Oral History of Raymond (Ray) Ozzie

Interviewed by:
Marc Weber

Recorded via Zoom December 10, 2020
Manchester, MA

CHM Reference number: X9392.2021

Weber: I'm Marc Weber of the Computer History Museum and I'm here on December 10th, 2020, with Ray Ozzie, who's a pioneer of groupware and many other things. Thank you so much for doing this.

Ozzie: My pleasure. Thank you.

Weber: Let's start with your full name and when and where you were born.

Ozzie: Raymond Edward Ozzie, Chicago, Illinois, 1955. Yeah.

Weber: Tell me a bit about your family.

Ozzie: So we were kind of a middle-class family first growing up in-- within the city in Chicago, eventually moving out to the suburbs out near O'Hare. Father was an insurance agent, said, "I don't care what you do in life as long as you're not an insurance agent." My grandfather we lived with them and he worked in the railroad and I spent a lot of time with him because we lived in a two-flat together. He was I think in retrospect probably quite influential [on Ray] because he was a builder. He would go to the railroad, he was a sheet-metal worker, would come home, had the shop in the basement, and I would just sit in that shop watching him do things for hours on end, days on end, and I think I probably have ingrained in my memory, the subconscious, this-- the smells of metalworking and what it's like to just be building.

Weber: Did you help out or mostly watch?

Ozzie: I would help out but I would mostly watch. I was pretty young at the time but it-- you never know how things influence you but I was at a young age very into electricity. My dad would bring home dry cells and wires and light switch and light bulbs and I learned about all of that probably in the third-grade, fourth-grade age range. I remember being very proud of myself in terms of what I knew and I had-- made a little-- a few quarters on the side by selling new doorbell buttons for-- to the neighbors and things like that.

<crew talk>

Weber: You were talking about selling doorbells. I was about to ask how about the classics of either model trains or--

Ozzie: Oh, yeah, very into model trains especially again because my grandfather was-- worked for the Illinois Central. My parents got me trains and built a big train set on a massive piece of plywood. It was probably a pretty typical way of growing up in that generation. As I grew older, as the-- as we started into the '60s, very into what was happening with NASA and obviously followed each and every launch but we all did at the time. They brought TVs into the classrooms and we got to watch each <laughs> one of them. Yeah.

Weber: Radio? Was that--

Ozzie: Yeah.

Weber: --an interest or--

Ozzie: I had a-- one of my gifts: My uncle bought me a crystal radio and I built that and early on my grandfather had a short-wave and we played with that a little bit so-- uh huh.

Weber: What did your mother do?

Ozzie: For most of my youth she stayed home with us, my older sister and my younger brother. Before having children, and after we left for school, she was an Executive Assistsnt.

Weber: Tell me a little bit about your neighborhood and what the environment was like physically.

Ozzie: Sure. In Chicago fairly deep in the city near an amusement park called Riverview, for those in Chicago from that era they'd know kind of where that is, but in a two-flat, very modest in a very ethnic German, Greek neighborhood. By the time I was in seventh grade, we moved out to the very edges of Chicago near the suburbs, a little area called Edison Park. It was a tiny, little home but it was our first house and eventually we just hopped over the border into Park Ridge and that's where I lived in-- until I moved to the East Coast after college and it's where my mom still lives.

Weber: Can I ask you what values you were raised with? Were politics, religion, ethics a big part of your family life?

Ozzie: Very religious, Catholic. I was an altar boy; everyone in the neighborhood was. We didn't have a ton of money, we-- but I didn't know that. We walked to school. My parents were very-- focused tremendously on contributing to society and doing so modestly, if you show modesty and respect, that was kind of at the core, and my grandfather who we-- my grandparents who we lived with were strict in that regard, my parents were very consistent with it, and that's how it was.

Weber: Talk a little bit about school. What were your favorite topics, least favorite, important teachers?

Ozzie: So I was pretty fortunate in-- to have gone to a great high school; it was Maine South-- Maine Township High School South and it was a relatively new high school, only open for a year or two when I went there, and it was a combination high school that had two very distinct tracks, an academic track and a trade track. If you were academic, you were heading off to college and it was all college prep, and my scores tended to encourage the advisors to make me want to go in that direction; however, my interests took me more toward the trade side and I did pretty much everything that I could on that side. There was wood shop, auto shop, print shop, but what interested me most was they had an electrical shop and these-- every time I say "shop" there was a very large area dedicated to these things with lots of equipment. In electrical shop, they used old navy-training films to begin teaching students the fundamentals of electronics; at the time it was all vacuum tube based but it's still the basic-- the same basic concepts of amplifiers and so on. We began with those films but eventually took apart TVs and

radios and tried to fix them and tune them and so on and this gave me a really healthy basis in electronics that ended up paying dividends later on in life, but what was probably most interesting and most coincidental is that the high school apparently-- now I understand this in retrospect was part of a grant program that the federal government had granted some number of high schools across the country both timesharing and computer-hardware resources to try to get the general population to start-- to begin to understand computers and computing technology. So in freshman algebra, the very first-- probably within weeks of starting high school my math teacher at the time, David Paisley, came into the classroom and just said, "Look. We were just delivered in the math department all of this equipment. We have no idea what to do with it and we're looking for a few volunteers who can just start playing with it and teach us what it's all about." And so I and two other people in different classes had volunteered to go in and begin playing with it and there were-- there was essentially a teletype connected to a prior timesharing system and it supported two languages, Basic and Fortran, and next to it there was a-- what I believe is probably credited as the world's first personal computer, Olivetti-Underwood Programma 101, and there's a couple of them back there, a wonderful machine. Today you would call it a programmable calculator but it had all the-- four registers and all the basic looping constructs and so between using this timesharing system and teaching myself Basic and Fortran and using the personal computer I began to have a good basis in what programming was all about and again that was fall of-- yeah, September of 1969. And my classmates didn't understand what I was doing, my-- the-- my parents didn't understand this, but the teachers in the math department were-- they encouraged us tremendously and if I-- if you map out the time frames exactly Paul Allen and Bill Gates had the exact-- were part of the exact same program at Lakeside in Seattle and so on so for some of us it gave us a little bit of a head start and a peek at what this would all become.

Weber: In that period, what did you want to be when you grew up and how did that change from earlier when you were small?

Ozzie: That's a great question. No. I think when I was small you don't think about these things consciously but I knew what I enjoyed and I enjoyed building. In high school I didn't spend a lot of time on academics. I was in AP courses and I didn't do all that well because I didn't put a lot of effort into it but I did well enough, but I-- but everything I would think about in the off time was either on the-- in my relationship with the person who ran the electrical-- electronics kind of track and on the theater side I spent an immense amount of time doing technical theater lighting and sound and that probably took more of my time and passion over the last couple years of high school. However, that's also where I began doing summer jobs at a place where my mom worked. She was a-- an executive assistant to the CFO at a-- at an insurance company, Protection Mutual, and their data-processing department had some summer jobs open and so I was-- I began to do some programming there also in languages that-- wonderful languages like RPG and <laughs> things like that.

Weber: It's very interesting that you're exposed at the same time to both something like a personal computer that was hands-on as well as being connected online; obviously both those things came in later.

Ozzie: Right. I have to tell you it was a funny era. We were allocated as students a very strict budget of funny money. For people, kids especially now, it's very difficult to understand this concept but online time

costs money and so they-- so-- and every time you would dial in, you would dial the phone, listen for the sound, put the phone in the acoustic coupler and then start typing online it would be deducting credits. I'm not sure how many cents per hour or per minute it was deducting but we had a very strict budget so I would-- a typical after-school or before-school session would be to quickly type in something or run a paper tape through to enter this program I had been working on, get the results and then hang up quickly be-- so you didn't deduct-- you didn't use up all your money for the month whereas the personal computer it was empowering. I mean you could sit there and play with it and play with it and play with it as much as you wanted and that was also instilled very early on and so later in career as we'll talk about it when I was exposed to the potential of doing personal computing you could just feel or-- that you were drawn to the personal computer just because of that empowerment.

Weber: On the time-shared one, you said you had written a golf game and--

Ozzie: <laughs> Wrote a very simple golf game and I found out that there was already one online and theirs was better so I stopped working on that, did some math problems that corresponded to what we were learning in algebra and so on.

Weber: And obviously because of the time constraints you were doing all of that offline--

Ozzie: That's right.

Weber: --quickly. On the Programma, could you sit there and really program the--

Ozzie: Yeah. What you would do—

Weber: <inaudible>

Ozzie: In both of those cases, you would do your-- you'd do-- you would do your work more or less on paper and then on the Programma 101 they had-- you would enter it and then you could save it on a magnetic card and when you began the next day you could just put the magnetic card in and start with that and use the keys to edit what was going on and then save it again. I'm pointing this way because there's a teletype over there and the magnetic cards are back there but on the teletype again you would-- essentially when you began your day you would load up your program that you were working on with paper tape, run it through and it just typed it as it was enter-- as it was uploading it, you would do your work, you would edit with line numbers in Basic and so on your program, get it working and then print it again to tape when you were done and then-- and there-- and your buddy might be waiting for you to get off the thing so he could get on and use it.

Weber: What made you think of the University of Illinois Champaign-Urbana?

Ozzie: My dad went to University of Illinois on the G.I. Bill and that-- we really didn't-- we had to-- from an economic perspective a state school was the only real choice. I applied to several of the state schools but it was kind of a fait accompli, if I got in that was where I was going to go, and I majored in electrical

engineering going in there at the time. I believe their math department had a nascent computer-science curriculum maybe even on the engineering side but it was very-- it was only several years old at the time I started. At Illinois most of what you are taking at the very, very beginning are not really-- are not courses that are pertinent-- deeply pertinent to your major at the beginning but I began-- because I liked building stuff I got a part-time job on the side at the reactor, the Department of Nuclear Engineering, and began building circuits that are-- were needed by grad students. They had a Lockheed SUE mini-computer so I started playing with that because I knew how to code from high school but this was probably the first time I had dealt with assembly language and one of the technicians asked me to help him build something that just kind of floored me. He had just gotten one unit of a new chip, an Intel 8008 chip, and he wanted to build a little computer around it so we had fun doing that. Eventually, we got an 8080 and it was mind blowing. I mean this is-- again this is-- to me this is the next generation of the kind of personal computer that I had dealt with with the Programma 101 but it was just so amazing to learn. Yeah.

Weber: You did computer science but within electrical engineering there were two or three flavors of CS. Right?

Ozzie: Yeah. The way this story went, as a freshman I did kind of what we were just talking about, the basic curriculum, and starting to work and as usual consistent with my personality of what I had done in high school I worked more than-- I worked on what I enjoyed doing more than I-- my academics and so I had a choice at the end of my first year to either drop out or be kicked out so <laughs> I withdrew, got a job in Chicago-- or in the suburbs at a company called Vapor Corporation working initially just in the factory, but I told them I had an interest in electronics and some background so they put me into the test lab of-- their electronic test lab and I learned more and more and more about electronics. They had an early mini-computer they didn't understand how to use so I started showing my boss how to use some of these things and they were-- I was extremely happy, I was tremendously happy, but the people who worked there said, "You are an idiot. If you don't go back to college, you're going to be stuck in this lab in 20, 30 years so leave; just leave." And so the next-- I think the second semester-- I think I was-- I had dropped out for a semester, the second semester of what would have been my sophomore year, I went back with my dad down there begging to get back in and I got back in and at that time the advisor basically looked at everything that I had done and he said, "Why don't you go into computer science? We're investing more and more and more in the electrical-engineering side of computer science" and so as I said there were three flavors. There was a computer science in the math department, a computer engineering in electrical engineering, which was more or less building computers, and computer science in the computer-science department so I went into that third one and it was tremendous; it was great; it was a dream come true. I got to graduate from using a teletype, using a keypunch. We would do our programming on paper and go to punch it in into Hollerith cards and have card decks and submit those card decks to the window with JCL on the front of it, it was-- an IBM 360 [model] 75 I believe it was, and you would wait hours-- you would just wait hours and hours and hours in the lobby of the digital computer lab at the time with everyone else and you'd keep checking and eventually they would put your listing in your box and so on. And at the time I was doing this you would go take a walk or go get a slice of pizza or something. Directly across the street from the digital computer lab there was this other building and it-- on the nameplate of the building it was called Computer-Based Education Resource Lab. I had no idea what it was but if you looked through the windows at night, which is when you do most of your coding, there

was this orange glow coming through the windows and I peeked in the windows and there were just people sitting at rows and rows and rows of these things I'd never seen before and I went inside and these things were interactive terminals. And so whereas in my curriculum I was using punch cards to submit card decks these were-- the system was called PLATO, which is an acronym for programmed logic for automated teaching operations, but this was a computer-based education system and these things were terminals. The terminals had 512 by 512 plasma panel graphic displays and keyboards, and all these people were sitting there. If it was before 10 p.m., they were doing a curriculum, they were students who were using those terminals for their English as a second language course or their chemistry course and so on, and after 10 p.m. a different set of people would come in and it was a mix between people doing programming and people doing gaming, and in any case I just was fascinated by this thing. I lost interest of course in punched cards because I'm like "What are these things? How do I get involved in this?" and I learned-- tried to learn more about it. PLATO was started by a creative eccentric named Don Bitzer and he-- well, actually there were several as in many of these projects. Dan Alpert got-- I believe conceived of the initiative and got the funding. Don was a real hustler and a real motivator but he was-- he took the project from PLATO One in the '60-- probably 1960—

Weber: --'60.

Ozzie: That's right, through the one that I was using, which was PLATO Four in the early '70s.

Weber: Your professors did not introduce you; you made the connection to this group.

Ozzie: Absolutely not. Yeah. There was a wall between the computer-- anybody who was doing serious computer science or computer engineering in the college I don't want to say looked down upon but they certainly shunned what was happening at Computer-Based Education Research Lab. CERL is the acronym that we refer to. It was just a different world. It was more associated with the education department and Don Bitzer was a physicist so it was very much associated with-- he also taught physics so there was an association there, but they weren't doing serious computer science, they were just using the technology as a solution, and Don was such a brilliant character in my mind. It's probably difficult for people to imagine right now but memory at that time was the core constraint. I haven't done the math lately but it would probably be no-- it would probably be accurate to say that it would cost at least a hundred K to put the kind of memory-- the size of memory that you would need to back up a 512 by 512 display-- graphical display into a terminal and these were not computers; these were terminals. They were built out of TTL logic, they were connected to a mainframe with a serial link, and so they had to somehow retain the image that was on the screen and the technology at the time for graphics was storage tubes. You would have an electron beam hit phosphor and the image would stay on the phosphor and you'd have to refresh it periodically but that was the-- if you had a terminal that was kind of the technology that kept the image on the screen, but Don basically said, "No. We need graphics terminals for teaching" and being a physicist he thought about the problem and he and a couple of collaborators invented the plasma panel and used it for that purpose. Each of those orange dots that I was talking about on the screen was a dot on the plasma panel array and it had self-memory, it retained the image on the screen, which was the only way they could make this affordable. So Don, being a hustler, they did the invention, then they did a deal with a-- an Illinois-based company, Owens Corning or maybe it was Owens Illinois at the time,

and they fabricated the panels and made these things and it was incredible and they-- Essentially Don had this attitude that he instilled in everyone who worked for him, which was if you can imagine it you can build it, that- that's it, that-- the-- and if you have a solution that requires technology you just build the technology to accomplish that solution. You do whatever you need to do. You don't have to worry about is it the right thing to build, is it the right architecture, you just build the solution and solve the problem, and he had a belief that computers could ultimately impact the nature of teaching and the nature of learning and that was his passion.

Weber: I am curious because for the CS people even if a lot of it was applications like education seeing essentially a graphical terminal at a much lower cost... I mean were they even aware of that? Was there so little communication? It seems like some of the technical things would have been interesting even independently of their use...

Ozzie: I am certain that the computer-science people looked at it. We had visits from Dave Liddle from PARC and others. I remember people coming out to understand it and there were so many nuances of what to understand because there's a-- an entire software fork and a hardware and systems engineering thread on the software side to do that learning software. There was another amazingly creative gentleman named Paul Tenczar who built a system-- an authoring system and a language called TUTOR; it looked very much like Python whereas it used indentation for structure as opposed to braces or things like that but it had a control flow that was very amenable to writing lessons. All the terminology was units and it would jump to units or call units and come back, and one of the most powerful things was a natural language-based answer-judging system. They did not believe in this style of education, they didn't believe in multiple choice, so you would set up a vocabulary of words and you would map out concepts and words to eliminate and things like that and it would put a little arrow prompt and ask a question after you learned certain material and it would say, "How would you compute the square root of"-- and you would type an answer and it would judge the answer and it would put little x's around for words that shouldn't be there, a little arrow where you're missing words, and just that until recently. Clearly, it wasn't using AI, it was using very straightforward procedural methods, but it was a fascinating thing and yes, people from the computer-science community looked at it but I don't believe it had the level of publishing that would be associated with a-- an academic project. They definitely published but they did not publish an immense volume and Don was a promoter so he would go-- he went on what at the time was a very popular show, "The Phil Donahue Show," and tried to explain to the general public what is-- what are computers; what is computer-based education. He would go and visit different governments and try to promote it and I think the nature of promotion turned off the serious academics and led to a lot of it-- a lot of the reason why it ended up filling a niche that was probably less heard of than other projects.

Weber: Was that comfortable for you being sort of--

Ozzie: I'm a builder; I love building. I didn't envision myself as an academic.

<crew talk>

Weber: No, but I mean the politics if you were in the CS department. I presume you'll tell me you were starting to spend more and more time with PLATO... how did your advisors feel about it?

Ozzie: <laughs> Well, great question. My formal advisor didn't quite know what to make of it. He just looked at the output. He looked at the grades; he looked at the amount of time I was spending on it; he was encouraging me. He said, "Look what happened to you. You're back. You were given another chance; don't blow it" so I allocated my time appropriately to do just barely what I needed to do but clearly B work, just barely B work, but I was spending an immense amount of time. Eventually, I begged and borrowed to try to get a job within PLATO. I went to Paul Tenczar because I'm a software-- well, I was kind of between hardware and software but I envisioned myself more on the software side as a future so I went to Paul Tenczar and I just-- I literally begged. I mean I just said, "I'll do anything that you want. What kind of projects do you need?" and they paid minimum wage, I think it was $1.75 an hour or something like that, but he gave me a project. I asked my advisor was anybody in the computer-science department familiar with PLATO, supportive of it, and there was one gentleman, George Friedman, who had an account and tried to teach his course using PLATO, CS109, and so I spent some time with him. He thought it would be great to have a Logo interpreter built on PLATO so concurrently I did the Logo interpreter project for George and I began working on a project for Paul Tenczar as a job and I just became immersed, just tremendously immersed, in the system and the environment.

Weber: By the way, I forgot to ask you. What was your parents' reaction when you had to leave after the first year.. I presume there was some pressure there--

Ozzie: Huge disappointment. Huge disappointment. My dad was quite cynical in terms of he just thought I'd never go back and he-- having gone there on the G.I. Bill he loved Illinois. We went down there for football games when I was a kid all the time. He put it on a pedestal, very high, and he was just disappointed that I would have let that opportunity go by. Everything was about opportunity, you don't let-- you have opportunities, you take advantage of those opportunities, you make the best of them, and very disappointed and that's why when I did want to get back in he used every bit of himself from a guilt perspective <laughs> in me to try to make sure that I took it seriously and with the dean, trying to persuade him that I was a good kid and that I wouldn't blow it this time, but I wish I could say I—

Weber: It worked, right?

Ozzie: Hm?

Weber: I mean it worked in the sense that in the second year you were keeping up even though you found something even more fascinating to distract you —

Ozzie: It worked— it worked but the history of my schooling including all the rest of college was just riding on the hairy edge, just riding on the edge of what I needed to accomplish so that I could do what I was passionate about. I ended up being on the five-year plan not just because I was out but because I had to drop a couple courses to keep the average up, which postponed the ultimate graduation, but I loved-- I absolutely loved PLATO. I loved the fact that it was the first time I did-- I was part of a team, a

development team working toward a common objective. It was the first time I was-- I understood viscerally the concept of leadership and followership, the fact that you are-- you collectively are believing in this vision and the person who's giving the vision through their inspiration and other techniques is getting you to work like crazy to get-- to accomplish this joint goal, the first time I learned about politics-- organizational politics and backstabbing and all the things that are just part of any organizational culture, but I learned a lot about technology. I was on the system staff of PLATO. I was not developing lessons; I was working on the TUTOR language interpreter and runtime.

Weber: It was higher status within that crew.

Ozzie: Oh, much higher status, yeah. In PLATO you had a two-- instead of just having a username like many computer systems, there was a username and a group, and the group was usually the course you were taking. Like if I was in CS109 as a student, my group would've been CS109, and my name would've been Ozzie. The group P was student system programmers who were part-time, and S was full-time system programmers, and there was an immense status associated with those, they called them sign-ons, but, you know, credentials, and so I got to be a P, and you know, I worked on language features. I worked on one feature that took advantage both of my hardware and software knowledge. They were-- it was the first programmable terminal that I don't know if you can see back there, but there's a-- that tan-colored thing is a PLATO IV terminal, you know, in TTL, but this was the-- we were developing the first Z80-based terminal, and it still had a plasma panel but it used a processor to do the communications and to interpret what was coming down the stream. But also I wrote these commands to do local intelligence, like draw a circle. So that was moving the circle logic from the mainframe, which was sending dot commands down to the terminal, where it could just draw a circle, and I added an animation command where you would load a what today we would call a sprite, and you would say, "I want that sprite to move from here to there with this velocity," and the lesson, you know, the early users of those terminals, started to take advantage of these things. So it was immensely fun.

Weber: Really a sort of client server architecture, in a sense.

Ozzie: Absolutely. Absolutely, and as a part of this I was learning about systems programming, because if you were-- if you wrote-- this was an interpretive language, and the interpreter's running on a Control Data mainframe. Very expensive mainframe, you know, with about a thousand terminals connected to it, 500 on campus, 500 in pockets of four around the world, and if you make an error in your assembly language, it crashes the system and a thousand people are <laughs> very disappointed.

Weber: <laughs> No pressure. <laughs>

Ozzie: Yeah. Well, and it-- and, you know, you make mistakes and you only do work from 10:00 P.M. until 6:00 A.M. That was what was called non-primetime. That was where we all worked, but I had to learn certain programming disciplines so you don't-- you aren't dangerous, and there were very strong figureheads in the development team, full-time people, who would really make it difficult for you if you were, you know, first of all, if you are dangerous, you do it once, everybody does it once. You do it twice or do it a, you know, little bit, they'll try to mentor you. You do it a little bit off and you're off, you're out,

you're fired, that's it, because the damage is just too great. So they were mentoring in programming style through their code. You would look at their code. You would ask one of the other junior programmers who reported to them why they did it this way, and, you know, it was you were just upping your game, upping your game progressively.

Weber: Now, there was a microfilm reader in some of the PLATO terminals, right?

Ozzie: That's right. The top of the terminal is a microfiche. It's a small-- it's a small piece of, you know, acetate that has a matrix of 8-by-8 images, and essentially the back of the plasma panel is ground glass, and while you would be writing text and graphics like arrows on the plasma panel, the lesson could select an image that would be projected onto the back of the ground glass. Furthermore, you can't see it from here, a subset of the terminals that were used in the Foreign Language Department, have a disk reader. It's essentially a floppy disc, a piece of magnetic material that is about that big, and it would have audio tracks on it, and so when you would execute the TUTOR command, put this image up, you could also say, "Play track 7," and it would seek and play it into the students' headphones. It was a multimedia terminal in the, you know, in the early '70s, and it was immensely innovative. But the part that, you know, to try to shift this conversation a bit, the part that fascinated me the most was the fact that there was the beginning of communications and online community. I don't think we talked about it that way but you had-- because there were 500 terminals on campus, on a very large campus, in pockets all over the campus, and 500 terminals in other places around the world, people needed a way-- the developers of the lessons, the authors, would need ways to collaborate with other people and share information with other people. So one of the students, or a couple of the students, David Woolley, and I'll think of it in a moment, but a couple of students, built a system of communications. We'd initially started editing shared text files as the way of communicating with one another. Then there was this thing built called Notes. There was a thing called Personal Notes, which is today we would call email. It was notes that you could leave. It would show the person who is sending it and it was a queue of these things and the body. So you had Personal Notes and Group Notes. In Group Notes, you would create these things called Note Files that were topical in nature. Like one of them might be called Public Notes, and Public Notes would be notes of general interest. Then there were System Notes, where if there were changes in the, you know, this was directed at the developers. You know, if there were changes in the programming language, you would have it there, and people would write a topic and people could enter responses to that topic. So essentially it was shared discussions among the community, and finally there was this thing called-- it's an awkward name. It's called Term Talk. Term Talk was a-- there was a button on the keyboard called Term. Remember, this is an education system. And the purpose of the key was that a student at any time, if they didn't know what a term meant in the lesson, they could hit the Term key and it would come up with a prompt and they would ask the, you know, they would type in the word and it would give a definition of that word. But there were system terms, so if you did Term and typed "talk," it would erase the bottom two lines of the screen, ask you the user on PLATO that you wanted to talk to, and immediately you were in an online real-time chat, and it wasn't the kind of messaging that you and I know today where you send a line and you hit Enter, you know, and the other person sees it. It was character at a time, so you, you know, I would type A, B, C, backspace, backspace, you know, and that's what the person at the other side would see, and finally there was this thing called Talkomatic, which essentially took that concept, but it had the concept of channels where you might-- you go into the Talkomatic lesson and there would be

four channels or a number of channels. You select which one you want to go into and then you are with a group of other people all talking like that. So to me this concept of people communicating, it really, that combined with the discussion tools, really you started to get to know people very, very well at a distance. You'd get to know their personalities. You'd get to know what other people think of them. You know, "Oh, this person's impressive." "This person's a bozo." You just started to get used to being on the system and we started to envision the system as a place as opposed to just being a tool. At the time, there was one thing that happened to me that was particularly noteworthy, and it kind of was career-- ultimately career defining. I was assigned by my boss, Paul Tenczar, to work on a project temporarily with this gentleman named Gary Michael. Very short-term project but he needed some resource. He was remote. He was in the Champaign-Urbana area but he was not at CERL, where I worked. So he sent me through personal notes and he sent me specs of what he wanted to work on. I could see the source code of the partially implemented feature. I worked on it. I would send him questions by personal note. He would send me a reply. Occasionally I would get stuck and I would Term talk him to do it in real-time, and even though his notes, his Personal Notes were extremely eloquent, I mean just-- they were very concise-- when I talked to him online he was just the worst typist you could ever imagine. Like you would see a character typed, wait, a character typed, wait. I mean, it was just so frustrating, immensely frustrating, and eventually I stopped doing it, but I just couldn't picture myself-- how could this-- somebody who's such a bad typist write up these long specs? Anyway, I forgot about it. I got the project done and he hosted a little party after the project to kind of celebrate, and I drove out to his house in Champaign, went in, and there were a bunch of people there, but immediately I was, I mean, to this day I remember this. The reason why he was typing so slowly is he's a quadriplegic, and he was typing with a [mouth] stick, and I had no idea. I just had absolutely no idea. I suppose I should've asked somebody or whatever, but I had just been assigned on this project and it really, even at the time, even as a college student where you don't think about these things, it made me question my biases. Like it made me question, you know, am I treating people equitably? Am I-- you know, because if I had seen him like that I never would've gone to what his mind is like. I would've just been preoccupied with the fact that he's in this wheelchair just barely able to get around, and yet I had seen the other part of him and gotten to know the other part of him and his sense of humor before I ever saw him, and that, that really made me want to work on communication tools after I left school. It just-- to me the nature of the computer being a communication tool was just-- it was just it was something I wanted to do. Yeah.

Ozzie: Look. It was a fascinating-- it was a fascinating system. Everyone was there based on merit. Everyone was there based on interest in what they had to do, and if you were in a group, people left you alone and you were in that group. So for example, there was a thing called the PLATO Corrections Project, and the PLATO Corrections Project was someone had a passion that perhaps you could rehabilitate by putting PLATO terminals in a prison and writing tools to-- repurposing many of the lessons that were used in college for them, and, you know, there was a lot of skepticism at the very beginning but would these expensive PLATO terminals be trashed? No. Of course they weren't trashed. These people treated them very well, but the users, some of the prisoners began writing lessons. They became part of the game playing and discussion community, and so you just had an immense, you know, immensely interesting mix of people online. Again, just extremely appealing. Those of us who were on PLATO at the

time, and this, the era that I was on was from ’74 through ’78, ’79, we got a peek of what the internet would ultimately become. You know, in a different form, in a different way, but the good things, the bad things about online community, you know, the entertainment aspect of it, the productivity aspect of it and so on.

Weber: And how-- I know that there were some connections between PLATO and, for instance, there was, what, the forums on the ARPANET, Jacques Vallée’s-- what was it? Forum, I guess it was called? How aware were you guys, the average PLATO user, were you aware that there was anything like this anywhere? I mean, you mentioned Licklider’s and Taylor’s paper, “The Computer as a Communications Device.” So were you thinking you were in this island or that it was part of a bigger...?

Ozzie: Well, first of all, we knew we were in an island because people told us we were in an island. I mean, the-- it felt like an island, but even the people who came in from PARC and others, you made it very clear there was nothing like this out there and that it was an interesting curiosity. We knew that there was someone out there, I believe the system was called Participate, but there was another conferencing system out there, and we didn’t think about it. I mean, we were immersed in our own community, and I was honestly not thinking of it academically. I was thinking of it because of the joy of building, because of the joy-- the immersion in this whole environment, and only after I left did I have any idea, number one, how much I would miss it, and number two, how many innovations I had been fortunate enough to have experienced. You know, I-- so in the ’78 time frame, the first wave of people who were my vintage, who started school-- who graduated from high school in my year, were leaving because they did four-year-- <laughs> they had degrees in four years, and at that point in time if you were in our field you went east or west. You want to the Bay Area or Boston, because the Bay Area’s the Bay Area and Boston was the heart of the minicomputer era. You had Prime, Digital Equipment Corporation, Data General. It was, you know, the-- we were still coming off the high of 128 being the-- such an immense ecosystem in the defense industry and so on, so it was-- and there’re bunch of amazing schools out here. So everyone who graduated had this decision, “Am I going to be a West Coast person or an East Coast person?” and you’d interview with companies. Well, the people who were my best friends, my buddies who I would hang out with all the time, they both went to DEC, to Digital Equipment Corporation, on the East Coast, and they started sending back listings of what they were working on, like stacks of paper just to more or less entice, and the stuff was amazing. I mean, it was amazing, but I had not graduated yet, so, you know, I was kind of torn. I was trying to think about what’s next and so on. I did-- so I interviewed at DEC. Four groups at DEC. I was-- and on that same trip I interviewed with three groups at Data General. I got my first rejection letter, and that was from DEC, and I was just immensely, you know, disheartened. Just it was so bad. You know, my contemporaries were over there. They were systems programmers on PLATO. I was rejected. This really good, new group who was building a new computer and new operating system at Data General sent me an acceptance, so I took it. Then I got three more acceptances from DEC <laughs> and two more acceptances from Data General, but I stuck with Data General because I just had this thing about DEC because of the <laughs> rejection. So anyway, I moved out to the East Coast and within-- it probably took a month, no more than a month-- I started going through the same withdrawal that my friends had gone through, which is there’s no online community. These are computers being used for data processing and scientific reasons, you know, engineering and scientific reasons, but nobody is using it to communicate with anybody. So I did, and Len Kawell at DEC, did what we all did.

We just wanted to recreate the communication. So in the context of the computer that I was working on, I built a little Notes thing. Len wrote a thing called K Notes, for Kawell notes. A guy by the name of I think Rob Kolstad wrote a UNIX Notes, and it just started, you know, it just started infiltrating the environment. You know, the combination, the-- having the combination of Personal Notes and Group Notes, email and discussions, in some way, shape or form, started, you know, kind of taking off as a pattern, and I would party with those guys every weekend and all we would talk about was, "We're going to do a startup." "We're going to do a startup and we're going to recreate that online community," and we knew we couldn't recreate it in general, because there wasn't general purpose connectivity, but within an enterprise or within an organization, a business, we could all see that ultimately there was going to be a computer at every desk. Of course. Why not? And a terminal at least on every desk, and people would use it to communicate with each other. So we started, you know, like most every weekend, we would just start talking about how that might take shape.

Weber: And why was it immediately a startup and not something to bring to either DEC or Data General? Or why were you so-- what made you so sure that it would not fly with those companies?

Ozzie: I think a couple of reasons. Number one, because you would read stories about the Bay Area, and not so much about Boston, about the Bay area and how people were beginning to do startups, whether it was hardware companies or this or that. I worked-- Data General at the time was considered a rebel startup trying to compete against DEC. DEC was a rebel startup against IBM. So there was this ethos, you know, of "that's what you do," and they were hardware companies. Both of-- both DEC and Data General at the time gave away software to sell hardware. That was just what they did, and we had this, you know, really the thing that we wanted to create was the community more than anything else, and I don't think we really even considered for a minute that those companies would be interested in it, and so Len and Tim basically-- I kept talking about, "Let's do this." I developed a business plan. "Let's do this," and we talked about this for several years. I had this tremendous boss at Data General. His name was Jon Sachs. Is Jon Sachs, <laughs> and our core group developing this new computer and new operating system was three people, of core developers, and there was a, you know, a second tier, but it was really a core of power architects/developers, and it was in new computer architecture. It was based on this concept of a local area network and intelligent workstations and print-- intelligent printing workstations that sat off the network. This was not common at the time. We thought we invented the LAN, which we didn't, of course. But it was great. It was wonderful, but two years into the project my boss, who I loved, left the company to do a startup, and he told me all about this new thing called VisiCalc and how he was going to do a thing like VisiCalc, except a little better, and he had this concept for it. Yeah.

Weber: And you had talked before about Dave Cutler and hanging out with him, I guess.

Ozzie: Yeah. Yeah, yeah.

Weber: And looking up to him.

Ozzie: Yeah, while-- so as I said, I hung out with my friends every weekend. We were on the West side of Boston. That's where I lived. They were on the North side of Boston, so it was a decision to go hang out

together, and every Friday, most every Friday, I would go meet them after work at this place called the Rainbow Spa in Chelmsford, and that was essentially where a bunch of people who are our age-- because hiring was a big deal. It was a lot of people right out of college who had no-- who came from somewhere else, and Dave Cutler. You know, he was-- he would kind of hold court with a pitcher of beer, and, you know, you would drink a lot and in many cases end up over at Dave's house and talk, and I met him, at that time, those listings that I had said were sent by my friends back to PLATO, those were kernel listings of Dave Cutler's-- of the beginnings of what he-- what they were working on, which eventually would be called VAX, and they were working on this operating system called VAX/VMS. Dave had just done a product called RSX-11M, and he had just started working on that. There were many people involved. Gordon Bell, you know, others on this project. Tremendously exciting project, so I, you know, we would compare notes. I would show them this LAN-based operating system and hardware I was working on. They were doing this minicomputer. Even though the companies were-- had very strict NDA's and it's East Coast, so they took it seriously, the people on the ground didn't care about that stuff and they-- and even though we were competitors, they, you know, I saw all the stuff, and I learned my early assembly language programming style by reading Dave's listings. It was another level of sophistication beyond what I had seen working as a systems programmer on PLATO. Jon Sachs, my boss, had a very-- he was also an amazing developer and was very into tight code. I would say Dave's strength was not just tight code but architecture and organization of large projects of assembly language, and so when you're doing anything of complexity, it's so wonderful to have that kind of mentorship. So eventually-- eventually Jon Sachs, who had left, made me pay more attention to what was going on out there, and the microcomputer era was kind of beginning to happen. You know, the Apple II had-- or I'm not sure timing-wise Apple I or II. But we were looking at processor architectures. The 68000 came out. This thing called a 16032 came out, which was an amazing little computer. It's a shame that it never took off, but it was almost like a VAX on a chip. It was just wonderful, and I started to get the itch to want to go be part of the microcomputer revolution. I was working on, you know, a MicroNOVA on my, you know, in that Data General project, but I wanted to be part of that industry, and so a recruiter had called me and told me about a job in Cambridge for a company called Software Arts, and I interviewed there and took a job. They-- and that's Dan Bricklin and Bob Frankston, who did VisiCalc, and I was employee Number 29 of that company, and they had assigned me, you know, a first job of doing the port of VisiCalc to the TRS-80, the "trash" 80. It was-- it had started out on Apple and-- this was on the TRS-80, and so I began that project, and it was amazing. Software Arts was incredible. It was the-- it was the first time I worked for a small company, and I loved it. You know, to, in some ways arguably, you could say, in some ways to their detriment, they ran it like a family. You know, everyone just loved-- they were decent human beings. Loved working with each other. VisiCalc was the most successful software product in the industry at the time. It, Software Arts, was an engineering-only organization. At that time in the early industry it was very common to have publisher, author-publisher relationships, so the-- there was a company on the West Coast called Personal Software that did the publishing of it and Software Arts was pure engineering, product management and engineering, and essentially was compensated with a royalty on every copy that got sold. So this was just a wonderful job.

Weber: One question. Why did you not go with Jon Sachs, or maybe he didn't ask, but I mean, he was going off to co-found Lotus.

Ozzie: Well, first of all--

Weber: But why?

Ozzie: Yeah. No, it's a great question. First of all, the first place he went was actually not Lotus. He started a company with someone else to do--

Weber: All right.

Ozzie: --to do a spreadsheet for minicomputers. He was going to take what was going on in microcomputers and go up the stack a little. However, Mitch-- he then met Mitch, and the reason that Jon, I believe-- he didn't tell me this-- but I believe the reason he didn't do this was the core of the product that he created at Data General was three people, and he left, and from an ethics perspective I just don't believe he wanted to cause that project to fail. The moment I went to <laughs> Software Arts, he called, and he said, "I'm doing this startup with Mitch Kapor. It's called Lotus." They were in the basement of a Franklin Street house at the time. I think they were eight people, and I said, "Look. I just started at this company working on spreadsheets." He says, "Well, just stay in touch." So we stayed in touch every few weeks. We only worked a block from each other. I was in Cambridge; he was in Cambridge. But I loved this company and these people so much, I just wanted to do a startup, and I was still hav-- I was still meeting with my friends, and I said, "Now is the time. It's going to be too late. We've got to do this product," and so Tim and Len said, "If you can do a startup, if you can get funding, we'll quit our jobs at DEC and we'll go do that Notes thing for microcomputers," which was the core concept.

Weber: And then help me with the timing. You had talked about meeting VCs around a different startup idea, something like a mini...

Ozzie: When I was still at Data General**,** Tim and Len and two other people over there started thinking about doing a startup around the National 16032, which was a little VAX, and the concept was do a VMS-like operating system for this microcomputer. Each one of us falls into our own role. I ended up writing the business plan for that. That company was called Microcosm. You know, we would meet at somebody's house, keep developing plans. But eventually that one fell apart, not because of lack of funding. Just because we couldn't get-- we just couldn't get momentum among what would've been the founding team. Glad we didn't. Glad we didn't do that, but that, that taught-- that got me into understanding what the-- that there were these things called VCs. That you write a business plan and that you bring it to VCs and so on, and so when I was at Software Arts, I began writing the business plan for what would eventually, you know, what we wanted to build that would eventually become Lotus Notes. I took a course on venture capital. I managed to at that course get a couple of contacts. I went and pitched-- or sorry. I sent business plans to Greylock, TA Associates, a few of the local Charles River maybe. At the time most of them did not ever contact me back. The guy from TA did call me back and I said, "Can you just explain to me what's going on?" He goes, "Well, do you have a pencil? Because there's a lot of reasons. Number one, you're obviously green. You've never been in a leadership position before from a business perspective," and this and that. But really, we were a little intrigued, except that nobody understood what you were talking about. This concept of connecting people together, the concept of assuming that there's a PC on

every desktop, that's insanity. That's absolute insanity. That we have, you know, a seven-year window in our funds and, you know, that's going to be 30 years from now, so don't, you know, it's just-- it's too out there, and I really appreciated the frank feedback, but, you know, it was disheartening, and so I told Jon Sachs this at lunch at one point, and he says, "Come over and work with Mitch and me. Come over." You know, "We're working on this--" they ere about to ship 1-2-3, Lotus 1-2-3, and again, it was only-- I think he was the only core developer. They had several other developers working on printer drivers and things. He goes, "Come join me. I'm burnt out. Just come join me." I didn't want to, but I went over and talked to-- he introduced me to Mitch Kapor. Mitch was great. He gave me his pitch as a visionary, and it was tremendous. But I said, "I just don't want to work on spreadsheets." Like, "Dan and Bob are great visionaries also and they-- and I'm working on the world's leading spreadsheet." He goes, "Look, just-- why wouldn't you do it?" and, "Tell me what you're excited about?" so I told him the-- I gave him the whole pitch about this Notes thing, and he said, "Look. I don't really get it. I really don't. I don't get it, but I will commit to you one thing. Right now, I will commit to you one thing. You come to join us. You-- Jon's burnt out. You come and deliver 1-2-3 Version 2, and when that thing ships, I will figure out how to get your thing funded. We'll figure out how to get you funded with what you want to work on," and, "Is that okay? How does that sound?" and that was the best deal I had going. You know, that was-- I couldn't figure out how to get it funded, so I said, "Great. Terrific." So, you know, Dan and Bob knew I was working on a business plan at the time. This is a slight digression, but there was a guy who worked at Software Arts named Tracy Licklider. His dad was J.C.R. Licklider, who had written this seminal computing as-- computers as a communication device. I told him about it. He got it, you know, but again, it was just one person's opinion. Not J.C.R. Tracy. But--

Weber: Did he talk about taking it to his dad?

Ozzie: No, no, no, no, no. This was all just socially over lunch kind of thing, you know, and Dan and Bob were very, you know, they're-- as I said, it was family. They knew Mitch very well. They knew Jon. They said, "If that's what you want to do, that's what you should go do," and so, you know, as I said, they were just great, great people, and so I went to work for Lotus.

Weber: And they were not interested in backing you or doing anything related to...?

Ozzie: No, no, no. I told them about it, but they were-- they were quite immersed. One of the most interesting things that I neglected to talk about my little stint at Software Arts was the fact that so many opportunities cropped up in such a small amount of time. I think I was only there nine months from beginning to end, but even right at the beginning there were some bugs I was trying to fix and they were bugs not in our product but in the underlying operating system, which was MS-DOS. So I flew out to Seattle and I met Bill Gates and Paul Allen. Microsoft was about 30 people at the time, and that was-- I think that was '82, and that was-- it was tremendous. We got to know each other. They turned me on to the developer who, you know, introduced me to the developer who was doing this stuff and they fixed the bug. Several months later I was working in my office and I started noticing Bob Frankston going into this big closet right across from my cubicle, and he kept going in there and going in there. Then construction people showed up, and they were working in there, and always the door closed. You know, it was just strange, and then finally these suits came in and they brought boxes in to the closet, and, you know, I

was <laughs> like, “What is going on?” and Bob’s like, “Can’t talk about it. Can’t talk about it,” and one night, it was probably like 10:00 P.M., you know, he came out. It was just us there, and I said, “Bob, you’ve got to tell me. What’s going on?” and so he says, “Okay. Okay. Come in here,” and I went in the closet and there was a big stack of listings, a plywood board-- a PC board attached to a piece of plywood and a monitor. I said, “What is this thing?” He says, “This is going to be the IBM PC,” and they were one of the early anointed few who would get access to the hardware so that the VisiCalc could be on it at launch. So it was tremendous but they were so immersed in the potential future of both VisiCalc and a new product that they had called TK Solver, which is an equation solver, kind of like the next generation of what you would do with numbers, that this concept of doing things with words was not really core, so...

Weber: And so...?

Ozzie: Yeah. So I went to work at Lotus. I worked on site there in Cambridge for two weeks and I asked Mitch and Jon, “Do you want me to get the job done or do you want me to work here in the office in Cambridge?” because there were just so many distractions. It’s a long commute and there were distractions, so--

Weber: Where were you living at the time?

Ozzie: Harvard, Mass., which is about an hour northwest. My wife was an engineer at DEC. On that same team. I met her <laughs> at those parties at the Rainbow Spa. She, I think at the time, was working at Wang Labs, but that was north and I was, you know, when I met her I was on the West side, so Harvard was a great place to live. But it was still like an hour commute into Cambridge, and so Mitch said, “Just do whatever you’re going to do to get it done.” So I opened an office in a little town called Littleton, Mass., and Jon joined us for the first several months, and there was a small team, a team of three people, myself, Barry Spencer and Matt Stern, and the three of us churned out a quarter of a million lines of assembly language in nine months. It actually shipped nine months to the day after I started, and that product by that time had diverged from 1-2-3 enough that we called it Lotus Symphony. It had lost macro-compatibility with 1-2-3. But it was one of the first what you would call suite products. It brought together word processing, database and spreadsheets and graphics kind of all in one, and the day that it shipped-- you know, was kind of heroic among all of us to get the thing out there. The day that it shipped, literally the day that it shipped, Mitch came down. I was in the basement of 161 First Street in Cambridge, Lotus’s headquarters at the time. Mitch came down. He shook a-- put a-- extended his hand, and he said, “You did your job. You did exactly what I had asked you to do, and so put it away, dust off your business plan, and let’s talk about what you want to do and how to get it funded.” There was nothing written down. There was no deal set in stone. There were-- there was an amazing number of bugs in the queue to be fixed, but he, he lived up to his word without anyone reminding him, and he gave me that opportunity, and it was just tremendous. Yeah.

Weber: Before we go on to Notes, Symphony – I mean, the feature set, the idea of doing a suite, who was driving that process? Who was-- how did it go from being 1-2-3 second version to being something much more?

Ozzie: Sure. Well, that company, as many companies since then and until today, was shaped around a hacker/hustler pair. You know, the-- Software Arts had Dan Bricklin, who was the product manager and the product designer, and Bob Franktson, who was the hacker extraordinaire. He, you know, he wrote the code, and of course it's always a negotiation about what can be built or whatever, but the vision, the product vision, that-- really came out of Dan. At Lotus, Mitch was the de facto product manager. He knew what he wanted. He had worked on spreadsheets before. He was at Personal Software at-- earlier and he had written a little add-in for VisiCalc called Tiny Troll, and that gave him the vision for doing 1-2-3, which was, you know, three products in one. It was kind of a mini suite. So he had this vision of where he wanted to take it. At the time, it was not just his brilliance. Everyone kind of knew that was the next direction, and so a company called Ashton-Tate was building something called Framework, a great individual called Robert, named Robert Carr, was at the core of that, and it was so well architected. It was, you know, infinitely better architecture than the stuff that we were working on. It had a core data model that was very elegant, whereas 1-2-3 and Symphony worked with the spreadsheet as the core data model of all of the modules. There was a company that was heavily VC funded called Ovation at the time, that was threatening to build something that would take over the momentum of 1-2-3, so there was a lot of pressure to get it out to market on time, and that was why we kind of raced, and so we took very strict roles. You know, they laid out the features. I implemented them the best that I could. It's not that I'm shirking responsibility of all problems that were, <laughs> you know, in the product and-- but that was kind of the allocation of responsibility.

Weber: Great.

Ozzie: Yeah.

Weber: Go on, but, yeah, I would've loved to know, for instance, what was the atmosphere like at Microsoft? What was it like at Software Arts, you know.

Ozzie: <laughs> I mean--

Weber: But--

Ozzie: --let me just give you the major things, because maybe you can interpret it from this. So we're just entering Notes. Notes, it just depends on what your questions are and how many nuances, because that was the longest single segment, that was like 13 years of life, and I, you know, I'm not sure how you want to do it. Then, just in quick succession, there's Groove, which is a little bit interesting. It's not hugely interesting, but there's a-- it's a little tiny segment because it's still collaboration and it's a different style of collaboration. Then it-- then we head into Microsoft, and I don't know how much-- I have zero idea how much you want to talk about Microsoft at all.

Weber: Oh. Definitely do.

Ozzie: And then, you know, and then post-Microsoft there's a startup talked Talko-- well, post-Microsoft there's a personal interest thing, which will be connected to what I'm working on now, which is the Fukushima, the Safecast thing, which I don't know if we've ever talked about.

Weber: No, I'm sure I know... Yeah.

Ozzie: Yeah, there's-- yeah. Anyway, then there's a startup called Talko, which is, again, collaboration, and then there's the startup, and then there's the startup that I've got right now, which is Blues.

Weber: Internet of things--

Ozzie: It was after the Lotus Notes experience that I finally understood enterprise and leadership and the difference between entrepreneurialism and working for a big company, because until that time I had never really worked for an enterprise. This is still pre-Microsoft, but the most significant thing probably, of the Notes thing, for me personally, was understanding the role of collaboration within organizations and the difference between small team and large process stuff.

Weber: You know, you tell me, but Microsoft also seemed-- you were obviously bringing your experience with collaboration to that. So, I mean, it all seems of a piece, but you'll tell me about it.

Ozzie: The Microsoft part, if you're viewing it from the perspective of-- at the macro-level perspective, as opposed to the specific product, my job there was, and I'll say this again, my job was a transformation job, an organizational transformation job. You know, the, you know, even before our company was acquired, Bill and Steve told me the goal is, "We are a PC company. You have to take us beyond what the PC is," and, you know, "You've done this as an entrepreneur, and here's where we are as a company. Go do it." It's at the change management level. At the product level, you know, yeah. I built Azure. I seeded the, you know, Office 365 and Xbox Live and all that, but that's not the interesting part. I mean, the interesting part to me is what they're writing right now about Satya, that was the goal. People were-- at the time, Bill and Steve knew that the company was not going to thrive if it was a PC-focused company, and getting it on the right track at that moment in time was kind of my job, so anyway. But again, you--

Weber: We'll get to that.

Ozzie: I don't know how much you want to get into that stuff, becau-- if the whole thing is about collaboration and entrepreneurship. The entrepreneurship thing is mostly we haven't covered, which is having worked in big companies and small companies, why do you still go to seek entrepreneurship? Why do you go-- <laughs> still seek to build small companies? So...

Weber: Okay. So talking about the beginning of Notes.

Ozzie: Yeah. So kind of in a nutshell, Mitch came down and said, "Dust off your business plans." I wrote up, I rewrote, what I had written before. I borrowed a Mac, a pre-release Mac-- this was 1984-- from a-- one of the groups down the hall who was working on a to-be-announced Mac product and I turned them

from character mode to graphical mode and came, you know, did imagery of what I thought the product would-- should be like. That was kind of mid-1984. We finally reached agreement on an organizational structure by December of 1984. It was a-- I'm not going to talk extensively about this, but it is notable in one sense. Mitch was an entrepreneur. He started Lotus. The people within Lotus had an inclination of wanting to allow me to do my product as a-- kind of subsidiary of Lotus at the time. I wanted a separate company, and Mitch straddled it. He was in the middle, and he ultimately enabled it to happen the way that I wanted, but it was a fascinating relationship, because my startup, Iris Associates, had no equity ownership by Lotus. They funded us with an initial check of 1.2 million, but it was an option to purchase rights to the product at a future time if they would be willing to commit to marketing it with the full power of their marketing organization. At the time we did the startup, essentially there was a huge amount of trust involved. They wrote a check and they had optionality on something that didn't exist, and at the time, if they failed to take advantage of that option, they would've lost rights to it and we could just do whatever we wanted with it. Was a fascinating deal structure, but in any case, December 7th, '84, I called my-- Tim Halvorsen and Len Kawell, very good friends of mine who, you know, from college and to today, and I said, "I got the check; let's go," and they quit their jobs and they joined me in January, and the three of us set out to building what eventually would become Lotus Notes. At the time, all of us, but Len in particular, believed passionately that the PC was going to switch from character mode, which it was at the time, to a graphical user interface, and there-- on the PC platform the Mac had just been announced, or launched. I guess that was November 20th of '84. The Mac was out there. People finally understood the Star and what PARC had been working on, but the PC industry, it wasn't-- the pundits in the PC side of the world did not-- it was not a foregone conclusion that graphical user interface was going to dominate, because it was so slow and resource intensive, and so on. But I thought so, and Len passionately believed in it, so we did a little tour, trying to figure out what graphical user interface we should bet on, because there were three primaries at the time. One was called Visi On, and it was-- VisiCorp, the company that published VisiCalc, had renamed itself from Personal Software to VisiCorp, and they had a graphical user environment, and it was quite mature, and it was reasonably performing given the resources that were there. I stopped in at, what is it, Pacific Grove, California, to visit Gary Kildall, whose company Digital Research, which was well-known for CPM at time, they had one called GEM, and then I went up to Seattle to visit--

Weber: That was GEM or-- but GEM would've committed you to CP/M.

Ozzie: That's right. That's right. It wasn't clear what Visi On would be. Visi On was going to primarily be MS-DOS at the time, but-- or PC-DOS. Yes. GEM was on CP/M 86, but I think that they had an MS-DOS version planned or whatever, PC-DOS, and I went up to Seattle and Bill introduced me to his team, who was working on Windows at the time, and it was really the poorest choice. <laughs> It was technologically not as-- clearly not as well thought out. It was rushed. There were five people working on it, and they were doing the best they could, and the best thing you could say about it is that Bill was absolutely committed, absolutely, unquestionably committed that this was the future and this is where it was going to go. Not analyzing the psychol-- without analyzing the psychology of why he was so committed or whatever, it just-- it showed, and I knew--

Weber: Why? A step back. Why were you so committed, and why was he? What did this mean to...?

Ozzie: I was committed because after using the Mac, which I did during the development-- Steve Jobs-- just step back a little. I met Steve back in the Software Arts days, and we had done work when I was at Software Arts on VisiCalc for Lisa, and at one point Steve had called us to come out there because he was working on this new secret project, flying a pirate flag over the <laughs> building and, you know, Mike Boich, Joanna Hoffman, Andy Hertzfeld, Bill Atkinson, they showed us the Mac that, you know, what was going to become the Mac, and was transformational, and he, you know, and Steve, of course, was very passionate about that, and Steve wanted-- in an era where software was either $395 or $495 for one seat, Steve was passionate that it was going to be $99. Ninety-nine dollars. No piece of software will be more than $99, and so we never did anything on that platform when I was there, but because I was at Software Arts and because I was at Lotus, where we were working on Jazz as a spreadsheet for the Mac, I was around this platform all the time, and, you know, once you use it there can be no skepticism. The Lisa was pokey. It was very pokey. It was laggy. The Mac was honestly snappier than today's PCs. I don't care what you put in, you know, put in it, they-- Andy Hertzfeld had done so much work to do everything in one screen retrace, so everything was just instantaneous in terms of how you manipulated everything, and so I loved it. Len had done a, at DEC, a workstation product with Dave Cutler. He had moved with Dave Cutler to Seattle and done a thing called VXLAN, which was a workstation product, and he was passionate about it because he worked on workstations. Tim Halvorsen had worked on the VAX workstation, a different workstation, a nascent workstation, in New Hampshire. So the bottom line is we all believed in it. The real question was, "Is the timing right? If we're developing this new product, is the timing right?" and so it was a big bet, but I think Bill, in his heart of hearts, probably was passionate about it from the competitive angle. You know, he was paranoid, as you should be as a businessperson. He's very smart, and he could see that the biggest threat to MS-DOS was a graphical user environment that reduced the MS-DOS to what Andreessen I think called "a poorly debugged set of device drivers." You know, he had a healthy paranoia about that. So anyway, I essentially told Bill, "But I can't-- it's the least mature, and I have to worry about my app. I can't be worried about your app being stable," and he said, "Look. Let's handshake on two things right now. Thing number one is we will develop a runtime version of Windows that you can include with your app so your app can be graphically independent of whether the OS becomes graphical. So we'll-- we will develop that for you," you know, and we'd been talking about it anyway, because one of our other early developers would love that also. The second thing is that we'll give you the source code and--" under NDA, "and ship you new floppies every week, you know, every couple weeks, and you can debug using the source code," because there's no doc and it's not done, and to me that was it. That was the defining, you know, the defining thing, and so we made the choice to work on Windows specifically because of source code. We never could've done that had we not been an independent company, because they could not allow the source code to get to Lotus, and so the fact that I had known Bill before and that-- and he didn't know what we were working on. I couldn't tell him about Lotus Notes, but he knew me. He was very flattering about his belief in whatever I was working on, and so we just decided to do that, and that began a very fruitful collaboration by which we helped them make a better product and a viable product as an early developer, and it was. <laughs> So...

Weber: And it sounded from the pre-interview like it did create some tension with Lotus, the fact that you were kind of in the middle, right?

Ozzie: Well, it didn't create tension in 1985, which was the first full year that we worked on the product. The first full year we delivered the first floppies with the product on it as early as, remember, we started working on it in earnest in January of '85. I believe it was May that we had much of the product completed, because it was three people who were just-- <laughs> we were working 24/7, and that had-- and that was based on an early Runtime version of Windows that we were working on, and they, the people who saw it in there were fascinated by it. But as the year dragged on, by the end of '85 and early '86, it was very clear that Microsoft was going to use Windows as a strategic weapon with what would become Excel and Word against 1-2-3, because they're competitors, and so Jim was approached by-- Jim Manzi, sorry, the CEO of Lotus, was approached by IBM saying, "Well, we're developing a next-generation graphical user interface environment also called OS/2, and so suddenly OS/2 became strategic to Lotus, whereas Windows was what Lotus actually contractually committed to allowing us to work on, and that's why we worked on it with <laughs> Microsoft.

Weber: And had you done-- because I think you had said in the pre-interview, that Mitch had initially wanted to give you guys a stake, just like 10 percent or something, and have it be really a subsidiary. You couldn't have done this...

Ozzie: That's right. That's right. Yeah.

Weber: Right. Yeah.

Ozzie: We had-- specifically because we didn't do it as an internal spinout or a internal business unit, we had complete control, and so much control that the nature of the contract essentially had a section in it that defined what the product was at the time, the concept was at the time. But the way it was drafted legally, whatever it was we could declare that it was ready for them to make a decision, and the option is going to be up. We could unilaterally state, "Okay. This is the product. Do you want to exercise the option or turn away the option?" If they exercised the option, they had to sell what we built and they had to use best efforts, meaning not just business reasonable efforts. They had to use best efforts, meaning as much as they used on 1-2-3, to make the thing a success, and they couldn't kill it. It was written in it that they had to bring it to market. They could not kill it. If they wanted to kill it, if they were hesitant, if they didn't want to use full efforts, then they-- then the way it was written they should not exercise the option, at which point they owed us no more money. We didn't have to return anything and we owned the IP, but they bought that option. So it was a very tense, increasingly tense relationship, as they saw, number one, that what we were developing was against the platform strategy of the company and it was in alignment with their competitors' platform strategy, and number two, they didn't know how to sell it. It was very clear-- it became clearer and clearer that this was a product that needed to be sold to an organization as a solution for a collaboration problem that they had, and Lotus's entire go-to-market infrastructure was shrink wrap products sold to computer stores. They had no such thing as a direct salesforce or they had a channel for shrink wrap product. So by the time we declared that it was ready several years down the road-- we spent probably--

Weber: Yeah, we'll get to it.

Ozzie: Yeah, we probably spent three years fighting against the memory management challenges of operating within the Windows environment. For the technically minded people who are watching this, <laughs> that was a 250K byte working set for both the code and the data, so it was just a struggle to take megabytes of code and the operating system itself and make sure that they ran acceptably in that kind of a working set on the IBM PC. But anyway, by the time that we declared that it was ready, Lotus was in a quandary, a real quandary as to what to do, and it almost led to a lawsuit and so on, and the settlement was that they would bring the Windows version to market if we took a year and also did an OS/2 version, and that we would ship both the OS/2 and Windows versions concurrently and equally, and that's what happened.

Weber: Before we get definitively to that...

Ozzie: <laughs>

Weber: So tell me little bit about Tim and Len, how you came up with the features, how different they were from PLATO Notes, and also, you know, you brought up the shrink wrap, versus kind of selling to an organization. One thing I wondered is this is already the era of, you know, CompuServe's information service, sort of consumer online services. So I'm wondering, you know, you must've thought of that, but I guess there must've been reasons you didn't go in that direction.

Ozzie: Well, that was-- okay. So during the rewrite of the business plan of what would become Lotus Notes and the project plan and so on, it definitely brought up Prodigy and CompuServe, maybe MCI Mail, I'm not sure, as early indicators that email was going to become more of the academic niche, more than the academic niche thing that Multix Mail, UNIX Mail, UUNET, you know, UUCP-based mail was, and that was a core part of the, you know, of laying it out. However, I very clearly laid it out, mainly because of my-- of what I did at Data General. At Data General I worked on a LAN-based workstation architecture, where there were file servers, print servers and intelligent workstations. I believed passionately that that was going to be the architecture in every major organization worldwide. You know, Ethernet. You know, IBM was behind Token Ring. There was this other thing called Sytek which was a token-passing bus, and Ethernet was there, and so all of the signs showed that that was going to be the architecture of deployment within enterprises, and so the fundamental product definition was email and discussions and address book and certain things like that. When I wrote up the plans, you could conceptualize it as a suite product for communications. It had discussions, mail, contacts, calendars and so on, but it looked like a suite product. Once we started working on it, the different roles... Tim was a very deep database and systems type implementer. He's very good architect. You know, as I said, he worked directly for Dave Cutler and they-- he learned a lot from that and that was his core strength. Len was the one who was operating at the graphical user environment level. He was the closest to the user. I had a role that was-- that spanned both of those, meaning some infrastructure and some UI, but I was the one who was trying to envision with the customer what they were also trying to do. So I implemented this thing that today we call a NoSQL database. It was a-- I did a bunch of dabbling around in what databases should be done, but I implemented that layer. Tim built an indexing layer on it that would-- that could be used to drive the UI, and so on. We operated as an amazing, you know, collaborative set of folks. It was very clear maybe six months on that in order to make it operate in a distributed manner we needed to take the concept of

synchronization and replication much more seriously, so we hired another guy, Steve Beckhardt, who dealt with the replication of the NoSQL database that I had done. Security had to be part of it because it needed to be distributed security. So we brought in this guy named Al Eldridge, and we can talk about <laughs> security, but that's a very extensive thing. We were the first implementers of public key cryptography in a mass market product, and so he brought, you know, he brought that knowledge in, but anyway, it was an amazing core team of five that built the majority of it, you know, during that period of time. Yeah. I'm sorry, I lost-- <laughs> I lost--

Weber: No, no. That's perfect talking about the roles of the people. But I'm still a little-- so, I mean, all this makes sense for an organization or company, but why wouldn't it also make sense working with someone like CompuServe, you know, something with a more--

Ozzie: So Comp--

Weber: Connected to a network and you have a consumer product as well? Or more like later MSN or something?

Ozzie: First of all, there was no concept of a consumer product for this product. The product at its core was about people working together in an organization to get things done together. That was the core mission. It wasn't about consumer email or things like that. People were using dial-up at the time. This would require a faster network. There was no such thing as... a fast network. You know, at that time, this was all pre-internet, you know, and so, you know, that was at the core. The biggest single design decision I, you know, I will-- again, on-- you don't forget these things. The riskiest, biggest design decision was I had envisioned it as more or less initially a suite of stand-alone communication tools that were woven together into the same UI. As we developed the database more, the possibility became the-- it became a possibility that you could actually develop it as a generalized tool and build those applications as modules on the same underlying database, and if you did that they would all replicate the same way. They would all, you know, it was a very coherent architecture, and so there was a very big transition where we decided we were going to make it an application development platform as much as having those core applications, and that was-- it turned out that that was determinative of the product's success, you know, over time, because-- and we can talk about this more. This was post-release, but really the strength of Notes ended up being around the business processes and practices that it automated, not merely around communications, so...

Weber: That's the flexibility that...

Ozzie: Yeah.

Weber: Gave you the flexibility to do that.

Ozzie: Mm-hm. I mean, at the time-- just to put a little end cap on the consumer thing, we were incredibly fascinated and had good ties with-- there's a guy named Tom Jennings who had a-- who built a system called Fido, and it was a-- it was, you know, the common man's bulletin board system that was trying to

compete with "the Man" who was CompuServe, Prodigy and so on, and it was amazing. It was wonderful, and we took many of the concepts as validating the fact that this core notion of messaging and bulletin boards was a, you know, an essential element to almost anything that we did. We saw it on PLATO, but again, now it was happening in another environment. Yeah.

Weber: But it's true. If you wanted the speed of a PC with a network card in it, yeah.

Ozzie: Yeah, yeah.

Weber: There was only one way to go at the time.

Ozzie: That's right. That's right.

Weber: And then talk just-- so Tim and Len are such important figures here. Say a little bit about them as people and kind of your relationship--

Ozzie: So Tim and Len, who I've talked about a few minutes ago, Tim and Len were more than collaborators or important figures. Tim Halvorsen, Len Kawell and I, were essentially peers and equals in the creation of this product and this opportunity. Just like in a marriage, you're brought together for a common purpose. You choose to be together. You're different and you do different things, but you are brought together by common beliefs and whatever. Tim and Len and I are different people who have different skills, who had a common belief that was brought together because of the vision that we saw in PLATO, and Len is a more right-brained thinker who thinks about people and experience and interaction design. Tim is a more left-brained engineer. He's an engineer's engineer who is deep into architecture, and I am more of a solution-focused person. You know, we're all good engineers but I brought the fundraising to it. I was thinking about, "How are we going to get this in the hands of actual users and people?" I managed the relationship with Lotus, but each one of us probably wrote, of the initial three million lines of code that was Lotus Notes, we each probably wrote a million-ish lines of code-- this is, you know, C, by this point in time-- ourselves, and the areas of the product that over time I gravitated toward were, you know, more, "How do you build an app?" The things that Len did was usability related. You know, "Is this thing going to be usable by human beings?" And Len-- and Tim was more, "How can we make it perform and operate within the operating environment that is there?" But there would be no Lotus Notes without Tim Halvorsen, Len Kawell and I, and also Steven Beckhardt, who did the replication, and Alan Eldridge, who was on the security side. Those were all ultimately core elements to our success.

Weber: And did you guys hang out together as well in those days?

Ozzie: Absolutely. Yeah. Absolutely. Well, I mean, we hung out in college when we were associated with the PLATO project, and we hung out, as I said, at-- when they were at DEC and I was at Data General. By the time we founded the company, each one of us had become married. I had one child at the time, our first, and it-- life was a little bit more complicated in terms of when we hang out together. It was more the families getting together and things like that. Tim's girlfriend at the time had introduced me to Dawna Bousquet, who became my girlfriend and my wife, and we've been married for, what is it, 37 years now?

Weber: Ever since.

Ozzie: <laughs> Mm-hm. Yep.

Weber: And so you really ran the company kind of, I mean...

Ozzie: It was--

Weber: In a collaborative way, at any rate.

Ozzie: We ran the company collaboratively and fell into our patterns. You know, Steven Beckhardt, he enjoyed the finance aspects of-- finance and operations aspects. I, as I said, dealt with the product management aspects of dealing with Lotus, and ultimately the go-to-market aspects, all the legal aspects of our relationship. You know, Len, as I said, from a human interface perspective, he dealt with all the test and usability aspects of it and so on.

Weber: And you had talked about, pre-interview, Ben Rosen and funding. Could you talk a little bit about...?

Ozzie: Well, Ben Rosen-- Ben Rosen was one of the key-- he was very prominent venture capitalist at the time, and he was the key funding source for Lotus, and Lotus, I'm not sure how much you are aware of the early days of Lotus, but the initial funding plan, I believe, for Lotus, was that it was going to sell 3 to 5 million dollars' worth of product in its first year and it sold 53 million in its first year. It was-- it took off like a rocket, and by the time Mitch and I were-- began talking about me doing this startup, Lotus was flush with cash. I mean, they were investing but they were also significantly profitable. So I didn't have to go to a VC. Lotus was funding it out of its operational earnings.

Weber: But I thought you had-- so maybe I misunderstood. I thought you had talked about Ben Rosen having some role... around Iris.

Ozzie: No.

Weber: Okay.

Ozzie: No. The only funding for Iris came from funding related to the option agreement that we had in place.

Weber: Okay.

Ozzie: And that was before the option. Then at a certain point they exercised the option, and as a part of the operational agreement between us, post-option exercise, they paid all of our costs. There were no profits until there were ultimately royalties. We got paid royalties on gross of the actual Lotus Notes product, but our operational expenses were borne by Lotus, and that was done so that we would organize

activities between Iris and Lotus as they would make sense for the best product and the best experience and we would make those decisions in a neutral, neutral manner.

Weber: But I mean, Iris, I mean, it was a separate corporation.

Ozzie: That's right.

Weber: Effectively such a unique status. I mean, you had a board of your own?

Ozzie: There was no board. It was a-- at the time I believe it was a sub S corporation, and Len, Tim and I-- I believe the top five people were probably the only shareholders, but the actual shareholders didn't-- wasn't really a material thing because even when royalties came in, we shared the royalties in more of a partnership type of style where it didn't really-- we passed around earnings in a way that matched contribution within the company as opposed to equity. And the-- it was an amazing partnership. Iris lasted for 10 years. Ultimately, the royalties that Lotus paid us became too burdensome because they had to invest in the business, and so essentially the only real way of solving it was for Lotus to buy us, eliminate the royalty stream and give us equity to-- and we became part of Lotus. Little did we know at the time though that IBM, who was going through a restructuring of their own, internally under their new CEO, Lou Gerstner, they had been looking at Lotus becau-- at Notes because Notes was the first of what ultimately became many products that could be sold to an enterprise and sold with professional services, and they viewed that it could be a pivotal asset. They had thought about buying Lotus earlier on for Notes, but they found out about this complicated relationship between Iris and Lotus, so they had just stayed by. Stood by. When Lotus bought Iris they reinitiated a guy by the name of Frank King, reinitiated the concept internally and they just said, "We're going to buy this company" and so they made an overture to Lotus sometime in early 2000 and-- sorry-- 2000-- 1995 and Lotus-- Jim famously at the time said-- Lou-- they had dinner and he said-- Lou said, "I'd like to buy your company." He says, "Why don't you buy me dinner instead?" And about a week later there was a hostile takeover made of Lotus by IBM that ends up getting consummated for three and a half billion, which at the time was actually a lot of money. <laughs>

Weber: No one at Lotus or Iris realized that when you formally were bought by Lotus didn't trigger this.

Ozzie: Oh, not a clue, no.

Weber: There's no way that--

Ozzie: Not a clue. No. Lotus was in-- kind of in trouble at the time because they were coping with the decline of the desktop business. There was now real competition with Excel. The market was kind of stabilizing; it hadn't reached its peak yet but there-- the early days of growth were-- the growth rate was shrinking and Notes was taking significant cash because it was a new go-to-market model. There was a lot of market development both in terms of building up a partner ecosystem for the product and a direct sales force. I should say that this was a period of education for me. I had never sold products to enterprises. There was a gentleman by the name of Larry Moore who came in and made some very strong positions with Manzi about how the product was going to fail if it was marketed per seat and as a

desktop product and it needed a solution sales force and you needed to raise the price. So we raised the price to $62,500 for 200 seats and he sold it with a solution sales force but that's what helped it catch on, and the two people on the go-to-market side that were instrumental in its taking off were Larry Moore and a gentleman at Price Waterhouse named Sheldon Laube who bet-- kind of bet his career on making a big deal at our product launch of buying a million dollars' worth of Lotus Notes to transform Price Waterhouse, at the time it was tremendous, but it was only at that time and that period when I actually started to understand the enterprise and how it was being deployed. And it was only around that time that I started to meet people like-- we had hired Irene Greif. Irene Grief is a-- was an academic at MIT who coined the term "computer-supported cooperative work." She was very, very instrumental early on in coalescing the academic community around the importance of what the category at the time was referred to as groupware. There was a gentleman at MIT named Tom Malone who had written a very influential paper on coordination theory that brought the importance of Notes and the category around Notes up because his pitch was that you had to look at the macroeconomic impact of these products on a corporation. These were not just little products that let people talk. This let you redefine your business process by looking at the coordination flows within an organization and using the technology to act as a superconductor to make the number of iterations or the fidelity of the interactions among people within an organization much higher. It was a tremendous, tremendous era. At that point in time, I started paying more attention to I think it was Oliver Williamson wrote a pretty defining paper on transaction cost economics and transaction cost economics went back to Ron Coase, his work in the '30s, very-- it gave us a new way of explaining what this was that we were pitching and why it was important. At the same time, so-- just for those who aren't familiar, Lotus Notes at its core was two things, e-mail and an application-development platform that let you build things like discussions and work flows that were very key to a certain application of the product. So if you put it in an audit firm they would build an application where auditors would collaborate with one another and they would shape the application to have fields that were relevant to auditors. Salespeople would use it in sales force automation, which wasn't a term at the time, or CRM-- they would build CRM systems on it. One of our early resellers built a great CRM system; his name was Mark Cuban. <laughs> There were all sorts of small businesses that were growing up as an ecosystem around Notes customizing it for different companies so it was really taking off in the Global 2000. At the same time, there was this little company, Microsoft, who came in and said, "Oh. We're going to compete with this thing" and they came out with Microsoft Exchange and Microsoft Exchange had a public folders function so they came-- we pitched our product as groupware and e-mail for the net, Microsoft pitched theirs as e-mail and groupware for the net, but suddenly there was a head-to-head competition and suddenly our sales skyrocketed because once the market saw that there were two people who were essentially telling the same story then everything was a battle between one or the other and it benefited both of us. We ended up taking the—

Weber: What year approximately? What year was that?

Ozzie: Ninety five, ninety six, ninety-- yeah, ninety seven, somewhere in that realm.

Weber: And you had said before that IBM had the bigger clients and Microsoft had the smaller ones. Is that--

Ozzie: That's right. The—

Weber: The two ends of the market.

Ozzie: That's right. The way that it ended up working-- excuse me-- the way that it--

<crew talk>

Ozzie: So the way that the competition between us, which was great, ended up: Microsoft because it had a tremendously broad channel infrastructure ended up succeeding in the mid market, small- to medium-size businesses. The average Exchange organization probably had 250 seats-- 250 users connected to it whereas Notes came in enterprise down and so we had most of the Global 2000 enterprises and we had very little presence in the low to mid part of the market but it was great. I mean we were both making tremendous, tremendous progress.

Weber: With IBM it had some huge number of users, over a hundred million, something like that.

Ozzie: I think when I left-- ultimately left IBM, which was '97, it was roughly 125 million users and these are users within enterprises and there were server sales and things like that.

Weber: Stepping back, when did Notes first get actually sold to customers and in what year?

Ozzie: We launched and our first sale was on December 7th of '89; it was five years to the day after we had-- I had spun out of-- I signed my contract with Lotus. As I said, the first three years of that was hard-, software engineering, the fourth year was getting OS/2 up and running, and the fifth year was really the-- getting ready with the go-to-market, and it had a slow growth at the beginning and it was probably only '92-- 1992 or 1993 that it really-- there was an inflection point in the curve, V3 of Notes.

Weber: Who were the main competitors early on?

Ozzie: There were no real competitors in the-- until Exchange came out and when Exchange came out Exchange was the only competitor. Before IBM bought us, IBM threatened us and tried to flood the market with something called OfficeVision or OV LAN, a LAN version, a non-mainframe version of an office automation system, but really we were alone in the market and the-- our biggest challenges were developing the solution sales, how to explain what you're selling to a customer in a way that's relevant. And I know this is very mundane but getting LANs pulled inside of organizations was extremely painful. New York and Chicago were particular challenges because unions prohibited people from drilling holes in their walls so you had to have unions do that and it was a very expense-- getting a-- I'll never forget one of our best early customers was Merrill Lynch and Merrill just struggled; it deployed-- it delayed their deployment by over a year just getting the LAN cable pulled within the organizations.

Weber: You could run over several different things at a time.

Ozzie: But they all required wires. They all required you to pull a coax through a wall. There was no real viable twisted-pair infrastructure at the time and it was tough.

Weber: For the e-mail portion, by the end of the ‘80s there were a number of-- Novell and various others, MCI. There were various sorts of e-mail.

Ozzie: That’s right.

Weber: Your pitch was--

Ozzie: Well—

Weber: --e-mail plus this environment that you could do your own custom applications in essence.

Ozzie: That was the core of the pitch. Just because we’re computer scientists we knew that the client server architecture of e-mail was more secure than the other predominant way, and I know this is hard for people to believe but the predominant product at the time was not phone-based e-mail; it-- in business it was either mainframe-based e-mail like VAXmail if you already had a mainframe in your company or a LAN-based e-mail, predominantly cc:Mail. And cc:Mail had a file server and the client would open that file-- that shared file and write it-- stuff into it so any e-mail client could open it up and write garbage to the file and completely overwrite the company’s mail but it was cooperatively-- you don’t do that; you get fired if you do that. <laughs> So there was no security to speak of and no assurance of service so we knew that client server mail where you run a server and have protocols that send stuff and in our case with encryption we knew that was the right architecture but it was much more resource intensive than cc:Mail which still ran in character mode on most laptops-- I’m sorry-- PCs. Ultimately, Lotus bought cc:Mail and folded it in and it became part of the Lotus communication suite and we developed very good interoperability between cc:Mail and Notes so that became a very good sale.

Weber: Security was the main selling point.

Ozzie: Yeah.

Weber: You had worked with RSA I think.

Ozzie: Right. When I initially was working on the specs the late summer of 1984 before I spun out I was grappling with how am I going to do security in a distributed environment. The data flow was modeled against a very popular system at the time, UUCP, which was the way that in the Unix community news groups and discussion forums were replicated in a very broad, wide-area way. One server would use dial-up-- telephones to dial up—

Weber: Store and forward.
Ozzie: --another one and do store and forward and that-- because there was no Internet that was the way things that-- the way things worked so-- or that wasn’t ubiquitous in a-- commercially so we modeled it in

a dial-up model like that. You would have a Notes server that had a modem on it but how do you have any assurance of security? There was no Unix operating system or distributed way of assuring security so I had read a-- an article in the-- when I was in the Lotus library. There was a journal called *Dr. Dobb's Journal* and I read an article in it about this paper on public-key cryptography written by Diffie-Hellman, Diffie-- Whit Diffie and Marty Hellman, and... The magazine wrote about this new algorithm, RSA (named for its inventors Rivest-Shamir-Adleman), that had come out and essentially in *Dr. Dobb's* they printed the Fortran source code of a sample app that did public-key cryptography and it took minutes to generate-- to do a public-key operation, you know, to do a public key operation, um, you know, in Fortran on a 286-based PC but it worked, and I said, "Well, this is it. This is the solution" so when I had the solution I put it in the back of my mind, it-- I wrote it in the business plan before we got funding, and that was it. Then in late summer when we finally got to the point where we needed to start working on it, the summer of '85, by that time Ron Rivest had written a letter to Mitch Kapor just out of the blue introducing himself and saying that he and a guy named Jim Bidzos were starting a company, RSA Data Security, and they'd like to have a meeting and pitch him on one, two, three using it or something. Mitch just forwarded it to me saying, "Do you know-- do you understand any of this stuff?" and so I took the meeting and it was a match made in heaven. I already knew what he was working on and we agreed right then and there that they would-- that we would be their customer, that they-- we would figure out a contract and they would write a tool kit that would do both a symmetric algorithm and an-- and a RSA and an asymmetric-- sorry-- an asymmetric algorithm RSA and a symmetric algorithm DES, and we would start using that in our product, DES for the bulk data and RSA to sign and encrypt the keys or whatever; you understand. So we did the contract, they built the tool kit and so on, and that was what we used all the way until we were in beta test in '86 and the lawyers told us-- the Lotus lawyers told us that we would need an export license to ship this thing outside the U.S. because it was a munition; I didn't know this. So I went with the lawyer to D.C. to meet with a organization called the BXA, Bureau of Exported Administration or something, in reality they were NSA employees, and we talked about shipping this mass-market product. They were fascinated and humored at the same time by the whole concept of actually doing this so within a month or so they gave us their proposal and the proposal had two-- it was essentially a bunch of small points but there were two important points: Number one, you cannot export DES, that is absolutely forbidden, we'll never allow it, there's no negotiation, that's it, and number two, when you do have an algorithm it will have a maximum key width-- they said-- an RSA key width; I forget what the RSA key width was but of the symmetric algorithm they said, "Regardless of the algorithm it'll have a maximum key width of 20 bits" and so we were very frustrated. We went back to Ron and essentially it became a negotiation where Ron Rivest gave us-- armed us with what to say next and we would meet with the NSA people and talk with them about it. Eventually, where it came to was Ron had to implement a DES replacement for us so he implemented two algorithms for us, RC5 and RC4, one to be used for storage of Notes on disk for disk-based storage, that was the RC5, and RC4 was the transport encryption, fastest in XOR over the wire, and that was acceptable. We got them up from 20 bits to 32 bits and when we shipped Lotus Notes v1.0 it had 32-bit encryption, really powerful, <laughs> but it only shipped in the U.S. Several months later we did an incredible amount of engineering work and we began to split the keys so that we could have a different key width for what was exported versus what was imported, and only months after our v1.0 we shipped that version and we went up to 64 bits domestically and we got them to go up to 40 bits abroad. And so if you had a collaborative group the product without any user interface changes would detect the key ring and the info in the directory about users and it would send the things abroad with a lower

encryption and the domestic ones encrypted higher, very complicated. I testified before Congress because of this with Phil Zimmermann and some others, spent an immense amount of time in there, was on a National Research Council study to try to make recommendations. We came out with this crisis report that was supposed to help change legislation and ultimately Clinton relaxed the exports so that we could do the same 64 bits abroad that we did domestically.

Weber: But writing to disk, you could use--

Ozzie: Well, we used this-- we stayed with separating the algorithms because of computational requirements. It's still way less computationally intensive to have a transport of the-- of RC4 versus the on-- the message encryption in the local store where we used RC5. And I was very proud of Notes. You know, right now it's very popular to talk about end-to-end encryption. I'm very proud of the fact that in that era even though people didn't realize the value of it every user had a key ring stored on your PC that was generated when you became a Notes user; it wasn't even stored in the central directory. That key ring was encrypted as a derived function of the password when you ran the product and all your keys; your private keys and things were in there. We had a certificate infrastructure. We had k of n certifiers so that the organizational certifiers-- you never had one person who had access to generating keys for an organization and could undermine the security of the whole thing. The only people who appreciated our end-to-end encryption were our government customers. Department of State was a huge user of it; maybe they still are. I don't know because essentially all the embassies ran and had their own servers and they wanted to make it so that the server admins of the communication infrastructure could not read the messages that were being composed by ambassadors so it was-- the encryption infrastructure was something that I think was quite significant.

Weber: How about banking customers, did you have... or finance?

Ozzie: We had some finance customers but the-- we didn't have huge penetration because the regulatory requirements required archiving that was easier to do in a centralized manner than in a decentralized manner and so they tended to be late adopters of decentralized technologies. I think arguably they skipped it and all of their mail, chat, everything ended up being centralized so they could log it all.

Weber: By the time you actually launched the product, it was really the height of the client-server craze. Did that help?

Ozzie: Yeah, it kind of helped fuel that craze. Microsoft even though they competed with us they loved us because the Windows client and the NT-- the nascent Windows NT server was the sweet spot of everything that we shipped so we were out there helping push NT servers and everyone was beginning to catch on that the NT server file-- for file service, print service and database service was the architecture of the future.

Weber: When you started with Microsoft obviously they were working on Windows 1.0, right, but by the time--

Ozzie: When I started working with Microsoft— yeah. When I was working with Microsoft they were starting with Windows 1.0. By the middle of Notes development, they were co-developing OS/2. Then that relationship soured and they began working on NT. Then <laughs> because I know Dave and-- <laughs> and so they started sending us the source code for NT so we could start getting that server; they didn't want to us to make OS/2 server the default server of choice. So we had a very good collaborative relationship with Microsoft on the server side of the world. Yeah.

Weber: When you first released it was still under DOS on the--

Ozzie: When we released it was a DOS-- on the original floppies it was a DOS client set of floppies, an OS/2 client set of floppies, a-- an OS/2 server set of floppies, and an NT server set of floppies so-- <laughs>

Weber: But the client was... even Windows 3.0 was still '91 or '92...

Ozzie: Right, and there was a Windows set of floppies. You would install the Windows floppies, then you would install the Notes floppies and so on, and I think in the v1.0 you might have even had a TCP floppy, an optional if you wanted to use TCP in your organization, that was kind of nascent at the time, but anyway, kind of moving forward with this, we were sold to IBM. IBM did amazing things with it over the-- and I stuck around for two years and I began to get itchy. As I said, I was beginning to learn more about the different social and organizational mechanics that drove successful installations and failed installations of the product and some of-- what was most exciting to me was that there was a real divide. There were organizational processes that drove the deployment of Notes like a supply-chain infrastructure and there were some people who were trying to use it in a dynamic-collaboration viewpoint where you would have a small group who was trying to develop a product so they wanted to get the engineers together and collaborate with the product managers and maybe people in different facilities. It was almost as though they were two different products that were satisfied by one infrastructure but the-- IBM was pushing it more and more and more in the direction of satisfying the enterprise scenarios, the large-scale coordination, global sales force automation and so on. So I had a product concept, started thinking about it in '96. I spun out in '97 to do a new startup called Groove and Groove's core mission was to leverage the Internet, which was a real thing by that point in time, to enable people to do the best dynamic collaboration that they possibly could, people coordinating in a very loose way, people from different organizations. If you had a job to do and you knew the people to do it you could get them to form a group very quickly, select the right tools to do it, I need a files-- a shared-files tool, I need shared discussion, I need chat, I need a calendar to assemble those tools, do it and then vaporize the collaborative environment. We made a very fundamental core bet early on-- and I realize this is a deeply technical statement so <laughs> I apologize for that in advance-- we made a bet on the Internet but we did not bet on the Web. People conflate the two right now but at the time it was hard for me to believe that in global-- that global organizations or multiple organizations would actually trust the Internet to hold their entire directories and contact information like the concept of something like Salesforce was-- and SaaS in general I just did not believe it; I just-- I could not believe. That was the core asset of a company, how could they trust that to another company, but the global transport I was a big believer in and there was-- this was the Napster era, peer-to-peer—

Weber: You bet on the Internet and not the Web.
Ozzie: That's right. I bet on peer-to-peer. Peer to peer was the way that <laughs> this thing was going to go so how do you get a bunch of people to collaborate peer-to-peer without a central server and how do you have the tools, synchronize their discussion documents, their files, all of that in a peer-to-peer fashion? You've got a dozen people who are working with one another but there's no server. How does this work? And they're not even online all of the-- at the same time so-- <laughs> - let me just pause for one second. This is going to, I've got a fax. Believe it [or not]...

Weber: Shall we take five minutes?

Ozzie: Yeah, let's take five minutes.

[Interviewee and interviewer took a short break]

Ozzie: Alright. So how if you're going to do this dynamic collaboration do you securely get connections among all the participants and how do you synchronize the data amongst all these participants? In essence, it's a problem of more or less global consensus, how do you get consensus that this version of the file is the same as that or this message that you sent arrived in that order, allowing deletions and all that so what we built was a-- this is 1997-- a distributed ledger system with cryptographically signed transactions that were essentially broadcast-- encrypted and broadcast among the participants with a Lamport's clock type of identities-- identifier scheme for global ordering so that essentially every node was secure. As a matter of fact, by encrypting it and inserting it into this transaction log, which today we would call a blockchain, you would know the identity and you would have the ordering. So this thing was-- it was amazing. It took quite a bit of time to get it right, about three years to get it right, and we came to market with it as a peer-collaboration tool probably sometime in the year 2000.

Weber: Where did the idea come from for the ledger--

Ozzie: A whiteboard. <laughs> No. What we started the company with was the concept of dynamic collaboration. Very good question. The primary technical cofounder is Eric Patey. He was primarily the architect of the overall system and overall environment. I would say the people involved in the distributed ledger aspect of it were my brother, Jack Ozzie who was working for me at the time. He was also at Iris. He came in a little—

Weber: You had mentioned--

Ozzie: Yeah. He came in the middle of Iris. Well, he was actually pretty early; he was probably 25-- employee number 25 or 26, worked on much of the surrounding infrastructure that made it necessary for Notes to connect into other systems, but he was one of the cofounders of Groove as was Eric, but there was another gentleman by the name of Ransom Richardson who was a very strong engineer and he had a very good understanding of the cryptographic elements of what was necessary. So amongst us kind of as the core group we generated the-- kind of the ideas that led to the log. There were multiple ways of synchronizing that we entertained from computer science because there are just different ways of dealing

with it, but a distributed transaction log based on signed transaction and global consensus around that seemed to be the best alternative, and essentially what Groove was if you called it today it's a DApp, it's a distributed app based on a common blockchain, and all the apps in it were built on this, the directory of users and their certificates, the-- everything that was necessary to build that infrastructure up was good and it was a-- just a very strong group of engineers who brought it up. Unfortunately— well, go ahead.

Weber: Oh, I was just going to say that if you could talk a little bit else about how the business started, who funded it, who was the board. By this time, you were a successful entrepreneur leaving IBM at that point--

Ozzie: Yep. Yes.

Weber: --and some of the people you'd worked with in the past had gone on to other things. Where did it come from? What's the business story?

Ozzie: Well, the-- IBM didn't want me to leave and they basically said, "You can have a blank check to work on your next project within IBM" and I said, "But I'm an entrepreneur. I've worked for big companies and I've worked for small companies." Now each one has its own unique character. For example, within a large company you have immense distribution but it's extremely difficult technologically to build a product because as you're trying to build the product you have all sorts of people coming in from the woodwork saying, "Does it comply with this architecture? Is it compatible with this?" because ultimately they're going to have to sell it and if it looks like it isn't coherent and consistent with the rest of the organization it's not just needless bureaucracy, it's necessary bureaucracy, but that-- but they have distribution and resources. On the other hand, as an entrepreneur you have infinite flexibility. It's wonderful. It's like a new baby. It could be anything. It can be infinitely successful. You have all this flexibility of a great team to work with unconstrained but distribution is horrible; it's just immensely difficult to get the flywheel turning. They've both got their trade-offs but for this one I wanted-- I knew that Notes would always be the thing that I was competing with internally because Notes was the answer to all collaboration within the company and within that market and I was trying to pick off a niche and do a better job than Notes did in the dynamic-collaboration niche. So I went the independent way. I seeded it myself for the first period of time. I then decided to take VC and I knew a bunch of people on the West Coast, and we took money from, well, I'll say number one, Mitch Kapor, who invested again. Mitch has always been a supporter. You know, I never could've done anything. There would've been no Notes if it wasn't for Mitch and if Mitch didn't blast through the bureaucracy of his own <laughs> organization to make it happen. But he supported me in that. Mitch introduced me to an organization that I already knew some people in, but he was very close with at the time, which is Excel, and so Excel became the lead of the first round after the first seed, where I had seeded it. I participated in that one and it was Excel. The next round, it was supported by Intel Capital. Intel liked it because they liked peer-to-peer. They saw more PCs, more computers at the edge, so they liked it, and Microsoft. Bill, I know Bill pretty well, and Bill was-- has always been a supporter, and, you know, they were very gracious and they came in. My, you know, Steve Ballmer was the primary negotiator and the primary-- my primary interaction point, aside from their corp dev, you know, group. But he had a lot of passion about it. Again, they didn't know what I was working on until it was launched, but once it was launched, that was when they came in, and we started talking about

potential integrations with the things that they were working on. Like SharePoint was the ultimate outgrowth in the collaboration side of what they were doing with Exchange. So anyway, I had a great group of backers. I had a great group of people helping work on it. As we began to try to bring it to market, we experimented with a few different mechanisms. I hired a great head of sales. Name was Brian Halligan. His is the-- he had since gone on to become the founder of HubSpot, or co-founder, one of the two co-founders of HubSpot. But going to market was extremely challenging. We came to market in 2000. This was where the bubble was bursting. Enterprise customers were hard to come by. There began to become an architectural competitor, which is the web. Centralization, you know, was a little bit <laughs> more popular than I think we had anticipated at that point early on, and it started to push us into niche scenarios within enterprises, and those niches were those people who had to collaborate with people outside the company. Very dynamically. Because when you're working with people outside the company, they don't have any common infrastructure, so if you're work-- if you're in a major company working with an auditor or working with outside counsel on a deal, suddenly you can get everybody together, you do your thing. It's all encrypted. They're, you know, really happy about that, and then the thing vaporizes. There is no server that it's sitting on, and they were very happy about that.

Weber: Who were the-- sorry. Who were the centralized competitors and who-- you're saying that it became a niche case.

Ozzie: Primarily SharePoint was the emergent competitor. You know, they-- Notes was still Notes. Notes was continuing, and I don't think we were really-- we really achieved a whole lot of traction in those accounts, but anyone who had gone with Exchange, I want to say this respectfully, they were not deeply fulfilled by the collaboration capabilities of the native Exchange, and when SharePoint came out, SharePoint tried to take on more and more and more of that, you know, "We're the collaboration add-on to Exchange, so much so that we'll just give it to you for free." So, you know, for the centralized ones--

Weber: Hard to compete with.

Ozzie: Yeah. Yeah, and you don't really try to compete. When there's something that's a locomotive like that, you try to find the areas where they're not-- their strength isn't, and the cross-organization things they were not succeeding at, at all, and so it was picking up. It was picking up. Then 9/11 happened, and 9/11, while obviously being immensely impactful and transformational to everything that was going on, it caused an explosion in growth for our product, and that was because we had some users that we actually didn't know that were in certain agencies within the government who had been experimenting with using it as a secure way of bringing people together. The moment 9/11 happened, we had an information sharing problem, and in that era-- hard to imagine in this era-- but in that era every organization, the FBI, the CIA, the MSA, state and local governments, they were all silos, and they were silos by design going all the way back to the Church Committee of the '70s. You don't want information sharing among people whose job it is-- if you're doing foreign work, you stay away from domestic work. Like, just stay far away. So when 9/11 happened ... the government said we must share, there was no infrastructure in place to do it. None of the client/server environments that were built at that time had infrastructures that thought about this. So we had very big uptake in three areas, domestic collaboration, trying-- in investigations domestically.

Internationally we had NGOs, who began using it because around that time there were also some disasters. There was a tsunami that happened in--

Weber: Indonesia.

Ozzie: Yeah, exactly, and the third one, after the quick, relatively quick war in Iraq, there was rebuilding, and they needed infrastructure to help the ministers, the new minister-- the ministries that were trying to set up outside the Green Zone, communicate with the people in the Green Zone and in D.C., and among coalition partners. They all were trying to collaborate, and organizations like USAID, who were working with them, and the Red Cross, they all had to collaborate, and there was no infrastructure. The infrastructure was destroyed, and there was no real internet infrastructure to begin with. So people were using Groove on sat phones, Groove on ad hoc IP networks set up by someone, not internet connected but IP networks, and the nature of Groove technology was that all it needed was a network with peers. It didn't need central servers, so it really took off. But commercially we had taken more money than we could sustain with that audience. We had-- by the time it was said and done-- we had taken 144 million is VC, and we were not going to be a unicorn. You could just tell at that time. So we were purchased by Microsoft, who saw the dynamic collaboration being a good way to plug in to the bottom end of SharePoint.

Weber: You always had a separate client. It was not browser-based.

Ozzie: It was not browser based and it couldn't be browser-based, because it was a distributed peer-to-peer app. Microsoft in SharePoint had a really hard problem in that SharePoint was web-based and many of the customers wanted to take documents offline. Again, if you're an auditor, you know, you connect to the network, you download a bunch of documents to your laptop, and you need to work with people when you're not necessarily connected to the corp net, and we did that in our sleep. That was just a core scenario, so they-- so Groove became Microsoft Office Groove, which eventually became SharePoint Groove, you know, then SharePoint. <laughs>

Weber: But just talking about Groove itself, so-- and the feature set, how different was it from Notes, and also, how-- you said there was 140 million or so of investment. Did you have a significant share of that? Was this...?

Ozzie: Yeah. I probably put 15 in it, into it. Yeah, yeah. Yeah. I mean, you know, it was a-- everyone believed in it. If you pulled any user of the product, they were passionate, absolutely passionate, because there was nothing like it. The features were not even in the same ballpark as Notes because Notes was an application development platform. This was actually more like the original Notes specs. It was a collection of pre-built tools that could-- that you could assemble into the solution that, you know, in this interaction with you, I need a simple file sharing and discussion tool and maybe a calendaring tool. In this other one, I need forms with, you know, a little thing where I can have to-do lists. There were just different tools, but because of the difficulty of writing a distributed app on this ledger infrastructure, we never were able to make it a generalized development infrastructure, which ended up being good because we weren't competing with it, you know, but again, its use was very focused.

Weber: For documents, it was powerful. You'd say you could download a document, you'd work on it, upload it.

Ozzie: It was very-- it was very good in--

Weber: ...And did you have collaborative editing of the same document?

Ozzie: Right. It was very good in three dimensions. Documents, which have been solved other ways by now, messaging. You know, it even had a push-to-talk function. Like you could-- it distributed voice clips just like it, you know, distributed other things, but the other one, which actually hasn't been solved at all, even to this day, is that we've developed a system that was very, very advanced in terms of the user list within the workspace that you're working on. So if I'm collaborating in a workspace with this person outside the-- let's say it's a deal. I've got this outside counsel, this inside counsel, this person who's on the other side, you know, et cetera. When you go and share something in that shared environment, it's extremely important to know who you're doing it with, because you might-- you don't want to have this explosion of permutations of, "I've got to share a channel with you and I've got to share a channel with you and this pair and--" so we had ways of doing sidebars, and in the member list, we used a combination of color, position, bold and audio cues to let you know who was there, did they just enter, did they just leave, are they in your organization, outside your organization, and so on, and the people who used it, once you started using it, with that, it brought a dimension of peripheral awareness that I don't believe has ever been seen in a tool like that before, and, you know, it's great. It's just, you know, every product dies eventually. <laughs> Lotus Notes had a long death and there are things-- or, you know, or long life, and there are things that are still in there that are not implemented elsewhere. That's probably one of the things in Groove that I hope somebody finds a way to reproduce in some collaborative infrastructure, because it was just-- it's one of those things you feel when you're in it. That was great.

Weber: Okay. With the Microsoft, so you sold it.

Ozzie: Yep.

Weber: And you were happy to-- well, you had to sell it.

Ozzie: Yeah. It was one of those things where you-- I as a leader, over holidays, made a decision that we couldn't be doing the same thing that we were doing for the last year or two, you know, again, so I called my leadership team, and basically, we broke up into three Tiger team-- three teams. One-- I can say this now-- one was going to work on the potential of a Microsoft acquisition. One was going to work on the potential of an IBM acquisition, and one was a radical change that was going to lay off much of the company and turn it into a pure open-source package, and we genu--

Weber: Mozilla, kind of?

Kind of. Yeah, kind of like that. But we would lay off the sales and marketing side and try to fund it with a core, you know, group of maybe 25 people, and that would be it. Mitch Kapor is very well connected to

the open-source community and always has been, so he facilitated incredible discussions in that, because I had dealt with open-source before, but I don't think I really understood what it takes to do community development, a developer community development, if you really want contribution, and that was great. It was tremendous, and both the IBM and Microsoft ones panned out. They both wanted to do it, and Microsoft moved quicker, and so 90 days later it was consummated, and I went to-- it was a fascinating exercise. Microsoft allowed the team as a part of it to-- and this was a core negotiating thing-- but to stay primarily in Boston area, because there were a lot of good people, and essentially that ended up being the core of a-- of what is today the Cambridge Development Center of Microsoft, which is thousands of people. But the stipulation for me was that I would go to-- that I would go on a schedule where I spent two weeks in Boston and two weeks in Redmond, and I would begin as CTO, but we'll see where the role goes from there, and the two weeks here, two weeks there lasted two weeks. I realized I had-- that my wife and I realized we had to get a place in Seattle and I would spend most of the time out there, and we would figure it out. Our kids were, you know, in school at that, you know, by that point in time, and so we just made the best of it as a, you know, a commuting couple across <laughs> the country. But it was great. It was wonderful, and I began, and soon after I began, I met with Bill and Steve and Bill said, "You know, I'm thinking about something. I'm thinking about leaving the company and starting a foundation," or giving full-time to the foundation that Melinda and I had done, and, you know, "What do you think about being," <laughs> you know, "splitting my role with Craig Mundie?" And, you know, he had a-- you can't replace Bill Gates, for crying out loud. Like, the guy, he's the founder of the company.

Weber: Tough act to follow. <laughs>

Ozzie: Yeah. You can't do it, and he didn't really word it in exactly that way, but the point was essentially this. He wanted to devote his full-time to something else. He would be-- he's open and encouraging an extended transition so that he isn't-- so that it was viable, because I'm a new guy coming in, and I'm supposed to be giving technical leadership to this large organization, and as an outsider, you know, he knew that it was a difficult concept, and essentially they said, "Look--" and again, I'm paraphrasing, because they wouldn't have used exactly these words, but, "We've built the organization in a PC-centric way. We are, you know, we have Windows and Office on the PC. We have a Windows server. It's a PC in a rack, and we have a, you know, we even have Xbox, which is essentially a PC wrapped up in a gaming OS, you know, and housing, and everything is PC-centric. You, Ray, seem to have ridden the wave of change that our industry has had. You know, you were ahead in the client/server. You saw this peer-to-peer thing coming. You must have an opinion, so why don't you take the company? Why don't you figure out what your opinion is and help lead the company in that direction? And it's up to you how to do it," and it was a, you know, obviously you're not going to say "No" to that, but it was-- I think we all kind of underestimated the, you know, the challenges involved, so we announced it, and Craig took Bill's outwardly facing jobs and I took Bill's inwardly facing jobs. So Craig was-- he traveled around the world, you know, 300-and-some days a year meeting with governments and customers. I was focused on internal change, and I, within several months, I wrote a memo, which is culturally how you do things in there, to try to set the company on a path of pivoting from a box-centric to a service-centric model, and, you know, stating that is obvious in retrospect, and even at the time it's like, "Yeah, okay. What's the big deal? We know the internet is coming." But then began the hard part of trying to work with the different

groups to define agendas, you know, development agendas that progressively lead you down that path. Some of it could be done with existing groups who were willing to do that change, and, you know, strong leadership takes strong followership. You can't do-- without both, it doesn't work, and the groups, some groups were very receptive to wanting to change, and they really defined their future. I didn't have to, you know, I'd nudge a little and give them air cover, and that's great. Some groups are kicking and screaming and some groups are just blocking, and that's just the nature of large organizations. Steve gave me an R&D budget that I could use to experiment and build things that either the groups were unable or unwilling to do, so I created this, this-- one of the projects I created was constituted with two individuals. One guy's name is Amitabh Srivasta, and the other one is Dave Cutler, and the project was called Red Dog, at the time. Dave named it. But that was what would-- that became over time Azure. It was-- the goal was cloud computing. That was January of 2006 that we--

Weber: And fully centralized. I mean... you had sort of given up the peer-to-peer.

Ozzie: Well, yes and no. I believe-- I came to believe that it's a pendulum and based on the solution there are certain things that happen centralized, certain things that happen decentralized, no one thing is going to win, but certainly cloud computing is the dominant force and that SaaS and those things, that, by that point, that was pretty clear to me. It was not clear within the industry. There was a whole part of the industry that was selling servers and client/server architecture was still pretty prevalent in 2005. AWS launched, I think, in '06 or '07... I can't remember. But yeah. That's where that came from.

Weber: What was-- what were the terms of the sale to Microsoft for Groove, and I mean, what-- you were offered the Chief Architect once you were there. I mean, the initial role was not that, right?

Ozzie: The initial role was CTO. It was Chief Technical Officer, and I was one of two or three.

Weber: Sorry. I thought it was Chief Soft... yeah.

Ozzie: No, no, no, no. I was initially one of-- ...the nature of the deal was most of the investors, not all, got their money back, at a macro level. It was not a great event from a VC perspective, but it wasn't the worst that it could've been, and I took a role. You know, all the people who were working there got great jobs and really good packages, retention packages that were quite good. That was a proxy for what they did for the past five years at Groove. A proxy for the equity that they had, and I took a Chief Technical Officer role, one of three. Craig Mundy and David Vaskevitch were the other two. It was several months after I was there that the concept of Bill leaving was discussed and in June-- this was '05 when we were acquired. It was June '06 when we announced that I would become Chief Software Architect and Craig would become Chief Research and Strategy Officer, and Bill would be with me, helping me, for two years, as part of the transition, and so for two years, Bill and I went to every meeting together. <laughs> We sat in ungodly numbers of meetings and reviews side by side, where everyone was pitching Bill and Bill was trying to say, "You're supposed to be talking to Ray," and that, you know, some, you know, some of that changed over time, but really it happened the day that Bill walked out the door. <laughs>

Weber: Did you and Bill ever disagree about the, I mean, how did that work out... because you're both sitting there listening with different opinions, right?

Ozzie: Never in public. Yeah. Yeah, never in public. <laughs> And look. The guy's the founder of the organization. It's his organization. He can do what he likes. He didn't fully grasp, not intellectually grasp-- he's a polymath, like. He did not fully internalize all the implications of pivoting the whole company to services at that moment. Obviously eventually he did. So when we were in meetings, sometimes I would be trying to give feedback to steer the group in one way and he was giving feedback based on his intuitions from his past, and that's just the way it was and we just worked through it. It's not, you know, and of course there were times of conflict but never, never with Bill. It was usually some group several layers down was trying to protect its turf and such a transition would mean change and somebody's cheese would be moved, and so, you know, there's a lot of thrashing. Change is hard. Change is extremely difficult, and it was the, you know, essentially by late-- by 2009, I had gotten in motion pretty much everything that needed to get in motion. Office 365 was moving. You know, it had just-- I had started it as Office Live. There was Office Live, Windows Live and Xbox Live, but eventually it migrated to its own, you know, cloud-based productivity strategy. Things were moving along and the bicoastal commuting was a little old, and so I talked with Steve and we worked out that by the end of 2010 I would pass the torch, and a guy named Satya Nadella took over my Azure stuff and as a development lead, and they've done an amazing job at transition to services. Tremendous.

Weber: In terms of the groups that either went, were happy to adapt or resist it, it seems like, you know, some types of functions are just easier to adapt to a service. How could some of the ones that were most rooted in, kind of, PCs, how could they adapt?

Ozzie: You can always find some way of adapting. The question is how much you drive it from within versus how much it has to be driven from outside. So Office, for example, drove it from within. You would start by saying, "It's a PC product. How can you adapt?" However, we were fortunate that we had this apparent existential threat that appeared in the form of Google Docs, and--

Weber: I was going to ask you about that.

Ozzie: No, and so you go-- every time somebody would say, "What do you mean? What are you talking about?" you could just say, "Look at that. That may not be perfect, but that's what it means to take productivity into a different context." You know, it's-- sadly, they-- Google Docs was a complete failure from a re-envisioning the concept. What they did was they said, "We're going to take exactly what users view as spreadsheet, presentation and database," or whatever. Spreadsheet. Yeah. Word processing, "And we're just going to put it in a browser," and yes, they added some critically important functions like collaborative editing and going to a URL. But for the most part, the paradigm is largely the same. That helps the existing group reconceptualize their own stuff, you know, as it might go out there. So they-- it took many generations but they, they adapted. You know, they first started saying, "How does the UI adapt?" Then it's like, "How does-- what happens to the storage paradigm in the future?" You know, it starts in files. Then it goes to a centralized database. Then it goes to a distributed database. You know, like, they had it-- they had a-- because the leaders were able to envision that future, they were able to

incrementally get their team to it. It was much harder for the server people. The Windows server people had servers on racks. They had a systems management server that could scale to an ungodly number, like 50,000 servers, <laughs> which to them was like, “What organization is going to have more than 50,000 servers?” and they had a very difficult time within themselves understanding, “No, no, no, no, no. There’s going to be millions of servers, and it’s all going to be sold as a service.” So that one was a little bit more difficult to, you know, to navigate, and there are other groups with shades of gray, you know.

Weber: Do you think your PLATO experience and the idea of collaborative editing online and doing, I mean, you’d had some experience in many guises with doc-- putting things like documents online and sharing.

Ozzie: Look, the biggest help that my past gave me in going to Microsoft was the fact that being in the collaboration business, I developed a-- and having dealt with a lot of organizations with failed and successful deployments and trying to understand the essence of leadership and the importance of leadership in process change that is necessary, you can’t just bring a tool in. You’ve got to drive a process change around the tool. By the time I got to Microsoft I was, you know, around 50 years old, and-- yeah, my birthday was my first year out there, and I took on the role knowing that it was not a technology job, it was a transformation job, an organization transformation and leadership job, and the tool of change was technology. Bill and Steve had an appreciation that I could go to anyone at the edge of the organization and immediately bond with them on the technology that they were building. That’s necessary for the job. That’s what Bill-- that’s culturally what Bill instilled in the organization. “Don’t trust leadership that can’t read your code.” But the change was a people thing and a collaboration thing, and my assistant did a-- who managed my schedule, did an analysis in retrospect. In 55, you know, plus or minus, percent of all my working hours in six years at Microsoft were spent on one-on-one, you know, to four-on-one, meaning either individual one-on-one or a couple of people who were trying to cope with something and-- or were-- I was trying to drive something. That, it’s just a-- it’s a transformation thing, and it is-- it ties into collaboration. I think if you asked someone there, you would-- I remember Butler Lampson and Jim Gray took me aside at one conference. They said, “We love your style. We love what you’re trying to do, but you’re destined to fail, and we want to help,” and I said, <laughs> “Why am I destined to fail?” Said, “We’ve been here a while. This is a cutthroat thing and you’re going to be dealing with people who are going to give you passive aggressive. They’re going to give you active resistance. They’re going to give you everything over the-- and your style is a collaborative style.” You know, “You need to take on a more nuanced leadership style,” and I worked <laughs> with those guys and I worked with a number of others to help effect change, because I was an outsider coming in, and, you know, the fact that Axure and Office 365 in particular were able to nudge the future three degree-- you know, change the course of the super tanker a few degrees, that ended up being a good few degrees, is just a fantastic thing. It’s a fantastic feeling to see 10 years down the road, you know, after I left, that they did make that leap, and I didn’t do it. I was, you know, I was-- I contributed to that, that nudging, but...

Weber: And what about-- search was another big area at that time.

Ozzie: Absolutely. Yeah. I mean, we-- <laughs> – search is a really tough one because there are times when you’re running a project of any kind, in any company, where you have a competitor and it’s your

instinct to chase after them, like, to do what they do, and the reality is in most cases what you end up is you're chasing taillights and the taillights are getting dimmer and dimmer because they're in the lead and they're gaining momentum. If you've got a situation like that, you have to leapfrog. You have to do good enough on some things and then you have to leapfrog in some dimension or plot a different course. They're going down these guardrails. You have to take it into-- and search never did that, at least in my era. Google just executed so well that, you know, it didn't happen. Microsoft tried to get into that. They tried to, you know, take the ad monetization model seriously. It didn't change the-- neither of those ultimately changed the course of the company, but I will say this. The investment in search and the investment in something people laugh at, the MSN, in retrospect, you know, the fact that they chased after AOL and all of this with MSN--

Weber: The original MSN.

Ozzie: That's right. That's right. MSN Messenger was very popular. You know, probably the world's most popular messenger outside the U.S. Those investments were key to the services transformation. I don't think people realize how hard it is. You have one group of people who did enterprise software in the company. Enterprise productivity and enterprise servers. You have another group who are-- who did high-scale services. The reason they were able to do Azure was because we took the DNA of both and put them into the common group. I used to hold a conference annually called SoftServe. It was like software and services, and you would have each one get up and tell horror stories about "my worst night operating a service," "my worst enterprise," and you got a level of empathy among the people who were working on the things. I think that was quite, you know, quite key to that, that transformation, when all's said and done.

Weber: And when you came in, some of the antitrust stuff was quite recent.

Ozzie: Yep.

Weber: I mean, had that-- the impression I got is, you know, Microsoft would've done more with integrating things online, obviously, if, I mean, that-- that put a stop to quite a bit of the plans for integration, or am I--

Ozzie: No, you're--

Weber: --overstating?

Ozzie: You're absolutely right. That's an alternate universe that never panned out. Like, it's a, you know, the Microsoft pre-antitrust was a different company, much more aggressive. Way aggressive. Super aggressive. Like the people at Lotus and the PC industry really hated Microsoft because they competed so aggressively and they had all the-- all the gears were turning. You know, the today analogy is Facebook. You know, they're just doing so well, and they're-- they're executing so well and they're dominant, and they're super aggressive. The antitrust thing really has two primary components, and in my opinion-- everyone's got an opinion-- but in my opinion, yes. There's the alternate technical product and

market reality. That's one thing to ask yourself, but the other one is Bill, led by his father, deeply believed they were doing nothing wrong. Like, he was aggressive but he deeply believed they were doing nothing wrong by giving more value to the consumer. But it's like putting stories in Instagram, you know, taking stories out of Snapchat and putting them in Instagram. But how could it be wrong by putting more features into the OS? And yes, they tied it all together. Leave the word "tied" out. That's a bad example. But they technically integrated it very strongly and all this stuff. When the antitrust decision came down, Bill was affected by it, because his belief system had been shaken up. He decided for his own reasons that he didn't want to be CEO anymore. So he gave Ballmer CEO, and the two of them-- Ballmer, arguably, I wasn't there, but arguably either he wasn't ready to be it or Bill wasn't ready to let go of it, they're very human things-- but they were not on the same page for several years. That caused a bunch of great senior leaders to leave and a bunch of other people to ascend that redefined the company, and so I think the biggest affect of the antitrust thing was not what was the alternate universe? It's what happened to the culture and leadership of the company as a side effect of the chaos that happened as of that decision, so where--

Weber: But the alternate, because, like, the original Internet Tidal Wave, tsunami, memo, that Bill Gates wrote, that's what you mean by the alternate universe as well--

Ozzie: Yeah.

Weber: --where he was talking about just making everything an integrated web service of some sort.

Ozzie: But it was more than one--

Weber: Or I mean webifying function of the company.

Ozzie: The natural path of any dominant technology provider is going to be to look in every corner and see how you can integrate that function to enhance the value of what you do, and if you focus on that, you don't even have to worry about competitors, okay? Like, you can say they-- you do it be evil, to stop competitors, but if you stay focused on increasing the value, the rest doesn't matter. So you see that in Amazon today with the ever-expansion of AWS, and the ever-expansion of their marketplace. You see it in Facebook, in the ever-expansion of its communication tools, and it's a natural-- it's the natural evolution of that, of that thing, and it's super-charged by the natural network effects that are present in social and in platforms, so...

Weber: And you talked about the advertising-based structure. Said that Microsoft did not succeed or do that much with, but I mean, I've thought that has become important in recent years, right?

Ozzie: Well, it's... I think it's a large business in the absolute but a very small business for Microsoft. I mean, if you just squint your eyes for a second and say-- just compare their products and do a little game of removing the product and see if you still believe in the company. So, I believe in Microsoft. What if you removed Office? Okay, it's a different company. I believe in Microsoft. What if I removed Azure? Little different company. Those are massive. If you removed Xbox, well, I think it's the same company, but it's a

little bit smaller. What if you removed Bing and the advertising? Yes, machine learning and a lot of the talent that search brought into the company is helping them be a great AI company now, but I don't believe that advertising contributes a material amount to their profitability.

Weber: Okay. Makes sense. So it might, in the future, but that's certainly not in the current picture, you're saying?

Ozzie: Yeah, yep.

Weber: So then anything else about your time at Microsoft or...?

Ozzie: No, no. It was a great learning experience for me, and I hope I did a little bit-- contributed a bit. I left in 2010 and decided I was going to take a little bit of time, rather than just doing another startup, and I'll just touch on this very, very, very briefly. In March of 2011, right after I had left, there was the tsunami in Japan, and Fukushima melted down, and someone who I had worked with and knew in the past, Joi Ito, ended up bringing together myself and a group of about 30 people to Tokyo a week after the meltdowns happened. So things were still very shaky and high anxiety. But he brought a bunch of people together, saying, "How can we as technologists help?" that simple question, and by the end of the week, we had a prototype of a system. We identified a need and had a prototype of a system in place to satisfy that need. The need was that people were being asked to-- everyone was afraid of radiation, and people were being asked to relocate. People didn't have any readings. They didn't know if the table next to you is irradiated or not, and neither the government, nor the power company, TEPCO, were releasing radiation data. I mean, some of it, the government, is-- some of it is cultural. They wouldn't release un-vetted, scientifically vetted info, but everyone was just scrambling, trying to contain the situation, and citizens were just freaking out. So we decided that if we could get facts, uninterpreted facts, out to people that people, scientists and others, would be able to interpret, is it safe or is it not safe? So a nonprofit was created called Safecast, and what we did initially was get laptops and a Geiger counter and a GPS and put some Tyvek suits on and strapped those things to a car and started driving around the exclusion zone, measuring, and eventually build generation after generation after generation of IoT device to do that measurement so that individuals could build from a kit their own Geiger counter and tracker, put it on their bike or car, go around places that were relevant to them and their family, and every time they measured it, as a side-effect of measuring it, it would also go to the cloud so everybody could see the situation everywhere, and if you go to Safecast.org right now and click on Maps, you'll see that even after all these years, the 10th anniversary is coming up, there's an immense amount of data, immense amount of measurement, both of radiation and, increasingly, air quality, roughly 10,000 volunteers, roughly 1,000 devices that're out there. It's a nonprofit I believe in. It's an open-data, citizen-data nonprofit, not pro-nuclear or anti-nuclear. People work side by side. Anyway, I got turned on in that project into IoT. I'm a director of the nonprofit, but I'm also an engineer, and I started coding things up and helping them, and it was like returning to the '90s. Coding for microcontrollers now is almost the same as coding for an AT-- a 286 chip back in the day. You're coding C. The operating system isn't really an operating system. It's like DOS was, FreeRTOS and-- it's like DOS. It was like going home again, and the hardware piece was like the hardware that I did back in high school and early college. I started building devices, and all of a sudden, hey, yeah, I know how to-- I understand all this stuff enough, well enough. So I helped build them

a-- something they did not have volunteers on staff with enough talent to do. I developed them a low-power cellular Geiger counter, solar-powered, that they could deploy into the exclusion zone, where they had occasional access, and it would have to kind of stand alone and maintain itself. This was to augment the drive-based radiation measurement. As a part of doing that, I couldn't believe how hard it was to do the cellular. Cellular is high power. It has arcane commands. The business models is you have to pay per device, per month. It was crazy. So that more or less motivated the birth of my next startup, which is Blues Wireless, which is what I'm working on now, and essentially-- what do I have? I don't even know what I did with it. It's a tiny, little hardware card that essentially embeds cellular capability onto it, and you buy the little card and embed it into your product, and you don't have to pay a carrier. You never have to talk to a carrier. It has crypto-keys burned on at manufacturer and certificates, so it operates with TLS from birth with no management. It's a fixed price, and it's got very simple JSON interface to send data. All you have to worry about is your data, and the results appear in the cloud. So, anyway, I'm pretty excited about it. It's my Nth startup. As I said, from an entrepreneurial perspective, it's got all the wonderful attributes of entrepreneurialism. It's a small group. The technology is wonderful, and the product is wonderful, and I know that we hit the mark, but building to go to market and customer base is going to be challenging, and it'll be interesting to see where it goes from there.

Weber: Very neat. I hope you'll contribute one to the museum.

Ozzie: Absolutely.

Weber: You skipped over Talko..

Ozzie: I'm sorry. Yes. Absolutely. Well..

Weber: … which might hearken back to PLATO just a little bit.

Ozzie: Yeah, it's a very good point. What led me to talk about Blues was because Fukushima and Safecast made me aware of that pain. But I was not motivated enough to do that company at that point in time. In 2012, I had some other collaboration ideas brewing. In essence, just like Groove had a slice of-- it took a slice of collaboration focused on a certain social mechanism, the dynamic collaboration piece, I was noticing in the emergence of cell phones that people were stopping-- they weren't using it as a phone anymore, that texting was becoming dominant, and the concept, the very concept of calling someone without prior coordination-- dialing someone and having it ring your phone was becoming evil. People don't want to hear the phone ring. They want a text message from somebody first saying, "Is it okay to call?" or, "Is now a good time?" The whole social mechanic around real-time interaction was changing, and I believed and I believe that a lot is lost right now in interpersonal communication, both in a social and in a business context, when you can't hear the nuance in someone's voice. Sometimes you can hear, is someone angry? Is someone concerned? Is someone showing empathy? Is somebody showing sadness? These things are very easy to perceive when you hear somebody, and we're using-- we are developing this immense affordance of emojis and things like that to make up for the fact that that side-channel is not present. So the goal of Talko was to create a messaging app that would be accepted by people in a work context as messaging but had voice woven into it so seamlessly that people would make

a natural transition much more often that started with text, and you could just immediately shift into voice, and it also bridged very seamlessly ephemeral and persistent. That is, the thing about messaging is you can go and look back at something. The thing about voice is after you hang up, whatever happened to that conversation is gone, and if you want to share it collaboratively with someone, it's just very difficult to do so. So, the concept of seamlessly going from texting or photos or whatever to voice – to talking over voice recorded and transcribed to ephemeral, that whole seamless transition was the problem space I was trying to take on. It did all the things that you'd expect, push to talk and all of these things, trying to help people collaborate using phones with an audio focus. Talko was a great startup. The first co-founder was my son Neil, a program manager who helped greatly in conceptualizing the experience. We were soon joined by Matt Pope, Eric Patey, and Ransom Richardson, all amazing people I'd worked with at Lotus, Groove, and Microsoft. Talko had a bunch of passionate users who were all vision-impaired. It turns out that the biggest community that picked up on it were the people who appreciate audio. But we really didn't get a whole lot of traction beyond there. Messaging is a very busy market, and so I sold that company also to Microsoft, and that became a core of the real-time communications that became part of Skype and Teams. I did not join Microsoft.

Weber: Some of the functionalities like where you, for instance-- today you can get a voicemail transcribed, and you can sort of go back and forth between the audio or the text, right? Is that..

Ozzie: Yeah. Some of it is. Okay, so what has happened since then is that all messaging apps have added a voice-record function in their apps, which enables you, if you really want to do that, to do it. All of the call apps, meaning the native Android and iOS call functions, now have integrated a message function that if somebody's calling you, you can message them back saying, "I'm not here." People are trying to cope with the seams between mediums in different ways. It's not woven together the way that Talko did it, but that's okay. You've got to do experiments, and that's the way I do it. So...

Weber: So, last, we have some of these sort of looking-back overview questions. Is there anything else you wanted to say just about the more... biographical...

Ozzie: No, not about the biographical thing. In terms of entrepreneurship or in terms of-- if I were talking to one of my kids or a friend or-- who's a young engineer, and I'm trying to look back at my own career and see what-- see if there any patterns, any good patterns that might be still relevant, I've been very fortunate to have been able to maintain my core passion of building things throughout. I learned midcareer something about myself that may not be relevant to everyone, but it's about myself. I enjoy being in a leadership role, even in very large companies. I'm on the board of HP. I'm a senior adviser at AT&T. I like being in a senior role, because you can make decisions that have a very broad impact. They're very small decisions that have a very broad impact, and you can at least in theory make a big difference in a very leveraged use of your experience. But I also like being hands-on. I don't have to be coding, but I have to be very close to the design of a product. There has to be some project that fulfills the builder piece of me, and because I managed to discover that midcareer, everything I've done I've had the luxury of structuring my job to-- when I was chief software architect at Microsoft, I had my labs where I could have-- I had a wonderful person named Lili Cheng, who-- I funded a team to do all sorts of experimentation in social concepts. They did something called Bubbles, and they did this, and they did

that. I got to scratch my itch in that way. That's where Azure came from and things like that. Right now, as I said, I'm on boards, but I'm doing my startup. So I'm getting both of those taken care of and so on. So I guess my biggest thing is really to try to understand yourself. Many startups have hacker-hustler pairs. There's the person who's the product manager and the go-getter, and they get funding, and they understand the customer, and they love the customer, and the engineer who loves building and loves the complexity of the machine that they're debugging and all of that. Really, try to understand who you are, because you can be successful in any of those things. A hacker does not have to become a hustler. You do not have to do any of these things. There are places in accomplishing these things, in doing all of it, and learn to appreciate the value that others bring in that task. Some people naturally are transactional salespeople. That's what they do. That's what they're good at. They like the competition of it all. Some people are solution-sellers. They love the customer. They love to understand what the problem with the customer is. They love to make the customer happy. They're not transactional. They love that. Some people are in the middle between that and projects-- product. They're product managers. They love shaping the product. Just understand. Try to understand who you are, because that will determine the opportunity space that's in front of you, and I know this isn't directly relevant, but one of the things I recommend to people out of school all the time is go work for a big company right out of school. Don't have the very first thing that you want to do make a million dollars doing a startup. Ultimately, you have to understand who you're building for, and we all know for consumer products what it takes to build those. You can look at the next person over and understand what a chat app does, and, yes, you can immediately go build that. But if someday you're going to build something that companies use, restaurants, dry cleaners, insurance agencies, SpaceX-- if you want to understand the machine that is an organization, there is no better way to do it than working in one and understanding the organizational dynamics, just like you understand social dynamics kind of intuitively. They are different. There are all sorts of mechanisms that come into play, and it's great to understand it before you make your decision how you want to set your career. So...s

Weber: Really nice. There's several of these global questions that-- Jon, do you have a feeling on how short or long-- I mean, Ray, I'll read-- one is, as you consider the challenges facing people and the planet, what new tech innovations are you most optimistic about making a positive difference and why? What're your hopes for the future of technology for the benefit of humanity? Then, of course, the next question is, conversely, what're the perils? What're the ones you're most concerned about?

Ozzie: So in terms of looking forward, I'm a technologist. I've spent most of my career in the collaborative aspect of it, the social and organizational collaboration aspects and in pure systems infrastructure. So with that filter, I would say the biggest technological-- the biggest peril I think is pretty apparent right now, and that is that many of the social tools that we've been using to help us reduce transaction costs between-- in processes and as social tools are now being used to fundamentally change society. You've seen what's happened with elections, what's happened with Facebook, what's happened with activism. They can be used well, and they can be used destructively, and I am-- I don't think-- there's obviously no right and wrong from a pure technology perspective, but when you connect people with zero friction, there are unintended consequences, and I think any entrepreneur right now, any student, anyone who's a technologist who's not thinking about unintended consequences of what they do I think is just simply

irresponsible, because we might've been able to gloss over that in the past. We really shouldn't be doing that at this point in time. If you're building a project, at least try to go several steps forward and think about it. What I'm most excited about, and this is a bit self-serving because of the Blues, but I do what I do because of the future that I see. We've been collaborating with people. I think we're moving into an era where we're collaborating with machines to get things done and to get things done and institute change in ways that we couldn't do before. One of the things I'm most excited about is the fact that inexpensive IoT technology is going to be ubiquitous. It's going to be something that individual citizens can take out and instrument streams and instrument mountains and hillsides and other places that they might care about as individuals who are passionate about the environment or who care about a local ecosystem, and you can use citizen science and the technology to learn about the environment, to feel more connected with the environment and to share that information with others so the aggregate trends can be measured. I think this is extremely exciting, and it comes at a time when governments are de-investing from environmental initiatives and where both governments and commercial enterprises are predisposed to only give good news and to sit on bad news. So from a technological perspective, there are many things that're going to happen, but, again, I'm pretty psyched about how-- about the systems that can be done to promote citizen science.

Weber: How can we ensure that innovative new technologies are inclusive, accessible and ethical to benefit everybody?

Ozzie: Wow. That is a deep question, and I don't know if you mind if I defer on that one, because it would be a rambling answer that would sound like a white man who's powerful, is trying to spout platitudes. I don't know what we can do. I mean, what we need is fundamental change, and that change starts in early education, and it's really not a technological topic. It's something computer museums should do, but the one who I wish you would bring in, because nobody is bringing him in-- I don't get it. Mitch Kapor has done more personally and investing through his Level Playing Field Institute in the area of helping underrepresented minorities using technology as a tool. He and his wife, Freada, are just such good people and have done real investments, and in this era, in this BLM era, it's a shining light of what-- it's not the answer. He's working different programs that, if scaled more, could work really well, I think, if you could find a way to scale it. But it's just absent from the conversation. I don't know why.

Weber: We can talk more offline, but actually we are talking to them. There is some conversation there, so happy to...

Ozzie: Good. Anything you can do to scale that message, that's great.

Weber: … talk about that afterward. But, yeah, that's a very good point, and what would you like to be remembered for?

Ozzie: <Laughs>. Oh my God. Wow. The thing that makes me feel best about my career and the thing that I would like to be remembered for is the impact that I had on individuals who either worked for me, worked with me or were part of the ecosystem of my products, where they were able to build more thriving businesses, particularly small businesses, around the technologies that we built. I personally feel

the best when people would come up to me and say, “It’s 2020, and I’m still working on Lotus Notes, and I built a company of 30 people, and I put my kids through college that way, and this person did this, and this person did that.” That to me is the ultimate reward. Technology will come and go, and if I didn’t develop a piece of infrastructure, someone else would. These things follow natural paths, but the impact that you can have as a small business or a large business owner on other people’s lives is irreplaceable, absolutely irreplaceable.

Weber: Here’s an even easier one. Drawing on your wealth of life experiences, what one word of advice would you give to a young innovator or entrepreneur? You can think about this for a little bit, but what is the word, and can you tell a story that illustrates why you chose it, and that, I should-- normally that’s asked-- people are given some time to think about that, right?

Ozzie: Yeah, but I can tell you that right off, to me. I mean, to me the word is build. The reason I say that is because build kind of implies that you are either pursuing something, a scratch that you have to itch-- it’s pursuing something in the realm of your creativity, or you’re trying to solve a problem that somebody else has. You could build an organization. You could build a team. You could build a software product. You could build a hardware product. You can build almost anything, but building is the best form of activism that you can have. To me, personally, you can talk about things, and you can make a difference talking about things. You can sway people’s opinions. You can shape culture; however, what we build is an action that we take that I know can impact things in very concrete ways. So, from my perspective, that’s what it is. It’s build.

Weber: Excellent. What does this award mean to you to become a fellow, and I think you know who some of the others are, to join those other pioneers as a CHM fellow?

Ozzie: Well, this is a little odd. I am flattered beyond what you can imagine. As you can tell by the nature of this interview, I do what I do, because I am passionate about it. I am very fortunate to have had some success in it, but I don’t view myself generally-- I view myself as a builder and as a solver of problems. There are people who I view in a completely different dimension from an achievement, an accomplishment perspective. When I look at the fellows list, that’s what it’s dominated by, and it’s incredibly flattering to be perceived to be part of that group, and all I can do is thank you for that perception.

Weber: Very nice. To face the pressing current needs of today’s digital citizens, what role do you see for CHM as a leading museum to help inspire and inform technology citizens in order to shape a better future?

Ozzie: That's a really good question. I mean, there are history and archival functions that CHM uniquely provides, and I think it’s important for the reason-- the obvious reasons. You should always have a good sense of history before you go off and do something new. There have been people who’ve made mistakes that you can learn from and so on. But I actually believe there’s also an educational component that CHM could uniquely provide, particularly in communities that may not have had the opportunity that some of us have had in being exposed to technology so early. I mean, I was exposed well before my

peers, because I was fortunate in a neighborhood that happened to get a grant. There were many schools and neighborhoods that did not get those grants, and that exact same thing is happening today, and I believe that CHM, in bringing together people who have been accomplished in this area, has an opportunity to get that leadership out in front of people who might not have otherwise been exposed to it, potentially to serve as role models in terms of things that they could aspire to.

Weber: For the last, really, two that are related, we've both talked about the advertising-driven model and some of the excesses of Facebook, things like that. Obviously online communities can have different characters. Do you think the sort of badder uses of this in ways that certainly you seem to agree can be destructive-- how does this affect this kind of basic idea of bringing people together online that's been such a theme of your career? I presume you didn't foresee this back then, but, I mean, how does it make you think about sort of bringing people together in this medium, now that it's revealed that there's kind of two sides to it?

Ozzie: Well, not that it's an excuse or a place to hide, but the vast majority of my career has been spent in what Irene Greif coined computer-supported cooperative work. It's the use of social mechanisms and tools to help an organizational goal or outcome that that organization might have. I believe that the dangers of social tools within the organization are different than the dangers that they are externally. Technologists are contributing to both, and so there are-- all the cautions are necessary. But within the organization, essentially what you can do is draw the same lines that you do that we have on the outside in pure social, which is, what're the undesirable mechanisms that are latent, that are currently happening within organizational constructs, and how do social tools either reinforce that evil, that bad thing, or how are they used to mitigate it? So if there was elitism or something like that or racism or any characteristic within a given workplace, are those social tools that the workplace is providing helping make things better or make things worse, and social is now woven into every communication tool that we do online. If there were people who were... protectionist of their territory or heavily political internally, those are going to take on a different tone when augmented by tools. So all can say is anyone who's implementing these things should be thinking about these secondary effects, and it's easiest to just simply-- just as I said earlier that it was easy for a group at Microsoft to look at Google Docs to understand what change might look like, people within the workplace should be looking at Facebook and Twitter and certain tweets that're out there and say, "Well, wait. How could this impact the organization in ways that are unforeseen but we could've directly drawn a line to?" The most obvious of those has already been shown. I think everyone now is suffering Slack fatigue or Teams fatigue and certainly Zoom fatigue because of the overuse of the tool and the complete lack of empathy and respect of the people on the other side of the wire. There are mechanisms that the tools could incorporate to help mitigate some of that, but they're not there yet, and..

Weber: For instance?

Ozzie: I'll give you a really silly one, and you'll probably laugh at this, but when I got to Microsoft, I went from a startup to receiving 250 non-spam e-mails per day, and it's difficult to have a life and be a leader and go through that. Most of the leaders would just ignore it, but I have a very difficult time doing that, and so what I did was I had one of my teams implement an Outlook add-on that computed a cost of a message by looking at the to and CC list and the number of words in the message. It was very crude, but

it would attach a footer to each message so that when you do a reply to a message, it said, “Are you sure? This message will cost $672 in”-- because to process that message times those recipients, and so just giving awareness-- yeah, you could put penalty mechanisms in, but it all begins with awareness. What is it? It begins with awareness, then understanding, then empathy, then-- there’s a step. You have to do something in the tool to help people appreciate the burden that they’re imposing on the other side, and just getting people to hit Reply instead of Reply to All in e-mail is a-- at that time, when e-mail was so prevalent, I would view it as victory putting somebody in a Cc list instead of a "To" list. There’s a different cost, because socially if you’re copied, it tends to be okay to ignore it. But if you’re in the to list, socially you really should reply unless you’re a jerk. Anyway.

Weber: Okay. That makes sense. So there’s different ways to mitigate. Last question, which is related, but the-- do you think the business model-- I mean, certainly the contextual advertising has been blamed a lot for some business models. Do you think there are better business models behind online communities? Most of your work has been around ones under license to an organization. But take this or don’t take it if you want, but do you think there are better business models?

Ozzie: I don’t believe there is a viable business model. We’re focused on the word business, meaning profits and funding. I don’t believe there’s a viable business model for online community that doesn’t have the negative repercussions of what’s going on. I believe you can create paid communities, but by definition they are bubbles. You’re only going to pay for something that you want to be part of and that has information that you really want, because it’s your money. So if there’s a filter-bubble issue right now, this is-- paid communities make filter bubbles even more prevalent. You hang with people in that community that you know, that you like, that are like you. So if you want to encourage diversity or diversity of thought, that’s not likely a solution, and that’s why I think if there is a solution, it’s probably not associated with the word business. I’m not sure exactly what the solution is. Our current dominant communication model, Facebook, Twitter and so on, is driven by the necessity to make a business out of it, and therefore it will focus on game dynamics, on dynamics that are self-reinforcing and looping, because that’ll give better returns. If there is going to be a healthy community, it’s got to be somehow open and moderated and not motivated by profit. I don’t know what that is, but that to me is really-- it’s the only alternative in that realm.

Weber: People brought up before the Internet, CompuServe, Minitel. Various ones were pay by the minute or by time. But you can’t really go back there, because once you have free out there, though... you can’t get people to pay again.

Ozzie: It’s that, but also we have a much bigger awareness these days of inclusion also, and at this point in time, we can’t even get past Internet access being something that-- people are doing right now what they did then. They’re paying the very low amounts for Internet access, but that still excludes a bunch of people. So I don’t know if you start raising the bar of paid communities, I don’t-- I think you would end up with good, healthy discussions, just like if you get the Atlantic or some publication, you’ll end up with well-thought-out arguments, but it’s going to be a niche audience that has access to it. I am most interested in the industry finding-- and it’s difficult, because I said industry, but someone finding something that will

become a substantial communication platform that is-- where the underlying mechanisms are open and free and inclusive, and I don't think that can happen commercially..

Weber: All right. Well, really wonderful interview...

Ozzie: I hope your memory card still had some room on it

END OF THE INTERVIEW

