hsu: And so was that your first ever encounter with Steve?

Horn: Yes. Yeah, absolutely. Yeah.

Hsu: So describe what that was like, meeting him for the first time, and then being part of the reality distortion field.

Horn: So part of it was I had come from PARC, where the technology was already in place. The technology of the future was there, and I was using it. Smalltalk and bitmap displays and the mouse and laser printing and ethernet and email and everything. And here I was coming to Apple, where they had a little personal computer that couldn't talk to anything else and you could program it in Basic and so on, and that's what you had. And so I was kind of like, "Huh, I don't know. I mean, this is interesting." And Steve basically said, "Look, we're going to bring all this great technology to the world." And I just remember being-- I was a little skeptical, kind of a little bit-- I was standing back like, "Well, show me," in my mind. His enthusiasm and his passion were just overwhelming. And he brought me around and he showed me, "Here's what the box is going to look like. Jerry Manock is designing this amazing box. Check this box out. This is how it's going to be built, and here's George Crow doing the analog. It's

amazing. Look how small it is. Look how well-designed it is. And here's the software team," which was, like, six people or something. Andy Hertzfeld showing me a demo of this thing. And then I was actually really blown away by the demo that Andy showed me, because it had the mouse and menus, and it had little bouncing graphics. And I thought-- I remember thinking at the time, "Well shoot, you guys are almost done." Of course I was completely underestimating how much work there was to do still to make that real. But I would say that Andy's demo of what the Mac was, even then in '81, was the thing that kind of put me over. It's like, "Wow, I could see having that in my house. I could see my mom using that, and I want to be part of that." But even after that, I did tell Doug that I would go to VLSI Technology. And then Steve called me on Friday after I told Doug, and Steve said, "What do you think of Apple?" And I said, "Well, I decided to go to VLSI Technology and work with Doug Fairbairn." He said, "You did what? You come back in tomorrow morning. I have so much more to show you. Blah, blah, blah." And that's when the whole reality distortion thing came in. He kind of, I guess, bullied me, even, into coming back to Apple the next day. And then I spent the entire weekend with Steve taking me around, talking to more and more people. And by the end of the weekend, I was convinced. And I had to call Doug on Monday and tell him I'd changed my mind. And that was very, very hard, actually. But Doug was understanding, and just the great guy he is, he got that that was probably a better choice for me. Yeah.

Hsu: Yeah, Steve doesn't like to take no for an answer.

Horn: No, he doesn't take no for an answer. No, absolutely not. Steve gets what he wants, and he's very convincing. And he has good arguments. He knew what he wanted, he knew how to get it. I mean, Steve is a genius, right? Brilliant guy. Not just in his vision for what the Mac was or what Apple should be, but how to find the right people and how to get them to do things. He wasn't always perfect about that. He might have used one technique a few too many times. Maybe that's not the right technique to use on Bruce, where it might be the right one on somebody else. But he did get things done.

Hsu: Yeah. I do find it interesting that-- was it because some of your work at PARC was on pretty low-level stuff. Like you mentioned the BIOS for the NoteTaker, and the microcode for the Smalltalk Dorado. Was that the reason that Larry thought that you were better suited for the Mac, instead of the Lisa?

Horn: It's possible. I can't ask Larry anymore, sadly. But that may be the case, because I ended up doing most of my work in assembly code on the Mac. Very, very low level. And so that suited me, actually. For me, it was a lot of fun, and actually the 68000 was a tremendous instruction set, very easy to work with. Orthogonal. I could literally think and type at the same speed. So it's a good match to my thinking speed. But I wrote a ton of assembly code for the Mac.

Hsu: Yeah. So most of the code that you wrote was in assembly?

Horn: Ninety-eight percent of it.

Hsu: Wow. So when you first started on the team, what were you assigned to do?

Horn: So they had this concept of the Finder. Bud Tribble had-- the idea of how do you find things on the disk and so on. So Bud had this idea, and so Andy and I talked about it, and I ended up doing this thing. I don't know what we called it, the mini Finder or something. And it looked like a floppy disk, and it had little tabs that were the file names. And it had a "Do It" button from-- because of course, "Do It" was one of the things in Smalltalk. It was like, "Execute this thing." So you would just click on a tab and then click "Do It" and it would launch that program. But then I had an idea of the files and folders idea. And I had never seen the Star, but I was aware of the concept of-- that you would want to file things, of course, because I had used all of the PARC systems, like Bravo and Gypsy and so on. And I knew about filing and hierarchy for files and so on. So I thought I would do that instead, and the idea of you could open a folder and look at it and close it and so on, I did a demo of that actually from not even looking at the file system, but just looking at a text file that kind of described what a file system might look like. So it read in the text file and you could kind of navigate this. And that was my demo of what I think it should look like. And then I ended up writing the real Finder after that, and then toward the end, Steve Capps came in and we pushed it over the line. But at the time, I was-- because I kind of think of foundational things, I was thinking, "Well, how am I going to store the icons? How am I going to store the strings?" and so on. And one of the things that I cared about was I wanted to make these programs be able to be done in different languages and not have to rewrite the code, right? So I wanted to separate out all the language stuff from all the code itself. And I started thinking about, well, if I had a little object-oriented database, which I-- since being in the Smalltalk team, I was thinking about objects. And we could actually do that. So I ended up going off of my track. And I kind of stopped working on the Finder and working on the fundamental part that would support the Finder and everything else with this little object-oriented database, which was called the Resource Manager, actually. And so I said I wanted to do this thing, and I talked to Andy. And Andy and Larry Kenyon basically got on board. And Andy ended up rewriting a lot of the Mac Toolbox to use the Resource Manager, and then of course the Finder used it for what was called the Desktop Database that kept not only the icons but how [file] types and applications would connect, and you would know that if you click on this thing and you want to open it, it should run this application. All that was stored in the Desktop Database. And that was all managed by the Resource Manager. Yeah.

Hsu: Okay, so the-- for the file type and information for this type of file belongs to this application, so that was something that you and Andy created?

Horn: Yeah, I thought of it. And I had this idea of a Type and a Creator. And Andy was-- I thought of Andy as the Yoda of the Mac group. Andy was like the wise, smart, guy that you would bounce ideas off of. And Andy used the Resource Manager for windows and menus and all sorts of things. But the concept of a Type and a Creator was unique and new to the Mac. Where the Creator would basically be what an application would export as, "This is who I am." And the Type was-- the file would say, "This is the kind of thing I am." And you would bind those together. So the Type and Creator system was the thing that allowed you to, for example, name a file anything you like, and all of the type and creator information would be hidden away in the file system that Larry [Kenyon] did. And that was just a bonus. We wanted to name anything you wanted to name and type it with spaces, and we didn't want to have filename suffixes or any of that stuff. We just wanted it to be very natural. And so Type and Creator came out of that.

Hsu: Right. So from the beginning, the design was, no, we're not going to do extensions, we're going to just do normal names.

Horn: No, no. These things are objects, right? They have types. They're of a particular kind, right? We want to file to be an object. And so with the file as an object, where do you keep its type, and if it's going to be manipulated by an application, how does it know? And all that had to be part of the system that was built and supported by the Resource Manager, the Type and Creator, and Desktop Database.

Hsu: Right, because the whole thing was object-oriented in your mind.

Horn: In my mind. Yeah, I grew up thinking object-oriented because I came out of LRG. And I wanted to do as much as I could that would make the system be as dynamic as it was in the Smalltalk world. Of course, you can't do that if you've got a tiny, tiny computer, your operating system is all in assembly code in ROM. How do you make things dynamic? And the best I could do was this object-oriented database, and the concept of separating out the non-logical part of the program into a separate area called the resource fork, where all the strings for the program and all the dialogs, menu bars, all the things that you would change when you're changing the program from English to Norwegian, you could just change all that separately and all the logic would stay the same, and you wouldn't have to recompile it. So that was kind of what I was trying to do is how do we get dynamic capabilities into this tiny machine? And that was the best I could do at the time.

Hsu: Yeah. So then the idea of saving that information into a separate fork, you came up with that as well?

Horn: I think-- maybe that was something that Larry came up with when I was talking with him. Larry Kenyon. The idea that there'd be a separate part. I probably was thinking that that would be the only thing. I used to think-- maybe I thought the file was just the object-oriented database. But Larry decided that there should be two forks. And it made it easier, and you could open the resource fork separate from the data fork. It turned out, for an application, the data fork was empty. So it was the case that the resource fork was basically the entire program. And there were code resources that would be loaded from that part, too. For documents, you would have the document data in the data fork, which the type of that document represented in that data fork. And the resource fork was for other things. So the other things might be, there are fonts that need to be used for this particular document, for example. Things like that. It wasn't really used so much in the end, but that was the intent.

Hsu: Right. Okay. I remember later on, the Mac had the ResEdit program that would allow you to edit these. Was there an early version of that back in the original Mac? Or was that something that was made later?

Horn: No, that was-- I think we had ResEdit in the original Mac. I mean, if I had to guess who wrote it, it would probably be Steve Capps. I don't really remember. I should call Steve and ask him. But that's the kind of thing, he could bang these programs out in no time, but that was one of the things that we used to kind of show that we could actually change things without having to recompile a program. And ResEdit

became one of the fun things people would do with their Macs. It's like, "Oh, gosh, look what I can do. I can totally customize my Mac however I want. I can change the menus. I can change how this program works or what the program looks like, I should say. And people did a lot of that, and there was a ton of creativity expressed through ResEdit. A lot of fun.

Hsu: Yeah. I remember-- my first computer was a Mac. And I remember starting to play with ResEdit myself. So it was definitely one of those early experiences where I was like, "Oh, I'm actually getting into the system and actually getting to explore the fonts and the different pictures and whatever the various resources and assets that are in this program."

Horn: Exactly. Exactly. It was amazing.

Hsu: Yeah. Let's see, okay, we did that part. Okay, so the-- oh, here's one thing that I was interested in. So I think Folklore.org mentioned that some of the Finder design might have been influenced by something that was done at MIT called Dataland.

Horn: No. I knew nothing about Dataland. No. I mean, the only influences I had at the time was PARC, right? Was Smalltalk. So I knew nothing of Dataland. The Finder came out of my interpretation of what one could do with a little computer based on what I grew up with at PARC. I didn't know anything about Dataland.

Hsu: Okay. So your decision to stop working on the Finder and spend a lot of time working on the Resource Manager became kind of a contentious issue with management.

Horn: Right.

Hsu: Is that something you can talk about?

Horn: Well, so Bob Belleville, who had been at PARC, he was our manager. And Bob was a very solid kind of software manager engineer type. Also very kind of old-school. And most of the people in the Mac group were pretty young. We were all in our 20s, and none of us really had a whole lot of experience with authoritative management, especially me. And so Bob and I disagreed on the importance of the Resource Manager. And I just wouldn't yield. I basically said, "No, we have to have this." Right? Because I was counting on it for the Finder. Bob thought, "What's this computer? It's just going to be another computer down the road and another one after that. What's so special about this Mac? And I thought it was-- here was my chance to really do it right, the way I thought it should be done. And so I basically just told him, "I have to do this." And I had backup from Andy and also from Steve to finish the Resource Manager. And I do remember at one point promising something about being done in two or three months, and I put stickies up day by day all around my cube, three months' worth. And I'd pull one down every day. And Andy recalls that after the last one was pulled down, I kept working on the Resource Manager anyway. But by then, it was obvious that it was the right thing to do. But yeah, there was some contention. It was tough, because we were a little bit renegade in the Mac group. We were kind of on a mission, thanks to Steve. He told us we were on a mission, and so we kind of felt we had latitude to do whatever we thought

was necessary. And that didn't necessarily match with the standard software production method or prioritization that one would bring in from someone with a military background, as Bob had, or kind of an old-school, big-company background.

Hsu: Yeah. So I mean, curious, though. So if you had not done the Resource Manager, what would have been the technical alternative to solve-- to provide the same kind of functionality?

Horn: Yeah, see, I don't actually know. Even now, I think it would have really hamstrung the Mac. I mean, I don't think it would have been possible to do-- have the internationalize-able system that we had. I think that we probably wouldn't have been able to do the complex programs that we'd had, because the Resource Manager would allow you to kind of swap in and out objects as you needed them. And we were so constrained with memory that we needed that capability. And it slowed the system down, but on the other hand, you could do things. If you didn't have a little object-oriented database, everything would always have to be basically in memory, or we'd have to have some other mechanism, which would be simply as complicated as the Resource Manager, if not more, but limited, right? So I wanted to do the generalizable thing-- foundation that would actually make all these things possible. So I thought that was the only option at the time. And even now, I have to think about what were the options. I still would probably go with something like that. Maybe something a little more sophisticated, but I only had three k-bytes of assembly code to work with. That was what was left in the ROM to build an object-oriented database in assembly code. So it worked out, but it was a struggle to make it fit.

Hsu: Yeah. Wow. Let's see, so who else did you work really closely with in the team?

Horn: Well, pretty much everybody. The team was small, right? We had 12 people at the time, I think, when the Mac shipped out. Twelve people in Smalltalk. Smalltalk? Twelve people in the Mac group, right? In the software group. So I worked, obviously, Andy, the Yoda of the team. And Larry Kenyon did the file system and a bunch of other things. Steve Capps, of course, I worked side-by-side late at night listening to Violent Femmes finishing up the Finder. Capps is probably one of the best programmers I ever met, along with Andy. I mean, and Larry. These people are amazing. Of course, Larry ended up marrying Patti. Patti King became Kenyon and she was our software librarian. We were a very close team. And of course, Burrell Smith, who did the hardware. So even though he did the hardware, he was part of our group. He was a good friend of Andy's. And we all-- we'd have dinner together all the time. It was kind of a family thing. So I would say who'd you work closely with? Well, we're all working hard on our parts, and we all had to collaborate because it was such a tight system, and we were just like family. It was great.

Hsu: Yeah. You mentioned the way that Steve would motivate the group. Is there some example of that that you could give us?

Horn: Oh, sure. Absolutely. I mean, he would come by and he would look at everything, and he would have comments. I think, for example, the title bar of the window should look a certain way, and I think Susan [Kare] did dozens of examples before Steve signed off and said, "That looks good enough." One thing I remember is Larry Kenyon had done the formatting software. So you format a disk when you put it in, and that was another innovation of the Mac is if you have a disk and you put it in, it's not formatted, it

would format it for you. On the PC, you were out of luck if you were trying to save something and you didn't have a formatted disk. It couldn't do that. So Larry built the file system and wrote the formatter, and we demoed it to Steve. And I remember Steve saying, "How long did that take?" or something. And it was like, three minutes. And he says, "Three minutes? That's way too long. We're going to have millions and millions of users. If you save two minutes off of that, that will be years. It will save years off people's lives. So you have to make it faster. And so Larry went and he just optimized the heck out of it and made it a bunch faster. So Steve would think of all these creative reasons that are kind of outlandish about why you should do something. It's like saving lives by making a faster formatter. Right? That was kind of crazy, but we got the point. We got the point that it was important. And he would come by and one day, "You guys are the greatest." And the next day, "You guys are idiots." And he kind of kept you on your toes by being critical, but in a way where he was motivating. We took criticism in the right way. It didn't stop us. It was like, "Okay, we can do this better." Yeah.

Hsu: Yeah. And one of the stories about Steve is that he sort of pitted the Mac group against the Lisa team in some way. There was this rivalry that he stoked. Is that something that you felt?

Horn: So I didn't feel rivalry so much as kind of a race. So it's like we both want to get this thing out, let's see how we can do. Let's get this thing done. Basically Bill Atkinson was from the Lisa team, and he was also [on] our team, right? So he was our bridge between Lisa and Mac. And he actually brought all the work he did on QuickDraw from Lisa to the Mac and he had to rewrite it for the Mac and make it tiny and assembly code and fit in the ROM, which basically it took up half of the ROM. But I didn't remember feeling a rivalry. Mostly we were very focused on what we were doing, and just getting it done as fast as possible and getting it out. With that kind of focus, you don't think much about what other teams are doing.

Hsu: Let's see, so you also worked on the Dialog Manager?

Horn: Right. So the Dialog Manager was just how do you ask questions of the user, get some answers, fill in some fields and so on. The Dialog Manager ended up being an extension of the Window Manager that Andy had written. And I designed it so that it was compatible. So a window—a dialog window was still a window. So it was a subclassing concept that I had learned in Smalltalk that's like, "Well, okay. If you make a-- if you extend the window data structure with some things on the end that make it a dialog, it's still a window as far as the window manager is concerned. So you can do all the window things, but also you have a little bit new functionality that makes possible-- is made possible by the additional things at the end of the data structure. So I worked on that and wrote that code. And I'm pretty sure it's one of those things where a bunch of people had concepts that they wanted to put in and helped me with some things. And I'm pretty sure that Capps wrote a little piece of that here and there as was necessary. And we all kind of pitched in and made things happen when needed. And I was kind of overloaded. We were all a little bit overloaded, but Steve Capps basically had the ability to dive in at any point and work on any part of the system and make it better. He's an amazing guy.

Hsu: That's an incredible skill. Were there any other parts of the Macintosh Toolbox ROM that you contributed to?

Horn: Really, so the Resource Manager, the Dialog Manager and the Finder, and the concept of Type and Creator, Desktop Database, just how the Finder worked. That was primarily what I did. I helped a little bit on the demo that was shown at the introduction. Again, we all kind of dove in on that. Andy and Steve Capps and I, and Susan Kare and a bunch of people worked on little pieces of that. But I think Finder and Resource Manager, Dialog Manager were my primary contributions, and Type and Creator and that whole kind of architecture.

Hsu: Right. That's a lot.

Horn: It was a lot of fun.

Hsu: That's a lot of stuff.

Horn: A lot of work.

Hsu: I mean, the Finder itself is huge.

Horn: The Finder was 46K bytes. Forty-six K bytes, right and the Lisa Filer that they had done-- they kind of redid the Filer after they saw the little pre-demo that we did of the Finder-- that I did of the Finder. That weighed in at, like, 360 K bytes. So ours was only 46 K.

Hsu: So what was it like when the Mac was introduced and the announcement and then the 1984 ad, and just all that hype surrounding the launch. What was it like being part of that.

Horn: It was-- well, we were pushing hard for that demo, right? I think we had-- I think Steve told us only a few days before the actual introduction, "Oh, we need a software demo. We need an intro demo." And again, I think Steve Capps and Andy kind of led that effort to have this demo and run something that was exciting to put on the screen. But they also had that MacSpeech that introduced itself. So that was not very many days before, so we were working all the time, up to the point. So when we were sitting in the front row during the introduction. It was more of a sense of massive relief. It's like, "Well, it's out there and it's running," right? It's actually functioning. We were worried it might crash or something during this demo. But it was actually running. A sense of relief, and then pretty much right after that, we all went back to the office, because we knew we had more work to do. And we ended up doing another release about four or five months later. I did another release of the Finder, to fix a bunch of bugs. I did a lot of walking around and going to computer stores to see how people used the Mac, right? What do they do? Do they understand how it works? What kinds of things do they find difficult? And I found out-- and there's a story on Folklore, Andy's site, I found out that people would just start typing. And the way that I had set it up is the disk that you put in that it launched off of would be highlighted when the machine would start up. And because it was highlighted, if you started typing, it would actually rename the disk. So I had to make a quick fix to make it so you had to do one more step, so that not all the disks would get named some random garbage. But those are the kind of things you only get in user testing. And we didn't do any user testing. We just made what we wanted. Of course, nowadays, that's not possible. You have to do massive user testing and all sorts of things before anything gets shipped. But back then, it was just a handful of us

doing the software and a bunch of people doing the hardware and marketing. Less than 100 people total. Yeah, amazing.

Hsu: Now, it seemed like it was such a death march at the end. Were you all really burnt out finishing the Mac?

Horn: I think so. I think a lot of us had just worked ourselves to death, right? And because we believed in it, when you read about what burnout is, it's not that you don't believe in what you're doing, it's because you do believe in what you're doing. And you just put your all into it. And so we all believed in it to the max, and we just put every bit of our effort into it. So in some sense, burnout is an indication of the passion and commitment. But yeah, after that, you're kind of tired and maybe you want to look at something else for a little while. And you saw that people would kind of move on after the Mac came out. And some people would stick around for a while, but it was a hard push. A hard push. In retrospect, what I wish I had done is just kind of taken a year off and then come back. That would have been-- I don't know. The history might have just been the same anyway. But I still-- even to this day, I'm grateful that I get to use a Mac in my everyday work. And there were some times in the '90s when it was a little touch-and-go with Apple. So I just give a ton of credit for the people who made it through and kept that company going. And of course, when Steve came back, when Apple bought NeXT, that obviously-- Steve's genius and passion basically resurrected Apple and made it a real company again. <01:07:44>

Hsu: Yeah. So you mentioned you should have taken a vacation. So you did not take a vacation after the Mac shipped?

Horn: No.

Hsu: You just stayed on and kept working?

Horn: Well, I stayed on for a little while, and then I ended up leaving. So that was in '84. I ended up going and working at Adobe for a little while and then I went to grad school at Carnegie-Mellon.

Hsu: Oh, okay. Right. Was that because you felt like it was time, or the team had already kind of split up? What was the reason you decided to leave at the time?

Horn: I think the team was a little tired. It was hard to keep together. I was tired. And I also felt like I'd given everything and maybe that was enough for me right now. And I went to Steve and I said, "You know, it's been a really great run. I feel like I've delivered a lot, and I need to move on." I told him that. He tried to get me to stay, but I just had already made my decision, and I wasn't going to change it at that point. And I'm glad I went to Carnegie-Mellon. That was another tremendous experience. I couldn't have picked a better place to go. And I also had a good time working at Adobe for Chuck Geschke and Warnock, of course ex-PARC people. That was also a tremendous experience, but very short. I was just a consultant. And then I ended up going to grad school.

Hsu: Right. So the Adobe-- I mean, so you went there primarily because of Geschke and Warnock, or because-- I mean, was it because they were working on PostScript for the LaserWriter?

Horn: Right. So they were doing the LaserWriter and they were doing PostScript. And I ended up-- I can't really remember how I got connected with Adobe, but it was a really small company at that point, you know, 20 or 30 people. And they were out near Palo Alto Airport on Embarcadero Road. And so I came in as-- I don't think I was even an employee, but I was a consultant. And I worked on the first LaserWriter spooler. The idea that you could spool printing jobs to the system and it would-- you could do other things while it was happening. That was a learning experience, too. Because it was something I didn't really know how to do, but I figured it out and I got help from some great people at Adobe. But it was really funny, because at Adobe, I met a couple of guys from Carnegie-Mellon who were there for the summer. Randy Pausch and Mike Young. And so they were basically my age and they were grad students, and I had just come from Apple. And I invited them over one day for dinner and I said, "You know, I'm kind of thinking about going to grad school." And Randy said, "You should definitely go to grad school, and you should go to Carnegie-Mellon." And so basically Randy was one of these guys who had his opinions, and they were strong ones, and they were well-supported. And so I ended up applying to CMU and going. And also talking to Geschke and saying, "I'm thinking about going to grad school at CMU. What do you think?" And Geschke is CEO of Adobe, and if I had stayed there, I'd be retired by now, with lots of money. But Geschke looked at me and he sighed, and he said, "If I were your dad, I'd tell you to go to Carnegie-Mellon." And that's all he said, right? So kind of the implication was, "I'm not your dad. You should probably stay here." But he said, "If I were your dad." So I took that advice and I thought, "Well, that's good advice." And I went to CMU and I don't regret it. I mean, you go through life once and you make choices, and your path through life is what it is. And CMU was a tremendous experience and I met a lot of great people. But I'm still working. I'm still working today.

Hsu: Yeah. So what did you study at CMU, and who was your advisor?

Horn: So I was a computer science grad student. Actually, Jim Morris and Jeanette Wing were my co-advisors. Jim Morris actually invented a number of interesting algorithms like Knuth-Morris-Pratt. Really interesting guy. He has a book out. I think Tales of a Computer Scientist. I can't remember the name of his book, but it was a great book, kind of of his life. And Jeanette, I went to Jeanette as I was getting close to finishing and I asked Jeanette if she would be my co-advisor, because I just needed to get finished. And Jeanette had a reputation of being the dragon lady. Just super smart and super tough, and she got her students out. So Jeanette agreed to be my co-advisor and guide me through the finish. So the two of them, I can't thank them enough for making my time at CMU productive and fun, actually. But I did work really hard. Jeanette in particular really was a great-- at the last bit of my PhD work. Morris was fantastic at getting me into the system and figuring out-- helping me figure out what to do and supporting me. Yeah.

Hsu: Yeah. It's incredibly important.

Horn: They are just tremendous people. Yeah. I feel so lucky when I think in retrospect about the people that I met from PARC and Adobe and Carnegie-Mellon. I can't imagine how lucky I am for the person that

I am and what I learned and the opportunities that I've been given by these people. So I just am grateful every day for that path through life and the people I met.

Hsu: So what was your thesis about?

Horn: So it's funny, my thesis was called Siri, S-I-R-I, just like Apple Siri. It's a girl's name in Norwegian.

Hsu: No relation, though, right?

Horn: No relation. No relation to Apple Siri, but it was kind of an exercise in how small can you make a programming language that did object-oriented programming and constraints at the same time? So constrained objects. I was influenced, of course, by Smalltalk. I was also influenced by work done in Denmark on a system called Beta. And I ended up with a little programming language that was more of a toy programming language, but showing how small one could express the constrained object concept. And so that's what it was. And actually, it did run, but it was super-duper slow. I would have had to have a lot of work to make it into a real language. And it was more of a kind of exploratory paradigm. It was interesting, though. It was fun.

Hsu: Was Eiffel also an object-oriented, constraint-based language?

Horn: I don't think so. Eiffel was-- I don't know enough about Eiffel. I can't remember, but it was-- the other object-oriented constraint language was-- it was very contemporaneous-- was Kaleidoscope by Bjorn Freeman-Benson, who happens to be one of my best friends now. Bjorn and I were doing similar theses at the same time. I got Bjorn into flying. We've stayed close ever since. I've known Bjorn forever. But he had done Kaleidoscope, and that-- he was advised by Alan Borning, who had been at PARC and had done ThingLab, which was one of the early object-oriented constraint systems. ThingLab was an eye-opening and amazing system at the time, and Bjorn's thesis was kind of taking that to the next level. And mine was kind of going to the other way. It was like, well, how small can it be? How expressive can you be in the smallest possible space?

Hsu: Right. So you had already been aware of ThingLab from before?

Horn: Oh yeah, I think I was aware of ThingLab. And I thought, "What a neat idea." But I don't really remember it influencing me that much, because I didn't know about the internals of ThingLab. I just thought it was an interesting idea. I was more, actually, motivated by the idea that a constraint system would be useful in graphical user interfaces, like for saying things to have certain relationships on the screen and so on. Which is kind of not a very important part of what constraint systems could do, but at the time it was like, how do you build interfaces in an easy way? How do you describe them and express them in an easy way. That was what was driving me, less than the simulation of the real world that ThingLab was so good at.

Hsu: Right, okay. Yeah. So that's actually kind of like the way auto layout works today, right?

Horn: Right. So I'm sure auto layout's way more sophisticated than what I did, and similarly I thought Kaleidoscope had fundamentally better ideas than I had. But mine were different enough that they were interesting and could have been explored more.

Hsu: So you finished grad school when?

Horn: I think it was 1994. January of '94.

Hsu: So then was that-- after that is when you went to-- on your resume, you've listed that you were sort of doing a lot of different consulting for different companies.

Horn: I was doing a ton of consulting. So kind of from '94 to 2007, I was just consulting and doing little projects. I had this idea that I could work remotely. So I had bought a house in the eastern Sierra at Mammoth Lakes. And I was trying to work remotely and do projects. So I did lots of consulting projects from there. And I also had a project of my own that was called iFile. The idea was what would a fresh take on the Finder, desktop, look like, if everything was an object, including your emails and your photos and your files and your messages. What if they were all a uniform object space. What would that look like, and if you could then kind of read the text in your documents and so on, what kinds of things can you do with some rudimentary natural language processing to do automatic collections. So that was what iFile was about, and I was working on that 100 percent on my own, writing it in C++ on the Mac. Again, that was probably something I could have done in one-fifth the time if I had been using Smalltalk. And it never shipped, but I used it for quite some time myself, and I think the ideas even today are worthy. I'd have a different approach to doing it, but I think having a kind of unified space where you had everything has value. Apple and other companies went a different way, which is for every type of media, your photos, your messages, your texts, there's a different app. So there's the Messages app or there's the Photos app, and then there's the Finder, which is for files. They are just different apps. My idea was there would be one app. There would be this thing called iFile, and all that stuff would come in and be automatically organized for you.

Hsu: Right, but I think in the mid-'90s, there were attempts at doing things like OpenDoc that [were] more document focused, that sounds similar to what you were doing.

Horn: Right, right. So actually, one of my friends is Kurt Piersol, one of the architects of OpenDoc. I think it's just a really hard problem. It never really made it at Apple because it was just, I think, too hard for developers to get engaged in. And also, it's easier to write an application that you own end-to-end than to write something that gets plugged into a bigger system where you don't actually know if your application is running. I mean, yes, there's a photo editor in your document. Whose photo editor is it? How do you pay for that? How do you get compensation? It kind of goes against the market reality, whereas if you have a branded thing, like Photoshop, you can pay for it, right? So an integrated system kind of goes against the market.

Hsu: I mean, in a lot of ways, Smalltalk is also an integrated system. It doesn't really have individual applications, although I guess you can make individual applications. But you've never done any commercial Smalltalk work?

Horn: Never done any commercial Smalltalk work, although there are commercial Smalltalks out there. There are commercial Smalltalks. And there are Smalltalks with the capability of building an application and then exporting that application as a standalone thing, although I don't have any experience with those things. But I think that's the way to go, is [to] do all of your development in a Smalltalk system, push a button and get a built application out that you can market as a separate entity. I kind of hoped that that would happen with Smalltalk back in the day. Smalltalk had kind of a dual purpose. One was like this evolving system that kept changing, and you had Smalltalk '72 to '74 to '76 to '78 to '80. All of these different systems were really radically different from the previous one, and they were evolving at a tremendous rate. And as soon as Smalltalk became kind of a thing in the outside world, it stopped evolving. So that's kind of an unfortunate situation, where you would like to keep doing the next Smalltalk that has evolved within the environment of the internet of today. But that hasn't happened. There haven't been really market forces that would force that to happen.

Hsu: Yeah, okay. One thing that you consulted on was for Apple's Advanced Technology Group?

Horn: Right. Right.

Hsu: It was something called LiveDoc?

Horn: Yeah. So Jim Miller's group at ATG was working on the system called LiveDoc. I wrote a lot of the code for that. And what it was was what kinds of things can you recognize in text that you can have actions on? Like phone numbers and addresses and things like that, or even just words to look up meanings, for example. So I wrote a little bit of that system, and we built a version of TextEdit that was live, live TextEdit, and you could actually do that. And that was a precursor to what we have on the phones today, where you can tap on a phone number and it will call it and so on. But yeah, I was a consultant at Apple. I guess, I can't remember when that was. That was probably late '90s when we were doing that.

Hsu: Yeah, okay. And then you also worked for Maya?

Horn: Yeah. So Maya is a company that Jim Morris and a number of people founded in Pittsburgh. And they were working on things that were kind of along the lines of stuff that I was interested in about information visualization, user interfaces that were novel and advanced. And they had a program called Hyperfax, and I helped a little bit with that, with basically 3D rendering of documents, for example. And one of the things that I did was because they were running on a PC, and we were writing the software that needed to do certain things with bitmaps, there was a particular feature in the Mac OS called MapRgn ("Map Region"), which would take any region shape and you could map it to some other rectangle. It would translate and modify the shape, as different rectangles. So it would shrink it or expand it or stretch it or whatever. So MapRgn was this function, but it didn't get copied in the Windows PC OS,

right? It didn't get copied in Windows. So I ended up reverse-engineering it. Gosh, now I'm going to have to remember his name (Joe Newcomer). With a Windows expert. I would love to come back to this if I can remember his name.¹ Anyway, and we reverse-engineered the Windows region concept and we ended up-- I ended up rewriting it, and making a version of it, and we wrote an article in Dr. Dobbs' journal about this, actually. So that was-- and we ended up using that in the Hyperfax system.

Hsu: Do you remember which issue?

Horn: No, no idea. Gosh, I'm really-- I feel terrible, because right now I should know the name of..

Hsu: We can look it up later.

Horn: Yeah, please. And then maybe we can rerun it a little bit.

Hsu: We can put it in the transcript.

Horn: Yeah. Because he was a great guy to work with, and the name is on the tip of my tongue.

Hsu: Yeah. We can fix that in the transcript. So you also did video codecs for Eloquent?

Horn: Yeah, that was interesting. So this is a company based in the Bay Area, right? And they were doing-- I'm getting things a little bit mixed up, but they were doing instructional software and needed to have video compression and so on. So it turned out that Telenor in Norway had a video compression system that they had done. And we actually could use it. So I decided that I would go over there and talk to them there and talk to them and see if we could license those. So I actually went over to Norway. I happened to be there anyway, and I went to Telenor and met the guy who had done this codec. And I said, "Hey, we would really like to license this." And he kind of looks around, and he says, "We don't know how to do that, so we'll just give it to you." So they just gave us this codec and we integrated it into the Eloquent system. That was pretty surprising.

Hsu: Wow, yeah. Pretty rare. So no royalties paid?

Horn: They just handed it over.

Hsu: Wow.

Horn: That was in the day, right? Because they probably didn't have any mechanism within Telenor to do any of that at the time.

Hsu: Yeah, wow. So then in 1999, you started a company called Marketocracy?

¹ <Bruce, if you can remember his name, we can insert it into this footnote here.>

Horn: Marketocracy, right. So Ken Kam I had met through friends and so on. And he was a very successful fund manager, and he had done a fund that-- a technology value fund that had done super well. Okay? And so he wanted to move on to the next thing, and we kind of had this idea together of, "Well, what if we could actually take the world's best investors, kind of get them off the net and pull them together into a team and actually use the combined brilliance of these people and make a fund," right? And so the idea is you would have a model fund that you would do yourself. It would actually have to abide by all of the mutual fund rules, and you could buy and sell stocks in real time, but with fake money. It would be using a delayed feed, so we wouldn't be having to pay anything for the feed. And then we would rank people's performance over the months and years, and over time, certain people would end up the best people. And then we would pull them together and create a fund based on their trades. And we would follow them. We would basically follow their trades and then create this fund together. And actually, it did very well for a while. Ken had some issues with how it was trading, because it was doing a lot of trading because you might have had 50 people who were bringing this in, and so you have a lot of stocks. And so it ended up not being fully automated, and the performance suffered a little bit. And I had brought a number of people into the company, and so it's one of those things that's like, "Okay, the vision has diverged." His vision, our vision has diverged, and so we went our separate ways. But I thought Marketocracy was a really great concept, and it was one of the early internet crowd-funded or crowd processing systems. We wrote the whole thing in WebObjects, which was an Apple thing at the time. And so we ran it on little servers. And it was really quite interesting. We learned a lot about what it took to build a website and a website that had a backend that did the actual work. It actually turned out to be a lot of compute to bring that all in and deal with real-time streams that were going to be right. Or right now, and, "Oh gosh, that price I gave you ten minutes ago was wrong. It should be this." And so you had to go back in and fix everything. And there was a lot of managing errors and the unreliability of the data streams that were getting in. So it was a real big learning experience for us about dealing with a real world that wasn't perfect. Yeah. Very interesting stuff.

Hsu: Why was WebObjects chosen?

Horn: I think because Blake Ward and I, who was one of the guys that I brought in, we were both Apple people. We believed in WebObjects because WebObjects was an object-oriented approach to doing web systems. It was based on NeXT, kind of NeXT thinking. And that was the most advanced thing at the time, and we believed in it. And we believed in the people who had designed it. And actually I would have done it at the same-- at the time, I would have basically, looking back, made the same decision. It was really a good system, actually. Really neat to see.

Hsu: So then you worked at WildPackets after that?

Horn: Yeah, I was just a consultant. WildPackets was a neat little company doing kind of packet analysis and system management. And one of the things I helped them do is how do you take it to the next level in terms of analysis? How do you look at things in terms of trends and monitoring and kind of figuring out what's going on and basically setting alarms. And so I kind of helped them design a little framework. And actually, I brought in the idea that why not save a bunch of these packets as they're coming by and use

the database and so on. And so I kind of brought that thinking to them and helped them design that. WildPackets was a nice company. I really enjoyed working there. Great people.

Hsu: How did you start getting into AI and natural language processing?

Horn: So I'd been working on iFile, which was doing some rudimentary natural language processing. Very rudimentary. And of course, it was English-only. I had always been interested in languages and thinking languages were kind of an interesting topic of study. And I remember I was reading news and I saw kind of an announcement in a little interview with Barney Pell, who I'd known through the grapevine that he had started this company called Powerset. So I sent a note to Barney. I said, "Hey, I wish you would have called me. I would have loved to join your company. I want to join your company now." So I ended up joining Powerset, and that was my kind of first introduction to the real world of NLP at the time. And so I learned a lot. I came in to build tools. I ended up running a chunk of the company, and I learned a lot, because we had Ron Kaplan was there. Livia Polanyi was there, people from Xerox PARC and from places that I knew and I respected, and they brought in the Xerox PARC technology called XLE, the Xerox Linguistic Environment. And so we had XLE, we had CFSSM, which was their finite state system, and I learned a lot about how you model and what a natural language pipeline would look like. And that was just really interesting. And I also learned a little bit about search, right? Because it was a natural language search system and the people who were building that part were ex-Yahoo. And so they were extremely up on all of that, and I learned a bit about search and what mattered in search. And learned a lot about what relevance was versus just "can we do natural language search." You still have to get the right answer, a relevant answer. Yeah.

Hsu: Yeah, so then Powerset was acquired by Microsoft.

Horn: Right.

Hsu: In 2008. So then you continued on and became part of Microsoft.

Horn: Right. So when I ended up going to Microsoft, one of the things that happened was they looked at us as kind of an acqui-hire. It's like, how do we make Bing better? And one of the big things that Bing was having trouble with was the search engine result page. After you do the search, you have a display. What does that look like? And they were having some trouble with that. So they basically came to us and said, "We're taking half your team." So they took half of Powerset and put them on the problem of making Bing's search engine result page awesome. And so our search team, the people who did the search side of things at Powerset looked at the code, and they ended up saying, "Well, we're going to do a new one," so they rewrote the search engine result page, and so that's where a lot of the features that you're used to now, where you have the structured results about who this person is and the related results and all these things, that came from the Powerset team. It went to Bing and then brought that to the masses with Bing search and a result page, which was, I think, copied by the other search engines. And so I think that Powerset kind of changed the world by going through Bing and showing what's possible in terms of what a good search engine result page would look like, a useful one, rather than just a set of pages that have potential information that you would want.

Hsu: Right. So you were part of that move to the-- Bing?

Horn: I was part of that move, but I wasn't part of the 30 people that went to the search page rewrite. So I stayed with Ron Kaplan and his team, kind of the natural language side. So it was Ron Kaplan and Martin Vandenberg, and I think Livia Polanyi was there. And we were working on how you get more natural language-ness into Bing. And so I mostly just worked with Ron and was working on things like summarization and natural language projects for Bing on that side.

Hsu: So then you left Microsoft in 2011?

Horn: Right.

Hsu: And then you joined Intel.

Horn: Right, I joined Intel.

Hsu: So how did that happen?

Horn: Well, the funny thing was is I had gone to another little company in the meantime trying to help them with their natural language problems, and that didn't turn out. And I was kind of thinking, "Okay, what's the next place?" And I called some of my friends. And I called a friend, Scott Love. And I said, "Scott, so I'm thinking about moving on to the next thing." He goes, "You should come to Intel. It's super great. We're doing awesome things at Intel now." And I'm going, "Intel? Big chip company, right?" Well, Scott was right. They were doing a bunch of great things. They had hired Mike Bell, who was leading kind of a brand-new effort. And they had hired Steve Holmes also, former Apple and Nike. I ended up working for Steve. And we had a group called Smart Device Innovation. And what I wanted to do is bring into Intel a kind of end-to-end capability in AI. So that if we knew what every piece of the AI pipeline looked like, we could write software and build chips that would support that, to build kind of an AI capability within Intel. And the way I decided that that would be best to do is to do a personal assistant. And I was very interested in kind of the personal cognitive assistant. Again, going way back to Doug Engelbart, how do you use computers to make the world a better place? To augment human intellect? And I thought a personal cognitive assistant would be the way to go. Well, the way things ended up doing, we ended up doing something slightly different. We ended up doing the Oakley Radar Pace, which was a running and cycling coach in glasses. And so I hired people. We hired some brilliant people for my team. We did a few acquisitions, put together a team with the right capabilities and confidence and passion to do something like this, and we built this product, which was really quite interesting. It had a full conversational interface. It modeled 12 different features of dialogue. We did our own speech recognition system. We licensed a speech-- a text-to-speech system. And all that ran on your phone. So it ran locally on the phone--it didn't require an internet connection. And of course, it had a backend on the phone that did the coaching, figured out at what point should it say something? Are you keeping up with what it wants you to do that day on that workout? It was very sophisticated and really quite good. And actually, I used it for a year or so and it actually got me into running, it was so good. But that was kind of-- that was Intel's foray into a consumer product with Oakley. And I think Oakley and Intel both chose to discontinue that, because it

really wasn't in their main line of business. Oakley didn't really want to have computerized glasses and have to deal with a web backend. Right? That was not their thing. And Intel, again, Intel has had challenges with its consumer outreach. But it was a great product at the time. I had a lot of fun, and again, hired some brilliant people, people like Jim Firby. Again, Martin Vandenburg, Livia Polanyi were involved in that.

Hsu: So interesting. So this would be-- you would wear the Oakley sunglasses, and they would be-- basically they would be smart sunglasses.

Horn: Right.

Hsu: And they would be connected to your phone?

Horn: Right.

Hsu: And so you would be running wearing the glasses and in your phone? OK

Horn: Right. And you would say, "Okay, Radar. What's my workout today?" And it says, "You're going to run for seven miles and climb 800 feet." And you would say, "Start workout." And as you're running, the glasses would be doing all the sensor fusion and checking your heartrate and how fast you're running, and it had a bunch of sensors in the glasses. And it would decide how well you're doing. And it might say something like, "Take shorter steps," or, "This is not a sprint workout. Slow down." Or it would just kind of speak up and say-- it would be quiet for a while and then it would notice that maybe now's a good time to tell you about your overall goals, and would just start to speak up about, well, the goal of this workout is blah, blah, blah, and it would tell you these things. It was very engaging and very natural. And people started to really enjoy using this coach that was real-time, and it would give you advice as you're doing your running or cycling. I think it was one of the most innovative products I worked on since the Mac. And that got me very excited about the whole idea of personal cognitive assistance, and I'm still working on that.

Hsu: Yeah. Well, I think it's also very interesting, because I think this must've fit into-- Intel was really big into Internet of Things and quantified self-type stuff at the time, right? So this was all part of that whole push?

Horn: Right. A part of the whole smart devices, right? And Steve Holmes was running the smart device group, Smart Device Innovation. So there were a bunch of different little projects that were going on. This was kind of the major project, probably with the most people at the time. Again, when you're focused on one thing, you don't see a whole lot of what else is going on. I did a lot of work with Intel Capital doing technical due diligence on companies that we might or might not acquire. And one of the companies I did bring in was kind of in support of the intelligent cognitive assistance area, a company called Saffron Technology. And we actually ended up using that technology within Intel and doing a lot of cost savings. But it didn't ever get integrated into any of the other systems we built.

Hsu: But the system that you worked on actually did become a shipping product?

Horn: Yeah. It became a shipping product, yeah. In fact, I was overseas one year, and I was walking through a town, and I saw an Oakley store. And I walked into the store and I said, "Do you guys have a Oakley Radar Pace?" And they said, "Sure. Let me tell you all about it." So that was pretty neat. And yeah, it sold for a few years, and then I guess they just stopped selling it. If you search for it, you'll still land on the Oakley site, but it'll say, "We don't have that anymore. That page doesn't exist."

Hsu: Maybe we should get one for the museum's collection.

Horn: Yeah, yeah. It'd be a good thing to have.

Hsu: It'd be something pretty interesting.

Horn: You can still buy them from other places, yeah.

Hsu: Oh, okay. And let's see, that comes pretty close to the present. So you continued to work on AI systems and personal assistants.

Horn: Right. And I'm still very interested in trying to use computers to help the world be better. How do we run simulations to see where the world is going? How can we steer it better? I'm very, very interested in doing what I can about the climate crisis, and anything we can do to allow the scientists to be better scientists, allow individuals to know more about what the truth of the world is, I think is worth working on. So I'm still very interested in the whole concept of a personal cognitive assistant and something that can reach a lot more people. I mean, we're going to have, what? Three billion more people on the planet by 2030, and a lot of these people, I think their interface to the world will be a handheld device. It might not have a screen, but if you have a conversational assistant that could really engage with the person and provide the information they need, that would have huge value. Something inexpensive, but reaching out to the whole internet in a way that basically meets them where they are. Speaks their language, understands what they're saying, knows about them to the point that it can provide proactive assistance and say, "By the way, I noticed that the price of such-and-such has gone down. You might consider selling your product in this other area. Helping people in their everyday world, but also bringing them up in terms of their understanding of the world on a broader scale.

Hsu: Yeah. Oh, I forgot to ask, so you're no longer at Intel. So you left in 2018.

Horn: Right, right. And I did a couple of startups in the meantime. One of them was Robust AI. I worked for them a little bit, learned a little bit about robots and robot planning, and now I'm back at Apple after about four decades. And I'm very happy to be back at Apple now.

Hsu: Okay. The reason you left Intel was just simply because the Radar system was discontinued?

Horn: It was discontinued. Also, again, Intel is one of these companies that they need to shift their focus on a regular basis at Intel. There are redeployments, people get moved around, or you're told your project isn't going to happen anymore. And so you have a certain amount of time to find a new job at Intel. I was in that situation, where our group was basically dissolved, and I basically chose to move on. I didn't find anything at Intel that was going to match what I was interested in working on. I loved Intel, though. That was another great place to work. Again, great people. I'm super happy to have been there for the time I was there. And I'm grateful for having had that opportunity. But at this point, I think I'm going to be at Apple for the rest of my career. That's the hope, anyway.

Hsu: That's an interesting homecoming.

Horn: Yeah, I think so.

Hsu: So you've been working in the computer industry since high school. And you've been in on the ground floor of the PC revolution, and now you're part of this AI revolution. How has the computer industry changed in the last 50 years? And do you think it's changed for the better or for the worse?

Horn: Well, clearly for the better and for the worse. So for the better, I would say the fact that computing has become commoditized, everybody has access to amazing computing devices and Macs and PCs that give you leverage, allow you to do things you never were able to do before. So much has changed. So in the sense that computing is everywhere and makes things better for people, that's great. When that happened, and the shift from research to commodity, which was late '90s, maybe even early '90s, that also slowed down innovation. Because when you had a small group that could move fast, like at PARC, innovation happened at this breakneck speed. And you can look at Smalltalk's innovation over the years, and how that was at breakneck speed. It just changed faster and faster and became more and more capable and could do more things and was more amazing. And you could build bigger and more amazing things. And then that kind of stopped as soon as that became kind of out in the world. It became static. When you're subject to the vagaries of the market, you're highly constrained as to what you're allowed to do. And a company has to have a certain amount of forward-looking courage to go outside of the mainstream and say, "We're going to try something very different. Something that might or might not have a financial return." So Alan Kay always said you should basically put ten percent of your money into something that you have no idea whether it'll succeed or not. You just need to invest in that, because ten percent of that might turn into something. But you have to have no expectations. You have to be able to think big. So I think right now, the thinking big is still constrained. Everyone's impressed by a lot of the things that are going on in AI. Things like Dall-E and GPT-3. But again, I think that's all incremental progress. It's not thinking outside the box with a goal toward doing something that is great for humanity. It's more like, "Well, what if we add another billion parameters to this big language model? What can we do?" Or, "What if we bring a bunch more data into the system?" Or, "What if we take this computer and make it 20 percent faster?" That's all great and everything, but it's not invention. It's not a new thing. It's the same thing, better. That's what innovation is. But it's not invention. It's not a new thing. So the question is, what is the new thing? What is the new thing that we're all going to look back ten years from now saying, "Oh, that was inevitable." But we don't have any idea what it is right now. So Alan would say, "Go ten years out, look back." Say, "What are we going to have in ten years that we look back and we

say, "Oh, that was inevitable," right? That's where, I think, we stand now. We need invention. We need people with the courage to go after things that they don't really know what the form will be. How will this actually be developed? Try things that you have no idea whether they'll succeed.

Hsu: That's interesting, that you mentioned that innovation is really just evolutionary, rather than invention, which is revolutionary. I don't think I've ever heard it put that way.

Horn: So Alan taught me that. I was watching some of his presentations, and I always thought innovation meant a brand new thing. But no, invention is the brand new thing. And the point of-- Alan's point making that distinction is that when you invent something, it's as hard to explain to someone else what that new thing is as it is for you to envision it, because you have no reference, because it's a brand new thing. There's no reference in your current experience that you can say, "It's like this thing, but different." So it's a very big step, both for the inventor, and for the inventor to explain it to other people. And that's what makes it hard, because it is a significant leap against your own background of knowledge. And so you can't really talk about invention just kind of in casual discussion, because it takes massive work to get out of your own framework, your own frame of mind, your own frame of reference. How do you get out of that and put yourself in another place that's unfamiliar. It's very difficult.

Hsu: Yeah. I mean, that kind of goes back to when Steve [Jobs] saw the-- Smalltalk. The fact that he recognized that there was something new there is kind of a big deal.

Horn: A huge deal.

Hsu: Because he saw something that other people might not have seen, and yet he also missed things, too.

Horn: Right, right. He missed a lot, because he didn't really-- he saw what was possible, but he didn't see how that was made possible. And so when you look at the early Apple operating systems, they were making that thing possible in a completely different way. We didn't have an object-oriented programming language underlying our entire operating system, where that's the case in Smalltalk. But we did kind of the surface stuff. We had menus and so on, and direct manipulation. So we were able to implement some of these things, but because it's kind of a snapshot of reality at the time, it's harder to evolve that thing than if you had a complete dynamic system underneath where you could change things very quickly. But invention is going to be very tough, because if you look at all of the James Bond movies now, and you look at all of the things from the past, it's like, "Yeah, we have all of that stuff." All that super-magic technology. We have it all. So what would a Bond movie of today show us? What would that be? And that's very hard to imagine. So we need the science fiction people, and we need the people who are really futurists to kind of imagine those things for us, and maybe we can help make some of that reality.

Hsu: Yeah. So where do you think the next big thing is going to be? What do you think it's..

Horn: Well, I mean, I can't really talk about what I think it'll be, because I'm now..

Hsu: Well, that's true.

Horn: And what I hope it will be is something that will allow everybody on the planet to understand our common destiny, and to act accordingly. We have massive, massive combining and merging crises. So we have environmental crisis of basically-- the climate crisis is major. We have issues with water and desertification. We have issues with population, and basically overusing the natural resources of the planet. And we need to pull together and say, "This planet itself, the thing that brings us life, is worth protecting." We need to have the concept of ecocide being something that is understood and real and used as a protection against further degradation of the planet. Biological diversity is going down at a tremendous rate. What are the consequences of that? We should basically be able to see into the future and say, "If we do these things, this is what life will be like." But everybody needs to be on the same page there. There are a lot of deniers, a lot of people who think, "Oh, it can't happen in my lifetime, so I don't care." I want to bring the future to everybody now, so we can all make the best decisions we can and be on the same page about that.

Hsu: In your view, what has been the significance of the creation of the graphical user interface, and the windows, icons, menus paradigm?

Horn: Just direct manipulation and modeless interaction. I think that made it possible for regular people who don't want to talk in code to computers to do real things. I think it absolutely was a seminal event, right? The bringing of the graphical user interface to humanity changed everything. And the reason it did that was what the graphical user interface does is it provides common ground. It meets us where we are. We know how to point at things. We know how to give commands to things, do this. It has a lot of context about who we are, right? Old computers, in the old, old days, the context you would have was whatever the commands you could type at the terminal. Or even a deck of cards. That's not very much of our reality, whereas a piece of paper on a desk is our reality. Pointing at things is our reality. And I like to say, "A click is not just a click." So if you think about a click, the entire meaning of a click is contextual. It's like, "Where did I click? When did I click? Did I double-click? Is it a menu bar? Is it in some text? It's all contextual, but that context is shared with us. It's something we intuitively understand. That's why the graphical user interface made it possible for regular people to learn to use a computer. Figuring out what the next "more context" is, how much more context can we learn and put in the computer to embody-- think about language. Now that Siri and other personal assistants can understand our language, that's another level of context. You have to get to more and more of what we are as humans to have a better user interface. And I think we're kind of on our way with these conversational assistants.

Hsu: Has there been anything in your career that we've missed that you would like to add?

Horn: I don't think so. I mean, I think we covered pretty much everything. It's been an interesting career. It's had ups and downs, but I've been very lucky to have the experiences I've had. I would say that some of the things I've been passionate about, I'd like to still work on. I still am very much an environmentalist. How can I use computing to help the world in that way? I think that's really important. If I retire, maybe I'll work fulltime on the climate crisis. I don't exactly know, but I've had a good run so far, and I'm grateful for it all.

Hsu: Thank you. Lastly, what advice would you give to a young person starting in the industry today?

Horn: I would say, "Just get involved. Get in and do stuff." Right? There's no substitute for doing things and trying things out, being fearless. I think being fearless is really important, and knowing that failing is not failing, it's just feedback. It's like, "Okay, that didn't work. What did I learn from that?" I think being fearless and realizing that failure is not a bad thing, that's my advice. I think just go with your passions, make sure that you work with other people and learn from other people. I think having mentors is really important, that you can skip a lot of learning by hard knocks if you have the right mentors. But I think just dive in, do stuff, make things happen, and be aware of other opportunities as they occur. Sometimes, if you have your head down all the time, you don't see other things happening. So every once in a while, look up and say, "Okay, what's happening out in the world? Is there something that I should be aware of?" So I know I did a lot of heads-down for many, many years. Probably okay at the time, but there might have been other things I could have done if I had looked up.

Hsu: Thank you very much.

Horn: My pleasure.

END OF THE INTERVIEW