MIPS Oral History Panel
Session 1: Founding the Company

John Hennessy, John Moussouris, Edward (Skip) Stritter, and Larry Weber

Moderated by: Michael Malone

Recorded: February 18, 2011
Mountain View, California

CHM Reference number: X6042.2011

<Introduction by Edward (Skip) Stritter>

Stritter: Hi, my name is Skip Stritter. In 1984, I got together with friends of mine, John Moussouris and John Hennessy, and we founded a new company called MIPS Computer Systems. So in 1984—I'll try to set the context here-- minicomputers have completely established themselves over the former mainframe dominance, and it's been a few years now since the IBM PC was announced, 1981, and before that the Apple II and Apple III computers, so personal computers were starting to spread, and starting to have an influence.

But personal computers didn't solve all the problems for intensive computer users. In a new type of computing, a new industry was building up that was called workstations. Workstation companies, such as Sun Microsystems, Apollo Computer, and in 1984, just recently established Silicon Graphics, were starting to introduce new technologies and new ways to think about computing on your desktop, but as powerful as the world of minicomputers and mainframes before it. Up to this point in 1984-- which by the way, is the year that Apple Macintosh was introduced-- workstations and high-end PCs had been based on 16-bit microprocessors, Intel 8086; and by 1984, Intel 286; Motorola 68000, which was used in the Apple Macintosh, were for the day the powerful 16-, almost 32-bit microprocessor generation. In the research labs, both in industry, at IBM and other places, and in the universities, Stanford, Berkeley and many other places, a number of trends were being developed as interesting research projects, and starting to converge in a way that created brand-new opportunities.

Three important trends led to the establishment of MIPS. First of all, VLSI design was being popularized by-- in academics-- by the work from Carver Mead and Lynn Conway, in particular, making it possible for graduate students and anybody outside of the Ivory Towers of big industry in the semiconductor businesses at that time, to create new circuits, play with ideas, and develop new technologies. In addition, VLSI technology, fabrication technology, had improved, so that the world was ready and the technology was ready for a new generation of 32-bit microprocessors. And then finally, in the development of semiconductor technology, new fab lines---fabrication for semiconductor circuits---were being established not just for the production of in-house proprietary products, but there was the beginning of a possibility of having other parties, third parties' designs fabricated so that the creation of a new kind of company-- which we came to term, fabless semiconductors-- was just becoming possible.

The next big trend coming out of academics, or coming out of the research labs-- in this case Bell Labs-- was the UNIX operating system. Prior to this, operating systems had been proprietary to the companies that produced them and sold them, but UNIX was, for its day, an open, readily available, and most importantly, portable operating system, that could be put onto a new microprocessor, or computer architecture, without rewriting the complete operating system.

The third trend coming out of the research labs, both commercial labs and in academics, was called RISC, Reduced Instruction Set Computing. This was a new idea for computer architecture. The basic concept being, if you could design and architect your computer with a very simple instruction set, reduced instruction set, especially tuned to the capabilities of the implementation fabrication technology, the semiconductor technology, so that those instructions could be run especially fast, using techniques such as pipelining, which had in prior designs only been used in large mainframe computers. Instead of the then conventional implementation technique of micro-coding, which was used among others, in the Motorola 68000, where the hardware interpreted higher level instructions, and executed them with a sequence of low-level instructions or microinstructions. RISC architectures just provided the low-level instructions. The designers made sure that those low-level instructions could be executed as fast as possible. And the programmers relied upon new compiler technology to compile their code down into these simple, quickly executing instructions.

The MIPS team, myself, from Motorola worked on the 68000 in an obviously commercial microprocessor development environment; John Moussouris had worked for many years at IBM, Yorktown, Heights research labs, working on the IBM at that point, research, RISC projects, the 801 project; and John Hennessy and his students at Stanford, had been working on a project called MIPS, which was also a RISC architecture effort. So beginning in 1984, John Hennessy, John Moussouris and myself started to get together over breakfast to discuss the possibility of starting a new company based on these breakthrough ideas: RISC architecture, UNIX operating system, fabless semiconductor manufacturing. And the hope, or the challenge, of creating a new generation of computer architecture that would expand and develop the workstation industry.

The [3 taped oral history sessions] that follow [this introduction], take you through the history of MIPS Computer Systems, from its beginnings in 1984 with John Hennessy, John Moussouris and myself. Our first employee was Larry Weber, and he joins us. By 1987, when we had produced our first chip, started shipping it, started getting customers, and started getting traction as a real company, we were ready to hire a real CEO.

That man was Bob Miller, who joins us on the second part of these [OH sessions]. We hired Joe DiNucci from Digital Equipment Corporation, to head our sales and marketing operations. And with these new players and industry-seasoned executives, we were able to expand our business to attract large Japanese semiconductor companies to make our products. And most importantly, of course, to build a large and loyal customer base using the MIPS RISC architecture.

In 1989, Bob Miller led us in a very exciting initial public offering, IPO. And then in 1992, we merged the company with our biggest customer, that had one of the most exciting companies at the time, Silicon Graphics. And so beginning in 1984, the people at MIPS took these new trends in computing: RISC architecture, the UNIX operating system, and VLSI technology, to put together a new kind of company, a fabless semiconductor company, to produce the first RISC architecture microprocessor sold on the public markets. And to help the world build a new generation of high-performance workstation computing and computing capabilities. At the time, we called it “minicomputer performance at a microprocessor price.” MIPS Computer Systems.

<Panel discussion begins with Michael Malone as moderator.>

Mike: Let’s begin. I want to walk our way around the table. Each of you introduce yourselves, give your background and how it related to MIPS, and then what your current position is. So we’ll start with you, John.

Moussouris: Okay, I started out as a physics undergraduate at Harvard. Then I went to Oxford as a Rhodes Scholar and got a masters and a doctorate in mathematical physics with Roger Penrose. Spent a couple of years at MIT Laboratory for Computer Science learning how to design computers. Then at IBM Watson for five years. They sent me to Stanford to teach. I was recruited into Watson by John Cooke, who was the inventor of RISC within IBM. And I ended up spending that year in Hennessy’s group in Stanford. The MIPS, Stanford MIPS project. And then I joined MIPS.

Stritter: I’m Skip Stritter. I did my undergraduate work at Dartmouth and then I came to Stanford to Computer Science PhD. When I finished that, I went to Motorola, it was down in Texas. And I got to be on the 68000 microprocessor project at Motorola. It was a special time, because microprocessors to that point had been eight-bit processors. And everyone was racing Intel with 8086, Zilog and Motorola were racing to get the 16-bit generation. So I was lucky to be at the right place and an interesting project. We did well with the 68000, because that was the beginning of the workstation business. So with Sun, Apollo and others. And in fact, a little after we started this company, the Apple Macintosh came out using the 68000 processor.

Hennessy: John Hennessy. I was certainly at Stanford during the time that MIPS was started. I led the project. It was a follow-on to the geometry engine project, which became Silicon Graphics. And I’m still at Stanford.

Weber: I’m Larry Weber. I spent 12 years at IBM doing programming languages, system programming languages. And then I got involved in an offshoot of the 801 project, where we tried to commercialize the 801 in the product divisions. I left IBM, went to a startup for a couple years, and then joined MIPS.

Mike: Let’s talk about Stanford in those days. John, you’ve done so much to change the place, that it’s almost unrecognizable from what it was back in the early ’80s. Let’s talk a little about the MIPS project there. The beginnings of RISC. RISC has obviously been assimilated into our culture at this point. But it was a real novelty [at that time].

Hennessy: It was.

Mike: Almost a radical idea back then. So background. Tell me how it began, and gentlemen, fill in the rest of the story.

Hennessy: So we started when the geometry engine project, which as I said was the beginning of Silicon Graphics, was sort of winding down. Jim Clark was getting ready to start Silicon Graphics. And we started looking then for another project. We were getting funding from DARPA as part of the VLSI program that they had started. And we were then beginning to think about what to do next. I had actually been working in the compiler area with a student, Fred Chow who later on came to MIPS. And doing advanced optimizing compilers and trying to push the state of compiler technology.

So we sat down and said, “Well, what should we do for our next research project. And given that we had all this big investment in compiler technology, I think a perspective that VLSI was going to change computer architecture, and that perhaps it needed to be rethought. Rather than just take the minicomputer tradition and embed it in a single chip microprocessor, we should rethink what the architecture should be. And obviously, trying to do a 32-bit processor at that time, while most of the processors that were out there were 16-bit, was going to push the edge of what was doable in a single chip. So early on, many of the ideas that came to be part of RISC, like eliminating microcode, were driven by the fact that we simply didn’t have enough space on the piece of silicon to do that. And we also felt from a compiler perspective that we could compile down to a level of instructions that could be directly interpreted in the hardware without the need for microcode. And that was the beginning. It began with a class project. We had a brainstorming project, where everybody threw ideas on the wall and we converged around what became the Stanford/MIPS architecture.

Mike: This was kind of gestalt switch, though, wasn’t it? Typically, you go to greater complexity. You don’t go to less complexity.

Hennessy: Yes.

Mike: It was kind of retrograde to the common thinking at the time. Why?

<overlapping conversation>

Hennessy: Well, there was-- I think there was a notion, and in fact, embedded very much in the VAX architecture that the idea was to close this gap-- the “semantic gap” it was called. And it was the gap between programming languages and architectures. And the idea was the hardware should come up. Well, to us, that didn’t necessarily seem like the right thing. I had actually worked a little bit on microcode generators and realized that in some sense, it was just as easy to compile down to the hardware. And of course, the advantage is you compile down to the hardware; you do it once at compile time. If you bring the hardware up to level of language, every time you execute an instruction, you go through an interpretive cycle. So I think we had that idea, but I think as you said, it was really radical. Probably the point that convinced me how crazy radical as such it was was a panel that we were on with Nick Tredennick who was then at IBM.

Stritter: Who, by the way, Nick is the one who, with me, wrote the microcode of the 68000. So he was entrenched in the microcode at that time.

Hennessy: And that microcode environment and everything.

Moussouris: And Nick also did the Micro 370 microcode at Yorktown Heights where I was before I went to Stanford.

Hennessy: So we're on this panel, I'm there, Dave Patterson, my colleague from Berkeley, who ran the Berkeley RISC project and Nick. This was against conventional wisdom. So at one point somebody says to Tredennick-- we had just started the company-- and somebody says to Tredennick, "Well, I hear Hennessy and his colleagues have a million-and-a-half dollars to go do a RISC company. Given all these bad things you've said about this technology, Nick, what would you advise?" And Nick says, "Take the money and go to South America." So there was a broad view that this technology was purely academic and it was a toy, and when you-- I still remember guys from Digital Equipment Corporation saying to me, "Well, when you put floating point in, or you put in page faults and virtual memory, all the advantages will disappear. And it's just a naïve academic project."

Moussouris: I had had the experience at Yorktown-- I joined Yorktown in '78, and that was when the Arab oil embargo was happening, and there was a hiring freeze, and IBM had just gotten out of its huge anti-trust suit, so it was stressed. And I remember looking at the semiconductor trend lines and presenting at Armonk a story that IBM needed to become a leader in microprocessors in order to remain a leader in computing, because you could see that microprocessors by around 1990 would overtake minicomputers and mainframes in performance at a much lower cost, and Armonk did fund it. And during the four or five years I was at Yorktown, the project grew from-- I was the first person up to about 100 people. And we-- I strongly felt we actually had to build microprocessors, not just do research around the fringes of them. And in the end, we had to decide which to build, and in the end, it boiled down to 801 versus 370. And Tredennick was the head of the Micro 370 project. I don't think he knew that I actually encouraged management at Yorktown to fund it, because I felt it was important enough for IBM. And in fact, they build 370 follow-on products even today as microprocessors. It was very important for them. But he felt so competitive that he really felt like he had to slam RISC in order to attract resources into his Micro 370 project. But those chips back then, you're right, they were all microcode. Half the area of the chip was microcode. It didn't all even fit. The problem state is all that would fit. Whereas, the 801 could get quite a bit higher performance. But the company took a kind of funny left turn and went into this other product that became the PC RT. I think partly to protect the mainframe business.

Stritter: And not to forget the influence of the compilers on the MIPS project. Because all the microprocessors up to that point had been designed by semiconductor companies. They were the 8-bit generation was designed by a wonderful VLSI engineers at the time, with no ideas about compilers, and no influence from the compiler or from the operating system.

Mike: In retrospect, did RISC have to be driven by a startup? Simply because, as you said, the existing companies they were dragging along this anchor of existing products, existing architectures and everything else. The traditional thing of the legacy problem. They couldn't get around it. It required a new company.

Stritter: If you look through those companies who were in microprocessors at that time, eventually, they all, to a man, they all tried to do a RISC project. Motorola did, Intel did, TI did. And it was very hard for those guys.

Hennessy: But that was after the evidence was there...

Stritter: But that was after the evidence was there. And it's very hard for those guys.

Weber: At IBM, the 801, it was started after the demise of FS, which was the world's most complicated everything. And George Radin, John Cocke, basically wanted to sort of go the other way, and then they focused on trying to use compiler technology and software technology to simplify things. They had a problem at Yorktown, because Yorktown was known for creating great things, but never got into product. And so the product division in San Jose said, "Hey," and actually it was Bill Worley specifically, said, "Hey, we're going to go out, and we're gonna sort of be the flag bearer to show that, in fact, a technology developed at Yorktown would be useful in the product divisions." But even though we were the product division, we still found a lot of embedded resistance to change the technology, and enter into something different than what everybody was already familiar with.

Moussouris: And the irony was that all those folks - Worley, Carruba [who managed the original 801 project], and Birnbaum, who was the head of the Computer Science Department when I joined Yorktown, they all left IBM to go to Hewlett Packard, to start what became the PA RISC program at Hewlett Packard Company.

Hennessy: I think the answer, though, John, is probably "Yes." And I think actually if you look at the kind of one of the forcing functions that led to the creation of the company, it was insight from Gordon Bell, who came to me and said, “You’re going to have to start a company to do this. This technology is too disruptive. It upsets too many existing markets. I mean, look, we had a processor that we could sell that was probably between five and ten times the cost performance of a VAX. And you know, DEC at that point, was making tons of money on VAXes. So it was such a disruptive force that you probably did need to do a startup to get it going. I mean, eventually, perhaps it would have gotten there, but it would have taken a lot longer to come along on its own.

Mike: There’s a certain irony that Gordon did that, because it was the VAX that drove the 8080, too, I mean, initially those...

Hennessy: He’s a scientist.

Mike: He keeps parachuting into the microprocessor business. And turning things upside down. Now, quick question before we move on to the actual project itself. Every major revolutionary idea like this has intellectual antecedents. When you look back now in retrospect, what led to RISC in the years and decades before?

Moussouris: I would say IBM, from what I could-- I spent a lot of time with John Cocke and the other RISC folks at IBM, and I think John, and IBM in general, had “Cray envy”. What Cray had done in the very high end supercomputing area - it really disturbed IBM that they couldn’t get anywhere close…

Hennessy: High end floating point. High-end scientific stuff...

Moussouris: Yeah, what Seymour Cray had done. And a lot of the RISC ideas, I think came out of, perusing-- you know, looking at techniques that Seymour Cray had developed.

Hennessy: So I think there was that motivation. I think the Berkeley crowd was motivated by some experience that Dave Dizl had on this wonderful computer called The Symbol, which was a computer that didn’t even have a compiler. It interpreted the text. And of course, it was the slowest thing you’ve ever seen on the earth, right? Because of course, you’re reinterpreting everything every time. The breakthroughs that occurred in compiler technology were critical. And in particular there’d been a major breakthrough in register allocation called graph coloring (done first by Greg Chaitin at IBM). ...and then we refined it. And it became a major breakthrough in how to do register allocation. So there were breakthroughs in compiler technology. And then I think my experience, and also Dave Patterson’s working on micro-coded engines, where we found, “Hey, we’re doing something at hardware time that we can do at compile time.” And it just didn’t seem to make sense.

Moussouris: It wasn't just a matter of reducing the complexity. It was a matter of moving the boundaries of complexity between the hardware and the software. Putting more complexity into the compiler, in particular. And...

Mike: That's a fundamental philosophical shift.

Moussouris: If you did it very tastefully, the sum total of the complexity when you were all done would be a little bit less. But you were mainly moving the boundary. And you had to do it tastefully with an understanding of both compilers and operating systems. And the horrible fact of microprocessor design is that it's totally unforgiving of mistakes. If you'd leave the wrong transistor disconnected, your chip is worthless. And if you don't discover it until after you ship millions of them, you have to get them all back. Intel's doing that right now with one of its peripheral chips. So, it's not like software where fixing the bugs is the basis of your economic relationship, ongoing relationship with your customer, you actually make money fixing the bugs. You know, with microprocessors you got to get it right. So debugging is critically important to a design of a RISC microprocessor.

Weber: The interesting thing about putting the compilers into it, as it turns out, it's really not that much more work for the compilers. Because the compiler has to do the same stuff. What you want do-- whether even if you have a complicated CISC architecture, you want to do register allocation. You still want to do all of those things. But now you're going to do it and you're going to do it and you're going to get a bigger bang for the buck, because you don't have this other sort of processor. The microprocessor---or the microcode processor ---sitting between the compiler and the real hardware.

Moussouris: Especially back then when there wasn't a single dominant computer language. You know, C, for example, is much more dominant now. C and its variants, than it was back then. Back then we had Pascal.

Hennessy: Pascal, Fortran and C.

Moussouris: ADA for the Defense Department. You know, we had all sorts of, you know, Fortran, we had all these different things. And you know, compilers would have to dance around the weird machine instructions, of complex instructions set machines. Implementing all those different languages, and implementing capabilities. And the operating system.

Hennessy: Well, a common backend also was a key breakthrough in terms of writing one-- the big work is in the optimizer of the compiler. That's the really hard intellectual thing. So the key idea was you translate to a common intermediate form, and then you build one optimizer that works for all the languages. So knowing that you could do that dramatically reduced the effort. You didn't have to replicate the really hard part of the job.

Stritter: There’s another factor, Mike, that I think we should recognize is I think of it as the demystification or the popularization of VLSI. So Carver Mead and Lynn Conway and that stuff had been happening for a couple years. And now student groups at universities could make real chips. Jim Clark did it. The MIPS team at Stanford did it. And many others. But the idea that just students could make a real chip that really worked…

Mike: So you could instantly expand the population of designers, and could bring more intellectual capital to the...

<overlapping conversation>

Hennessy: Absolutely. And that was the whole view. It was about the liberating designers that why does the person who makes the silicon at the foundry have to be the same person that does the design? And I think as Skip said, the amount of software knowledge in the microprocessor companies then was abysmal! Really abysmal. They couldn’t have written the compilers on their own. There was no way they could do it. So they had to hire outside companies to do it.

Mike: Well, this is intriguing, because it suggests-- in retrospect, when you look at these things, they look like intellectual exercises, when in fact, this is a very pragmatic solution to the situation and challenges of that era. You know, multiple languages, lack of software people in the companies, the rise of VLSI. Very much a product of the era…

Hennessy: Absolutely.

Mike: Even though now it looks like a logical process resulting from something, far larger forces.

Hennessy: Yes, and in fact, it’s interesting you mention that because one of the problems we had in convincing people about this technology is we didn’t have, until quite a bit later, a good solid scientific engineering explanation of why RISC was faster. And it actually didn’t happen until a famous paper was published analyzing the VAX 11/780. And showing that, in fact, contrary to what people thought, the number of clock cycles per instruction was twice as high as they thought. And so that gave us the key insight to then understand that role and to understand the role of pipelining and everything else. But we didn’t have it at the first. We kind of stumbled into a great idea without really a good understanding. Simple was better. But why is simple better? And as we eventually understood it, then it was much easier to explain what was happening.

Mike: Well, one of the lessons of Silicon Valley is no matter how great the technology is it ultimately comes down to a battle of marketing. And we'll get into that in a few minutes, but seems to me this became a marketing challenge.

Stritter: Sure.

Hennessy: Absolutely.

Mike: …especially when MIPS became a company, you had to sell that sizzle, and had to learn how to sell that sizzle. Not just selling specs. Let's talk about the RISC project at Stanford. The physical nature of it. Where was it? Where were you guys located?

Hennessy: We were kind of split up. We had just built a new building called The Center for Integrated Systems. Some of us were in that. Some people were still in the old ERL Building. And then we had a few people in something called "The Boys Town Trailer", which was an expansion trailer outside some of these buildings. And we were just shifting over to the CIS Building. So people were kind of moving around at that time. It was an incredible time, because there was a lot of interaction and dialogue among the universities who were working on this big VLSI area that DARPA had started up. So we were using tools from MIT, and tools from Berkeley and things, other ideas. And it was an intellectually vibrant time. With lots of meetings and interest among people, and this wonderful thing that occurs when you get that kind of focused energy and that many smart graduate students really working together. Namely the field can move at a rate that's much faster than it normally moves when people are working kind of more or less alone.

Mike: Let me ask you, since you run a great university, does that still exist? Or is it more silent now?

Hennessy: It still exists. It still exists. I think it exists whenever there's an opportunity. I think recently you saw it around the issues of autonomous vehicles, where there was lots of collaboration and interaction going on among the various people trying to build autonomous vehicles. And the field all of a sudden makes a quantum leap because there is that kind of interaction.

Mike: Was academic life different in those days, gentlemen? Than now? Has anything changed?

Hennessy: The equipment was pretty sparse, embarrassingly. <laughs>

Mike: Technology and innovation become much more institutionalized at universities. Was there more a seat-of-the-pants feeling in the early '80s than now?

Hennessy: I think when you look at a lot of the great projects, you'd find a similar kind of thing. You'd find the same thing if you go and look at the roots of Google, and how Google got its start and the first algorithm at Stanford. You'd see a lot of the same kind of...

Moussouris: I think there's been a shift towards software.

Hennessy: Oh, yeah! There's no doubt about that.

Moussouris: As being easier to fund, less capital intensive, lower-- as I said, it's much more forgiving of mistakes.

Hennessy: And maybe a bigger opportunity now.

Mike: Is that a good thing?

Moussouris: That it's forgiving of mistakes?

Mike: Yeah.

Moussouris: I mean, it's good for the funders of the software. I think it's one of the reasons why in other areas that are less forgiving of the mistakes, one should be careful to take guidance from people who became incredibly rich and powerful selling software because they tend to be almost fail-blind. Because in their field, there was no, you know, it was very low cost in having something go wrong. You know, I wouldn't want to have a software entrepreneur be responsible for designing nuclear power plants, for example.

Hennessy: But it was a unique era. I mean, here a university team could build something that, while it was still a prototype and a university project, not a complete product, had enough elements that it could embody the key intellectual things. If you said now in a university, "You've got to build a mini version of something that's competitive with what Intel does," it's impossible. The size of the teams, the effort required. I mean, we had four graduate students doing the design! That's hard to accomplish in hardware. And I think that's why we've seen a lot of this shift to software.

Mike: Now how far were you guys going in the actual work you were doing? Were you designing circuitry completely? I mean, were you laying out?

Hennessy: Yeah, mm hm, we were laying out. Fabbing it.

Mike: So literally, that could not be done today in a university setting.

Hennessy: You could do the layout and fab, but to do something that's competitive with an Intel microprocessor, that would be extraordinarily difficult now. The scale of that project is just so large.

Mike: Well, as Gordon Moore says, "The other Moore's law is the cost…"

Hennessy: Yeah.

Mike: "..Of fulfilling Moore's law." And it just becomes astronomical, too. And it gets beyond the range of even great corporations at a certain point.

Hennessy: Yeah.

Mike: So day-to-day life. What was it like there when you went in each day? What were you working on? What did the kids look like? How many, you know?

Stritter: <laughs> The kids!

<overlapping conversation>

Hennessy: We had a group of graduate students, you know, a couple of our colleagues are still hanging around. Forest Baskett was hanging around a bit and a couple of other guys.

Moussouris: He had gone to DEC.

Hennessy: He had gone, by DEC by then.. He had gone to DEC to start Western Research Lab just about then. We had four core graduate students that were on the hardware and low-level compiler team, and then Fred Chow was doing the optimizing work that had started before. And really three guys did the design and hardware, and then we brought in a couple of other people to design the first board. So our idea was to get to a prototype, which eventually was a board. We got it fabbed both through the DARPA MOSIS system, but also on the Stanford fab line as well. And they got it shipped back, and we built a board, demonstrated the board worked, and that was the end of our project. Then we published papers, Mike. And we thought, "This is so obvious! We're going to be heroes! Everybody's going to do this now!"

Mike: That was that “etcetera” moment. Everything else will be taken care of.

Stritter: Whatever.

Hennessy: Everything else would be taken care of. Like good academics. We do our work, and you know?

Mike: What kind of infrastructure you guys have there? What kind of equipment?

Hennessy: Xerox Park had donated a bunch of Altos to Stanford. So we had those for doing black and whites, or doing layout design on the Altos. And then we had some early versions of what eventually became the Sun Microsystems product that we had designed at Stanford. And we had some of those around, and beginning to come along. A big VAX 11/780 as sort of the design rule checking and simulation software for doing that kind of stuff. A mix of CAD tools, primarily university-based CAD tools from Berkeley and MIT and Stanford mixed in.

Mike: So now you have a prototype. One of the more mystical parts of all of technology is the transformation of the-- in this case, academic prototype to a real product in a real company that is funded, has to actually sell these things and turn a profit. Walk me through slowly, this transformation. Because this is the part that’s rarely ever documented, it’s rarely ever remembered. It seems almost like a hallucination even to the people who were part of it. So fill in, gentlemen, how this thing came together as a real company.

Moussouris: Well, the story we told when we went to the VCs, we started out telling the story, it was John who told the story of, “Here’s the Stanford design. And here’s a design we can do later in CMOS.” The Stanford was an NMOS design. The CMOS design would be a new…

Mike: And John, was it a technology story, or was it a technology-slash-business story you were trying to sell?

Moussouris: It was a technology/business story.

Hennessy: This is a lot faster and cheaper than anything else out there, so it must be right.

Moussouris: Skip has dug out the...

<overlapping conversation>

Mike: The spreadsheet on the back saying, "Our five-year revenues..."

Moussouris: No, there is a spreadsheet.

Stritter: There is a spreadsheet actually.

Moussouris: Surprisingly. <laughs>

Stritter: I found the MIPS Business Plan, which these are the papers, or copies of with John Hennessy's handwriting that we presented to the VCs. So here's a handwritten chart, and you can't see it, of course, but the VAX and 68000 are down here in the corner. And here's MIPS going way up here.

Mike: Obviously, that's the growth curve every VC ever wants to see.

Stritter: Exactly. Exactly.

<overlapping conversation>

Moussouris: That's performance. It's MIPS per thousand dollar...

<overlapping conversation>

Hennessy: And of course, these are on these acrylic slides with the pens, remember the old way we used to do this?

Moussouris: Yeah, I remember this.

Stritter: It was technology. It was about technology.

Moussouris: Price performance relative to everything else that was out there. But to talk about the transformation, in the end, what finally happened, to just kind of cut through the guff of the story, we hired people. We raised some money based on the story, and we-- you know, John was on sabbatical for one year. So he was kind of the Chief Technical Officer. We knew he had to go back. Skip was in charge of the software and the board and box design. And I was the one crazy enough to take charge of the chip design, which was circuits plus architecture. And I went out and hired, you know, I asked, "Who's the best chip designer to do a really fast chip in Silicon Valley?" And the answer I got from multiple people was Ed Hudson, who was a Berkeley graduate actually, not a Stanford graduate who'd spent years at Intel designing SRAMS, very fast SRAMS. And Intel got out of the memory business, so he left. And he was starting a small company to do very fast floating point to plug into PCs. And I convinced him...

Mike: That was also serendipity, too, because Intel had just gotten out of the memory business…

<overlapping conversation>

Moussouris: Yeah, they'd just gotten out.

Mike: So there was a lot of talent floating around the Valley.

Moussouris: That's right. And so what we ended up-- you know, and then he hired some other people, he hired Dan Freitas from-- who had also been at Intel. And then we got a bunch of the Z8000 people to come. Dean Carberry and Roger March, Mike Wagman and Bob Patrie, and so on. We got a great chief architect from Weitek, Craig Hansen, and a top logic designer, Tom Riordan from Hansen. So there's a team of about ten people whom I recruited with the help of John and Skip and Steve Blank, who was our VP of Marketing.

And we had a big, long debate about what we should do. Steve Blank had the idea, you know, Stanford chip needed to be changed. And it needed to go from NMOS to CMOS, from Mead/Conway design to SRAM-like design. You know, very high speed, balanced path, dynamic circuits. Very much more advanced circuit design. And it also needed to have its architecture changed - ideally suited to compilers and operating systems. And you know, the question was were we going to do the Stanford first? Steve found a contract design house that was willing to do the Stanford redesign. And that kind of broke the deadlock, because when we really scrutinized that, we realized we shouldn't do this. We should just all focus on the CMOS design. And that's what we did. And we did a completely new design. There wasn't a single rectangle in common with the Stanford design. There were other things that needed to be changed. The Stanford one only had 16 registers. We had 32. It was word-addressed, we did byte-addressing.

Mike: Did you know that was going to happen when you went into it? That you were going to have to throw the entire thing out?

Moussouris: Well, ironically, no we didn't quite. We thought we might be able to get an early market entry, but...

Stritter: That was part of the story…

Hennessy: We didn't have much money. That was part of the reason.

Moussouris: We didn't have the money, and ironically, while part of it was going on, one of the ironies of it is that the Stanford design actually had a lot of features in common with the design that I said IBM took a left turn going with the PC RT design…! And so I was kind of stunned when I got to Stanford to discover here's the PC RT coming out of Stanford sort of. And I felt that it was really important that we do a cleaner, easier to debug. And there was also the need to have on-chip memory management, you know...

Hennessy: Floating point coprocessors.

Moussouris: Floating point coprocessors. Room for other coprocessors. Very fast arithmetic. You know, and then later on, Hudson added some other things that became best paper award-winning inventions. Like we did the first phase-lock loop synchronized interfaces to the external caches and coprocessors, that allowed the bandwidth of the chip to be 64-bits per cycle. You know, the Stanford chip was in an 84-pin package. We were in 144-pin package. Phase lock loop. So we had 64 bits per cycle at the time when everyone else had only 32 bits every two cycles. You know, the Intel, the 386 and the 68020.

Mike: The packaging was new as well at the time, right?

<overlapping conversation>

Hennessy: Yes, packaging was new.

Moussouris: Packaging was new.

Hennessy: There were a lot of things changing. The cache RAMS were changing. When we started, we thought NMOS was going to last a little longer than it did. But by the time we got the company started the switch was already onto CMOS. And then as it became clear that we would have to redesign more and more, we said, "Well, let's just start from scratch. We've got a lot of experience, we know a lot. We can take a lot of the compiler technology and we can really easily retarget it, and let's just start from scratch.

Mike: So it almost seems like you could not have done this two years before.

Hennessy: Well, two years before we probably would have been forced to do an NMOS design. And then we might well have been tempted to just take the Stanford design. I think the truth was when you start with a clean sheet of paper...

<overlapping conversation>

Mike: Because it's seems like it's a nexus of what you guys are doing. IBMs doing the same thing. New packaging technology, the suddenly availability of really bright memory folks pouring out of Intel after they shift to a pure microprocessor business.

Hennessy: The arrival of UNIX, so there was a standardized operating system that could be...

Moussouris: That was another key thing…

Hennessy: … the shift to high level languages. All these things were kind of going on at once.

Mike: I want to go back to the money real quick before we go onto anything else. Because you say you didn't have much money. Where did you get the money? Where did you go?

Stritter: So we went around to the lead VCs.

Mike: It was all VC money. No angels, no mortgaging houses? None of that stuff.

Hennessy: Well, there weren't a lot of angels then, because there weren't that many successful people in the Valley.

Moussouris: I have to say for myself, I had to leave IBM quite early. Because IBM had a huge number of lawyers after they settled the anti-trust, that they were going-- you know, they made certain that people who left the IBM PC project, for example, and tried to go off on their own got-- you know, were very disciplined not to...

Mike: Like this was the year of Boca and IBM introducing the PC.

Moussouris: That's right, so…..I had to leave early…

Hennessy: And so John made the decision that he-- before we could really seriously talk to anybody and he considered whether or not to do this, he had to leave the company early.

Moussouris: I left early. I remember going-- you know, Skip, I remember you were working for the VCs as a consultant at the time. And John, you know, still professor at Stanford. But I remember half my life savings disappearing between the time I left IBM and the time MIPS got its first funding from a VC.

Mike: Because you were funding yourself.

Moussouris: Yeah, I was funding myself. Yeah. I was funding my-- you know, I was trying to help MIPS get...

Mike: You had to live.

Moussouris: Yeah.

Mike: So you were working with a VC? Consulting with a VC?

Stritter: I came back from Motorola from Texas, and I was consulting wherever I could find work. Some of it was Apple, and other places.

Moussouris: You had been at Nestar, too, right?

Stritter: Yeah, and we had done a networking company, and hanging out with the VCs to see what was going on and where I might hook up. But also to make some money.

Mike: So give me a sense of the road show. How many of you would go? You had your business plan with your vertical.

Hennessy: Well, the three of us would go.

Stritter: The three of us.

Moussouris: The three of us.

Hennessy: Occasionally one of the students went on the project.

Mike: You had your vertical….

Stritter: This was it. [hold up original business plan]

Hennessy: We had our chart. It was a technology pitch. I mean, I think we basically said, “We know how to design a technology that will change the face of computing and how these machines work.” We don’t know anything about managing, or organization, or anything like that. We need help putting together a management team. And that was our pitch.

Mike: So you guys walked in with the obvious tech creds, so when they saw that chart, they believed it. And you were honest enough to say, “We don’t know how to run a company.”

Hennessy: Yes.

Stritter: Yes.

Mike: We need some management fire power. Who’d you go see? Like Don Valentine?

<overlapping conversation>

Moussouris: It was-- we did see Sequoia.

Stritter: Sequoia.

Moussouris: But they weren’t...

Stritter: Mayfield.

Hennessy: Mayfield, Kleiner.

Stritter: Sutter Hill, TVI.

Stritter: You know, all the lead guys at the time.

Mike: You saw some real heavy hitters. It must have been a tough little run through the gauntlet to see those guys.

Moussouris: Yeah, and they beat us up a little bit. Which I think they kind of have to do in order to make sure that the team can cohere. But you know, I think that the ones that were really interested were the ones who had big investments in the workstation companies. Sun-- you know, SGI, primarily was Mayfield; Sun, KP and TVI; and Apollo, Sutter and Greylock. Those were the-- they were the ones that were most interested. They knew that the Motorola 68000 line that was used in all those workstation products was not keeping up with its schedule. Apollo had gone to the extent of building a board level implementation of the 68020, to try to get a lead over everybody else.

Stritter: So that's the market context here. The workstation business has taken off and people see it as a big opportunity. So everybody was looking for a way to get in and fund technology.

Moussouris: And they were all afraid...

Mike: The CPU supplier is becoming unreliable….

Moussouris: The workstations...

<overlapping conversation>

Stritter: And not only, you know...

Hennessy: It was as much a performance play, I think, as anything else.

Stritter: Right.

Moussouris: Right.

Hennessy: We said, we can give you minicomputer performance for a workstation price. That was really the value proposition.

Stritter: We showed them a performance improvement curve, so Motorola-- I mean, they were improving, and it was a good company, but ours looked like this. [holds up chart]

Moussouris: They were all afraid, too. The workstation companies were hearing the rumors of IBM and Hewlett Packard and DEC all having internal RISC projects that would be proprietary to them, that would only come out in the system level products. So they knew they'd be sideswiped if they didn't have a microprocessor that could outstrip what those systems companies would do.

Mike: How much influence did you guys have on the internal RISC projects at these companies? I mean, there's always been this cross-fertilization between HP and Stanford. We all know that one.

Hennessy: The HP team all came from IBM.

Moussouris: Well, it is interesting, you know, Craig Hansen, who was the Chief Architect at MIPS, working in my group, he had been the floating point architect on PA RISC at HP, which acquired Apollo. And then he'd been a floating point and graphics design-- graphics architect at Weitek. And it was actually the Weitek floating point that was used by SPARC, you know, by Sun. Sun didn't have its own floating point chip. It used Weitek. So by the time he finished at MIPS, he was the Chief Architect in all three leading workstation companies, all of which were differentiated by megaflops per dollar, particularly, because it was engineering applications. He was the megaflop designer of all three, and they all came to market at the same time. You know, so in essence, he'd gotten faster and faster. <laughter>

Hennessy: So then the DEC project was a group of people led by Forest Baskett that spun out of Stanford. They did a prototype out here in California in Palo Alto. But they could never sell it back East. They just couldn't get any traction going back East in trying to sell it.

Moussouris: In fact, his attempts to sell-- one of the starting events of MIPS from my perspective, was that John and I co-hosted a talk given at Stanford, with the title "Two VLSI RISC Chips." Or, "The 801 VSLI RISC Chips." I have a copy of it. It was given by a Yorktown Heights researcher named Al Chang, who was a PhD from Berkeley, had been at Yorktown working on the RISC chips for a long time. And we had not been forewarned that the talk was really not going to be about the 801, it was going to be about ROMP, this-- the thing that became the PC RT. And it was a totally different design.

Hennessy: So we all went in expecting to hear this great talk on-- you know, there were three groups in the country working on this. And few talks had come out. We were going to hear a great talk. Finally, we were going to hear something about what IBM was doing in RISC, right? It was really the first talk that wasn't just informal

Stritter: Very close, right.

Moussouris: And Baskett was there. I have an interesting doc-- this is a copy, the five-page email that he wrote back to DEC on the East Coast to tell them what he learned in this talk, and along with everything else that he could read between the lines, because he kept his ear close to the ground for a long time. So it was a frighteningly accurate rendition. And we didn't know that in order to get permission to give this talk, my boss of this 100 person microprocessor organization back at Yorktown, you know, who's like the second most powerful person in research division, in charge of university funding and everything - that he had agreed not to have Al Chang explain that this was a different chip. And no one told us that this agreement had been made. <laughter> So you know, poor Chang was getting pelted with vegetables because it was a different chip, and he wasn't explaining why. So I finally stood up and explained why. And when Forrest wrote about my explanation, he said, "One of his Yorktown Heights colleagues in the audience became so exasperated with this approach, that he made it clear that it was not done at Yorktown Heights." And then he added the sentence that was the end of my career at IBM. <laughter> It said, "His tone made clear the low regard that some of Yorktown Heights 801 people have for the Austin work." And this got from DEC to Austin to Frank King, who was the head of the Office Products Division, Texas. And he called up Armonk demanding my boss' resignation. Then my boss called up Hennessy and me at 6:00 in the morning, West Cost time.

Hennessy: Said he was going to yank all the research funding from IBM at Stanford…

Moussouris: And that's when I realized that I did not have a job waiting for me back in Yorktown Heights. <laughter> And I said, "Maybe I should start looking for a job." And Hennessy said that Gordon Bell is offering to fund us as part of Encore to do the MIPS chip. And we looked at the business plan, and I said, "Well, this is interesting but maybe we should just talk to some VCs."

Hennessy: And it didn't work out for lots of reasons. So Gordon said, "Just go talk to the VCs." Actually Jim Clark told me the same thing. So I had been consulting at Silicon Graphics since the beginning of the company, because I'd built some key technology as part of that project, as part of the geometry engine. And I sat down with Jim and Ed McCracken, and they said, "You know, we love this technology. We can't fund it. But if you went out and started something, we'd be interested in being a customer in the future." And so there was a lot of, kind of, pieces.

Moussouris: And it was really-- it was Mayfield that put in the 1.5 million. And they were on the SGI board. So they...

Stritter: So Mayfield was riding high with the SGI investment, and it had been going for a year.

Moussouris: And John had been a cofounder of SGI, too, so he knew…

Mike: So Mayfield was your Series B?

Moussouris: Series A.

Stritter: Series A.

Mike: For a million-and-a-half?

Stritter: One-and-a-half million dollars.

Mike: Wow, that's a fair amount of trust.

Moussouris: Well, we were supposed to get two-and-a-half but KP and TVI didn't-- you know, they were supposed to join in and they were delaying. And...

Stritter: So we just took it.

<overlapping conversation>

Moussouris: We ended up taking the one-and-a-half.

Hennessy: The cap table was funny, and there was a lot of stuff going around.

Mike: What was your lead to Mayfield?

Hennessy: Grant Heidrich.

Stritter: Gib Myers.

Hennessy: Gib Myers was on the board, too.

Mike: So you got the youngsters putting...

Hennessy: Yeah, we did.

Moussouris: Well, Gib Myers...

Hennessy: Well, Gib was a little more senior, but Grant was younger...

Moussouris: Gib was pretty senior.

Stritter: It was Grant's project.

Hennessy: It was Grant's third company or second company.

Mike: Okay, so now you have money! What do you do next?

Stritter: So the first thing we did-- I remember this, because I was writing the checks. For some reason I was the CFO. A third of that money went to buy a VAX. The first thing we did was spend a third of our money, the next day, basically.

Weber: That's the standard startup mentality back then. Get the money, buy a VAX. <laughter> Good for DEC.

Mike: Everything else follows. Facility? Building? Office?

Stritter: So we looked-- we stayed in Mayfield, and we had a little office in the corner, which was also the Xerox room, and that's where we kind of started hiring. That's where, I believe, may be the first time we met Larry. Because he'd been introduced to us.

Mike: So what was your first impression?

Weber: Well, Skip called me up. I knew Skip through a mutual friend. I had worked for Bill Worley, I had left IBM a couple of years earlier. Bill had left after I did. Went to work at HP, calls me up and says, "We've solved all the problems with the 801. Come to HP, and we'll do this new great technology." So I went through the whole thing and they made an offer and it was great. So I said, "Just give me one week. I need to check one little thing out." Which was exactly when Skip called. And we...

Mike: So you chose Sand Hill over Page Mill.

Weber: Yeah, we did.

Hennessy: We had to do a hard sell on his wife, though. <laughter> She didn't want him going to another startup!

Weber: Yeah, my last startup was rather rough.

Moussouris: You had a rough experience at Nestar [ph?], too, I remember you were telling me about it.

Stritter: Yes. People don't want you to leave wherever you are, and so I had a little bit of resistance from where I had been. But the key was we spent the rest of the million-and-a-half dollars on a dinner for Larry and his wife. <laughter>

Moussouris: Now wait a minute!

Weber: What really sold me on the company-- I mean, the idea of RISC technology, there was no sell there, because I was already drinking the Kool-Aid on that for years, actually. But we met over at John's house one Sunday afternoon. And Steve Blank was there. And they started talking about the idea of using foundries to create chips and create a chipless semiconductor company. And I just drove-- I was living in South San Jose at the time. I don't think I remember anything about the drive, because I was just thinking about what went on in that meeting. I just came home and told me wife, I said, "This is *really* it!" I was just blown away…

Moussouris: An SGI was already beginning to do that. The geometry engine.

Mike: The famous paper, "The Chipless Semiconductor Company."

Stritter: "Fabless."

Hennessy: Fabless.

Stritter: "Fabless Semiconductor."

Hennessy: Sort of same era. I mean, people were beginning to push it. I mean, the notion was that Mead/Conway would make this possible and things like that. But it turned out in the early days, it was a much more fragile relationship.

Moussouris: It was hard.

Hennessy: It was *really* hard! And most of these companies, they didn't want to sell fab services. So it was really tough early on.

Mike: So your first impression when you arrived.

Weber: The day I started, they were still in the copy room. They had a number of other hires from Stanford that had already signed up. But I think I was the first person to actually physically start.

Mike: Did you have the VAX in the copy room, too?

Stritter: No, so we rented a building.

Mike: I was going to say….

Stritter: It's a block from here. On Charleston Avenue.

Mike: Charleston Avenue.

Moussouris: Very close to SGI.

Stritter: Right across from SGI. The famous Tank Bunker, which had been a catheter company or something.

Moussouris: Next to the dump, I remember.

Stritter: With the smell of the dump, yeah. <laughter> I remember we moved in and John Hennessy, we drew straws or whatever. He was in charge of phones. And I was in charge of furniture. So somehow John, that night, got the phones delivered the next morning. We moved in-- there was no furniture, of course. I hadn't done my job yet. We were sitting on the floor with these phones, starting the business.

Moussouris: We also had rented potted plants on the floor. Those were the two things we had, phones and plants. <laughter>

Mike: Was it a prerequisite for the first 30 years at Silicon Valley that you had to start right in that area? I mean, that's kind of where Shockley started.

Hennessy: Yeah, it was relatively inexpensive space at the time, right?

Moussouris: Yeah.

Stritter: There were lots of buildings.

Hennessy: ACS had moved across the street to much bigger building...

Moussouris: They even gave us free rent for the first couple of months. <laughter>

Hennessy: Very typical.

Mike: How many employees you have at this point?

Stritter: Well, Larry was the first one.

Hennessy: Larry, and we had two of my PhD students. One came-- one took a leave and came right away. Steve Przybylski], and then Chris (Rowen) was kind of consulting for us, until he finished his PhD.

Moussouris: Chris had to finish in July the next year. We had Les Crudele, I guess, committed back then. We had Steve Blank.

Stritter: Les Crudele had worked at Motorola on the 68000, and then he went to Apollo and did some of the projects John was talking about. So we...

Mike: So you spent a third of your dough on the VAX. You spend probably another third on staffing.

Stritter: Sure.

Mike: You don't have a lot of money for running this company.

Stritter: That's right.

Stritter: Things were cheaper in those days.

Hennessy: Well, we had kind of a plan to get to a first-stage prototype with the million-and-a-half.

Stritter: Yeah.

Hennessy: And that was the goal. And we would get to that, and then we would get another round of funding.

Mike: Did you make that? Did you hit that benchmark?

Stritter: Well, that's part of the story. We did, actually, yeah.

Moussouris: We did. We had this-- during-- one of the effects of the argument over should we try to productize the Stanford chip, or should we do a completely new design, is that we each had to get a more and more aggressive budget and schedule. Right? Each of the alternatives had to bid down and say it could be done faster and cheaper, and so on. So we committed to do the completely new design in less than a year with ten people and a one-million dollar budget. And we actually did it!

Stritter: So this is the naïve techies from the university…

Moussouris: We actually did it, and we secured from Prime Computer, an agreement that they would prepay us one-and-a-half million dollars, which at the time was equal to our capital in, if we could deliver a working chip, running particular code on a board at a given clock rate by Christmas of that year. And I delivered the board. I lived on the East coast, my relatives lived on the East Coast. So I went to see them at Christmas, and I took the board up to Prime Computer. And I discovered-- I thought I was going to be greeted with celebratory welcome. And instead, you know, there was a guy -- I forget his name now-- with a very red nose (he'd obviously been drinking a lot) and a really pissed off expression on his face.

Stritter: At the Christmas party, right?

Moussouris: Because he had gotten the company to commit this payment under the premise that there's no fucking way in hell we would ever actually succeed in doing this. <laughter>

Stritter: That was-- he made a good bet, too.

Moussouris: Excuse me.

Stritter: But we did it!

Moussouris: Yeah, we did it.

Stritter: October '84.

Hennessy: And we had this MOU, and they would give us money.

Mike: So when you handed him that board, it was basically his career death sentence.

Hennessy: Yeah!

Moussouris: Well, the whole minicomputer business was already going down the tubes around then.

Hennessy: And we got Prime because they were in a dilemma, they had fallen behind, and they didn't have a way to catch up and get back into a top position. And they were losing customers. So they were really-- they were desperate. I mean, why else would an established company sign up with a startup that kind of hasn't demonstrated anything yet, but they did.

Mike: So it's the Bomar story all over again.

Hennessy: Yeah, yeah, that's it.

Mike: You have a desperate company throwing a Hail Mary. You guys pull it off and it ends up being more important to everybody else in the end than it was to them.

Stritter: Yeah, it helped them, too, yes.

Mike: Now were there any problems along the way? I mean, this couldn't have gone completely smoothly right at Christmas. There must have been a catastrophe here or there.

Moussouris: One big gigantic issue had to do with finding this fabless semiconductor strategy; and we thought we had some great unfair advantages, because Skip had a relationship with Al Stein.

Stritter: Al Stein was my boss at Motorola, and he had come out here a few years before, started VTI, VLSI Technology. And he was trying to build a foundry business.

Moussouris: So we were very confident that VTI would be a great foundry for us. And then our CEO, our first fulltime CEO, Vaemond Crane, had been from Sperry Univac and he was close friends with the man who ran the division in Sperry that was doing a very advanced semiconductor fab. So that was another great source.

Stritter: We were all set.

Moussouris: But I was evaluating them all. I was going with my circuit designers and evaluating all the fabs, and it turned out there was something about the relationship with VTI that disturbed me. And it turned out, we didn't really fit their business model, because they wanted to provide all their own circuits in a Mead/Conway kind of methodology. And because it was their IP that was going to be on your chip, they wanted to have the right, make it, market it, in their own catalog.

Mike: Oh.

Moussouris: And we didn't want...

Mike: They weren't just fabbing it, they were...

<overlapping conversation>

Stritter: Uh yeah, right.

Moussouris: And we knew we couldn't get to our clock rates using their Mead/Conway design methodology.

Hennessy: It was sort of a standard cell, an early precursor of that approach.

<overlapping conversation>

Moussouris: So we wanted to do our own designs, down the bare rectangles, which is what we did. And we also didn't have a business model. Since we were going to be a chip supplier, competing with this established supplier was not practical for us. And what we didn't know at the time is they already had a deal with ARM, where they could sell the ARM chip. And it was a competing chip. So that didn't feel right.

But luckily what happened is that one of the potential foundries was Sierra Semiconductor. And it was the only foundry where the CEO, Jim Diller, and the CTO, Tom Klein, directly dealt with us. And I just felt the bullshit quotient was incredibly low with them. And I decided I wanted to fab with them. But we didn't have the money to do it. The board wouldn't allow me to do it. And I went back to them and said, "I really want to work with you folks." And they said, "Well, we're really excited about what you're doing, and we want to work with you, too. So if you make just the masks, we will make the chips for free, and you only have to pay us if our yield is as good as VTIs." And we released our tape out to them just before Thanksgiving, and 17 days later, they gave us wafers that had fully working devices. They had working devices.

Stritter: Two weeks before Christmas.

Moussouris: We had to make one mask change, one metal mask change, and then we got fully working devices from Sierra. And for a whole year, after Sierra got us fully working devices, nobody else could make any chips. So Sierra was our only source.

Stritter: And they saved us.

Moussouris: They saved us. And you know, we had a big celebration.

Mike: They actually became a reliable sole supplier for a year?

Moussouris: If it hadn't been for them, we would have been dead.

Weber: Let me frame this time a little bit. It's a new chip architecture-- it's a new implementation. It's a new architecture. We did have Fred Chow's basic optimizing compiler. But none of the special optimization that went in for this particular chip. We needed to port to the operating system. We needed to get it all together. So it's not like, you know, we're really working on just the chip and really focusing on that. We had to invent on every single cylinder, and everything has to be built almost from the ground up, in that period of time.

Mike: So everything ran in parallel and it all had to converge two weeks before Christmas.

Hennessy: And it really was a thing where the weakest link, if something snapped in this, it really would have fallen apart. As in a lot of startups, everybody was jumping in to do everything they could. I still remember John (**Moussouris**) sitting there reading logic equations and checking them. I mean, it was--everybody. I wrote test software. You know, everybody just jumps in and does what they needed to do.

Mike: Tell me what hats you guys wore. Tell me your titles or was this just the motley crew making it work until you figured it out.

Stritter: You know, titles didn't matter very much. I was the Head of Engineering; John was the Head of VSLI; John Hennessy was probably Chief Scientist, I guess.

Hennessy: Chief Scientist and Salesman. <laughter>

Moussouris: And Recruiter.

Stritter: But titles and hierarchy didn't matter very much really.

Mike: Was there a mutual hire in there that made it work. Was there one guy, one or two guys that if you hadn't have found them you wouldn't have pulled it off.

Moussouris: More than one.

Stritter: More than one, yeah.

Hennessy: What was remarkable about this company is the quality of people that we hired. The first 20 or 25 people we hired. You know, Ed Hudson.

Stritter: John Mashey.

Hennessy: John Mashey.

Moussouris: Mashey was a critical hire.

Hennessy: Really, really incredible, Craig Hansen. I mean really incredibly good people.

Mike: Now this was right in the middle of a recession in Silicon Valley, right? It's always been said the best time to start a company around here is during a downturn because there's so much talent that's available.

Moussouris: That's true.

Mike: Was that critical to this company?

Moussouris: I think that helped.

Hennessy: It helped get people.

Stritter: That was critical. And a high hiring bar as well. With no compromise.

Moussouris: I know from being inside of IBM during the introduction of the PC, that IBM was really-- IBM chose Intel as the supplier of the processor, because Intel was at risk of going under. Intel was in the memory business originally, and the Asian memory manufacturers had pretty much clobbered them. And IBM was afraid to be reliant on Motorola. There was bad blood with Motorola back from the FS collaboration, and so on. And IBM actually made a hundred-fifty million dollar equity investment in Intel to keep them alive.

Mike: This was during Andy Grove's 110 percent solution era, or whatever it was. So everybody was working extra hours.

Moussouris: That's right.

Stritter: Yep, yep.

Moussouris: What happened a little bit later...

Mike: <inaudible> ...running around. The Oregon fab, because they couldn't finish it?

Moussouris: So Intel got out of memory. You know, Intel was stressed, and a lot of people left when they got out of some of the pieces of the memory business. Those events helped us indirectly.

Hennessy: But having really good people was really key. And we had the most wonderful...

Mike: How important was your board? And who was important on your board?

Stritter: So the board was Grant and Gib from Mayfield. Bob Wall, who was our first sort of interim CEO. Mayfield introduced us and he made a great career out of helping little companies get started and being their initial CEO and helped us hire the eventual CEO. And I can't remember exact timing, but one of the crew, it turned out was we were looking-- we decided we should have an external board member that would be useful. And I had competed at Motorola in this battle to get the IBM PC...

Mike: Operation Crush.

Stritter: Bill Davidow ran Operation Crush at Intel.

Mike: Now as Bill still at Intel, or had he gone VC?

Stritter: He was just leaving to start his venture firm.

Moussouris: Mohr-Davidow. He just joined Larry Mohr at Mohr-Davidow.

Stritter: So we got Bill, not only we got his money, but most importantly we got *him* on the board. Our first outside board member.

Hennessy: And then it morphed, the second time around, when we went out for more money, we had a lot of people come in. The first struggle was tough, but then the second time around was easier. Although we had some cash shortages along the way.

Stritter: That's right, yeah.

Hennessy: But we got some momentum there.

Stritter: Mike, back to the critical hires...

Mike: Having known all these guys, I mean, written the book with Bill, was there anything in particular they contributed on all of this? Were there any crucial moments where they made the right phone call to somebody, or connect you with the right person?

Hennessy: Well, they helped us hire CEOs, that was for sure.

Moussouris: Hire; fire. <laughter>

Hennessy: Hire and fire, exactly. Hire and fire the CEO. Certainly they did that. I think we were probably in all honesty, incredibly naïve about the amount of day-to-day help we could actually get from a set of board members. And beyond hiring the CEO, you know with sales marketing and all those other kinds of things that we didn't know anything about. You don't get a lot of help. And we were just naïve about that.

Moussouris: Mohr [Davidow] helped us recruit Vaemond Crane. Who after Sperry had been the head of the system's division over at Intel, to be our CEO, our first fulltime CEO. And Crane, you know, Crane had a very intense personality. But he really did focus us on schedule and on closing customers.

Stritter: Discipline.

Moussouris: Which at that particular time, you know, I find myself years later fantasizing about being able to hire someone like Crane to do what he did at MIPS. You know? Because it's hard to do, and very valuable.

Hennessy: One of the things he did when he came in, and he told us we would have executive staff meetings on Saturday mornings, because that would send a message to the rest of the team that Saturday was a working day in the company. And we would have our executive staff meetings first thing Saturday morning. He would always go and buy donuts on the way in for everybody. It was a-- but then we would be there at, I'd say, starting at 8:00 in the morning, 9:00 in the morning. And whenever, the people would roll in at 10:00 or 11:00 on Saturday.

Weber: We were definitely working six days a week. There was just no question about that.

Moussouris: The chip designers I was working with, their wives would show up in my office and say, "Is my husband still alive? I haven't seen him for three days." <laughter>

Hennessy: Oh, remember when we got the cots?

Stritter: We had cots in there. In the CAD room.

Moussouris: They would wake me up sleeping at my desk, you know?

Stritter: Sofas and cots.

Mike: What's the distance of time between incorporation and then to funding and then funding to the delivery of the prototype to Prime?

Stritter: October '84 is funding.

Moussouris: October '84; prototype, December '85.

Hennessy: Yeah. Incorporation is probably some time that summer, like August, something like that.

Moussouris: That's right.

Stritter: Just before the funding. Yeah, just before the funding, we got.

Mike: And then you delivered in December?

Stritter: December, 14 months later. Fourteen months later we delivered.

Mike: So it's 13 months.

Stritter: We taped out in November.

Moussouris: We didn't even really decide which chip we were going to do until late '84, you know, near Christmas of '84. So less than a year.

Stritter: And then there was a lot of design stuff. Larry should talk about integrating the compiler knowledge, and getting the operating system strategy built into the chip.

Weber: One of our early hires was a guy by the name of Earl Killian, who was from Lawrence Livermore, where they were working on a processor there. And what was interesting is Earl…is some of that is sort of the prototype of a lot of our people. Because he really-- although he was a software person, he was intensely interested in computer architecture, and in details of VLSI. So he'd spend all of his time over with Ed Hudson and with John, just because he was interested. And so everyone in the company was just sort interested in what everybody else was doing, in a good sort of way. One of the things that Earl did really early on, is he comes in one morning and says, "I want to show you something." And he produced this program that converted a object module and a MIPS binary into an equivalent program on a VAX. So most people use a functional simulator to simulate a new architecture. And functional simulators run at speeds that glaciers would be--

Hennessy: Fast.

Weber: I mean,they're just terrible. But we were running basically, the whole compilers at speeds of half of VAX speed, because it was so fast. It allowed us to completely create the compilers.

Mike: He did this on his own time?

<overlapping conversation>

Weber: Sort of a project.

Hennessy: He had this idea one night and...

Stritter: Saturday afternoon, yeah, right.

Weber: And it turned out that we got the compilers. We did a full boot strap every single night. So that stayed verified. So the compilers became very stable very quickly.

Hennessy: Way before we had the hardware.

Weber: Way before. And John Mashey joined us. And he went in and helped us construct the call architecture so it fit well with the operating system. And did a lot of the changes on the MMU and the cache design. So all those things were going on in parallel, so that when the chip came back, it really was sort of-- it's a non-event, because everything really had been checked out.

One of the things that Earl did that was really interesting is that the VAX architecture is a little-endian architecture; and we were designing a big-endian. So in his conversion program, he figured out how to switch the order of addressing. Which turned out to be the basis of a whole bunch of patents later on, when we started building chips that were capable of being both little-endian and big-endian. Which was important in order to get Microsoft in the end, because they wanted a little-endian architecture.

The other thing that he did was that same technology evolved into a profiling technology, we call Pixie, that allowed us to understand what was going on. And later on, we started optimizing things that would order things in the cache. We had a direct map cache, and we'd try to minimize the cache effects. And so this one technology that sort of came up sort of magically almost; suddenly resulted in getting the compilers running, getting the operating system running. And eventually, getting big- and little endian. And allowing us to do optimizations on a cache architecture that really had never been really even thought of before.

Moussouris: I think one of the really critical factors that resulted in MIPS surviving the various mistakes and succeeding ultimately, is that the three of us as founders, I think, helped to create an atmosphere where there was, you know, the spirit of cooperation between the groups, rather than competition or politics. It was, "Whatever we can do to help," and people would do, cross from one group to another. And one great example of that, Les Crudele was working in-- was managing the board operations in Skip's group. And he helped debug the 68020 by producing a board level 68020.

Stritter: TTL version before the chip came out. That was the standard.

Moussouris: So he had three or four people working with him to implement that same methodology within MIPS. But because the chip had to come out so fast, it turned-- it became clear after a while that we weren't going to be able to get the board running in time for it to really help, but he turned around, himself, and the people working for him-- and we had them coming into the chip group and helping us debug the logic directly on the chip with our simulators, and to get to this incredible deadline. Which was a very selfless, relative to his goal, and you know, it was a critical contribution to making it all work.

Mike: Did it help you guys at all to be basically somewhat naive about startups? Because what you did was so unlikely that if you had had too much experience, you probably wouldn't have-- you would have talked yourself into believing what you just-- what you accomplished was...

Hennessy: Yeah, we would have thought that we're trying to do too much. We maybe would have thought that. We would have asked for a lot more money, and who knows whether we would have gotten funding.

Mike: Was the board all over you constantly to file for these patents to get the intellectual property control of this stuff? I mean, it sounds like you were spinning off ideas by the hour. I mean, was there Brown and Bain guys sitting there with patent applications.

Moussouris: There were a bunch of patents.

Stritter: We did a bunch of patents, but I think it was the board.

Hennessy: Probably not at the beginning. We started thinking seriously about patents, but it was a little later, really.

Moussouris: We filed patents on some of the basic circuit and interface-- the inventions in the chip. You can go back and do a fast, quick search in the PTO with MIPS as the assignee and you can see the patents that started being filed back in ’85.

Stritter: A lot of good papers were written by you guys, too.

Weber: Yes.

Moussouris: And we’d published papers that won awards, both of the phase-lock-loop interface, like I mentioned before-- and also for the self-timed multiplier – we had a multiplier in both integer and floating point multiplier that was running six or eight times faster internally than the external clock rate of the chip. So incredibly fast arithmetic, both integer and floating point. And we won best paper awards for that as well.

Mike: So you make the Christmas delivery to Apollo. At that point, you now...

Moussouris: Went to Prime. To Prime.

Mike: And you are now a real company. You have a product, you have a customer. You have cash.

Moussouris: I think we had to threaten to sue them to get some of that. <laughter>

Moussouris: Yes.

Hennessy: We were close to being a real company.

Mike: And by the sounds of it, you probably were lining up with all the other creditors of Prime just to try to get your cash before they disintegrated. So this is a crucial moment in a company's history. Up until now everybody's on a team, you're rocking and rolling, you're not even looking up. You're in the trenches working away. Now all of a sudden, you've got a product and you got to start delivering, and you got to start getting your salesforce geared up and getting out there. What was that transition like? Did it change the company? I mean, there are moments when companies have to change. Was this one of those moments?

Hennessy: Well, we needed more than one customer, because we knew Prime had a lot of problems as well. So we were still, we were looking for other customers throughout that same time.

Moussouris: SGI was-- I think in the first year of shipment, they were more-- they took more product than everyone else put together.

Mike: Well, Clark had always been watching you guys all the way along, right?

Moussouris: Well, there's an interesting story here, but in mid-'85, I got a phone call from Jim Clark, out of the blue, telling me that he felt like he ought to let me know that SGI had decided to go with Fairchild's Clipper. And I was just astounded, because I had assumed that because of the closeness of the relationship, the common board of directors and all that, you know, I had assumed that this was a locked in relationship. And I said to Jim, "I think you may be making a mistake that could be lethal to your company. And before-- and let's get your top engineers and our top engineers into a room together for about a day, say, to make certain you have accurate, up-to-date information. And if at the end of the day, you still feel this way, so be it. But let's do that before you go down this path." And we had that very intense set of meetings, and we turned them around, and secured them as a customer. And thank God for them as well as us.

<overlapping conversation>

Stritter: Oh, yeah, absolutely, yes.

Hennessy: So we had challenges there because we built the initial chip, but we really needed a floating point chip. At first we were going to do a board, because we didn't have time to do a chip. And that was one of the reasons Clipper had done a floating point unit earlier than we had, so we had to recover and do a floating point unit real fast.

Moussouris: So we didn't get any sleep the second year, because we were doing the floating point chip.

Mike: You're dealing with Jim Clark, and you're competing against-- who was running the company? Was Corrigan running Fairchild at this point?

Moussouris: It was...

Hennessy: Sachs was the person running the project.

Moussouris: Sachs was the head of the Clipper project.

Hennessy: Who'd come from Cray, I think, actually.

Moussouris: Yeah.

Mike: You're up against a big experienced company.

Moussouris: Yeah. Fairchild had been the predecessor to Intel.

Mike: Yes, yes.

Hennessy: The good news was we had a lot of system expertise, and they didn't. And that, I think, made a fair amount of difference.

Moussouris: And compiler.

Hennessy: And compiler.

Moussouris: That was a big event.

Mike: So is Year 2 just we're going to give SGI what SGI needs. Is that what the business was for Year 2.

Moussouris: You know, on the chip side we had to do the...

Hennessy: …the floating point.

Moussouris: The floating point, and we had to finish the compilers and get the OS fully debugged. You know, John Mashey had been at Bell Labs and was one of the top UNIX people in the world, and that helped a lot.

Mike: Did you have any other customers? Or were you just burning that million dollars from Prime?

Moussouris: Well, we did get nine million dollars. Vaemond Crane closed around in May or June, in middle 1985 of about nine million more.

Mike: Who'd you get? The same guys? You went back to Mayfield, Mohr Davidow, who else?

<overlapping conversation>

Moussouris: KP led it. Kleiner Perkins led it.

Stritter: Merrill Pickard.

Hennessy: Sutter came in.

Moussouris: Yeah.

Mike: Who was running it from KP?

Moussouris: Well, a strange thing happened where they put Brook Byers in, even though he was the "bugs and drugs" guy as they say over there. I think it was because they needed to maintain some plausible deniability about the influence of Sun on us. You know, because they...

Stritter: He eventually resigned from the board, because of the potential conflict with Sun. We had Brook for a while.

Moussouris: You know, Sun, we didn't know it at the time, but they had committed this SPARC project with AT&T and Fujitsu. So we were headed for a collision course. And I think that became an issue that both Sun and SGI went public in the '86/'87 timeframe. '86, I think. So you know, our board ended up--

some of the board members had to balance their fiduciary duty to do really well to their fund on the one year after IPO-- what's it called?, the distribution dates for their limited partners on the workstation IPOs versus what we did and I remember, you know, like the DEC Station, you know, when the DEC Station came out from DEC, we secured DEC as a critical customer. When the DEC Station 8900, I think it was-- no, the DEC Station 3100 came out in 1989, Sun's stock took a bit of a beating. And we could have come out with that same product two years earlier-- or at least one year earlier. And I think the board was a little conflicted over not wanting us to do that, because they had a fiduciary duty to their portfolio.

Hennessy: Yes, I mean, it was difficult. We were trying to get other customers, I think, and part of the difficulty was, you're putting your architectural dependence now on this very small company, and everybody realized that that was a big concern. And there's always this macho attitude of, "I own my architecture." I mean, particularly among companies like DEC. And you know, for Sun, they were saying, "Well, we could do this thing with MIPS," and there were various reasons why they might or might not do that. "Or we could go do our own thing based on the Berkeley RISC work."

Mike: And we forget that's the era when everybody-- they were verticalized. They went all the way up to semiconductors, and all the way down to chips.

Hennessy: And they were all the way down, right. And I think we-- you know, for better or worse, what never quite happened, although we had a series of discussions-- we never quite got any of the big players to converge around one of the RISC architectures. We had a very close call with Motorola, they almost instead of doing the 88000 series, they almost came onboard. I mean, we were close to signing an agreement with them. We had several deals like that. If one of those would have gone, there would have been-- instead of getting four or five RISC architectures, which, then, of course, most of them ended up dying, we would have been able to have one early on that would have been triumphant.

Mike: I'm intrigued. You guys simultaneously had to sell a new product, a new company and a new overall design philosophy, at the same time.

Stritter: Yes.

Mike: What was your pitch?! I mean, it seems to be when you multiply all that risk, with a K, you guys are a bad bet!

Moussouris: You know, I do have to say there is…was a simple way to give the pitch. I don't remember exactly. It may be rewriting history to imagine that we had it back then. <laughter> But, I remember-- I mean, there is a simple way. If you look at our R2000, 3000, 3010, so on, the early chips at MIPS, the bandwidth at the pad ring was four times greater per cycle than any of the other-- than any of the CISC chips from Motorola or Intel. Twice as good as SPARC. And in essence, everything we did in the micro architecture, and in the software, in the OS, in the compiler, all amounted to using that bandwidth just as efficiently as the CISC machines used their bandwidth. You know, maintaining the same bandwidth efficiency in ultimate performance.

Hennessy: But in the end, that wasn't the problem. Because I think by then, John, we were winning the technical arguments. We were winning the technical arguments about the design. The problem, which I think we never understood, Mike, was how much the other potential customers saw the risk of this tiny little company. And we didn't see it, because we were killing ourselves! We were working so hard. We were very confident in what we were doing.

Mike: Well, if they had looked through you, they would have seen that the risk was even greater, because you were dependent upon a whole new manufacturing philosophy.

Hennessy: Yeah, absolutely.

Moussouris: Right.

Mike: So there's like four major areas of risk compounded.

Hennessy: We had relatively small companies doing our fab then, and not any of the big guys, right?

Moussouris: I think one of the things that I want to point out about this, the marketing department at MIPS answered the question, "How can you believe in this small company?" by saying, "Well, it's based on the work of the greatest engineering university in the world, Stanford University." So the credibility of Stanford University was a big comfort factor for our customers. Now the design we had done was not the Stanford design. I mean, if you look almost everywhere, Wikipedia or the MIPS website or anywhere, it all says it's the Stanford design. It was not the Stanford design. The downside of that, the people who were not associated with Stanford, were harder to retain within MIPS. And most of the senior people left between the time that the 3000 was done, and the time the 4000-- there were other reasons, too.

Hennessy: There were other things going on. <laughter>

Moussouris: There were other things. 6000, so on. But….

Mike: Did you guys ever consider a second source supplier?

<overlapping conversation>

Moussouris: We did, oh, we did.

Stritter: Sure, oh, yeah.

Hennessy: We did, yeah.

Moussouris: Oh, we had lots of them. Skip succeeded wonderfully getting second sources.

Stritter: We'll tell that story in the next phase, but we tried to get all the big domestic vendors. We almost got Motorola, and we had other discussions and we'll go through them. In the end it was the Japanese who gave us the credibility of world-class fabs. That was a couple years later.

Moussouris: NEC, Toshiba, Sony, all signed up. Skip signed them up.

Hennessy: We couldn't get Motorola, because they had had one customer, and that customer was in dead competition with several of the customers we had already gotten. So they couldn't do the deal.

Moussouris: I think it was Data General. Yeah.

Hennessy: And what happened was, Motorola was told by the customer, "We'll sue you if you go that way."

Moussouris: So that's the second session story.

Stritter: Yeah, we'll tell that story.

Weber: I went out to Austin, and I forgot the guy's name.

Moussouris: Oh, yes!

Weber: And met with him, and the guy calls me up on Sunday morning and says, “It’s a done deal. We’re announcing it tomorrow.” This was like ’88 or so. But it was like, we were all ready to celebrate, and then it just fell apart. It was like dust.

Hennessy: There were a lot of close calls like that.

Moussouris: There’s one thing I want to say, I think to help me overcome my bias, you asked the questions earlier, “Could it have been done two years earlier?” And I’m biased to say, “No, because I wasn’t around two years earlier.” <laughter> However, I think if you really take a step back and you look at what turned out in the end to be MIPS’ most important competitor, it was ARM. And ARM was a spinout from Cambridge University in England. That really did spin out the university design initially. And it was before MIPS that this happened. Before we started MIPS. And they got VTI as their foundry, and marketing ARM, etcetera, etcetera. Now they had horrible-- they almost died multiple times over. You know, they had big investment form Apple, from VTI, from Olivetti. They had restructured. They almost-- but in the end, because they were so distressed, when MIPS was less distressed, they got the design win in Nokia cell phone before anyone knew Nokia would become the world leader. And that became the foundation of a five billion unit per year franchise now.

Hennessy: I think, John, we thought of ourselves, MIPS thought themselves as CPU supplier to big computer companies.

Stritter: To computers. To the computer business.

Hennessy: But ARM, maybe flirted with that, but I didn’t last too long. And they moved much faster in the embedded space. If we had moved into the embedded space faster, we could have secured a...

Mike: Were you guys thinking of that at that point?

Hennessy: No. In fact, it was really the SGI.

Moussouris: Well, the software made it a good host processor. The fact that we had good UNIX and good compiler, and it was a uniquely good host processor.

Mike: Did you see any of the ARM guys when you were over at Oxford?

Moussouris: No.

Mike: You didn't get down to Cambridge to see them.

Moussouris: I got to Cambridge, but to [Seven] Hawking's Group, you know, I was in physics, not computer science while I as there. So no, I didn't know the ARM people. But I think if MIPS had tried to spin out a year or two earlier, it might have had a course like ARM, and if it had done the deal with Nokia instead of ARM, it could have been okay.

Stritter: We were all groping for a business plan.

Hennessy: Nokia is a lot later, John. That's much later.

Moussouris: That was much later. Ten years later. So you would have had to survive for ten years of misery.

Mike: So you spent Year 2 chasing customers and trying to convince...

Stritter: And make a floating point chip.

Mike: To make the move, and make the marriage. Give me the resolution of that.

Moussouris: We did a floating point chip that was able to do the work of the geometry engine faster than an entire geometry engine chip set-- there're like five or six of them, you know? There was a whole pipeline of geometry engine, GPU chips. And the one MIPS floating point chip could outdo. And it was much more programmable, so that's why we got the Playstation design win, I think, because it's much easier to program foreground action in a 3D graphical video game on a host CPU than it is on a GPU, a graphics processing unit. And that's what we did with our floating point chip.

Mike: So when did SGI finally embrace you guys?

Moussouris: Oh, like...

Hennessy: I think pretty early.

Moussouris: In '85.

Hennessy: They were-- and they were looking at side bets along the way if we didn't deliver. And Clipper was one of them, and things like that.

Moussouris: Which they should have.

Stritter: That's good business.

Moussouris: But I guess we delivered and they shipped probably by some time in '86, they were probably shipping MIPS based stuff.

Mike: Was there a moment when you guys felt like you were a real company? I mean, there's this experience where you're a startup team.

Moussouris: We were always a real company! <laughter>

Hennessy: Once SGI actually shipped through, and there were workstations out there with MIPS things in them, then all of a sudden you feel real. You see these real machines going out and people using it. So that made a big difference, that ship through. Because what happened in the end with Prime. What happened in the end? SGI ended up supplying Prime. That's what happened in the end.

Mike: Was there a shipping moment where you celebrated. You all went out and got drunk.

Moussouris: When the chip worked.

Hennessy: When the chip worked, we had a celebration, certainly.

Stritter: I remember when the chip came back and it was evening-- it might have been a Saturday, I'm not sure. And Danny Freitas

Mike: After your Saturday meeting?

Stritter: Danny Freitas had set up a test jig. But he knew that everybody would be hovering over. So he didn't allow anybody in the room. And he went in there, and he disappeared into the test room. There were probably some other people with him, I don't know. And then...

Weber: It was like having a baby. I mean, everybody was pacing outside.

Stritter: And we didn't know what to do. Exactly, right. And all of a sudden...

Moussouris: Does it have all its fingers and toes? <laughter>

Stritter: All of a sudden on the PA system, on the PA system comes Danny's voice, and he said, "One plus one equals two." <laughter> And it like erupted! And then 30 minutes, 30 minutes later, you know, "Two plus two equals four." And we were up and running, because it worked.

Hennessy: It was amazing when you think about it.

Stritter: One year after we got our money.

Mike: Let's put everything in context, since we've run out of time here. What did you learn from this experience? Personally, that you used later in your life. I mean, you all went on to distinguished careers in a very wide array of careers. I'm curious what you learned from the MIPS experience?

Hennessy: I can tell you, I had NO idea of how business -- how they were run, what they-- I didn't know what a gross margin was. So we learned a lot of things about finances and things like that, certainly. I learned some things about management. Recruiting people. Sometime later we had to do a layoff. Tough situation. And you know, a startup company, you do a layoff, it really destroys the morale, and how do you rebuild the morale?

Mike: When did you do the layoff? What year?

Moussouris: Oh, it was much later.

Hennessy: '87?

Weber: '87?

Hennessy: Yes.

Moussouris: I think what learned from, especially those first two years was that a very small group of people with very modest resources, united to succeed at something against whatever obstacles happen to be there, can beat hundreds of squabbling geniuses with a hundred times more money. You know, that all it takes is a small group that's really united to get the thing done. That you can do amazing things.

Weber: Motivation is a real multiplier. And large companies, the motivation isn't there. It's very dispersed. But we were very motivated, very focused. You know, I don't know how many times faster we were doing, but without a doubt it was the most challenging, most enjoyable experience I've ever had in my commercial life.

Stritter: What you learn is you surround yourself with the very best people, with no compromise. And then you can do these kinds of things.

Hennessy: And if we hadn't done that, I think the company wouldn't have succeeded. Because there were enough close calls along the way. If it hadn't been for the quality of people we would have never made it.

Stritter: It wouldn't have worked, yeah.

Mike: Let's go around now. Tell me the rest of your career at MIPS and what you did afterwards.

Moussouris: Well, one of the things, I should say again about the naivete of three PhDs. We didn't understand when we were setting up, like we thought that we had sold 30 percent of our company to Mayfield for 1.5 million. But we didn't understand that the 70 percent of which we had a fraction and the rest was allocated for future, current and future employees, those weren't shares. Those were options. Or options allocated. So they didn't vote, and they didn't give us board seats. You know? And there was a mechanism set in very early to shift control of the company away from the founders. That happens a lot. Sometimes it's healthy. To some extent it was healthy. But you know, I mean, John was-- I remember you saying it was an issue that you were going back in a year. But in other respects that was good for them. I think they felt-- they liked that idea.

Hennessy: Well, we had other complications, because we had to negotiate a license for some of the stuff from Stanford. So I didn't want to be on the board. Because that might complicate that license negotiation.

Moussouris: Skip and I both got moved into like-- we both had operating positions, sort of each with half of the engineering challenges of the company in the first year. But we both got moved into non-operating kind of advisory positions. And various polit-- as we grew, and particularly as we tried to absorb external groups, we absorbed this R6000 project, this ECL project, for example. Politics set in, and divide and conquer. And I remember, we all, I think, heard things about each other-- which luckily we didn't believe. You know? Divisive things. There was a divide and conquer going on. And that did make it more frustrating as the years went on.

And I ended up leaving MIPS after three years. I consulted to MIPS and its semiconductor partners for the fourth year. And then I started a media processor company called MicroUnity. And this time I made sure I did have voting control. Could control my board. And I got MicroUnity funded by all the big-- by Hewlett Packard initially, Cray, and then the four largest cable companies, you know, TCI, Time-Warner, Cox, Comcast, Microsoft, Samsung. I built a semiconductor fab. I designed media processors. But HP and Intel partnered with each other, and very strongly tried to kill off our manufacturing. You know, HP was supposed to be our manufacturer. They killed that off. Motorola also. Motorola was a big investor, too, but they killed off our manufacturing. And we ended up being forced to be an IP company. And we've been successful as an IP company. We've licensed to all the major PC and video game - our patent portfolios and we've been quite successful. But it's much harder, you know, when you do it the second time in the same domain, they see you coming. And they block-- you know, they have their own version of Crush. It's harder. <laughter> But it's been gratifying. I think the basis of the years I spent at MIPS was incredibly valuable.

Stritter: Absolutely. So I stayed. We leave this part of the story, we're out of money. It's the end of 1986, Vaemond Crane has been asked by the board to look for a new-- for his replacement. And we're about to go interview Bob Miller, which we'll talk about. But we stayed. We went-- I did. We got bought by SGI in the end. Well, I shouldn't fail to mention the IPO. The exciting IPO. And then after SGI took over the company, a bunch of us got together, we did a company with Bob Miller called NetPower, so that was another adventure, another story at some time. And then as that wound down for me, I got to do another project just like this. It was called Clarity Wireless. It was a bunch of brilliant people from Stanford. We again hired the very best people we could. We were designing WiMax, five years before WiMax, or the technology was ready for it. Luckily we got saved by Cisco, and Cisco bought the company. But it was another example where I was able to use these lessons to bring another company, another piece of technology in the market.

Hennessy: So I went back to Stanford for most of my time, and kept consulting for the company and spending three-quarters of my time during the summers there. And stayed through the IPO-- largely moving into a role of technical spokesman, because I was only down there a day a week. And continued there through the IPO, through the acquisition by SGI. Then SGI asked me to be chief architect there. So I did that for a while. And then helped MIPS spin back out when the embedded processor portion of MIPS became a very successful standalone business by itself. And SGI encountered a few bumps on the road after acquiring Cray and doing a few other things. So we spun it back out, and I helped them get that done, and then after about a year, I kind of faded out of that. And other things at Stanford took over my life.

Mike: One quick question for you. You, in terms of scale, you know, you're running Stanford University, this gigantic enterprise with billions of dollars in its budget. Are there ever days when you kind of miss the camaraderie of that little team?

Hennessy: Yes. Well, you miss the hands-on experience-- and you miss the opportunity of doing it yourself and being enmeshed in a team that's doing it. And it's a very different kind of experience. So now, replace that reward with, some new faculty member you hire that makes an incredible discovery. And you have to find that you helped that person achieve that incredible discovery, but you can feel good about what you've enabled in the process. So it's a completely different kind of experience in that sense. But the great thing is---both at Stanford and MIPS---I got to work with just incredible people. The best people you can find. And that, I think, is something that's always kept me excited.

Weber: I stayed at MIPS all the way through the integration with SGI. I kind of focused a lot of that on the software management and did some of the application work and managed some of the operating system work, and so forth. About the time the R4000 and everything started building up, I transitioned into more of a-- almost a marketing role with being sort of the technical voice behind the ACE Consortium. In the end I decided not to stay with SGI. I was in SGI for a couple of hours, to be processed out. I then went on to work at Sun Microsystems. Which was kind of a-- took me a little while to acquaint-- because they were the bad guys for so long.

Moussouris: Larry! I had no idea! <laughter>

Weber: And I ran their Development Tools Business, General Manager of that group. I then took a role, a bigger performance role in the company. I ended up running a couple VLSI project. I ran-- they have a microprocessor called MAJC, which was a very interesting processor. I even ran their JAVA engine microprocessor. Didn't go anywhere. After leaving Sun, being president of a digital set top box company, doing commercial set top box for the cable industry. And then I decided to retire and do some volunteer work in the general community.

Mike: Let me tell you guys an anecdote. I want to see if you've had the same experience. About a month ago I was going down to Fry's with my 15-year-old, my youngest. And I was driving down Arques, and I do it all the time. I just happened to glance over and saw the brick building. And I said to my son, I said, "That's where MIPS used to be." And he said, "Oh, okay!" He said, "But isn't that MIPS over there?" And he pointed at a new sign across the street from the old MIPS building. And it said MIPS on it. And I was astonished to see that sign! <laughter> So MIPS is back!

Stritter: Yeah, a new factor.

Mike: The long-standing truth that if you have a great processor architecture, it never, ever dies. It's immortal.

Hennessy: And it's really, I think, the truth was-- and I think this is where the decision to redesign it was really right-- because it gave us a chance to realize that some things we had done in the Stanford Project could be done better. And we could bring in that expertise, bringing in both more compiler expertise, but also John Mashey and the OS expertise. And we did something-- I mean, if you think about it, there's an architecture, it's going to be going on 30 years. And that's phenomenal.

Moussouris: 500 million a year is the number of MIPS cores that are manufactured.

Mike: How many MIPS cores do you think are in use out there?

Hennessy: Well, it's kind of hard to say how many. A few billion.

Moussouris: Yeah.

Mike: Which means that you guys helped create one of the most ubiquitous products in human history.

Hennessy: Yeah!

Moussouris: Yeah. And now what's happening, it's interesting, you know, the whole center of gravity that back then was in the workstation world, and then it went through the PC world, now is moving to smartphone, and smartphone are moving to broadband, and MIPS has dominated broadband, set top boxes, and Blu-Ray players and Atheros WiFi chips. And you know, MIPS is in Cisco routers and all of the -- so broadband has been dominated by MIPS. Cell phones have been dominated by ARM. And now there's an interesting contest being re-fought between ARM and MIPS. And MIPS appears to be slowly catching up on ARM. Is getting … just in CES, there were a bunch of MIPS-based smart phones being announced.

Hennessy: There we go. So.

Mike: Wouldn't it be ironic if MIPS' best days are yet to come?

<overlapping conversation>

Hennessy: It would be ironic. That business is so important now that there probably will be two major architectures, because it's such an important business. As we move to consumer-- I mean, let's face it, these things are going to replace a lot of what we usually were used to, whether it was desktops or conventional laptops. That's the future, and whether it's tablets and smart phones and things. That's where it's going.

Moussouris: There are predicted to be 2.2 billion mobile broadband users by the year 2014. That's only three years from now. And most of them will be mobile broadband platforms, not PCs or laptops. And they will-- it's up in the air as to what the fraction will be ARM versus MIPS.

Stritter: Interesting, like the fabless semiconductor strategy that we were groping for, that's what they all do now. It's well accepted and well established.

Mike: And because of what's happened recently, this, in many ways, this is your bit of immortality.

Hennessy: Well, we helped propel that along. We helped figure it out. But it was really hard then, Mike. People didn't understand this business plan. And I know that we'll talking about a lot of that in the next session.

Moussouris: I want to mention three high points for me in my MIPS experience that relates to this. A physicist friend of mine invited me to Fermi Lab, at the time that the biggest physics experiments in the world were there. And we toured, you know, there's this big apparatus down in the basement, and there are computers going up four stories, that are analyzing the terabytes of data coming off this apparatus. And on every level there were MIPS machines. Even, you know, they had DEC MIPS, they had SGI MIPS, and they even made their own multi-processor choosing the MIPS processor for their own design. And you know, so I realized, gee, I've helped my physicists friends here. That was one.

The second one was when we started MicroUnity, we needed-- instead of buying a VAX, and we also had to rent an Elxsi and buy all these Apollos, you know, we got one $20,000 MIPS server, and it had more simulation capability than a half-million dollars worth of VAX's and other things. That was a peak experience.

And the third was my six-year-old niece telling her friends in grade school that the Sony Playstation I had a chip that her uncle had helped design. And "Oh, wow!" You know? <laughter> Those are really big-- those are experiences where you feel like, "Wow! We did something," you know? It affected the lives of people. Not just physicists, little girls, you know, engineers, everything in between. You know?

Mike: Well, gentlemen, we've run out of time. Thank you very much. I think you've certainly added to the store of knowledge that we now have about MIPS.

Moussouris: Thank you.

Stritter: Thanks, Mike.

END OF PANEL

