Oral History of Tom Proulx

Interviewed by:
Marguerite Gong Hancock
Eileen Fagan

Recorded July 22, 2019
Mountain View, CA

CHM Reference number: X9130.2020

Hancock: Tom, thank you so much for coming to the Computer History Museum.

Proulx: Sure.

Hancock: Today is July 22, 2019. My name is Marguerite Hancock and I'm here with Eileen Fagan to interview you and to have your story added to our collection. We're thrilled to have you. So if you don't mind we'd like to, for the record, start at the beginning. If you could tell us where and when you were born and talk about your family or early influences that helped shape your life.

Proulx: Well, I was born in Long Beach, California. I grew up in Southern California. My parents were young. I think when my mom had me she was 18. My dad was 21. My mom was from Germany. Her sister had married a G.I. after the war and moved back here. She came over when she was 17 to visit her sister and met my dad, and ended up never going back. But they struggled, they didn't have a lot of money when I was young. When I was a baby, they were living in a cottage in somebody's backyard. I mean, it was pretty slim pickings in those days. And we moved around a lot because my dad would get a better job. We'd move and then he'd get another better job and we'd move. So up until pretty much until high school, we moved about once a year.

So I don't have any long-term friends from my early childhood until high school, and then high school we finally settled down. My dad had kind of made his way up to a middle manager kind of job at a corporation and was making a better salary by then, and we bought our first house. That was a big deal, buying a house, because we had rented up until then. And that was in Huntington Beach, California. That's where I went to high school. I went to Edison High School in Huntington Beach. I was really interested in math and science. I took all of the lab science courses that the school had to offer.

At that time, I wasn't even really thinking about going to college. It just wasn't something that my parents really talked to me about. My dad went to college and got a law degree, but we just didn't talk about it. I'm sure if I had asked "Do you think I should go to college?" they would have said "Yes." But we just didn't talk about it. So, it was just something that I kind of came to on my own. For a while in high school, I was not even planning on going to college. But I really liked math and science. I applied to a number of colleges and got in everywhere that I wanted to go.

Hancock: What made that difference for you to decide to apply and to think about college?

Proulx: I had friends at Edison who went. One friend in particular, who was actually my girlfriend at the time, who was a year older than me, went to Stanford. And that's when I first started thinking about going to school. I actually went up to Stanford to visit her and <laughs> it was like, "Wow. This place is amazing." That wasn't *the* reason I went to Stanford, but that was *a* reason that I went there. That was the reason I applied to Stanford. I didn't even know where Stanford was up until then. I thought it was an Ivy League school. <laughs> But when she got into Stanford, I learned that it was up here. I came to Stanford to visit her while I was starting to apply to colleges and really fell in love with Stanford.

But Stanford wasn't my first choice at the time. My first choice was Caltech. I had always been really interested in Caltech because in fourth grade, my teacher had read a bunch of stories about the senior ditch day pranks and stuff that they did at Caltech. <laughs> I just thought that was the most amazing, coolest thing ever. So, I always wanted to go to Caltech, and I did get into Caltech. But I visited it and it just wasn't as good of a fit for me as Stanford was. In particular, I wasn't sure what I wanted to major in. I knew I liked math and science but I didn't know what I was going to do. And if you're going to CalTech, you better be pretty sure that that's what you're going to do. So Stanford gave me a little more breadth. It was a bigger school. It wasn't big, but bigger than Caltech. And there were more women at Stanford. <laughs> Caltech, I remember at the time, had 60 undergraduate women. <laughs>

Hancock: The ratio's not as good.

Proulx: The ratio was not very good. That also entered into my thinking. So, I ended up going to Stanford and that was a great decision for me.

Hancock: How did you decide on the EE major ultimately?

Proulx: Well, in senior year of high school, I believe, was when I took my first computer programming course. It was in BASIC and I just loved it. I just really loved it. So when I got to Stanford--

Hancock: Can I interrupt for a second? What were you working on? Since we're the Computer History Museum.

Fagan: And what year was it?

Hancock: What year and what hardware were you working on?

Proulx: So the computer programming course I took in high school was in BASIC. It was on a mainframe somewhere in the district office and we just had Teletype terminals in the classroom. And it wasn't really a classroom, it was a little section of the back of one of the classrooms as I recall. But it was just so magical that you could type in some code and say run and it would print out really cool stuff. I mean, it was just really fun.

Fagan: What year was this, Tom?

Proulx: I graduated high school in 1979.

Fagan: Okay.

Proulx: When I went to Stanford, though, I wasn't sure what I was going to do. I didn't know about computer science as a major and in fact, at the time Stanford didn't even have a computer science undergraduate major. I started off as a physics major simply because that was my favorite subject in high school and I kind of just rolled into that at Stanford. I did that for the first couple years. I liked physics but I didn't want to dedicate my life to it. And I couldn't see how one would earn a living as a physics major. You know, I could understand if I wanted to stay in academia and be a physics professor, I could understand how that would work. But I didn't want to do that, and I just didn't see how a physics major was going to get me a job in industry. So, after my sophomore year, I switched to EE. Because I really would have liked at that time to have gone into computer science but they didn't have that, and the closest thing that Stanford had at the time was EE with a CS emphasis. So that's what I did, and I loved it. I took all the computer science courses that I could take and I really, really enjoyed computer programming. I just loved it. I love it to this day.

Fagan: What was the first program you wrote?

Proulx: Oh god. I don't know.

Fagan: Or what was one of the ones you wrote or worked on?

Proulx: Oh, I remember one really fun project I had, for what class was it? I think it was for CS107, and I mention that by name because that class still exists today. <laughs> It was then and is now still the class that kind of separates--

Fagan: It's the weeder.

Proulx: --the wheat from the chaff, or the men from the boys. My daughter and son both went to Stanford. Both were CS majors. Both took CS107. So a lot of the classes that I took, they took. And of course they've changed a lot since then, but not that aspect of CS107. It's still <laughs> just a real ballbreaker, you know. But I remember one really fun assignment was the jealous husband's problem. It was a variation on the fox, the cabbage, and the hen. You got to get all three across the river. This was a variation of this called the jealous husband's and the same sort of thing. You had to get three couples across the river but no man could be alone with another man's wife, that kind of thing. It was a fun recursive kind of algorithm. I really enjoy these recursive algorithms because you program how it should solve the problem but you don't actually know what the solution is, and then it solves the problem for you. It's really very gratifying when you see the answer kind of emerge from something you've created. So, I remember that was really a memorably fun project way back then.

And what else? I had some good summer jobs while I was at Stanford. After my freshman year, I got a job working at Rockwell International and I was working on some interesting stuff there in missiles. <laughs> We have these missiles in silos all over the country and in those days, maybe even today, they're guided by gyroscopes and little seismic movements that there always are would cause the gyroscopes to precess. And they had this software that would have to fix that so that the gyroscopes wouldn't get too far off because then you'd miss Moscow. <laughs> So I got to work on that and that was really fun.

And then after that, in the following years I got an internship at Hughes Aircraft Company working on chip design there. Then I became a Hughes Fellow which meant that they were paying for my school, which was great because I didn't have scholarship. My family made a little bit too much to get financial aid but not enough to really to be able to afford Stanford. So, my parents basically said, "Look. We'll pay for your room and board plus 1,000 bucks, but you got to do the rest." I had to come up with the rest, which I did through a combination of some scholarships and I worked summer jobs.

During high school, I worked at Kmart. I had busboy jobs at our local country club. And I just worked a lot to make money so that I could afford to go to school. So, when I got the Hughes Fellowship, which I think was junior year, that was really really helpful. Of course, I had student loans, too. So that meant that I didn't have to keep racking up student loan debt once Hughes came along.

Fagan: Was this back in Long Beach? Hughes Aircraft.

Proulx: It was in Newport Beach.

Fagan: Okay.

Proulx: Yeah. In those days, I mentioned my family had moved around a lot. By high school, we were living in Huntington Beach. And I was working at Hughes Aircraft in Newport Beach. <coughs>

Hancock: Can you create a little context for us, you had this intense interest in computer science and around the Valley there were people who were coming. So can you just describe what was exciting at the time?

Proulx: Well, I'll tell you--

Hancock: Who were you talking to and listening and watching?

Proulx: I wasn't talking to people in the Valley. I was talking to just my professors and things. That was it.

Hancock: Oh. Sure.

Proulx: But the guy that inspired me was Steve Jobs. My senior year at Stanford, I had actually applied and had been accepted to the coterm[inal] program. So I would have had another year to go, because it would have been five years to get my bachelor's degree and my master's degree. And the master's degree would have been in computer science. I say "would have been" because in fact, Intuit happened and I never ended up going back the fifth year.

But I had started working on my master's degree in computer science and I was inspired by Steve Jobs. I had decided I was going to do what Steve Jobs did. I was going start a company. I wanted to start a company but not a hardware company. I wanted to start a software company, and specifically I wanted to start a software company that would be in personal computers. And I wanted it to be some kind of mass market sort of product. I didn't want it to be a niche-y kind of product. I wanted it to be a mass market product. I didn’t know what it was, but I knew those things I wanted it to be. I wanted it to be PC software, mass market.

I had another year to go so I was starting to think about this. But there was no rush because this was my senior year and I was going to be back a fifth year for my master’s degree. And one day towards the end of winter quarter, I was working with a group of friends on a group project for some class, I don’t remember what it was. We were in the lobby in front of Terman Library, the engineering library at Stanford which is no longer there. They tore the building down a few years ago. But we were working in there and this guy comes up to me and he’s got a handful of flyers and he taps me on the shoulder and says, “Excuse me. I’m looking for computer programmers. I have a bunch of flyers that I want to post. Where would be a good place to post these?” And he was Scott, Scott Cook. So I told him. I told him where the computer science areas were and there was a bulletin board downstairs and everything.

I said “But tell me about this. What are you doing?” And he described this idea, actually it was his wife’s idea, Signe’s idea, for basically automating personal finance on a PC. He told me he had done a little bit of research on this, and he walked me through some of the research. He described to me with some specificity what they were thinking about, and it was basically what ultimately became Quicken. But it was basically a simple mass market product that would run on PCs and would automate the routine financial tasks that every household in America has to do. And it was like <snaps> boom, obviously that is the idea that I’ve been searching for. PC, mass market, that is the idea. I mean it just dropped into my lap.

Hancock: So, just painting a picture. You’re standing outside of the library and on a chance meeting you’re having this entire conversation?

Proulx: Yeah.

Hancock: And you knew?

Proulx: It was instantly obvious, because I was actively thinking about this over the last several months, right? And then all of the <laughs> sudden, it’s just like, here it is, you know. This is what you’ve been thinking about. So, it was immediately obvious this was what I wanted to do.

Fagan: What did you think about this guy?

Proulx: Well.

Hancock: There’s the idea and then there’s the guy.

Fagan: Then there’s the guy.

Proulx: Well, you know, I tell you--

Hancock: Co-founder.

Proulx: What <laughs> I remember about Scott--

Fagan: Yeah. What was your first impression?

Proulx: My lasting impression of Scott was--

Fagan: Yes.

Hancock: What was that?

Proulx: --he was wearing a button-down shirt and it was the stiffest, starchiest shirt I had ever seen. It was like-- <laughs> I had never seen a starched shirt before. <laughs> That was my lasting impression. And you know, where I came from, people didn't wear button-shirts, much less with starch. Certainly at Stanford everybody's wearing t-shirts and shorts. I had never seen a shirt <laughs> like this before and it was like he could barely bend his arms, it was so stiff with starch. That I remember. <laughs>

Fagan: That is so funny.

Proulx: Isn't that funny? <laughs>

Fagan: That is so funny. So that must've impressed you at least that maybe this guy knew something or was different in some way.

Proulx: No. The shirt didn't impress me one way or the other. It was just remarkable. But I liked Scott. He was clearly a smart guy. You can size people up quickly from talking to them. I could see he was sharp. I could see that he had spent some time already thinking about this idea, and like I said, he had already done a little bit of research. We did a lot more later on, but he had already done a little bit of research and kind of had figured out in broad brush strokes what the product was going to be. And it really resonated with me.

So he came to Stanford looking to hire a programmer. I told him I wasn't interested in being hired. I told him I'd be interested in partnering with him. That's what we ended up doing, and that was great. He bought me a PC. You know, one of the early IBM PCs. It was like this big and a big green screen and two floppy disk drives. I brought it to my dorm room so I could actually start working on it while I was still at Stanford. So I believe that that was the first PC on the Stanford campus in a dorm room, that one that Scott bought me. This was in 1983. And I started working while I was finishing up.

And then that summer, I had to go work at Hughes Aircraft. Still at this point we weren't sure what this was going to be. Was this a company or was this a project or was this a crash and burn? We didn't know. So I was still working at Hughes. They had paid for my school. I was going to keep working for them until I knew. That summer I was working 40 hours a week at Hughes Aircraft and then I'd come home and I'd work all night long on Quicken. We didn't settle on a name until later. The working name at the time was Kwikchek. K-W-I-K-C-H-E-K. Kwikchek.

Hancock: Very cute with a K.

Proulx: Yeah. Exactly. That was a working name just because we had to call it something. I would work all night long and then I'd go back to Hughes Aircraft the next day and try to do my job there, but I was so sleep deprived. I had all these strategies for getting through the day. Sometimes I would have to go to the restroom and I'd go into a bathroom stall and I'd do this and take a nap in this position, which you can actually do. And I rode my moped to the office. I didn't have a car, but sometimes I would just borrow the car of one of my officemates and go out and take a nap in the car at lunch or something. <laughs>

I remember at the end of the summer we used to have these brownbag lunches where you would all go into a conference room and everybody would bring their lunch, and then one of the engineers would give some technical presentation on some interesting topic. So I'd go to these. At the end of the summer when I was getting ready to go back north-- I thought I was going back to Stanford. Turned out I didn't go back to Stanford. But as I was winding up my summer job, they had a going away party for me. One of the guys there said the thing he's going to miss the most about me is watching me struggle to stay awake during these brownbag lunches. I didn't realize I was the source of entertainment. People watching me do this all day long.

So the summer ended. I came back up here thinking I was going to go back to Stanford and finish up my master's degree. I talked to Scott and we were not done with Quicken, Kwikchek. We still had a lot of work to do. It became clear that I really couldn't do both, and I had to make a decision. So the first thing I did was I told Stanford, "Hey. I've got to defer. I gotta take a quarter off because I'm busy." And they said, "Okay. We can only do that a couple of times." I said, "Okay. Well, push me back." Then winter quarter comes around. I called them up and said, "I gotta defer again." And, "Okay. Well, that was your last time." And then spring quarter came around. It was clear this was a company, this was a big deal. There was no going back to Stanford. I was going to have put all my eggs in the Intuit basket, which I was excited to do. So I called up Stanford and I said, "Sorry. I can't do it."

But I had one more class I had to take. I had enough units to graduate, but I had one required class for my undergraduate degree that I hadn't yet taken. So spring quarter of 1984, I had to go back and take this one class three afternoons a week. I'm going to tell you the reason. There's a reason I'm telling you this story. So this class was called "Using the Computer as a Laboratory Instrument" and it was where I learned assembly language programming. They taught us on HP microcomputers that were connected to various lab instruments for taking measurements and controlling devices and things like that, and that was all done in assembly language. So that's where I learned assembly language, which was really valuable.

But the most valuable lesson that I learned was on a project that I was working on. I was in the lab and I had this weird intermittent bug that I just could not figure out. I was just tearing my hair out about this bug. It's intermittent. It's so frustrating. I'm looking at the code. You know, you print it all out and you're just tracing through the logic. I am absolutely convinced that the code is right but yet it's intermittently crashing. And while I'm working on this thing, the professor comes into the lab just randomly and said, "Hey. How's it going?" I said, "It's going horribly. I've got this intermittent bug. I cannot figure this out." He said, “Oh. Let me take a look” and he looks at my code and it’s all fan folded <laughs> wide sheet and he’s looking down. He says, “Huh. That looks right. That should work.” I said “I know, but it’s not working.” He said “Well, what’s happening?” I described what’s happening. He says “Hm,” and he goes over to these little microcomputers and he pops the lid off. He takes the interface card out, he takes a pencil eraser and erases across the gold contacts, plugs it back in. He says, “Try it.” Tried it. Problem solved. Totally works.

Now the reason I tell you that story is because a few years later we’re working on the Apple II version of Quicken. We first launched with the IBM PC version and then a few years later we were working on the Apple II version. And guess what? We had this really weird intermittent problem. <laughs> This kind of similar sort of weird crashing intermittent problem that we could not figure out. Rather than spending days or weeks or whatever trying to figure this out, I thought “Hm. I wonder.” I opened up that Apple II computer and I pulled out the cards and I took my eraser and erased off the contacts and plugged them back in, and voilà. Bug solved. No bug. I mean, if I hadn’t learned that lesson in that class, I don’t know what we would have done. It never would have occurred to me ever to do that. Ever.

Fagan: No, because the thing you learn is that the computer is always right.

Proulx: Yeah. Yeah.

Fagan: <laughs>

Proulx: Yeah. Right. My kids laugh about this. I have a rule. I say, “Tom’s number one rule of computers: computers don’t work.” I said if you would just remember that rule, you’ll never be too far in. You’ll never get in too much trouble. What it really means is, you can never assume anything works until you prove to yourself that it works. Whenever I would write a new piece of code, I don’t care how simple that code is, I’m just pessimistic. There’s something wrong with it until I’ve tested it and made sure that it works, I just don’t believe that it works. Because you never know. Even though these few lines of code might be trivial, you add these few lines of code and they move memory around that could cause some other thing that wasn’t broken before to break. I mean, stuff happens. So you just never assume anything works until you’ve proven to yourself that it works. And that’s my number one rule of computers. Computers don’t work.

Hancock: Some irony coming from you. So tell us a little bit more. You’re working hard now on the software. Were there certain breakthroughs? You’re in this long process of development. How were you making decisions on what you were going to include? What your approach was? You’re really essentially creating your own--

Proulx: Yeah. We’re creating this approach from scratch. Basically what I knew about writing software was what I had been taught at Stanford, which was pretty good. But not for huge projects like this. Everything at Stanford was on the mainframe at the time, so you didn’t have to worry about memory problems or anything like that. Of course in those days on PCs that was a huge issue for us.

We launched Quicken version one. Originally, our target machine was a 128k machine and we tried our damnedest to make that work. We just absolutely couldn’t do that. So then we kind of let our belts out to a 256k machine. That was obviously still really a challenge to get that to work. You could not fit the entire program in memory on a 256k machine. So there was all kinds of funky stuff that we had to do, swapping pieces of the program in and out of memory and stuff. Stuff that you just don’t have to worry about today. It’s all done for you by the operating system. But in those days we had to do all that manually. And of course it could create all kinds of bugs if you brought code in over existing code that needed to still be there. So there was a lot of challenges there.

But going back to the beginning, the language that I was taught at Stanford and loved and would have liked to have used was a language that nobody’s heard of anymore called Pascal. It was meant to be a teaching language and it was great as a teaching language, but it was a high level language and I really liked it. You could write very very elegant, pretty code, which I liked.

But Scott’s wife, Signe, was working for a software company called Software Publishing Corporation, SPC, and Scott had talked to Fred Gibbons who was the CEO and told him about this thing we were working on. Fred got the misimpression that this was, “Oh, just a little checkbook program.” So he said, “Oh. Well, you should just write that in BASIC.” Which if it was a little checkbook program, yeah. You’d just do it in BASIC. But for what we were doing, BASIC was really a bad decision <laughs>.

But I didn’t know that at that time. I mean, I knew BASIC. I knew its limitations. I thought, “Boy, this sounds like a really bad idea.” But Scott said, “Yeah, but Fred Gibbons, the guy knows. And all of their programs are written in Pascal.” And so I said, “That’s what I would like to do.” But here’s this guy who is an expert, who if anything should say, “Yeah, you should do Pascal,” and he’s saying, “No, use BASIC.” So, I said, “Okay. They must know something we don’t know.”

So I deferred to them and we wrote it in BASIC. Boy, what a disaster <laughs> that was. We started out in interpretive BASIC. The interpretive BASIC is what was built into the IBM PC and you couldn’t write anything of any size in interpretive BASIC. It was really slow and cumbersome. But we started down that path and then by the time we realized this is just a horrible mistake, really what we should’ve done is just thrown the whole thing away and started over in Pascal. Probably Pascal would’ve been the right language at that time. But we didn’t want to throw everything away. So I was looking around, and I found that Microsoft had this compiler for BASIC. So that’s what we ended up doing. We switched over to using a compiled BASIC, which was better but it was still not great.

But that’s when Quicken version one came out running on DOS-- because this is way before Windows-- running on DOS on the IBM PC in compiled BASIC. We finally launched in October of 1984. So it started out in March of 1983. We thought it was going to be a few-month sort of project. <laughs> It took until October of 1984 when we finally launched it, and that was a big deal when we finally launched.

Hancock: How were you surviving financially? There’s pressures for your time but also money.

Proulx: We didn’t have a lot of money. Scott had put in some money. We had a little bit of angel money from his dad. Somewhere in that time, we hired an operations VP, a guy by the name of Tom LeFevre, and his former boss put some money in. I don’t remember how much total cash was invested in the company, but it’s on the order of probably two or three hundred thousand, I would say. Not very much. But Scott and I weren’t taking salaries for much of the time, and when we were it was very minimal salaries, and we didn’t have a lot of employees. By the time we launched, I think we had grown to seven people. So, we didn’t have a lot of money. The plan had been to raise venture capital.

As I mentioned, Scott had come from Procter & Gamble. We were applying a consumer products approach to a technology product, to a software product, and basically just doing exactly what Procter & Gamble would do if they were entering this category. And that means starting off with research. So before we wrote a line of code, before we designed a single feature of the program, we did what Procter does, and they really understand the market cold.

We did this by telephone interviews of people. We would call people just randomly out of the phone book and we said we were college students doing a survey so that they wouldn’t hang up on us. And we would ask them about, “What do you do with your personal finances?” There were two things that we were looking for. First, we were looking for, what do people actually do with their personal finances? There were a lot of things you could come up with, and we came up with a laundry list of features by looking at the competitive products that were on the market at the time. Quicken was not the first in the category by any means. There were a lot of other products.

Hancock: It was a crowded space.

Proulx: It was a crowded space and we looked at the feature lists of all of these competing products. So we had this laundry list of things you could do and we would ask people, “Do you do this? Do you do this? Do you do this? Do you write checks?” A hundred percent of the households we talked to said yes, because back in those days that’s how you paid bills. “Do you maintain a check register?” Most people did. Not all, but most people would maintain a check register so that they knew how much money they had so they didn’t bounce checks. “Do you reconcile your checkbook register against your bank statement?” Most people, not all, fewer, but most people. “Do you produce any kind of reporting?” Yes, most people occasionally, at least at tax time, wanted to find out how much they spent on charitable donations for the year or their mortgage interest or whatever. So, those four things, most households did.

“Do you produce a household balance sheet?” Zero. “Do you graph your household expenditures?” Zero. All of these more esoteric features that these other products that were on the market were focused on, we found very few, if any, people did them. So, there were the big four and a lot of features that nobody did. So we said, “Okay. Those are the things we gottado.”

The other question that was super important for us to understand, again from Procter, was what’s the key benefit? What’s the single improvement that people are most looking for when it comes to their personal finances? What we found overwhelmingly was, time savings. Eighty-five percent of households told us they just wanted to save time. They hated doing this. It was a chore. If they could get rid of it altogether, they would. But you can’t. You gotta pay your bills. So therefore, they just want to get it done as fast as possible.

And that defined Quicken for us. We knew we had to focus on doing these four tasks and we had to do it faster than anything else. Faster than anything else meant faster than these other products, but it really means faster than all alternatives. What we came to find out was the fastest alternative that we had to beat was doing it by hand. All these other competitive products, in fact, took longer to do these routine sorts of tasks than doing it by hand.

And we brought users of these other products into our office and we would give them a stack of routine tasks to do. “Here, pay these bills. Reconcile this bank statement,” et cetera. We would time them with a stopwatch. And then we would have them do it by hand, and it always would take them less time to do it by hand than to use their software to do it.

So the objective for us was we had to be faster than doing it by hand. That was a really tough objective because doing it by hand is pretty darn easy. It’s simple. It works. You don’t have crashes. You don’t lose data. I mean, the pen is a very, very good instrument, and it’s known so there’s no learning curve. So, all of these advantages that the pen had over us that we had to overcome. And we didn’t get it right the first time. It was an iterative process where we would design things. We would bring people into our office. We would time them. Have people do the same testing with a stopwatch on Quicken and by hand as we had the people doing the competitive testing. We would find where we weren’t good enough and we would then tweak the design to make it quicker and easier.

We also had to make it not just faster once people knew how to use the program, but it had to be fast and easy the first time they used it. Because if there was a big, long learning curve, people would just give up. They wouldn’t invest the time to do it. You’re not going to save people time if it takes them a huge investment of time to get to the point where they can save time. So we had the objective that they had to be successful with it <snaps> first time they used the product. And that meant some very big differences in how we approached software to the way that it was done by everybody else in those days. In those days, software came with manuals and you would learn to use the software. We thought, “That can’t be. That isn’t going to work.” Our software had to be designed intuitively enough that the design itself communicated how you would use the product. We take all this for granted today, but in those days it was completely groundbreaking.

Hancock: That was new.

Fagan: That’s what I was thinking. This was radical thinking of the time.

Proulx: It was totally radical and totally obvious in hindsight. But at the time, nobody was doing that.

Hancock: It was brand new.

Fagan: Did you and Scott line up naturally?

Proulx: Yes.

Fagan: That it had to be that way?

Proulx: We were totally in sync on that.

Fagan: Wow.

Proulx: It was obvious to us and the name of the company, Intuit, was obvious to us. I think Scott came up with the idea for the name Intuit, but he had a list of names. “Hey, what do you think about this one?” When he said Intuit, I said, “Yep, that’s it.” That was, to me, an obviously right answer. Quicken, different story. But Intuit, the first time I heard it, I was, “Yes, that is it.”

Hancock: Because of this notion of, it has to be intuitive?

Proulx: Yes, the software itself, the design should be intuitively obvious. So some of the things we did-- the main data entry screen, I laid it out to look like a check.

Hancock: No easy task with those old-style monitors, right? <laughs>

Proulx: Yeah, well, the IBM PC had these ASCII graphic characters. You could draw lines and corners and stuff. They had this extended character set so you could do it. So I did that. Whereas competitive products, the data entry would be “Payee: date: amount:” kind of in a vertical list like a database entry screen, because essentially that’s what this is. A special-purpose database product is what it is, right? We all had to enter the same data. I just rearranged them in a way that looked like a paper check, so it would say “date” up here and then “pay to the order of” and you’d type in the payee’s name. And then dollar sign, and you’d type in the amount, and then it would write out in words, you know, Twenty-seven and 93/100, and a bunch of asterisks to fill it out. Just like you do by hand, and it was a memo field where you could type what you wanted in the middle of it. So as a computer science exercise, it was trivial. There was nothing to it at all. But as a design exercise, this was this huge idea that nobody had done before.

Similarly, the check register, or the database itself where you would go back and see all of your historical transactions, I designed it to look like a paper check register. I remember I had my checking account with Wells Fargo Bank at the time. One afternoon I opened up my check register <laughs> from my checkbook and I literally copied it. Where they had double lines, I put double lines. Where they had single lines, I put single lines, and our checkbook register looked identical to the Wells Fargo check register. So those were a couple of things that we did that made it obvious to people how to work with the program.

Another thing we did, though-- prior to the Mac, and prior to Windows, there were no universally standardized user interface conventions for the PC. There were certain things, like “tab” would advance you to the next field and things like that. But in terms of how menus worked and what the different function keys would do, none of that was standardized. Every program wrote their own user interface. And we decided that makes no sense. Why reinvent the wheel? What makes a great user interface, obviously there are certain things that you want to do. You want to be consistent so that if the F1 key is the “help” key, it should always be the help key; not the “help” key on the main menu and on other screens it’s some other key. So there’s some obvious things you do like that.

But the main thing that makes a user interface intuitive is simply familiarity. I already know it. That’s why the Mac is so great. It’s not that the Mac is an intuitively obvious user interface. I mean, when you think about the Mac and using a mouse to pull down a menu, who would ever think to do that if you just sat them in front of it? Nobody would ever think to do that, other than, “Of course you do that because you’ve learned to do that and you do it on everything and they all work the same.” So we thought, “Why would we reinvent yet another interface? Let’s look at what’s out there, where there’s a big user base of a product that’s well done with lots of users.” Because to the extent that those users end up using our product, they’ll already know it. And if it’s a new user that hasn’t used this other product, at least we’re copying something that’s tried and true.

So we looked around and decided that Software Publishing had done a really good job of that. They were writing simple mass-market applications. What we were doing in our space, they were doing for a simple word processing program, a simple database program. And they had a consistent UI/UX across all of their products. So we just ripped them off, and we just <laughs> stole all their good ideas. It was absolutely the case of imitation is the sincerest form of flattery. It was exactly what we were doing. We said, “Yep, you guys did a great job. We’re going to copy it,” and that’s what we did. So, F1 was the help key. F10 was the key to confirm the whole screen. “Enter” moved you between fields. That kind of stuff. That’s what they did and that’s what we did. And that was something that I don’t think anyone else had done either, because everybody wanted to reinvent the wheel and put their individual stamp on it. We said, “No, no, no. We just want to make it easy on users, and so we’re going to copy the best of what’s out there,” and that’s what we did.

Fagan: So when the organization starts to grow and things get successful, and now you have multiple operating systems, what were the biggest challenges? How did you navigate through that? And how quickly did the engineering team grow?

Proulx: Well, there’s a long time between those early days and <laughs> when we got to that point. Let’s see. So we first came out with the IBM PC version. We did an Apple II version next. I don’t even remember all of the versions. At some point, we went back. We had to redo the PC version. We threw the whole thing out and rewrote it in C because the compiled BASIC version, there was just no way we could live with that. We started that over. So Quicken version two was written in C, which is a great language, I love C.

And we grew the company. We grew the engineering team. At some point, I think it was 1988, we did a really horrible, horrible project for a new operating system called DeskMate. You’ve never heard of it. But Tandy Corporation, which was the company that owned RadioShack, they had decided they were going to enter the PC market with a low-cost PC. But they were going to make it like the Mac. They were going to make it have a graphical user interface and a standardized user interface across all applications. And that was how they were going to compete against IBM, and Dell, and all the PC clone makers of the time.

So they came to us and said, “Hey, we’d like you to port Quicken to DeskMate.” I think they guaranteed us a million dollars' worth of sales through purchases of DeskMate versions of Quicken if we did this. So we agreed to do it. I think partly it was the million dollars and partly it was just because they were a big company and we thought, “Yeah, maybe this could be the next big thing,” and we wanted to make sure we were on it. They had gone to other companies and other companies were doing the same thing. So, there was a little bit of, “Well, if they’re doing it, I guess we should do it too,” and so we did <laughs>.

And it was such a disaster. It was an underpowered machine. It was buggy. It was poorly documented. It was just a nightmare. The DeskMate version came to be internally called Death Mate and it was just killing us. The lead engineer on it was Eric Shenk and he would just call his mom at night and say, “I can’t do this anymore.” <laughs> I would come into the office in the morning and he’d be sleeping under his desk. We would do that because the overhead lights would be on, it was too bright to sleep. So you just go under the desk where it was at least a little bit darker and get a nap under there. Poor Eric. That’s what he did. I actually don’t remember whatever happened with the DeskMate product. I believe we finally shipped it. But it was just an awful, awful project.

And I remember in the middle of all of that, I was also under a lot of pressure too. I was working with Eric on this and we had a deadline. We had to get it done. And I was also scheduled to go on a trip with my mother to Germany. My mom’s from Germany. It was like, “I just got to get this thing done,” and I’m working around the clock to try to finish thing up and I can’t get it done, and I can’t get it done. I’m just feeling so much pressure. And one day I come into the office in the morning and all of a sudden I’m feeling sick to my stomach. I go to the bathroom and I throw up and all this coffee comes out. All my morning coffee comes out. Everything. I go back to my office and feel a little better. Then I feel bad again. I go back and throw up again and more coffee. After a while, I’m thinking, “I didn’t drink that much coffee. Where’s all this coffee coming from?”

I called Ginny Boyd up. Ginny was working at the company. She’s my sister-in-law today. At the time, she was my girlfriend's sister, and she was the fourth employee of the company who was in charge of all of our operations. And I said, “Can you take me home?” I think I rode my moped to work and I didn't think I could make it home. So she went and got the car, and I met her in the parking lot. And as she was bringing the car up I threw up again in the bushes. She says, “I think I ought to call some help,” and I said, “No, just get me home.” So she took me home and I threw up again <laughs> and I was going to the bathroom and I just can’t get relief. She said, “I think I should call an ambulance.” I said, “I’m fine. Just leave me alone.” She finally just ignored me and called the paramedics and they came. It turned out I had a bleeding ulcer and by the time they got there, I had lost a third of my blood they said. So they put me in this suit that they could inflate to keep my blood pressure up and they took me to the Stanford Hospital. That was a close call, all because of Death Mate. <laughs>

Hancock: It was aptly named.

Proulx: Yeah, it was aptly named. And I remember I’m in the hospital, but I got to get this thing done. I asked Scott and Eric, “Can somebody bring my PC in so I can get some work done?” They refused to do it <laughs>.

Fagan: I was going to say, I hope they said no.

Proulx: Yeah, they said no. So, anyway. I didn't go to Germany and I guess my mom forgave me. <laughs> And at some point, we did finally ship the Death Mate version. But it never blew up, which is why you've never heard of it. I think the only sales we ever got were those "guaranteed million dollars of sales," we never got another one.

Hancock: We've talked about some of the difficult challenges. How about high points? Days to celebrate? Something when everything came together?

Proulx: So, we glossed over-- there were some dark days in there where we ran out of money and it was bad.

Hancock: You can say something-- it's running on fumes, if you want to talk about that before we get the high points.

Proulx: Well, we had originally planned to raise venture capital and we had all of this research that I mentioned and Scott had all these slides made up. This was before PowerPoint. We had these acetate slides that he hired somebody to make. They glued down little colored squares to make bar charts and stuff like that. But he had all of this data to go into our presentation with VCs to show them, "Look, we surveyed people and 85 percent of households want to save time, and look how much longer it takes to do this," and blah, blah, blah. And we just utterly failed. We never got even a second meeting and we talked to just about every VC on Sand Hill Road.

In hindsight, it's not surprising. Neither of us had any experience. Scott is a few years older than I. He had at least a little bit of business experience as a consultant, he had worked at Bain & Company. Before that he worked at Procter & Gamble. But he wasn't ever working on a technology product. At Procter, he was the brand manager for Crisco shortening. So, he was a fat salesman. And <laughs> then I was a bright kid out of Stanford, but what do I know about anything? So there was no way we were going to get funding, and we didn't <laughs>.

But that was kind of key to the plan was getting funding, and we were going to do this huge marketing launch with lots of advertising, and again following the Procter & Gamble playbook. But without the cash to do it, we couldn't do that. So we had to get creative, and what we ended up doing was selling Quicken through banks. Scott, while he had been at Bain & Company, one of his clients was Wells Fargo Bank and so he had a relationship there. In fact, in the early days of Intuit, after he had left Bain & Company, to get some cash in he continued to do some consulting for them on the side and maintained the relationship there. He showed them early versions of Kwikchek in those days and got them interested. He proposed a deal to them where they would sell Quicken to their customers, and that's how we launched it.

So in October of 1984, we launched Quicken through Wells Fargo Bank. And that was the plan. So they had this big press conference for us up in San Francisco and that was really exciting. We were on the news and it was really exciting. And then a couple of months later Bank of Hawaii did the same thing. We did a similar deal with Bank of Hawaii and we thought, “Wow, this is working.” We’d get this big initial order because the banks would, at least Bank of Hawaii was actually selling Quicken out of their branches. Wells Fargo hired this 800-fulfillment service to do it. They didn’t do a very good job with the launch and fulfillment. But Bank of Hawaii did a great job. They put a few units of Quicken in every branch of Bank of Hawaii throughout the state and throughout Guam and American Samoa -- everywhere where they were. So we got this huge order. And they had this, I think, assistant vice president at the bank who was just a go-getter and did a great job. His name was Tim Farr and he did a great job of training the branches on how to sell Quicken. They did a great job.

And in fact, in 1985 Quicken was the second-largest selling software in the state of Hawaii. Second to Lotus 1-2-3 which was the big spreadsheet program before Excel. First, there was VisiCalc, then there was Lotus 1-2-3 that ate up VisiCalc, and then Excel came along and ate up Lotus 1-2-3. But in those days, Lotus 1-2-3 was huge, and we were second only to them solely by virtue of Bank of Hawaii and Tim Farr's efforts there. So, we had this formula. Wells Fargo didn’t work so well but Bank of Hawaii did great, and we said, “Okay. All we got to do is replicate Bank of Hawaii in the other 48 states and we’re going to be great.” So that was the plan.

We started doing that and Scott would go out selling to these banks. The problem was that we could never replicate Bank of Hawaii and we tried to get them to do all the things that Bank of Hawaii had done. But what we later discovered and figured out was it wasn’t so much the specific things that Bank of Hawaii had done, it was Tim Farr. That was the magic formula. And no other bank had a Tim Farr, so we could just never get other banks to be successful with it. We’d get the initial order but there was never a second order, they would never sell through. And we got to the point where we just couldn’t, in good faith, keep doing this because we just knew the banks weren’t going to be successful with it. So we ran out of money.

We still had faith in Quicken because we had done all this research. We knew the product was great and the few customers that we had loved it. It reinforced that our research was right and that the product was designed right, and it was saving people time. We knew that we had a great product. We just had to stay alive long enough for the word to get out and for people to tell their friends. Ultimately, we believed we had to succeed. But we ran out of money. We had yet to put out our Apple II version and so we had to set some money aside to launch the Apple II version. And basically after we set that money aside so that we could launch the Apple II version, we didn’t have enough money to pay salaries at that point. So we had to tell everybody, there were seven of us, Scott and me and five employees, “Sorry, guys. We can’t pay your salaries anymore.” Three of the five people had to leave because they needed salaries to live. The two people who stayed were Ginny Boyd, my sister-in-law, and Susan Schlangen who had become a good friend and for whatever reason had the financial means to be able to afford to stay. She stayed out of loyalty to our friendship and wanted to see this thing through. I’m not sure of all the reasons that she stayed. But I know Ginny stayed because her dad told her, “You’re not going to let Tom down.” <laughs> She’s like, “Well, Dad. They’re not paying me anymore.” But he told her, “No, you’re going to stick this out and you’re not going to let Tom down.” It was a really great thing that she did because we couldn’t have done it with fewer than four people.

So we kept it going and just did what we had to do to survive and get that Apple II version out. Then when we did, a number of things happened. The Apple II version launch gave us a little bit of cash because we did have some banks who had given us orders for Apple II. So we got some initial orders out, got some cash in for them. Then we had three banks that we had been in discussions with for a long time, and all of a sudden they started signing up. So we got the initial orders with them and they gave us some cash. And then because we had an Apple II version, Apple came to our rescue. Apple loved us. The IBM PC was really a business machine, not so much a home machine.

But the Apple II was purely a home machine and they needed titles for home products. So they really loved what we were doing, and they wanted us to succeed and wanted to be helpful for us. They put us into Apple stores and they had some Christmas-time promotions that year where they had like “the 12 good reasons to buy an Apple II,” and we were one of the 12. So we started to get some sales of our Apple II product through that, and that’s where we started to turn the corner and started to get some cash.

So that was a turning point. It didn't completely bring us out of the woods. It brought us from the brink of bankruptcy to having some cash in the bank. And our expenses had gotten so low. There were only four of us, and we had done all kinds of things to reduce our costs. We had done a lot of things to keep our expenses down. My wife-- my girlfriend at the time-- was working for Crocker Bank, and Crocker Bank had just been acquired by Wells Fargo Bank. So all the Crocker stationery was getting thrown away. So she would go grab all this stuff out of the trash bins and bring shopping bags full of Crocker stationery to us to use so that we wouldn't have to spend money on paper or pens or things like that, or on stationery. So we would do things like that.

And back in those days you actually sold a physical product, so we had boxes and things like that, and we would use inventory as office furniture rather than using office furniture as office furniture. So for your printer stand you'd just put a box there and put your printer on it. So we just did everything we could to keep our expenses to an absolute minimum so that when we got this influx of cash, it wasn't a lot of cash, but it was enough that we were cashflow positive which was a huge turning point for us.

But it wasn't enough to cause the company to succeed, to blow up, and that’s what we were in it for. None of us was in it just to have a little lifestyle company that would kind of survive. We wanted to either blow this thing up and create a big company, which was what we were in it for, or let’s crash and burn and get on with our lives and do something else. But we were in this kind of steady state. We weren't going to do that. And so we started thinking about, “How do we get from the steady state to either this or this?” And we decided we've got to start advertising, we've got to spend money on advertising. Fortunately we had built up a little bit of a war chest. Like I said, we had built up to about $100,000. This is the summer of 1986 or ’87, something like that.

Fagan: A couple of years in.

Proulx: I think that’s right. And we decide we've got to spend some money on marketing and see if we can get this thing to go. The problem we were having, in those days software was sold through software stores, and software stores, retail stores, they want to carry products that sell. So you've got a real chicken and egg problem. They want to carry products that sell, but how do you sell if you're not into the stores? Somehow we've got to create demand to get into the stores. The traditional way that companies would do that would be a lot of advertising. We really didn't have a lot of money for advertising to do that kind of advertising, to create awareness among the stores so that they would want to carry us so then consumers would walk into the stores to find it. That just was going to take too much money and too much time.

So we started looking around to see if anyone else had cracked this nut, and there was a company called Borland International. Borland International had done a couple of radical things. They had a program called Turbo Pascal, which was a compiler, and they had done two things radically different than anyone else in the compiler market had done. Number one, the price was really cheap. I think the price was like $50 or something like that, versus hundreds of dollars if you bought a compiler from the big guys like Microsoft. And they were selling direct. They were bypassing the retail channel. They were running ads in computer magazines with their 800 number, “Call us direct and buy and we'll ship it directly to you.” They just didn't even care about the retail channel. We looked at what they were doing and said, “That’s brilliant, we should do the same thing.”

And so we had to decide, though, do we test it, see if it works, tweak it if it doesn't, get it right, and then expand it? Or do we just go for it? The reason that this was an issue is, as I mentioned, this was all happening toward the end of that summer of, I think it was ’87 or ’88, going into the Christmas season which was the big retail selling season. And we thought, “If we do the prudent approach, which is to try it, see if it works and then if it does, expand it. By the time we will have gone through the trial and decided that it’s working, we'll have missed that Christmas season.” So we thought, “Maybe we should just go for it.”

And by going for it, what that means is first of all, cutting the price of the product. We were previously selling for $99, and we decided we should cut the price in half to $49 like Borland had done. We also removed copy protection from the program. That was a big thing back then, programs were copy protected. Users hated it, it was a pain. But that was a big risk because if you didn't copy protect the program, people could install it on their machine then give it to their friends and they could use it. We worried that people would just steal it basically. But we thought Borland didn't do it, and if the price is cheap enough maybe people won't bother.

Finally the big thing was the direct response advertising, running these ads in magazines where we would put our 800 number for people to buy from us directly. Advertising is a little bit of a black art, it’s hard to know what’s going to resonate with consumers or not. That was really the big thing that you'd like to do is test the ad, make sure it’s working before you really heavy-up on that ad. But by the time we would have tested an ad, read the results, and decided that it’s working, we would have missed the Christmas season.

So we decided to take the gamble. We had $100,000 of cash in the bank. We signed up for $120,000 worth of advertising over several months in all the major PC magazines over that Christmas season. Scott and I and Ginny and Susan wrote the ad. Scott took the first stab at it and then we all edited it and edited it. We wrote those ads and, “Okay, let’s see what happens,” and we ran those ads, and the phone started ringing.

Fagan: I was going to say, you must have been the same four people answering the phone too.

Proulx: The phones started ringing and they started ringing off the hook. Those ads, the headline was, “End Financial Hassles,” and it worked. The ad was very heavy with content, right? It wasn't an image kind of ad like Intuit is doing today. This was all text, and it had to be all text because we had to anticipate and answer all of the questions that the consumer would have. If we didn't answer their question there, we would have lost the sale. They couldn't go into a computer software store and ask the salesman to answer the question because the salesman had never heard of us, and so we had to anticipate and answer all the questions there.

So we had our 800 number, we had a coupon in there that they could clip out and send in with a code on it, and the code told us what magazine, what month so that we could identify what ads were working and what ads were not and what publications were working and what were not. And when people would call us on the phone we would ask them for that code. So we were really always very analytical about knowing exactly how efficient our marketing dollars were going. What we learned was, the best ads and the best publications would pretty much double our money in 30 days’ time. So we would pay $10,000 for an ad, and 30 days later we would have $20,000 worth of revenue from that ad. That’s not a bad business. And that is when the company started to do that, and we did that for a long time. Those were the fun times. Those were the really fun times when the phone is just ringing off the hook and you're talking to customers, and we loved talking to customers too.

Even the support line, we loved talking to customers on the support line because when customers would call us with problems-- first of all, we loved talking to customers because it meant we *had* customers. [laughs] And we would learn so much from talking to people about things we thought were easy that they were finding not easy. Then we would incrementally improve the product. So it was always working to improve the product based on customer feedback.

In fact, from the early days we talked to customers because we had to, because there were only the few of us. But it was so valuable in terms of our product design intuition that for many years we had the entire product team, the engineers and the marketing people on a product team, were required to spend some number of hours per month on the support lines just like any other support person, talking to customers so that their guts would be as in tune to our customers as mine were in those early days.

Fagan: Yeah, that’s such a good practice. So, now the company’s taking off, and I know there's all kinds of things we're skipping over here, Tom. But Scott absolutely credits you with the idea of buying Chipsoft, and that was a game changer for the company. So talk about why, what was your inspiration?

Proulx: I think I discovered tax software because my wife and I always did our own taxes, and it was a painful task. I realized this was something that was absolutely a task that is better done on the computer, and--

Fagan: How did you and your wife do your taxes?

Proulx: By hand in those days. You'd get paper forms and you'd fill them out--

Fagan: Fair enough.

Proulx: And it would say multiply line 10 by--

Fagan: Fair enough.

Proulx: -- .103 and you'd do it and--

Fagan: I forgot about those days I guess.

Proulx: Remember? You can actually still do that.

Hancock: Yes, you can.

Proulx: You can go to the post office and get paper forms to do all that.

Fagan: My dad did it that way. Yes. I remember.

Proulx: That's how we used to do it, but it was clearly a bad way of doing it. And you know, to me it was such an obvious thing for us to do. It was such an obvious fit with who we were. All of our customers would have to pay taxes just like they had to pay bills, so 100% overlap of our customers. And there could even be some advantages to having the data in Quicken and later QuickBooks for your small business that we could just import it into the tax software. So to me it was so obviously a great next business for us, and I started kind of wanting to work on this. In the earliest days I was our head programmer, and then as we got bigger I was one of the programmers, and then I was V.P. and chief technical officer and began managing the program. But I realized when I got to that point, it wasn't as much fun for me, number one.

And number two, I saw these big opportunities, tax being at the top of the list, I thought that we weren't going to get to because nobody was working on it because we were so busy. I said, "Look, we can hire somebody to run engineering," and it had gotten to be a big organization. "We can hire somebody who I'm sure is going to do better than me just because I'm learning as I'm going. We can find somebody who's done it before, and then I can go do this." So we did that, we found a guy by the name of Steve Pelletier and brought him in and he took over for me in engineering. I took over the newly formed business development, which was about getting us into some other businesses. The biggest one was the tax business that I wanted to do.

And I don't remember where, Eric and I both met this guy Dan Caine, and he had a little company called Legal Knowledge Systems. He was a tax lawyer turned software engineer, and he had developed this program called Ask Dan About Your Taxes. It was a great little program, and it was exactly the kind of program that I envisioned for Intuit, which was not just a spreadsheet that would do the calculations for you but it actually had advice in it as we do today within TurboTax. But at the time that was new and radical, no one had done that before. It interviewed you, it asked you questions and then filled out the forms for you, which is to me exactly what I had in mind.

Hancock: Intuitive.

Proulx: And he had done it, right? So I thought, we ought to acquire this guy. And he really liked me, I really liked him, and he liked Eric-- I think we met him at some of these software conferences and stuff, and we just hit it off. Eric and Dan and I were just birds of a feather. We were all the same kind of guys. We just really liked each other and wanted to work together, and really to me it was just an obvious fit.

So I kind of lobbied internally that we should do this, and Scott was against it. Scott was absolutely against it, and I pushed hard on it. The reason he was against it was because he couldn't figure out how we were going to do tech support. Operationally, he said, “You’re going to have this huge tech support burden that goes to zero on April 16th. How will we possibly staff up for that and then get rid of all these people on April 16th? How can we do that?” My response to that was, “I don't know, we'll figure it out, we're smart guys. Other companies that are not as smart as us are doing it. We'll figure it out.” But Scott didn't think we could deliver world-class customer service if we got into the tax business, and therefore I couldn't convince him. The way we made decisions in those days was it had to be unanimous. If any one of us didn't want to do it, we weren't going to do it.

Fagan: Were you the leadership team?

Proulx: Yeah. We had an executive committee and, you know. But basically, yeah, it was the leadership team. And I just couldn't get Scott-- everybody else wanted to do it. Scott wouldn't do it, but you had to be unanimous and--

Fagan: And he was the CEO, right?

Proulx: And he was CEO, but--

Fagan: And this was what year? How far in were you now?

Proulx: God, I don't even remember. I think this would have been-- we ultimately did the deal finally in 1993, so this would have been probably 1991-ish I would guess, something like that.

Fagan: Okay, and you did the deal with Dan?

Proulx: We'll get to that.

Fagan: Okay.

Proulx: So anyway, I had to call up Dan and say, “Dan, I’m really sorry but we can't do it.” Dan, though, needed to get a deal done because he wasn't a marketing guy. He had a great little program and a great engineering team, but he needed a way of selling it. And since we turned him down he did what he had to do, which is he went to our competitor. We had a competitor at the time that actually mattered to us, a company called MECA. They had a product called Managing Your Money, Andrew Tobias’ Managing Your Money, and they were a big competitor of ours. They were ahead of us to begin with, and I think by then we had passed them up, but they were still somebody we cared a lot about. And they went to MECA.

So now what? Now our competitor has a tax product, so now we *have* to get into the tax business. [laughs] And so how are we going to do it? We didn't want to be in the tax business, or Scott didn't want to. But now we have to be in the tax business because MECA is in the tax business. We can't let them have the advantage. So we looked around and said, “Well, there's this company down in San Diego called Chipsoft, and they have a tax product. It’s kind of shitty but they’ve got market share and we can help them.” So we went down and met with those guys-- and I say “kind of shitty” because I mean, you know, that’s overstating. But it didn't have the friendly interviewing feature, it was just a glorified spreadsheet. So it was not an Intuit kind of product.

And so we allied ourselves with them. We didn’t acquire them, didn't want to acquire them, but we started partnering with them and doing bundles with them. So we would bundle TurboTax with Quicken, and these would be sold in retail stores and you get the bundle. From the consumer’s point of view, it was a good response to MECA, and they changed the name to TaxCut, MECA Managing Your Money and TaxCut. So that was bundled, and we had TurboTax and Quicken. TurboTax had more market share, not as good a product but better market share. Then with our marketing help we really launched them, and we would send our marketing people down to teach them how to do marketing as well. So we really invested a lot in that relationship, and they became the dominant tax software, we made them the dominant tax software.

Fagan: But as a partnership.

Proulx: As a partnership. And so we were just basically [laughs] building the price that we would ultimately pay for them.

Fagan: I was going to-- yes.

Proulx: So I’ve always said that clearly, we ultimately got into the tax business, it was a great thing for us to have done. But boy would it have been a better thing if we had done it in ’91 or whenever it was with Dan Caine’s company. We would have put TurboTax out of business because all that marketing help that we did to them would have been directed to our own product, which was the better product. MECA would have been out of business, and ultimately did go out of business. And I think Microsoft wouldn't have even tried to get into the business because it would have been too hard for them. When they tried to come into the business they only had to compete with Quicken, and if we had Quicken and TurboTax or, you know, Quicken Tax [laughs] together I think it would have been too hard for them to have come in.

So I think we would have ended up having a very dominant position. I mean obviously the company did extremely well with an extremely strong position, but even more so.

Fagan: It would have happened sooner for sure.

Proulx: Sooner, and even more so.

Fagan: So now we're talking about ’93. You're still--

Proulx: So in ’93 we decided, finally Scott was convinced we’ve got to do this, we’ve got to buy those things. We're just building up all this value in somebody else, we ought to get the value for it. So we had to do a couple things. First thing we had to do is we had to go public, because we needed a public currency to be able to buy these guys. We could have done it as a private company, but then you're getting into, “Well, what’s their stock worth, what’s our stock worth?” So we said, we're going public. When we want to know what our stock’s worth we’ll look it up in the *Wall Street Journal*. That’s what it’s worth, right. The market will tell us what it’s worth, so it takes that away.

And so, we decided that was step one. We had to do that, and we wanted to do it very fast. So I remember we had a board meeting in, I don't know, December of ’92 or January ’93, where we finally decided, “Yep we're going to do this.” Then a couple days later we hired our law firm, we changed law firms to go with Fenwick & West. Gordy Davidson was the partner there who John Doerr had recommended. And we decided we were going to do this, and we were going to do it as fast as we could. This was on the critical path for me to do what I wanted to do, so I dropped everything else that I was working on to jump into the IPO with Eric.

Eric was CFO at the time, so Eric and I tag teamed on getting the S1 put together and working with the lawyers and everything and drafting and all this stuff that we did. And we did it in record time. We brought some innovation to that process even, which I guess we don't have time for. [laughs] But Eric and I looked at the way things were done in those days, how the drafting is done, which was really an antiquated process. We said, “This is crazy,” and we brought some modern technology to that which sped the process up. And we did the whole thing from decision date to filing the IPO-- I think it was done in 60 days start to finish, which was unheard of at the time. And so then we went public and then came out of our quiet period. Then I started to have discussions with Dan and Chipsoft. I wanted both because I still wanted to have a dominant position in the tax business.

Fagan: But how were you going to get Dan? Dan had already been bought, right?

Proulx: Well actually, he hadn't been. What Dan had done was he had sold the marketing rights to MECA, but he retained the engineering for himself. He was still his own company and he still had the engineering product rights. So I thought “Well, I’ll buy Dan, I’ll still have these marketing rights to MECA.” [laughs] Well that’s kind of going to screw them up to have their competitor-- I would have never admitted that at the time, but that’s what I did.

So it's a long story, but we didn't actually have time to buy Dan's company. But what I could do very quickly, and in fact did it overnight one night, was I got a call option on his company and in return gave him a put option to us so that effectively he was locked up and the only way that he would not be sold is if both we and he decided not to go with us. That was something I could get done quickly, and we did it overnight one night.

And we did it overnight because I got word that the chairman of Chipsoft, Charlie Gaylord, was on his way, was flying from San Diego out to Massachusetts where Dan was to meet with him first thing in the morning. I knew how persuasive Charlie was, and I thought, "Charlie's going to lock him up if I don't get to him first." So I called up Dan, I said, "Dan, we're doing a deal tonight, and call your lawyer up, I'm going over to Gordy's office." I spent the night there, he spent the night with his lawyer, and we worked all the night. We finished the deal, signed it, faxed the signature pages back at 7:45 and Charlie was due to arrive at Dan's house at 8:00. So Charlie shows up and Dan says--

Fagan: Sorry, it's done.

Proulx: Sorry Charlie. He got a deal with Intuit.

Fagan: Sorry Charlie.

Proulx: So that's how we locked them up, and that was important. The reason Dan became so important is because we knew that we and TurboTax and Chipsoft wanted to get together, but whoever had Dan had a little more negotiating leverage [laughs] than the other. So that was what this was all about, because if we had Dan, then we could tell TurboTax, "Yeah, gee, we're going to—"

Hancock: Maybe, maybe not. Yep.

Proulx: "Maybe, maybe not. We can get into the tax business without you." And similarly if they had Dan--

Hancock: You would be locked out.

Proulx: We were locked out of the tax business. So Dan became the key negotiating issue.

Hancock: Fascinating.

Proulx: So anyway, we then did our deal with Chipsoft, negotiated that, came up with a good deal. It had to be approved by the Department of Justice though, because there's a filing called a Hart Scott Rodino filing that all big mergers have to go through because the Justice Department wants to make sure it's not creating a monopoly. Any time you're a big company or a public company acquiring another company, there's a mandatory review that goes on. So we had to go through this review. The DOJ was reviewing this and I believe they got input from a company called H&R Block that said, "You should not allow this to go through because it will create a lack of competition," and they were persuasive. I don't know for a fact that H&R Block did but I'm pretty sure it was them. And whoever it was prevailed on the DOJ, and the DOJ called me up and said, "Can't do it, you can't have both. You can one or the other, but you can't have both."

And this was really awful because I really wanted both. I really liked Dan's product, but by this time we had a lot invested in TurboTax. It had the greater market share by far, and we could not do that. So I had to tell poor Dan, I said, "I'm sorry but we're going to have to unwind the deal with you." Now I mentioned we had done a put/call with him. The only way to get out of that deal, he had to agree to get out of the deal because he could put it to us even if we didn't want it. So, we had to buy him off, and we did. It was still a great deal for us because having Dan, you know, it was great for him. He made a good return on his investment, which I feel really good about. And having that leverage in our pocket allowed us to negotiate a better deal for our shareholders with Chipsoft than we would have otherwise, so it was good for us. But I always look back and say, "Boy, if only Scott--

Hancock: If only at the beginning.

Proulx: -- had agreed to do this in the beginning, we would have owned the tax business entirely." And so when we paid Dan off to not execute his put, he then sold his company to H&R Block, which became--

Hancock: H&R Block, that's what they-- TaxCut, yep.

Proulx: -- their TaxCut product, and that product would have never become a competitor to TurboTax but for that.

Hancock: Well, something would have happened. All right. At one point do you exit the company? We got to get the short version of this.

Proulx: Oh, well, the short version is that in 1993, that was the year that we went public, I had this huge complicated three-way negotiation and the DOJ, and H&R Block and all of this going on. I was just constantly traveling, I had kids at the time, I missed Easter, missed birthdays, I was just-- we finally got it done. The Chipsoft deal closed in September, and we had a few months of transition and ramping up and getting integration. And then I just needed a break. I was so burned out and I just needed to take some time off, and so I took a sabbatical. I said, "I don't know how long I'm going to be gone. I'm just going to take some time off and recharge my batteries and get to know my family again. I'll come back when I'm ready."

So that was as far as my plan went, and so then I think that was like March of 1994. And I just never came back. I got so busy with some other projects, and then Bill Campbell joined the company. My biggest regret actually is that I didn't come back because I would have loved to have worked with Bill Campbell, and I never got the chance to do that. Bill wanted me to come back, but what he wanted me to come back and do is start the international business, which would have been--

Fagan: More travel, yeah.

Proulx: Even more travel and being away from the family and I just couldn't do it. And that was when I said, “You know, I think I’m just going to call it.” [laughs] I never came back at that point. So, I didn't set out to leave, but I just kind of faded away.

Fagan: Kind of like Stanford. [laughs]

Proulx: Yeah. Exactly. Yeah.

END OF THE INTERVIEW