# Oral History of Larry Tesler, part 3 of 3

Interviewed by:
Hansen Hsu
David C. Brock

Recorded February 17, 2017
Mountain View, CA

CHM Reference number: X8020.2017

**Hsu:** Today is February 17, 2017. And I am Hansen Hsu. I'm joined by David Brock and Mark Weber and we are with Larry Tesler, again, for part three of our interview. So a quick review of where we were last time. So you were part of the Lisa group doing user testing working with Bill Atkinson on the Lisa UI. Then helped develop an object-oriented framework, Lisa Toolkit on top of a new language Object Pascal. And then after the Lisa team merged with the Macintosh team you created the MacApp framework along the same lines as Lisa Toolkit. So how soon after working on MacApp did you help create the Advanced Technology Group?

**Tesler:** Well, just one quibble there, you talked about Clascal—

**Hsu:** Oh, OK. Oh, that's right.

**Tesler:** It was Clascal for the Lisa Toolkit. It was Object Pascal for MacApp.

**Hsu:** Oh, right. So they were two different languages. That's correct, yes.

**Tesler:** Yeah, they were similar, very similar. Object Pascal got the endorsement of Niklaus Wirth. And so—that's what we wanted. We wanted to get some adoption.

**Hsu:** Right.

**Tesler:** And so then the question is relative timing of the founding of ATG and the completion of the Lisa Toolkit or Object Pascal?

**Hsu:** Yeah. So I guess maybe just sort of continue the story from working on MacApp and Object Pascal up through to the creation of ATG.

**Tesler:** Right. They may have overlapped. In other words, I think the first release of MacApp 1.0 was probably before ATG was formed. But to me it kind of runs together. We were—before there was ATG, there was ADG, Advanced Development Group. And that was started by Wayne Rosing, I think. And then during a reorg—probably the one that he left the company in—it got moved under Ed Birss who was running a small part of that group and had run a part of the Lisa project. And that's B-I-R-S-S. <pauses> And so there were a few subgroups that were pretty small, the total size of the Advanced Development Group was somewhere between ten and thirty. I don't remember exactly where it started. I think it was closer to ten. And by the time we expanded it to become ATG it was more like the thirty people level. And so chronologically it was 1986 that I think MacApp was released. And I took over ADG. And then 1987 when they decided to invest more in research because they were worried that they were going to run out of ideas [that] they could get from other people like Xerox PARC. And they didn't have Steve Jobs around anymore to come up with ideas. So they, for some reason, decided that "Advanced Technology" was a better name, a more appropriate name for a technology innovation group than just "Advanced Development." But in practice I think we acted more like you might expect an Advanced Development Group to be acting which is we had a pretty short-term horizon. And there were always people who wanted to lengthen the horizons but I just thought that was unrealistic for Apple at that time. If we could make the product better by investing a few weeks or months in it in the Advanced Technology Group then I felt that was a better return on investment for the company than investing in something for ten or twenty years later. So eventually I structured it so that it was kind of 90 percent actionable in the next 2 or 3 years. And ten percent more speculative and longer term. And some of those projects would continue and become shorter term projects as the time went on. <pause> And so one of the groups in ADG that became a group in ATG was the object-oriented programming group. And originally, I ran that myself. And it was—it housed Smalltalk. It was a Macintosh Smalltalk. I think there was a Lisa Smalltalk before that but that got abandoned pretty quickly and the focus changed to Mac. And it was an obligation of taking the license, to be one of four licensees of PARC to have Smalltalk. We had to implement it. We couldn't just say, "sure put our name down in the ads." No, we really had to do it. We had a team on there. Initially it was Rick, Rick-something that worked on Smalltalk. And then we actually hired Dan Ingalls so that changed the whole nature of the project. And—because he was not hesitant to innovate, if that was what was needed. He wasn't trying to duplicate something else. He could contribute and do stuff to it without concern that he might upset somebody at Xerox. And then the other thing in the object-oriented programming group was initially a Lisa Toolkit but that became MacApp.

**Hsu:** So that was run out of ADG?

**Tesler:** It became an ATG project. Well, it just kept going but it always had a small audience. When it was Lisa Toolkit, this room and about this number of people, you could expect to attend a Lisa Toolkit developers meeting. But MacApp developer's conference was a big deal. And there would be hundreds of people there. And I guess at some point we might have moved it into the same organizational group as MPW, Macintosh Programming Workshop. But I'm a little fuzzy about the transition. But I guess I had handed it off to people and they did the right thing. Design patterns got worked on in the early stages there, when we were working on—maybe even Lisa Toolkit but certainly on MacApp—we kept finding repeated patterns of certain things that show up in object-oriented programs that make them more reliable. So, for example, the one I remember in the early days was when you allocate and free an object in memory, where do you put the allocation statements and where do you put the free statements? And if you put them in unrelated places, you're really asking for trouble. You're going to allocate it more times than you free it or vice versa. And a better thing to do is put them right next to each other in the same module and then organize your code so that everybody calls their—for example, you can count allocations and frees and notice that they're drifting away from each other and something's wrong. So some of the people in the group got interested at a meta level just in design patterns and object-oriented programs. And Kent Beck was involved in that project and became also somebody influential in the big broader software development world. Part of that was about design patterns. So I wouldn't say that we should take credit for design patterns. But some of the early thinking was done in the context of the most popular object-oriented development framework because it was—there were hardly—it was basically the only object-oriented development framework for a while. So does that answer your question plus a little?

**Hsu:** Yeah. I mean, I was actually—kind of wanted to follow up on that. So sort of this allocation/free problem, was that in Object Pascal because—didn't Smalltalk have the garbage collector?

**Tesler:** Right. In Smalltalk that particular thing wasn't as important but I'll say I think it was still important because at the lowest level of allocating a block, and freeing a block, you still want to do that. When there's no garbage collector present you still need to do that. But even when there is a garbage collector present, say it's in—jump forward a few decades, it's Java. If you're trying to understand a piece of code that has a bug in it, you don't want to be hunting all over the place. Say, did I find all of the routines that got called that could possibly trigger an allocate or a free? That's not always obvious from its name, for example. And one of the things we did, though, was have naming conventions. Sometimes in extreme cases it would be something like, so-and-so allocates. But just more typically you name things appropriately. And even if you know the garbage collector is going to garbage collect a particular item, you still might want to do some cleanups before you let the garbage collector add it or some things like that. And if you call something that you know frees something and it's got another pointer to it then that's a warning sign. So we did often do counting of allocated objects and see if they're what you expect. And that would be true whether you garbage collect or do it all explicitly.

**Hsu:** Right. So then let me step back a little bit, so Del Yocam was your boss when he tapped you to head the ATG?

**Tesler:** Yeah. And that was a surprise. There was a lot of political tension in the executive office. I think some of the executives thought John Sculley wouldn't last very long. And when he went off to do something else or burned out, they were jockeying for who would be the successor. Now, that again, assumes that you pick the successor internally, but there's always hope. So there was, I think, even though the obvious person to do it was Jean-Louis Gassée, and he thought he should do it, I think it was a wise decision to keep that separated. Otherwise, there would be constant borrowing of people from the other group, usually the product development group borrowing people from the research group and eventually making the research group doing no actual research or advanced development. It would just be helping out. And there was some of that anyway. And, in fact, we welcomed some of that but if somebody really didn't want to work on a project and a project they had in research was an obviously good idea that we should be working on, then I would maybe help find somebody else to do it. But their tendency would be to try to get the most experienced and best developer they could. And doing that by taking that resource away from ATG would hurt the company long-term. So that was probably the thinking behind that. On the other hand, Del was a totally non-technical person and very focused on anything but technical activities. The kinds of things he managed were, like, entire continents of sales activity. Not, you know, how many disk drives are going to be in the next Mac we ship, that kind of thing. Is that why you asked about Del?

**Hsu:** Yeah. Yeah. So what was it like managing an R&D group? And sort of what shape did it have? What kinds of resources were you able to get? And what was the culture like in comparison to PARC?

**Tesler:** I think I already told a story at an earlier interview which is that Wayne Rosing had this habit, when we turned in our weekly progress reports, it was always one page, what we were supposed to do, what we actually did, and so on. And there was something that he did occasionally which was, what is your goal personally for what you'll be doing five years from now? And I had never heard of that question being asked on a weekly report before. But one year he gave that question, I think, in preparation for performance review delivery. And I said I would like to be managing research for Apple or a research group at Apple. I don't remember the detailed wording of it. And almost exactly five years later ATG was formed and they put me in charge of it. So it was something that I wanted to do and I had developed a lot of knowledge and opinions—I'm not sure which was more than the other—about how a research group should be managed and what the culture should be like. And now, of course, that would vary with the company. So the kind of places I had worked, I got some experience working at the Stanford AI Lab and there it was very hands off, hire people that have the right skills, have maybe a very broad general discussion about what they're interested in, and then see them again next year. And then at PARC it was somewhere in between a disciplined group and a completely self-directed, individual-directed group. And so I saw some things going wrong there, and primarily we were unable to connect with the development or the product development portions of the company that we were a part of. And it's not like PARC today where it's kind of intentionally separated, not necessarily had [have] much to do with the main business of the company. At that time, it was intended to be a fundamental piece of getting products out in new areas. But the researchers were hired with kind of the wrong attitude. We wanted to invent the future and they did. You couldn't blame anybody for doing that. They said they wanted to do that and that's what they did. But it was demoralizing in the product development parts of the company, that there was this group in California, sun, volleyball, and that was just kind of doing whatever and it may or may not help their business at all, and they had to pay part of the tab to fund the research. So I got involved in that part, which I talked about before, when I was helping Ginn & Company and they wanted to get money—results for their money. And also, I was involved when we had to take a look at what we were doing versus what the hobby computer movement was doing. So I found you couldn't—unless you were Bell Labs or IBM or something, you didn't have the resources to fund far out research in the kind of companies that I was involved in. So when I saw the way that it was being run by—at Apple—by Wayne Rosing and then Ed Birss, I thought they had a good appreciation of that need. And so when it got handed off to me I continued with that—those—set of motivations. So in terms of what the culture was, it was it had to be connected with Apple's business. The line we adopted and it came from a different company and I'm afraid I don't know—I can't remember what company it was. It might have been 3M or Bell Labs or it might have been something completely different. But the criterion for whether we would consider funding a project would be whether there was a clear path to product from what we were going to do—if we did it successfully, would there then be a clear path to product? Or would everybody look at it and say, "well, that's kind of cool, but we're not really sending cheese to the moon in this company. And I don't see what connection there is with the business." But if you say, "oh but we will do this and we will do that and then we'll do this and then we'll partner with them and blah, blah, blah, we can go." And then "you see now why this is a hot thing." "Oh, yeah, okay, that could be. I believe that. But I'd rather we funded all of the short-term product stuff and be competitive." And then my job was to say, "but this is going to be something that's not on your radar right now. I'm not surprised that you don't think it's important because you don't hear customers begging for it and salespeople complaining that we don't have it yet. But one of the things that we're doing, you're going to walk in here in a few months and you're going to say, 'I just found out what our competitors are doing and we have to do something there too.' And so I want to have some of these things being looked at." So that worked pretty well. And budget time always was when the conflicts arose. Take all of that ATG money and put it someplace else. But the top management was very devoted to ATG. We managed to get things into product to some extent. And we had this, it turned out, the unexpected structural cultural thing that we did was that we managed an organization that on an annual basis transferred fifteen percent of the people in product development into ATG. And the same number of people which would be 100, some number that equaled 15 percent of the overall population of engineers, we would basically trade so that nobody's headcount would go up or down. But we would infuse some of the ideas in ATG that sounded interesting. And instead of competing with product development we would merge that group with product development. And that's how QuickTime got started. It was originally completely an ATG thing. And then a couple of people in product development decided that they wanted to do some stuff with multimedia video and so on. And they saw demonstrations that ATG produced that showed, first of all, that it was possible but secondly it wasn't really easy. There were a lot of problems that needed to be solved, particularly the synchronization of audio and video, which we take for granted now, was a hard thing to do in those days. And so at first they brought ATG in to share their wisdom, what they've learned doing some preliminary research. And then decided that there was really nothing to keep it from becoming a product except put it on the product list and fund it. Technically, we had enough stuff that we just had to define some standards but we knew what technology the standards had to implement. And so most of the multimedia group went over for at least a temporary stint in product development. And that happened in several areas. We had AppleScript, Apple Events which lasted many years, outlived, the Macintosh, the original Macintosh II generation. And that too started out as an ATG thing that was turned into a product development thing. Even though QuickTime was a success in itself, a lot of other things that we intended to be linked to QuickTime and part of the product were not successful. The same thing with OpenDoc. There was a thing called OpenDoc that was Apple's lame response to Microsoft…

**Hsu:** Object Linking and Embedding, OLE.

**Tesler:** OLE. Yeah, OLE, exactly. Thanks. So we weren't always successful but that was because we were taking risks and doing new stuff. And we were successful enough times that they kept funding us.

**Hsu:** Earlier you mentioned—so there was a dedicated object-oriented group at ATG working on Smalltalk. How many groups were there? And when these groups would sort of then transfer to product development were they just whole—completely transferred over and they were replaced by new people with new projects?

**Tesler:** Well, starting with the last question first. One thing I learned about management is, there's a lot of things you can borrow from previous things but there's always something different. And so it could have been anything. And if it could be that one guy was working on it and suddenly we get Rex to hire five more people because this is important. And that first person becomes the seed, maybe the leader of the group, maybe the technologist or the scientist in the group. Sometimes it was, we loaned five people to product development and then they took five people who were burned out from doing three products in a row with no rest. And give them a year in ATG to get a refresh break, think about what they'd like to do next, hopefully something competitive, hopefully something innovative. And then when they're ready, normally they would go back into product development and maybe with some other guys who they got to know during the time in ATG. And it would work either way. You could start with any person in either place. But the usual thing would be somebody who's proved that they are good at engineering something, wanting to augment that, that reputation and their contribution, by doing something that is recognized as innovative. And we wanted to give people that opportunity. A lot of times, like, oh woah, that guy yeah, but the reason he's done three successful or three projects in a row or three products in a row is they're all three very successful. We want him to do another one. But this guy's fried. I mean he just can't do it again. He needs a break. And sometimes the breaks were pretty short. You know, they just needed literally to take some vacation and when they come back from their vacation or maybe they went to SIGGRAPH conference during their quote "vacation." And they come back just full of ideas about how we could do that on the Mac. In terms of how many different groups there were in ATG, it varied over time but when I was running it, it was Graphics for a while, Graphics and Sound. I think they changed it to Multimedia because it was Graphics and Sound and Animation. And other things that were generalizations of where it started. And there was for a short time an AI group. And then we realized we were about 40 years ahead of ourselves and maybe we should come back to that later. And then the developer tool related things. Modular, the OLE competitors. ActiveX is the one I was thinking of. I mean and it is related, both of them are OLE, Object Linking and Embedding. But they also had ActiveX. And that's the thing, even a little more so than OLE, that OpenDoc was aiming it for. And it was a bad idea from the beginning. And I kind of sensed something was wrong but it wasn't until we started playing it out and realizing how big a project it was and how little we had thought about the benefits and what you would actually do with it. And people tried to address it in various ways. The first application of OpenDoc was related to Internet Web authoring and that kind of thing.

**Hsu:** That was CyberDog?

**Tesler:** CyberDog. And well, networking was a new thing at that time. And then they seemed to go together. But the group size was getting large. And the investment was getting large. And the motivation for doing it was kind of a typical Microsoft motivation. It wasn't something that Apple normally did. And we were, I guess, when they announced ActiveX and/or OLE—I can't remember the order those came out.

**Hsu:** I think OLE came out first.

**Tesler:** First, and then ActiveX?

**Hsu:** Yeah.

**Tesler:** But we were doing it because Microsoft was doing it and we needed to have something better. And that was before we learned, or one of the ways we learned, sometimes it’s good to just copy or get on the—create a standards group and agree on a standard—and that way you tie without spending a lot of money.

<group laughter>

**Tesler:** And so we were losing and spending a lot of money doing so. So that was a lesson there. Let’s see. Smartifacts—Harry Vertelney (who passed away recently—in the last year) was an architect. I think I talked about him maybe in another interview. I don't know. And when he came to interview for a job he was a building architect, not a software architect. And he had this idea that there ought to be little things that are battery powered that you walk around and do stuff with. And it didn’t take much selling. Everybody loved that kind of vision but we also weren’t doing it. And here was somebody who was adept for architecture in the physical world instead of in the virtual world. And he was envisioning ways we could take what we had done with the Macintosh and do derivatives, more specialized derivatives of that or peripherals. And so with no qualifications whatsoever we hired him anyway and he started a little group that just started cranking out little toys that did stuff. It might have a radio connection or a telephone connection. And one of the things that they did was a handheld Macintosh that ran Mac OS 1.0. We didn't actually even have a number system when we had that version of Macintosh, but it ran the Macintosh operating system, an older version, but one that we all would recognize from the 1984 demos. But it was just a little handheld thing. And… that might come up in the Computer Museum’s predecessors to the iPhone, who was doing what in that space before the iPhone. That’s going to be very cool. And that will probably be mentioned, I’m guessing. Or if you ask—if you have a lead into the questioner they could ask, “What about Smartifacts?” But during the early Newton days it came up as an alternative. It’s like why are we doing a whole new operating system? Why don’t we just do the Mac operating system? And people would say, “Oh, you can’t fit the Mac operating system on a small device.” And then Harry Vertelney would have his guy come out and say, “But we did.”

<group laughter>

**Tesler:** And it made you have to question your logic. And why do we need something different. And based on the success of the iPhone, it was the right thing to do to have something different. The iPhone has its own operating system. And there’s some overlaps with macOS, the Apple Watch, watchOS has some overlaps with iOS also. But in those days, we had the conviction that it needed to be its own—certain devices needed to have their own system software and user interface.

**Hsu:** Did you want to…

**Weber:** The first handheld device of the Mac OS, so what was the interface?

**Tesler:** I think it was a keyboard that popped up on the screen. <pause> But it might have been a physical keyboard, a small physical keyboard. And then in terms of pointing we had—or that was an era when we pretty much had single touch, not multi-touch. But yeah that's a very good question. I'm not sure I ever saw more than just windows opening up and I was already in the Newton group when they finished it. So yeah, you have to ask somebody else how that worked.

**Brock:** Do you recall if that was disclosed or presented outside of the company, the Smartifacts, in particular this Mac in your hand?

**Tesler:** I think we were keeping it quiet primarily because we didn't want to reveal our internal debates and conflicts. And we were trying to do that under the veil of secrecy. So people wouldn't say oh, "Apple already tried that, that didn't work. So let's go do this other thing." If people are going to make mistakes and learn from them that's fine, but we'd rather have them make their own mistakes.

**Weber:** Approximately, what year was that?

**Tesler:** Well, the Smartifacts were from 1987, '88 maybe even to '91, '92, when the Newton was under development. So it kind of existed in those years. It maybe continued a little bit past there.

**Hsu:** So last year we interviewed Ike Nassi who worked for ATG in Cambridge.

**Tesler:** Yes.

**Hsu:** And then later head of the developer tools group. What was it like to work with him?

**Tesler:** He was always getting ideas of his own and from people in his team. And you could pretty much count on a good talk about some new thing that they were working on whenever you ran into Ike—which he visited pretty often. And he's still that way. I just ran into him at the opening [of the Make Software exhibit] a few weeks ago. And I didn't get to see any of the exhibits because he didn't want to let me go until I heard the whole story about what he was working on. So I'll be back to look at the exhibits. But he was used to choosing his own path and his own projects and do his own organization. And he was comfortable doing that and got good results, I would say, about most of it. The one thing that he volunteered to do that in retrospect I think was probably just a bad thing to do, period, but I don't think he made it any better, but he did volunteer to take it off my hands, so I can't complain. That was the—what became known as Dylan, Dynamic Language. And—at Alan Kay's suggestion we acquired a small company in Cambridge, Mass. And so lesson: don't take business advice from Alan Kay.

<group laughter>

**Tesler:** Stick with the high-level stuff, vision. And we wanted them to do an interpreter for a language for the Newton. And Steve Sakoman supported the acquisition, which took a lot of cycles for a very, very small acquisition. The company had like four to six people in it or something. But it attracted some interest because it was a stake in the ground in another location than Cupertino. And it seemed like we're growing up as a company and able to do research and engineering in a new area, in a new geographical area as well as a new subject area. So what we were looking for was a new friendly language that was—had features that were good for a handheld device. And something that could be very small and fit in a handheld device. So that means it probably got an interpreter of some byte code language or something. But we were open minded to what could be done. What the guys in the company we acquired—was called Coral at the time, C-O-R-A-L—was they wanted to do a compiler to make sure that the thing was very, very fast. But the trouble with that was that the easy ways to do compilers generated a lot of code when you wanted a lot of speed. And no interpreter, just compiled code. And they wanted to do Common LISP which wasn't—and we finally convinced them that they should do a new language. So they took Common LISP and they just changed the syntax to make it more like Pascal even though we asked them to make it look more like C. And Ike served as a buffer, for good, I think, mostly for good because it was hard to communicate with these folks. But he would work with them on a complaint that he got from us. And see if he and the group in Cambridge could make some decisions or build something that would address our issues. But over time it became clear that it wasn't really working. They didn't really want to be doing what they were doing. And everything they suggested or questioned was a way to try to turn it back into the project they wanted to do. So in the end I got them organized and they got a book explaining Dylan. But in the process of doing it they kind of lost the interest of the Newton group and they were on a schedule and they wanted to ship something and new releases of the Dylan language would come out and they were buggy. And it was kind of the same thing we did abandoning the AT&T Hobbit chip because every time we got a delivery of it, it was very buggy and that was one reason. But that same thing happened to Dylan. It just was out of sync with what we were doing. And so, let's see, but if I keep going I'm going to diverge away from the ATG topic into a Newton topic.

**Hsu:** Right. Well, we'll get to that.

**Tesler:** So we'll come back to that.

**Hsu:** Yeah, we'll come back to that. But you mentioned Alan Kay. So I wanted to go back and talk about Alan Kay's joining Apple as an Apple Fellow. And maybe before that just maybe talk about how you started the Apple Fellows program and how you decided who got to be Fellows.

**Tesler:** Yeah. Around the same time as we founded ATG it was actually when I took over ADG, Advanced Development Group, again, there were the worries that we won't keep innovating without Steve there. So they were trying to figure out what to do with Steve Wozniak. And he was still on the payroll and he would propose ideas here and there, work on some of his own ideas, apart from Apple. But they also wanted to have a tier of the R&D organization that reached higher than we had employees up to then. They could get director pay but we wanted them to be able to get vice presidential level pay. And to do that we started a Fellows program, Technology Fellows at first. And there were a few fellowships given out immediately. Steve Wozniak, of course, I hope I'm not going to forget anybody here, but I probably am. Rich Page who was the hardware manager on the Lisa. And one of the Apple II early engineers I can't remember if it's one or a different one. I can't remember who was in that first round. But there were a small number, two to four initially. And—but also I think they were trying to figure out a way to get Alan Kay into the company. So they established this program and one of the rules initially was that you had to have made your notable accomplishment but made that notable accomplishment at Apple. Well, that rule didn't last even a year probably. John Sculley announced that Alan Kay was being hired and then Al Alcorn was being hired based on their previous industry contributions. And I mean, he was the CEO, so who was going to question it? But after explaining to people internally that if you want to get to be a Fellow someday you should start out in Apple and work your way up there, now we—or you could start at Xerox, and Atari. It kind of ruined the message. But nobody really cared because they were all excited about having these people in the company. And it was some person in HR [who] had to suffer rewriting the rules or something. Alan. Wow. And management. Okay. So what happened was after there were a few Apple Fellows, the Apple Fellows kind of went there together to management and said, "We need some organization leadership and you guys don't know anything about what we do. And so we want somebody who is technical or maybe one of us who has some management background to report to and organize things." So somebody showed up at my doorstep proposing that they report to me. So starting, I guess, about the same time as I took over ATG one of the things that was listed as part of ATG is the Apple Fellows and they all report to Larry Tesler; a dotted-line report into me, I think, and solid line, they were reporting to some other place in the company that varied from person to person. And so I organized, occasionally we would have a meeting, maybe quarterly, semiannually, I don't know, initially of all of the Apple Fellows. Later we started the Distinguished Engineer, Scientist and Technologist level, which was just below that. The DESTs we called them. And also later we added Apple Fellows that weren't even technology people, particularly Kristina Hooper who was a multimedia innovator. And kind of, she just could understand where multimedia was going and think of buildable examples you could create right now to show people what your vision is. And for that we actually had an office in San Francisco because that's where the media people were, not San Jose. Let's see. But when we met we gradually started talking about critiquing product programs that Apple had at that time. And over the years, we sometimes had suggestions of things that a particular group might do, work with this other group or adopt a new technology that we know about. Or just try to redirect Apple if we think they're going in the wrong direction. And sometimes it was very controversial. There was one towards the end of the process, towards the end of life of this Apple Fellows program. There was a dispute about—in the company or a debate about whether—get this right. <pause> Let’s see. Oh, yeah, the question was whether we should continue a project that had been started, that would do an Intel version of the Mac operating system. So John Sculley always had this notion that if we could use Intel technology and other standards we could lower the price of our machines, which were too expensive for the market. And PC clones were killing us. So everybody—it sounded like a good thing. Nobody differed with that type of goal. And they staffed up a project with some very good developers to develop an Intel-based Mac OS. Well, as it went on they started running into the difficulties of it. And at some point the senior members of the technical staff, who were usually not working on that project but working on something else, liked to poke their spears at it. Are you guys ever going to be done? And what’s performance going to be like? And is Intel going to give us—going to keep favoring the Wintel companies and not really do much for us. And all of those concerns were well founded and on the other hand it was still a great idea. So we brought it to the Fellows and DESTs joint meeting and had a debate about it. And a vast majority of the people in the room thought this was not the time to do this, to port the OS to Intel. It was—there had to be a bigger speed—a smaller speed loss because the—part of the solution was to emulate the Motorola instruction set. And…

**Hsu:** What they eventually did on the PowerPC also? The mixed mode—?

**Tesler:** Yeah. They had colors, the names of these different operating systems, Red, Yellow, Blue. But the thing we had trouble with, was the order they were doing things was going to leave us in a funny position where we have the slow emulator trying to emulate Motorola code. And this operating system is stepping back from the Mac OS. It really wasn’t an operating system for the future. And we started looking around and getting interested in either the NeXT operating system or the Be operating system as examples of something that would be a better thing to start with. And so that was—so the suggestion we came to was, first port everything to a new operating system. If you can do all of this stuff, why don’t you just do it on a new operating system, preferably that’s been already developed elsewhere. And then use the engineers that are interested in this other approach, have them finding ways to get stuff to run fast in the emulator. And in fact, they had a very fast emulator at that point, and anyway, there's a lot of details. I don't remember the details of it all, but in the end, we said, "This is the wrong path. We should acquire an operating system. And you give us a big start, otherwise, we'll be spending years trying to get parity." So.

**Hsu:** What year was that?

**Tesler:** After that, people didn't come to us much with their—<laughs>

**Hsu:** Oh.

**Tesler:** With their—So let's see, that was '95, I guess. Something like that.

**Hsu:** So pretty late, yeah.

**Tesler:** Yeah. And this project had been going on for a couple of years already and again, the best thing about it was this emulator that was—that was one—we knew it was one of the hard problems and somebody had done a really bang up job of it. But, but there were—it just exposed other things that became either performance issues or reliability issues. Yeah.

**Brock:** Just one quick—Isn't your—the recommendation of that group close to what would happen, what, two years later?

**Tesler:** It was—I mean, you could see similarities in what we were recommending versus what was done by the acquisition of NeXT. And—but I mean, you have to get the—you'd have to get the written version of the recommendation and then compare it with what they finally did. I don't think it's that simple that it was just, well, that's what they decided to do. Parts of it are what they decided to do.

**Hsu:** Okay. Huh.

**Tesler:** And I don't remember anymore what parts.

**Brock:** Oh, sorry.

**Tesler:** Yeah, that's okay.

**Hsu:** So let's go back and talk about Alan Kay's role. What did he exactly do at Apple?

**Tesler:** He hired a few people that he had met along the way in his career and people who had worked for him at Xerox. And he started a pattern of every few years going and working for a company like Apple or Xerox and then forming a team, a lot of the same people each time, and taking that team with him to the next company. And after doing that for several companies and realizing that you want Alan Kay when your business is running out of ideas and Alan Kay comes and his ideas don't really have anything to do with your business, and so you pull the funding and Alan goes and finds funding somewhere else. Everybody wants Alan Kay. But they forget that they need to be able to use what he does. And they never do that. They just think he'll bring innovation to our company. But do you want—do you want a programming language for kids? You know, otherwise—

<laughter>

**Tesler:** You're not necessarily going to use what he can produce.

**Hsu:** Well, so what does—was that what he was also working on? He was working on Squeak?

**Tesler:** Well, that was, this is before Squeak. One of the reasons for Squeak was, he wanted to make changes to Smalltalk and they had just to make it legally work, he renamed—when he was at Disney, Squeak was the--

**Hsu:** Oh, right.

**Tesler:** Mickey Mouse, so.

**Hsu:** Right. Oh, okay, yeah.

**Tesler:** He renamed it "Squeak," and then, and when he left Disney, they probably had some discussions about using the name Squeak when there's no mouse anymore.

**Hsu:** <laughs>

**Tesler:** And but he got away with that somehow.

**Hsu:** But he was doing that—he was sort of on that path already at Apple.

**Tesler:** Yeah. There was a project called Playground that was based in L.A. and he lived in L.A. and most of his staff was in L.A., but not all. And the Playground project would go into school, and they went into a few schools and changed the way that the curriculum of that school worked. So first of all, every kid had a computer and over the years, that evolved from a desktop to a laptop. And—but he had somebody else managing the Playground program and I can't remember the last name now… But he also would give advice sometimes and when he gave advice it was really excellent advice to get, and so everybody at Apple who had, you know, "Gee, but if I could just get five minutes with Alan Kay. I want to ask him this question." He had this terrible habit of saying, "I'm going to next visit, you know, Cupertino the 15th of April" or something, and then at the last minute, he would cancel or go home or leave early and go back to L.A. on an earlier flight or whatever. Somehow you just couldn't actually get him physically in a room and, you know, debrief him or challenge him with a question that he would be the best person to answer. We had the person, we just couldn't get the person <laughs> to be present. And so managing him was kind of like it might be managing a company that you acquired 19 percent of and you're trying to get the benefits of having that—owning that company, but you just can never quite get into a meeting with the people that were there. And I never could figure out what to do to make him show up.

<laughter>

**Tesler:** So I stopped trying.

**Hsu:** So, so how much interaction did he have with Dan Ingalls' group working on Smalltalk at ATG?

**Tesler:** Yeah, well, yeah, Dan was there first and then Alan was hired. And, yeah, and you're right, the obvious thing at that point is, is Alan going to be the manager or something of the Object Oriented Group. Alan never wanted to be a manager.

**Hsu:** <laughs>

**Tesler:** He was sure he was terrible at it and didn't want to do it. And when he was at PARC, he hired one of his college buddies, Chris Jeffers, who had better business instincts than he did to run his group, manage his group officially. And Alan had a placeholder title or a convenience title of being the head of the Learning Research Group, but the administrator of the Learning Research Group was Chris Jeffers and you went to him for anything you would normally go to a manager for, that was, you know, administrative as opposed to technical. Alan's kind of a unique force, force of nature, so.

**Hsu:** How much interaction between or how much influence did the Smalltalk group at Apple have with the rest of the company? Were there any—did anything that was sort of—product-wise come out of that?

**Tesler:** The only thing about Smalltalk that was, that anybody in the company was interested in, was that it was the language that the kids used in the Playground schools. And there were a few Playground schools around, but not many. In fact, there was particularly one that they spent a lot of time with and other ones less, and I don't—I've never even visited those other ones. But they also worked with ACOT group, which I forgot to mention before, as an important part of ATG. It was education research. ACOT stands for Apple Classroom of Tomorrow. And that was started by somebody who was in the Marketing, Education Marketing group, but they wanted it to be housed in something more product-oriented. And so ATG was where they decided that would be the good compromise. Do education—do technology-based education research in ATG for the—to work in partnership with the Education Marketing people. And so we had somebody running ACOT and had a few people working for him. Martin—I want to remember Martin's last name. Maybe it'll come to me [Martin Engel]. And then the Apple Classroom of Tomorrow was that one, and then Playground was one that Alan was funding, basically. And, oh, and then external research, where we had research that was being done by universities and Apple was contributing funding to those projects or equipment. And that was run by Barbara Bowen. And they, they all together formed the Education Research Group of ATG. And let's see, what was the question that reminded me of the other question?

**Hsu:** Oh.

**Tesler:** <laughs>

**Hsu:** It was just sort of was there any impact that the Smalltalk group had on the rest of the company?

**Tesler:** Oh. Yeah, that group wasn't really thought of as the Smalltalk group, even though that they had that in common with each other. It was just a means to an end of kids being able to master computers. And so—and Alan knew that it wouldn't fly to try to change any of the research or development tools at Apple into Smalltalk-based. The only people who actually did stuff like that were people working in Alan's group. It had no impact on the rest of the company.

**Hsu:** Right.

**Tesler:** That it was Smalltalk. And but what did have impact is the kind of things they would build and deliver to their school. That was interesting to people. And I think it did affect what happened in Education Marketing and Sales. But I was never closely involved in that.

**Hsu:** Right.

**Tesler:** I would say the people whose names I mentioned, the guy that ran ACOT after Martin, David Dwyer. Yeah.

**Hsu:** Mm-hmm. So then the—so then the—the Smalltalk group itself, there was—because I'm trying to figure out, like, there's all these different object oriented technologies. There's Object Pascal, and MacApp. And then there's Dan Ingalls and Smalltalk. And then there's also this Dylan thing that—Are these all separate, completely separate organizations that aren't really talking to each other?

**Tesler:** Well, partly, they're chronologically different.

**Hsu:** Ah.

**Tesler:** If a language doesn't catch on real quick, it gets abandoned by the whole industry.

**Hsu:** Uh-huh.

**Tesler:** It's sometimes, people have a good marketing campaign and they convince journalists that it's the language of the future or some segment of the market that this is the language of the future, and—but languages have their—have their purposes. And—but Python might be a great solution for a medical researcher or something, but for someone building an operating system on—a server-based operating system or something, it's not going to—not going to be the right tool. So Smalltalk never was really pushed or claimed to be the future of a practical language. It's always a research vehicle where you can do most of your ideas in a single page of code and—but it's going to be kind of slow. These days, it would probably be fast enough. But people gave up on the Smalltalk as a tool long ago, but Smalltalk inspired other things, like Java and Dylan for that matter. And, and also it inspired Object Pascal. And Smalltalk very much influenced Objective-C because the language has Smalltalk syntax in it. Anytime you want to invoke a method, you, suddenly you draw square brackets and you put in a little snippet of Smalltalk. And the NeXT OS was built on that and then recent macOS, well, let's see. It influenced the macOS that was based on the NeXT OS.

**Hsu:** Yeah, OS X.

**Tesler:** Yeah, OS X, yeah. And, and then also for iOS, it influenced iOS. But now the part of Objective-C that they threw away was the Smalltalk piece. If you—it's now more of, more like Java where it's built in.

**Hsu:** You mean the Swift language?

**Tesler:** Yeah, Swift.

**Hsu:** Yeah, yeah. But at—So then at the time, so the—you mentioned earlier that MacApp. So MacApp was part of ATG at one point and then was spun off into the developer's tool [group]?

**Tesler:** I think so.

**Hsu:** Is that what it was? And, and so it never had been part of Dan Ingalls's organization, it was always a separate part of ATG.

**Tesler:** Dan Ingalls' organization was side by side with it in the early days of ATG.

**Hsu:** Uh-huh.

**Tesler:** But <pause> yeah, but I can't remember—I couldn't tell you the years exactly.

**Hsu:** Okay. I wanted to proceed to talking about the Knowledge Navigator.

**Tesler:** Ah, that's a good topic.

**Hsu:** And, and of course, you know, we were talking about Alan Kay. And, you know, what was his role in that project and how did that project start?

**Tesler:** First of all, it's one of my favorite projects. Secondly, I know less about it than you could possibly guess.

**Hsu:** <laughs>

**Tesler:** I barely knew of its existence when they were working on it.

**Hsu:** <laughs>

**Tesler:** I did know. But, but I had no details and suddenly there was this video that was ready to watch. And, and I saw it for the first time, you know. But the instigators were, and I don't know who—who got involved first and who talked who into what, but John Sculley, Doug Solomon.

**Hsu:** Oh.

**Tesler:** Mike Liebhold and, I think, probably Kristina Hooper. People in the Creative Services Department. And, yeah, it was, I think it was John Sculley [who] wanted something to bring to a conference with him and show the future and he wanted Apple employees to see the future as we envisioned it. And, and so he picked people who seemed to be on the cutting edge, particularly multimedia. He was really into Hollywood and sometimes we felt—we thought he spent too much time there. But—and they got together and they produced on a pretty tight schedule, they produced a pretty amazing piece of work.

**Weber:** So I read that George Lucas was involved in the production. Do you know anything about it?

**Tesler:** No.

<laughter>

**Tesler:** I never heard of that. But I'm not—nothing would surprise me.

<laughter>

**Tesler:** But that wouldn't surprise me a little bit. <pause> I don't—I don't know in what way he was involved. I can guess maybe some ways that are predictable, but it could have been something—there could be some great story there I don't know about.

**Hsu:** So was it—it sounds like it almost was independent of ATG. Was it still run under the auspices of ATG or was it--?

**Tesler:** I think it was *not* run as—as an ATG project. Yeah, because I think I would have known more if it was—if that was the case. I think it was considered skunkworks.

**Hsu:** Okay.

**Tesler:** And there was a problem we had at Apple and other people have had outside Apple where somebody discovers there's a skunkworks in their company and, "What is going on here? I never authorized this project." "No, no. It's a skunkworks. One of the other executives was talking with one of your engineers and they got this idea and they're—they're just doing it in their spare time, evenings and weekends. It's skunkworks." And so, "Oh, that's interesting. That's an interesting idea." And then they get an idea at some point later and go, "How do I make this happen? I'll start a skunkworks."

<laughter>

**Tesler:** And, and without understanding really how you start a skunkworks or what you maybe shouldn't try to do. I know of one company that got in a lot of trouble because they didn't let the employee give up anything they were already committed to. They were told to also work in the skunkworks.

**Hsu:** Whoa.

**Tesler:** And gave them some ridiculous deadline. And the poor guys are trying to please the executives and somebody who has no idea of what or how much time it should take to do something is driving them to spend too much time <laughs> on the skunkworks project. Anyway, so it was—it was a skunkworks project and in this case, one that was done successfully. And it wasn't Sculley's first skunkworks. He had done a few before. That's why I think it was successful. Another thing that was a skunkworks project was HyperCard.

**Hsu:** Oh.

**Tesler:** And that, too, was pretty successful. But I think there were some before that where he made the kind of mistake I talked about. <laughs> And, "Oh, it's not—You can't really do that. It doesn't work. You need to get permission from the person's manager to use their time."

<laughter>

**Tesler:** "Oh, okay. Sorry."

<laughter>

**Hsu:** So then was Alan Kay involved at all in the Knowledge Navigator?

**Tesler:** I don't know.

**Hsu:** <laughs>

**Tesler:** He was probably still there. I'm sure he was still there. He could have been invited to participate and he could have not shown up, I don’t know.

<laughter>

**Tesler:** I'm not sure. But I would—I would think that he should have been involved.

**Hsu:** Mm-hmm.

**Tesler:** He was definitely a fan. But it wasn't clear to me what he did to make it happen.

**Hsu:** What sort of aspects of the Knowledge Navigator were sort of maybe later incorporated into or motivated parts of the design of the Newton?

**Tesler:** <pause> It wasn't much like a Newton. It was more like an iPad. <laughs>

**Hsu:** <laughs>

**Tesler:** So <pause> with Newton, you could write on it and with Knowledge Navigator, you would talk to it. Newton was black and white, I think. The Knowledge Navigator was color, video, drawings, so it was heavy on graphics and Newton was a telephone without a telephone connection, so. <pause>But there was—there was in spirit, there was a similarity in that there was, they both had a desire to be somewhat intelligent, serve the user intelligently. And on the Newton, that was very basic. You could just say, "Meeting next week with Jim. Lunch on Thursday at 1:30. And remember to call such and such," and it would put things on your calendar. And Newton Intelligence, we called it. And it required just a little bit of natural language understanding and handwriting recognition. And Knowledge Navigator was pushed into a—positioned into a much future—much more futuristic set of technologies that are kind of the ones we have today.

**Hsu:** Mm-hmm.

**Tesler:** And I've often thought, "Why doesn't somebody go get some Kickstarter money or something and make a Knowledge Navigator that would make it through that entire demo?" You could probably do it now.

**Hsu:** <laughs> So the project was never more than just a concept, because it was just too far in the future?

**Tesler:** Right. That was one of the requirements of whoever decided <laughs> what should be the positioning of it. It's visionary. Don't expect to see this from Apple or really anybody in the next few years. But this is what we're thinking about. This is where we think the industry is going, where the technology is going. And, and it was a great recruiting tool. It was inspiring to people.

**Hsu:** Mm-hmm.

**Weber:** And how much was the Dynabook concept floating around, the--?

**Tesler:** Dynabook?

**Weber:** Yeah, because it seems linked obviously to both, but.

**Tesler:** Yeah. The Dynabook had a lot of similarities, but <pause> I mean, we would—we would sometimes say if Alan Kay was in the room or something or one of his people, just as a context, contextual help, we would say, "Well, you know, just like—like a Dynabook. You could carry it with you to school and do your homework on it and that's what you could do with an eMate, which was an education version of a Newton. That's what you could do with an iPad, Siri. So sometimes we remind ourselves how old these—some of these ideas are, and everybody [is] just waiting for the technology to get there.

**Hsu:** So let's talk about the Newton. So was—so the Newton, was that also a private, like a skunkworks project started by Steve Sakoman and Jean-Louis Gassée.

**Tesler:** Yeah. It started out that way.

**Hsu:** So it wasn't out of ATG at all. It was--

**Tesler:** Never, ever part of ATG. But they wanted some help from ATG in some things, especially after I got involved. But even before I got involved, there was the desire for Dylan at some point, and they turned against Dylan later. But originally, they were working there. They were interested in using the ARM processor, which was kind of an ATG project in a way.

**Hsu:** This is after they had decided not to use the Hobbit? Or were there--

**Tesler:** No, the original—original—when I took over Newton from Sakoman, it had two Hobbits, each—each—inside of the device was going to have two Hobbits and an ARM. Or maybe it was a Hobbit and two ARMs. I can't remember exactly, but it was a multiprocessor architecture. Steve Sakoman was—knew that was coming, and was impatient. I think he wanted to make it happen, because that was—in order to get higher performance, it had a lot of advantages, but some limitations, as well, of course. And so, which question am I on now?

**Hsu:** Uhm--

**Tesler:** Oh, did we work—did Newton work with parts of ATG?

**Hsu:** Yeah.

**Tesler:** So the ARM was already part of Newton in that vision. But it was just a peripheral controller.

**Hsu:** Right.

**Tesler:** And I said, just make it the main processor. The only processor. And that'd get the cost way down. But they were worried the performance would go way down, too. And it actually was performing about the same as the Hobbit. But still not good enough for the handwriting algorithms that we had at the time.

**Hsu:** Hm.

**Tesler:** And then the other thing was what's now called Wi-Fi was—I'm sure it started several other places in the world at the same time, but at Apple it started out in the Newton project as the way to communicate in a workgroup. And Steve Sakoman had this vision of a conference room table, and everybody would be sitting at the meeting with their Newton, and networking things would happen. <laughter> Just transparently. And to make that happen, he thought that they should use spread spectrum RF, and there were people worried about whether that was going to run into spectrum licensing problems or various issues like that. And then there were people that were concerned about the dangers of radio frequency radiation to the user, and maybe we should use infrared, because that's harmless. And there was just lots of things that we take for granted now, were undecided at that point. People knew they were issues. It's not like nobody had thought about them. I'm sure there's always some IEEE group that studies—thinks about all these things. But eventually we decided that RF wireless was the way to go. And—but I was there, and I said, "This is not going to be ready in time for whatever aggressive schedule we're setting for ourselves. Let's transfer this into ATG. The one engineer who really is an expert on it, the hardware guy will go to standards meetings and make sure that what we're envisioning for the use of wireless, in addition to specific standards, doesn't cut out Apple." We always worried that we were going to adopt standards based on assumptions of Intel technology, for example. And we wanted to make sure that DVDs ran on Macs, not just PCs, that kind of thing. And those were struggles. But if you don't show up—if you show up, you can influence the decisions. If you're not there, the decision will get made without you. So we wanted to be involved in the allocation of spectrum.

**Weber:** Who was that engineer?

**Tesler:** Ah, yeah, I knew you'd ask. Jim something? If somebody wants to collect these questions, I can--

**Weber:** Yeah, I'll ask you after.

**Tesler:** Yeah.

**Hsu:** So during this time that Sakoman and Gassée were working on this, and you were still at ATG, what was sort of the interaction like? How did you sort of interact with them?

**Tesler:** Well, ATG people, in general, maybe it's because of who we hired, but they're technology enthusiasts. And so we were—most of us in ATG were fans of the Newton, even though we knew hardly anything about it, because it was mostly under wraps. But Steve Sakoman would sometimes come over and bend somebody's ear, or get the person to bend his ear. And not just Steve, but other people in the group [would] come over. And gradually we would get to know each other. But when I first—the first time I went to look at the Newton after it was in turmoil, Gassée leaving and Sakoman leaving, I was surprised at all the stuff that they were trying to do. I completely underestimated what they were trying to bite off. And how they were trying to do it. But I was glad to see some things, and other things that were appalling. There was a philosophical thing that was driving everybody in that group—they had an agreement with Gassée, and when I say they had an agreement, it wasn't just like somebody got him to sign a contract, it was a shared philosophy of—that you don't want marketing people designing computers. Engineers should do that. And Gassée is a marketing guy! So what does that mean? And sure enough, he stayed out of it! He protected the group, because partly what he was doing was running an experiment to see what would happen if you let engineers make all these kinds of decisions. Because [if] you look at startups, you know, Steve Wozniak and Apple, it's Bill Joy and Sun, and you never see marketing guys there. Well, Steve Jobs. <laughter> But anyway, there was that. And so it was, "You can't touch this project, if somebody comes in—somebody from marketing comes in, makes a comment that might throw the engineers off, and we were going to make—oh, and no compromises on the technology. We want the fastest technology. We want the lowest power consumption, we want the smallest size we can get made. And no compromises. You should never say this, "Oh, my Newton's too slow." Or "My Newton is black-and-white only." I mean, it's going to have everything. And I don't know exactly at what point in the process they assumed that you would compromise on anything, but when I got there, nobody was interested in compromising on anything! <laughter> The other thing was the rewards—is that there would be some special bonus program—maybe even equity in the Newton product itself that they had talked about. And I'm sure they had talked about it. I don't know how they—I never saw any evidence that anybody in that group had an agreement that was signed off by an executive and HR that resulted in them getting these extra special perks because they worked on the Newton. But they believed they were going to. And nobody had dissuaded them from believing that. So I'm not sure—I never did figure out where that came from. At first it was, "Oh, the manage—Gassée and Sculley and Sakoman have promised us all this." But I could never find anything. Anyway, so there was that. That was a morale problem. And [a] morale problem is the last thing you're going to need in a project where the head of the group is leaving with his boss to go compete with you in a different company with similar technology! So, yeah, it was a demoralized group when I got there. And a demoralized group in the end, because they couldn't get their vision delivered and accepted and supported.

**Hsu:** Right. So at that point, what was the shape—what was the actual vision of the product? Was—what was sort of the feature set of it? I mean, you've mentioned there was supposed to be wireless. There was supposed to be handwriting recognition. This is supposed to be a pen-based, sort of like a kind of a tablet-sized. What was sort of the vision at that point?

**Tesler:** Well, what you just said, was the vision. And it was pen-based because Jerry Kaplan was doing a pen-based machine in his company. Apple and Microsoft both thought we better not be left behind, "We don't really understand why this is a good idea. But we better not be left behind, in case it's a good idea." And Vinod Khosla came over one day, the VC, and had a meeting with us, and tried to convince—he had investments in some companies whose existence depended on there being a mobile handheld device. And he wanted to make sure that that was—that Apple was playing in that space, [so] he could promise to send over some of these companies that he was working with to add value to Newton. So it was going to be expensive. It was going to be like the Lisa, the original Lisa. Started out targeting a thousand dollars, and in the end it was twelve thousand dollars or something. Nobody would pay that, so we compromised on the price by compromising on the margins. But [we] discussed Lisa before. But the Newton was at a stage where it was more like a Lisa than a Macintosh. It wasn't the minimum viable product. It was the maximum "somebody's dream" project. But it was going to be seven or eight pounds. It was going to weight seven or eight pounds, which they were proud of because laptops at the time weighed more. <laughter> And it was going to have, yeah, as I said before, multiple processors, handwriting recognition, drawing recognition. And there was the preliminary thinking about intelligent things it could do. That was kind of in the vision, but they hadn't really got anybody really working on that at that point. It was too expensive, too heavy. And definitely not parsimonious.

**Hsu:** Could you talk about how you became the manager of the team?

**Tesler:** Yeah. I was fascinated when they showed me what they had, I was blown away. And on the other hand, I said, you know, "I, and some of you in the room here have been through this before with this, it's too big and too expensive, and then loses out to something smaller and cheaper, even though it's a little more limited in some ways at the beginning." And their answer was, "Well, yeah, we agree with that, but the management told us, ‘No compromises.’" And so anytime there was a meeting, and there was a feature, and somebody thought that'd be a cool feature, and somebody said, "Well, wait a minute, that'll add weight and money," somebody would say, "No compromises!" "Oh, okay. Well we'll just find a way to do it." And—but it had just exploded and outgrown its original—the original concept was more like a Knowledge Navigator. But that was way far in the future, and this was the best you could do right now. But yeah, there were people—some of the people working on the project thought it was a research project really, and they were happily working on—all by themselves on whatever that we had—the one hardware guy and wireless and one software, woman actually, working with him, and that was it for collaboration. It was going to be the collaborative handheld device. And there were two people working on the networking, one on hardware, one on software. There was one guy working on handwriting recognition. And getting nowhere, but—so it was—there was just something wrong with—and I think what was wrong was this notion that you shouldn't compromise, and somehow we have these promised bonuses that we can't exactly explain to you, but we were promised that. And that was kind of the—couple of the challenges that I had. And there were then specific things about technologies that weren't working as well as they needed to. And you know. So I started asking a lot of questions. And I got pretty—pretty quickly I figured out who was coming up with these specifications and who was critiquing them, and there was one particular guy that stood out in terms of being able to get along with everybody else. And when I would say, "Well, why is that part of the system this expensive?" You know, he was the only one that could really answer that question and tell you ways it could be made cheaper. He was also probably the youngest person in the room. And—his name was Mike Culbert. And so I adopted Mike Culbert <laughter> and we started going around together, with the help of the Hardware Manager, Rex Smith, and the Software Manager, Jerome Coonen, and I said, "Let's just find a way to make this cheaper, because otherwise the Board's going to cancel the project." "Oh, it can't. This is so important!" I said, "They're going to cancel the project. And I'm sure of it, unless we can come up with something that's practical." So some of the things we went after was, "Okay, can this be done with fewer processors? Instead of three, how about two or one? And can we have infrared do anything useful that we wouldn't be embarrassed that that's all it does?" But—and so we came up with beaming business cards, and messages. Handwriting recognition. "Maybe we should require printing instead of allowing cursive?" "Oh, but we want cursive! Everybody wants cursive. All the users want cursive." "Mmmm, let's leave that an open issue, we'll see what we can do." Anyway, the big thing for me was the processor, because what you're going to spend on that is going to determine all the costs you decide other things should be. If you're going to spend a lot on the processor, you're going to want to spend a lot on other things that'll justify that processor and vice versa. And I was fascinated by the ARM for no good reason. I think it was, like a lot of engineers who use the ARM at the assembly language level, it was—it just had a very unique and elegant instruction set, that was kind of RISC, but kind of not RISC at the same time. And they had at a very early stage built a static version of it, so that it would consume very low power. And there were some other processors around that were static, but this one was designed from the beginning to work well in a static implementation. And we had used it in ATG for a printer driver and—for a Apple II successor, which made a lot of sense because Acorn was using it as a successor processor for the same 6502 that we used in the Apple II, so this was the U.S. equivalent of what the ARM was actually built to do, is replace the Apple II processor. Anyway, we made—I convinced the board to make an investment in ARM and let it spin out of Acorn. And that turned out to be the most successful part of the Newton project was the ARM. We made more money off the ARM from our stock than we lost on the Newton.

**Hsu:** Wow. That's crazy. <laughs> So why was the Hobbit chosen before? Why had they settled on that processor before?

**Tesler:** Okay, what was the name of that processor when it was an AT&T? Started with an S.[1] Anyway, AT&T announced at one point a processor, that started with an S, and that was interesting. It had a—not a unique architecture, but an unusual architecture, that it was a stack-based machine.

**Brock:** Hm.

**Tesler:** And that was actually fascinating to me, because one of my favorite machines when I first started out was the Burroughs 5000, which was stack-based. And here it was many years later, what, twenty? Thirty? Twenty, thirty years later, another go at trying to have a stack-based machine. It was kind of a computer scientist's dream. But somehow between Jean-Louis Gassée and Steve Sakoman, they

[1] Larry Tesler adds: No, it was “CRISP.”

decided that AT&T was a great partner to work with. And AT&T was looking for somebody to justify the use of this chip. Anyway, so Gassée went to them, or Sakoman and Gassée went to AT&T with a proposal to make a special version of the chip that was static, and in other ways low power. And they agreed on the name Hobbit. And then it was going to be the static version of this other thing [CRISP]. Plus some other changes to the memory management and some other stuff. And Apple would finance the development, so AT&T wouldn't have to—which didn't make a lot of sense, but—and—but it was really no good reason. I think they went around and looked at the offerings from other companies, and they couldn't find anything suitable for what they needed, for what they thought the Newton should be. And AT&T was desperate for a partner, or they would lose their whole project. And for just what was not a whole lot of money for Apple. It was five million dollars, I think, they put into the development of the first Hobbit. But I got worried when I asked questions like, "Well, what's the processor going to cost per chip?" "We estimate $15." "So what did AT&T estimate?" "Mmm, they said it depends. We'll negotiate the price once we have some working chips, and they can see what their costs are." "Oh, okay." So there was that kind of discussion going on and so the first Hobbit chip was delivered after I got there, and they started looking—you know, testing it. And there were a lot of bugs. I can't remember the number. I won't guess, but there were a lot of bugs. A lot more than anybody expected. And so they were making lists of bugs that needed to be fixed, and the priorities and all that sort of stuff. And I said, "Well, okay, there's—at least they had some silicon, what is the—now can you find out what the cost is going to be? We've got to put in a cost model for the machine. What the cost is going to be for a Hobbit?" So they went to AT&T and pressured them very hard to come up with a number, and they came back $50. And I said, "That's not good. <laughs> That's not where we're trying to go with this product. We have a parts list. We're trying to get it down to $500, and ten percent of it is gone with a processor chip?" Anyway. The ARM started looking more and more interesting. <laughter> And they delivered us some silicon a few months later with zero bugs. And it sounded a lot better.

**Hsu:** ARM did, you mean?

**Tesler:** Yeah, Acorn. Well, ARM was spun off from Acorn by then. Yeah. But it was Acorn engineers who left Acorn to go to ARM and found it, who also did a custom version for us. We wanted a custom version from whatever company we worked with. And we couldn’t get the big companies working with us, because they didn't want a custom version for Apple. We're the smallest market.

**Hsu:** Hm. Let's see, so earlier we talked a little bit about Dylan also, and so I was curious to ask about why the Newton team was interested in having Dylan as a language? I mean, that sounds very much like a Dynabook kind of a vision, having a dynamic object-oriented language on a handheld or on a portable device. Why was that so important to that vision?

**Tesler:** Steve Sakoman wanted a forward-looking device in every way. And so if object-oriented programming was the way to go, he wanted one where object-oriented programming was built in and dynamic—dynamic language. As the speeds of computers went up he was looking ahead, as he should, what do we do with the extra performance we're going to get in the next five to ten years and it's a lot of growth. <clears throat> Excuse me.

**Brock:** May I ask a quick question? My question is about the context when you came in and assumed leadership of the Newton project. Just in terms of the whole landscape of the company, how important was—how important to the company overall was this job that you now had to come into this project to try and have a product come out? Was it a bet the company sort of moment? Or what was the scale, I guess, is what I'm…

**Tesler:** Right. It wasn't bet the company on the Newton. And I think throughout, probably one of the things that Gassée and Sakoman did best was to keep it from becoming a bet the company. That was one reason they were protecting it. They were protecting—also protecting the company from the Newton.

<group laughter>

**Tesler:** And so I just kept doing that. But I got a little more disclosure so people in the company wouldn't see it as a deep dark secret. It was a little—we were a little looser with the information than we had been—than they had been.

**Brock:** So it was more of a question of just devoting a lot of people and money and time and effort into this project, rather than…

**Tesler:** Well, for a while it was pretty small. There were—I can't remember the number, 30 people in it, something like that when I got there, 20 to 30. And in one building, a small building on Bubb Road I think it was. And the real money didn't start being spent until we actually had a product and we had to gear up for manufacturing and preordering inventory. And that's where we got killed because we based it on—we based the forecast on how much we needed to make instead of how much demand there really was. And so that led to the predictable outcomes.

**Brock:** Well, I don't want to get out of sequence.

**Hsu:** I was wondering, so Newton in the press was closely associated with Sculley. Why was Sculley so interested in the Newton? And at what point did he sort of stake his legacy on the Newton?

**Tesler:** Well, I don't know if he did that but you could find elements of that going on for sure. And I think that he just thought it was the place that Apple should go, the same reason that he drove the Knowledge Navigator. This was what the future was. He could see that. And he wanted a way to show that to potential employees and current employees to motivate them, and mainly to customers and journalists. And his specifications for the product were very undemanding. It's [not] like go to the moon and come back, you know? It was, I want to be able to fit it in my coat pocket. And we'd argue with him about what coat he wore that day.

<group laughter>

**Tesler:** But eventually it got to a size where he could show it off. The thing that I think was really unfair was how the press skewered him for predicting a 2 trillion dollar—or whatever number he came up with—industry would grow out of this. First of all, they would misquote him as saying that it was going to be a 2 trillion dollar product for Apple. He didn't mean that at all. He was talking about the entire industry and we well exceeded that by now, I think. But eventually it would be that big. And that was prophetic. I mean people couldn't picture it at that time. But I think that—remember it was—even though he was coming out with a series of Macintoshes, that was Steve's machine. And in order for him to be recognized, not only did he have to continue the success of the Mac and make it more successful, he had to do another product. I mean, he's kind of where Tim Cook is. So where's your next product? You did an iPhone. It was great. You did an iPad. It was pretty good. A watch, well, I think it's great. But—half the market, Apple has half the market there. But, you know, he wanted something that he could pull out of his pocket and show instead of having to show the thing Steve dropped from the ceiling. But I don't think it was a wrong thing to do. I think it was the right thing to do. It wasn't just solely for his own ego. It was a way that he could generate demand for Apple products.

**Hsu:** I mean, at the same time there was also General Magic going on. How would you compare the two simultaneous efforts and why do both of those things?

**Tesler:** Right. Well, the odd thing was that General Magic was started in ATG and Marc Porat—I hired Marc Porat to come up with something like that. And so it was very much a project that I did at a time when I wasn't on Newton yet. And Bill Atkinson and Marc—who was the other person?

**Hsu:** Andy Hertzfeld?

**Tesler:** Yeah, it must have been Andy, Andy Hertzfeld. Had this idea that was different from the Newton but it was, again, like the Smartifacts I talked about before. It was another way to do it and Porat was a very persuasive fellow. And I mean, he was an old friend of mine. Nobody else would have hired him, I think because he had no track record in technology. But he was a visionary. And so life got complicated when I was suddenly helping out Newton and he was going off on his vision. And we had a lot of discussions about it. Most people thought Apple should choose, do one or the other but not both. But they were—Sculley liked both. I mean, if we had two entries in the market, it would be a little less risk, and keep our competition jumping. I mean, it would definitely position us as the company with the most credible mobile product offering. And if you look at Apple now having the iPhone and the iPad is a little bit like having—or the Watch, maybe, would be an even better example. Having more than one product in the same family is a helpful thing. Well, of course, there was no family resemblance between the General Magic thing and Newton and that's where we kind of lost that ability. So Marc came at it another way. "Okay, we'll leave Apple then, just leave Apple. I'll take Bill and Andy with me and we'll do a totally different unrelated thing. We won't use any Apple technology in it." And Sculley supported that. It was a kind of odd solution, spin it out of your own company, but he did. And it had some early success but it wasn't really what people were looking for either.

**Brock:** Could you maybe talk just a little bit more about that arrangement because it seemed to be like a spin out and there was still some sort of relationship. I don't know if it's just on the level of ownership or whatever.

**Tesler:** I'm sure there was stock involved, ownership share. But I think it was—Sculley would be the person to ask or Porat. But when Sculley decided to support both of them, at some point in that process I think he probably expected it to all work it out. Somehow it will work out. One will take over the other or vice versa or we'll cancel one and do the other. But he didn't do that because there were very persuasive people and major investments in both sides. And sometimes it's the right thing to do. I mean starting ARM was the right thing to do as a company, spinning that out [from Acorn] and so it seemed like maybe spinning out this other thing into General Magic might have been a good idea. And yeah, but I'm not sure what would have happened if it was more successful. Would Apple try to buy it back? I don't know.

**Brock:** Just one last follow up on that, so from listening to you it sounds like that the group in ATG with—around Porat—while you were working on, you know, were helping with Newton, their mobile computing project is different from the one that they pursued once they spun out and went into General Magic.

**Tesler:** Right. General Magic was nothing like the earlier vision. It wasn't the Knowledge Navigator. It wasn't—what was the—Paradigm, was the code name for the project, Paradigm. And they hired a guy who had this idea for this Telescript. And it was fascinating to be seized [ph?], to hear about this Telescript technology. Agents floating around the Internet. And at that time there was—the Internet was not public yet. And so—is that right? Yeah. It [public Internet] was '95.

**Weber:** Why, I understood they were going to do their own proprietary network.

**Tesler:** Yeah. And they did working with—that's why AT&T was interested. But they were going to send Telescript coded agents out into the phone network, basically, and it's hard at the time to justify. If people knew then what we know now they wouldn't have tried doing it.

**Weber:** But, so then what was the earlier vision?

**Brock:** When they were still inside?

**Weber:** Because that—

**Tesler:** So it did not include Telescript, that's for sure. <pause> You know, I think it was just a general pitch in favor of handheld devices, it started out as. They had this graph showing the growth of desktop PCs and then the falling off. And we were starting in the—we were just getting to that peak and starting into our downslide. But there's a bigger peak coming if we jump on the other one before anybody else does. And a lot of it was very metaphysical. And Knowledge Navigator, it was kind of what the Newton guys were trying to get away from. It's like marketing driven product definition.

**Brock:** Okay.

**Tesler:** But that's when Porat hired Bill and Andy. I suggested that they look at it and see what they think and they both got very excited and wanted to work on it. And for a while there was a tug of war because there was the Newton and there was the Paradigm. And at the beginning it was, well, we can't have both. Which one is going to succeed? And when the Newton was funded by the board, that put General Magic in a defensive position. And that's when they said, "Why don't we spin this out?" And then just coincidence, who they happened to hire just led them into a different concept. It's hard to remember. I'd completely forgotten about General Magic and that factor in the Newton decisions, because it disappeared from our radar when they spun out.

**Brock:** There's one other factor, I think, floating around in the mix, too, which is a project on a telephone.

**Weber:** Well, that's earlier on, though. The Steve Sakoman, the Mac phone thing.

**Tesler:** Oh yeah. Right. I didn't know a lot about that. But I do remember that, Sakoman's Mac phone.

**Weber:** I read that's what got—what Jobs had hired him, for that and that sort of somehow planted a seed for an interest in mobile devices.

**Tesler:** Yeah, that's an interesting point.

**Weber:** But that's one interpretation.

**Tesler:** As good a theory as any right now.

<group laughter>

**Tesler:** Where are we in your question list? Are three-quarters done? Ninety-percent?

**Hsu:** Yeah, we're going to have to stop at some point. But I don't think we're going to get through all of the questions that we have. So I'm just going to ask things that are related to what we're talking about.

**Tesler:** And be done by four? Is that what you're aiming for?

**Hsu:** Yeah, we're aiming to be done by four.

**Tesler:** Okay. Because today is our—my wife and I are celebrating our thirtieth wedding anniversary today.

**Brock:** Congratulations.

**Hsu:** Congratulations.

**Weber:** Congratulations.

**Hsu:** So since we were talking about Bill Atkinson and Andy Hertzfeld, so Steve Capps who worked on the Newton, was close with Bill and Andy and worked on the Mac with them. But decided to go on the Newton. Sort of what—do you know why he decided to stay on the Newton? What was…

**Tesler:** I think so. There's no reason you can't ask him.

**Hsu:** Yeah.

**Tesler:** Just quickly I would say that my interpretation of what was going on was that he had a lot of loyalty to the group. They had been through a lot together. And people looked up to him and if he left they'd have nobody to look up to. That wouldn't be the case but that's how you feel sometimes, like I'm going to leave people that look up to me and what happens if I'm not there. But I think he also did kind of believe a version of the keep marketing out of—he got to be good pals with Michael Tchao, T-C-H-A-O.

**Hsu:** Oh, Peter Tchao, the product marketing guy?

**Tesler:** No. Michael Tchao, product marketing.

**Hsu:** Oh, so his name is Michael? Because I think the [book] *Defying Gravity* says his first name is Peter.

**Tesler:** There was a Peter something there but a different person.

**Hsu:** T-C-H-A-O, correct?

**Tesler:** Yeah. He went back to Apple to work on iPad. But you know Steve [Capps] and Michael Tchao head up a friendship that they basically defined the product. So I think Steve never completely bought the, "there shouldn't be any marketing people around." I think he was happy to have some. But I think he wasn't comfortable about Marc Porat defining a product. That was going too far probably in the marketer's direction. But I would ask him.

**Hsu:** That actually leads to the next question I wanted to ask which was about this whole marketing driven thing—you've been highly critical of the book *Defying Gravity*. And the impression that I got from reading that book was that—maybe it's a consequence of the fact that the people they hired to write that book were just following the marketing people around. But the story is told almost, mostly from the marketing perspective and seemed to have been driven by marketing. So was that untrue? Or did the Newton later become more marketing driven?

**Tesler:** Oh yeah. The Newton—when I got there and discovered that there were no marketing people, I said we've got to have marketing people, product marketing. And John Sculley came along and said, "There's a guy who bends my ear all the time that has just the right kind of thinking for this project." And he said, "Could you meet him?" And he came over and I met him and he met a few people in the group. And it was a great match. And also it was a guy that Sculley had picked, so he had his guy on the project. He got it up—Michael was good at knowing when to update Sculley on stuff when there were political things that needed to get resolved.

**Hsu:** So that was Tchao.

**Tesler:** Michael Tchao. And then there were a couple of other marketing people hired in addition. And I think on the org chart somebody else was higher up than Michael but he was actually the guy that everything was centered around. He was just a little younger and less experienced than some of the other guys. He was the first but not the only marketing person. And I had some tensions with both groups. I had tensions between me and engineering, and I had tensions between me and marketing. And the main issue for me—there were a lot of things that I advocated that other people in the group didn't like the idea of at all. But the thing that I didn't like was that it was a lack of usability testing and feature testing. In fact, I hired an anthropologist to—a business anthropologist, one of the few at that time, to find out what do people want in a mobile device? And what markets and what features? And it was one of the guys that she interviewed, she said, “You should talk to this guy because he basically is a customer who understands what other customers like him want.” And he said, “Look at my phone, I’ve got a phone. I’ve got to carry it around everywhere. Now, you’re going to give me this other device. I’m going to have two devices to carry around? Forget it. I want you to have all of the phone functionality built into it or I'm not buying it.” And she said, “This is typical. They keep saying, ‘but it’s got to be built on my phone.’” So then we would go to engineering and say, can you build a phone in to this Newton? “Well, we’ll go talk to Sharp because they’re doing our manufacturing.” And they go, “No, maybe in five years but not right now.” And marketing wanted to avoid that. They didn't want to know that it was too early to come out with the product because the product had to have a phone in it. And so that was the—the other thing was who the customers would be. And the customers were supposed to be guys like this, mobile professionals but they weren’t interested because it didn’t have a phone. But they [the anthropologist, Eleanor Wynn] did find markets where it was of interest. Police, fire, I think may have mentioned this before. But mobile blue collar workers, really, and not white collar workers, because they have radios and other—they’re used to having a bunch of stuff. And she rode along with a policeman for a couple of days, firemen for a day, and other professions. And found that—she said, “This is your market.” And we go to—Sculley is deflated, “I thought my market was CEOs, like me. Not firemen.” And “No, no, no, we don’t want that. That answer is not acceptable.” I said, “but that’s what the customer is telling us. Sorry.” So they would not listen to what the customers were telling us. And there was the famous fake focus group. They said, “We’re going to run a focus group.” I said, okay, is it going to be here in the Bay Area? “Oh, no, no, people here are too technology-oriented. We’ve got to do it somewhere else. We’re going to do it in Minneapolis.” Minneapolis. Okay. Well, I’ve got to attend. And I’ve got to bring my anthropologist with me too. And, “No, no, no, you can’t come. No, you need to be home here. Blah, blah, blah.” And they go to this thing and they come back and they say, “This is what we found out.” And they boil it down and it’s like, mobile professionals, no telephone necessary. Networking necessary. Handwriting recognition, important. It’s like, that’s funny, this looks just like the spec sheet for the Newton. They’re going, “Yeah, we’re building the exact right product.” I said, okay, do you have videos of the focus groups? “No, no, we don't have videos.” You know, he never ran a focus group or they ran it and they didn’t like the answers, whatever. They came back with justifying what they’re doing instead of finding out some insight that is a surprise. That’s what you want to—you shouldn’t run focus groups if you already know the answer. So, not happy. And we made a lot of mistakes on the Newton from me down. Everybody made a lot of mistakes. It was a little advanced for its time. [We] didn’t know how to come at it.

**Brock:** Do you think that’s the fundamental thing here that it was just—you know, you talk about the technologies having to kind of find this moment and it’s unsuccessful if it’s too late or too early. And this seems like an example of it being too early.

**Tesler:** Oh, the Blackberry came out a year later or something. And the PalmPilot maybe less than a year later. I mean there was—but they made the right compromises. And they also, if they want the market to be—if they want the early adopters to like their product—it’s not going to help them past the early adopter stage—but they have a real big impressive early adopter program. And I remember Donna

Dubinsky being in a PC Forum.[2] Everybody in the room gets a Palm, a PalmPilot.[3] It's this room full of influencers. You can't find more influential influencers than you can in a room attending this annual conference, Esther Dyson's conference. And every one of them walks off with a PalmPilot, what do you think they're going to be using next week? They'll be showing their friends. It was great. But they forgot about crossing the chasm or didn't know about that one yet. So they got the early adopters and that's all they got. <laughs>

END OF THE INTERVIEW

[2] Larry Tesler adds: Actually a Gartner Group Forum and an Agenda conference.

[3] Larry Tesler adds: With the conference schedule preloaded.