

2011 Panel Session

"Making Great Products Great"

--transcript--

Well, thanks, everybody, for joining. My name is JL Gray. I'm vice president at Verilab based in Austin, Texas. Some of you may know me from maybe reading my blog, Cool Verification, or some of my short missives on Twitter. Other folks may know me because you've attended one of my workshops or presentations and still others of you may have ' I may have gotten a chance to work with you on many of the client projects that I've been involved with while at Verilab, and as I've been working and doing all these things across the years, I've really had some questions

There's some key things that I wanted to make - wanted to understand about how to be successful with a new product development. What does it take to go from being the doer going down, sitting down, writing some code to actually achieving, creating, leading a team and making a great product with a great team, right? And that's what we've - I'm very, very pleased this afternoon to have a great lineup of panelists who'll be able to help us address this question.

As we get started here I just wanted to check a little bit about folks here. How many of you are verification engineers? Right. So this is the - as I told the panelists, this is the prime - primary audience. Any designers? A few. Software? Managers? Yeah. Press? Any press? Analysts? I see one. There she is. Watch out. Okay, there she is over in the corner. I'll help you identify them.

So all that being said, as we're doing the panel we're gonna start off with some introductions and things and about in the middle I may give you a little bit of introduction and say, "It's time to come up and ask questions if you like." There are microphones on both sides. Please feel free to come on up to the mic and ask any questions. With that, let's get started.

So the first panelist that we have is Mike Leary. He is corporate vice president of AMD and let's see here. I gotta - just gonna go through some of these - this extensive background history, so I'm just gonna read a little bit of it here to give you a taste. So Mike Leary is corporate vice president of AMD managing the development of array and mixed-signal IP for AMD's CPU, graphics, and South Bridge programs. Mike's been with AMD for nine years and he manages circuit design and SoC development for the Athlon 64, Opteron 64. Prior to AMD, Mike managed VLSI efforts at Procket Networks, Chromatic Research, and HAL Computer Systems, and he started his career at Digital Equipment Corporations, and one of the things that Mike told me about - I was asking for some statistics about the types of products that his team has done and his team is responsible for 500 million units sold over the last - since 2002. Is that right?

Mike Leary:

That's correct, over 500 million.

JL Gray:

Over 500 million, yeah, and I - that's outstanding, so thank you, Mike, for joining us. Next up we have Andy Wright from Cypress. Now Andy is a VP of design at Cypress Semiconductor. Earlier in his career he led a succession of major development programs including the PSoC 3, the largest in Cypress history. He has 15 patents, a BA in - now it says math, but do you mean maths or math? Just -

Andy Wright:

Math. Irish.

JL Gray:

And a BAI in EE from Trinity College in Dublin. Now I happened to get a chance to get some information from Paul Keswick. Paul was the executive VP of new product development and IT at Cypress, and Paul says this about Andy. He says, "Andy is one of the smartest and highest energy people we have at Cypress. He started his career as a CPLD design engineer and had an immediate impact on our projects.

"He has become an expert in many product areas - CPLD, TCAMs, PHIs and now PSoCs. His expertise of our PSoC development team has helped to transform our company. PSoC 3 is a revolutionary product and what we have

learned while creating it has driven Cypress to the forefront of SoC design and verification." So that's Andy. He had no idea apparently. You didn't write that yourself, did you?

Andy Wright:

Did you mix those up?

Mike Leary:

[Laughter] All right. Next up we have Kevin Thompson from Synopsys. Now when you think about Synopsys, you're thinking VCS, the various other tools, which I'll not name 'cause I'm sure I'll slip up and name the wrong one. Kevin has a different background. Kevin's background is in optics and he worked at a company called Optical Engineering Services. Is that right?

Kevin Thompson:

Optical Research.

JL Gray:

Optical Research, yep, and which was recently acquired by Synopsys, so Kevin is the group director of research and development optics in SEG at Synopsys and he joined as part of an acquisition. While he was at Optical Research Associates, he was the lead optical designer for the optics used to test, validate, and verify the new mirrors in both the COSTAR and the WFPC2 cameras used to make photos for the public and science as part of the successful Hubble first servicing mission, so he is joining us today and this is a real treat to give us a different perspective from what we would normally hear about from those of us doing the other kinds of chips and things.

Kevin is a fellow of the Optical Society of America and a fellow of SPIE and he was recently appointed to the OSA's board of meetings. Kevin is a graduate of Optical Science Center and the University of Arizona in the class of 1980. Yep. And one thing I wanted to demonstrate and, Kevin, maybe you can explain this for me. Kevin sent me an interesting slide. Can you explain what this slide is Kevin?

Kevin Thompson:

Sure. What you see on your left is a photograph taken from a very large telescope on the ground. It was taken around 1990. This particular photograph was chosen because it had the same format as the Hubble was going to have when it was launched, and they chose it originally to demonstrate an area of space that had the least number of objects, so this is a very dark place, and on your right is what the Hubble saw after it was repaired. Now in the interim when it was not repaired it did slightly worse than on the left and so one of the big disappointments with the Hubble when it was originally launched is it literally did not see anything new to science. On the other hand, once the repair mission was complete everything you see on the right was new to science and had never been seen before, and this is a region with very few objects, so you can get some perspective on just how dramatic the Hubble was to science in that sense.

Kevin Thompson:

Great. Thanks, Kevin.

[Applause]

JL Gray:

Next up we have Mike Frazier, so Mike is an 18-year veteran of Xilinx, joined the company in 1993. He's a corporate - or he's a vice president of IP solutions and leads engineering teams responsible for the Xilinx portfolio of embedded, connectivity, communications, DSP and video IP products, which covers quite a broad range, I guess. What sort of things are in that -?

JL Gray:

Right. And Victor Peng, a senior vice president at Xilinx says this about Mike. Says, "Mike and his IP engineering team are one of the three primary legs forming the foundation for Xilinx products: silicon, software, and IP. Mike's team's primary charter to deliver a broad range of IP products from basic IPs to _____, FIFOs, standard connectivity protocols," et cetera. He doesn't actually say "et cetera," but I'm skipping ahead. "And all three legs of the foundation are required for our customers' success, so ensuring the customers can move seamlessly from software IP development to _____ is paramount. So that's Mike, which reminds me of one thing. I had - I got almost everybody.

I got quotes for almost everybody, so I apologize for those of you I didn't get quotes for, but I tried my best. Devious nature and all, I tried my best. But I do have a quote about Kevin from Dr. Howard Ko, which is senior

vice president and general manager of - at Synopsys and Dr. Ko says, "Kevin and his team have delivered innovative solutions to the optics industry as some of the world's leading developers of optical design analysis software. They have provided imaginative, cost-effective solutions across the entire spectrum of optical design with more than 4,800 completed projects. Kevin has brought his experience, accomplishment, and vision to Synopsys and is aiding us in our expansion into market such as semiconductor, _____ equipment, and cameras."

And last but not least, we have Dr. Yu-Chin Hsu from SpringSoft and Dr. Hsu is a VP of verification technology and product group at SpringSoft, a former vice president of research and development at Novas. Many of us here in the room are familiar with the Novas and SpringSoft products. He has over 20 years of research and development experience in EDA industry and academia and I guess before this you were at Avanci. Avant? Sorry. Avant, yeah, sorry. I can't read my own typing.

Yu-Chin Hsu:
Avant!

JL Gray:
Avant! Yes. And Johnson Teng says this about Yu-Chin. Says, "Yu-Chin has successfully led his team to develop the Verdi debug system, the mostly widely adopted RTLD bug system in the industry, saving engineers incalculable hours of verification." So this is our panelists and, again, thank you very much. I'm super excited about getting a chance to grill you guys; I mean ask you some very polite questions. And let's - so let's go ahead and get started.

JL Gray:
So, Mike, I was wondering if you could just tell - I mean I'm going through the bios real quick and trying to just summarize as quickly as I can. Could you tell us a little bit more about what your team does and why it's so important to AMD?

Mike Leary:
Sure. What my team does is it develops a lot of the mixed-signal and array IP and what does that mean? It really develops all of the external interfaces that talk to all the other components in a PC system, the PCI express, the memory PHIs, display interfaces, all the things that'll talk to screens, talk to disks. SATA, USB, those types of things are done on my team as well as the large array, so the L2 caches, the L3 caches that are critical for the performance of the microprocessor cores that are also developed as part of our SoC.

JL Gray:
Excellent. And what happens if you get it wrong?

Mike Leary:
We try not to think about that when we get it wrong, but typically when we get it wrong we have to dig in in the lab and oftentimes this stuff, being analog in nature, is very difficult to debug, so it really requires us to put a lot of features into these different IPs in order to give some visibility into the chips, but it requires a lot of hours in the lab if there's something wrong and a lotta simulation time, especially in these advanced processes where things don't always look like they appear to look like and they don't always simulate like they should simulate, so that really the focus is getting it right the first time and clearly being a verification conference a big part of that is mixed-signal verification flow that really flushes out the IP.

JL Gray:
So now I was gonna ask everybody sort of an introduction easy question, but you've just triggered something and I'd like to expand it out. You said "get it right the first time." How often - no, let me rephrase it so you can actually answer it and not be under some sort of NDA. How - when you expect - when you're working on a complex project what is the expectation? I mean for - and I open this up to everybody, but, Mike, I'll let you start. What is your expectation about getting it right the first time and how likely is it - or how much trouble is it to do that? Why not just let it not get right the first time, have the chip come out, re spin? What's the problem with that?

Mike Leary:
Sure. I mean the key part, I think, in any product development cycle, whether it's in AMD or anywhere else in developing lenses or gate arrays or whatever the case may be is really time to market, and if you get it wrong the first time and you have to go back and re-spin that costs you about three to four months every time you do that, so in an industry that's very competitive, where new products are introduced all the time and you have to be constantly innovating in the industry, you just can't afford that, so getting it right the first time is really critical, especially if you miss a market window. At AMD or in a graphics program if you miss a market window you're in some serious trouble, so getting it right the first time is critical.

JL Gray:

Sure. Now, Kevin, obviously working on the repair mission there's a possibility of getting it right and there's a possibility of getting it wrong. How do you deal with that when the stakes are high?

Kevin Thompson:

My - I'm not in a university setting a lot of my time and what I've come to realize - and a group like this, I think, is very guilty of this - is we're educated in a very ideal world where everything is always right, and we don't really think about defending ourselves against things that will go wrong. For quite a number of years - and I'm not actually teaching a course called defensive engineering - but let's take the Hubble as an excellent example and the bottom line right now is I basically tell people the first thing you focus on is what is in this project that has not been done by anyone before. Now in what we do there is always something, and I said, "Look, focus on that." Chances are we're do - whatever we're doing the second time, well, probably we will get right, but what we're doing the first time, there will probably - something will go wrong, and it won't be 'cause we've done something wrong, it's be 'cause we've assumed something from the past that no longer applies. What tends to happen is the assumptions change when something new happens, and if you don't revisit that, that's what'll get you.

What happened with the Hubble was in the case they were making a mirror that had not been made that big before and so when you make mirrors like that you essentially have to make an optical ruler to define the shape of the surface you wanna eventually manufacture. The ruler that they had to make was different. Before then they used regular normal camera lenses or things that looked like camera lenses to make the ruler. This time they were gonna have to use mirrors, the kinda mirrors that in fact you see in the first lithography systems from Perkin-Elmer in the '70s, their Micralign series. In fact, the person that was involved in the design of the Hubble ruler was Abe Offner. He's the guy that designed the first lithography system that was commercially successful out at Perkin-Elmer.

The situation, the science was all done well. Everything was done well. What fell apart was one day a group got together and they were behind budget. They were behind time. The customer, NASA, came in and said, "You know, we had this big test to test the big mirror against the little mirror. We can't afford the time or the money. We're taking it out." That's fine. That was a rational decision.

What went wrong was nobody stood up in the room and said, "What are we gonna do instead?" They just moved right along and by taking that test out they missed the chance to understand that they made the ruler wrong. Now the ruler, that's a little bit longer story I probably don't have time for, but it was just kind of a normal cause of business thing. It was just things happened, very fluke incident, one-of-a-kind thing, but that's always gonna happen and what got them in the process was to take that test out and not think about the implications of that. So, in fact, in that period I was on five programs that were - all failed at that magnitude, multibillion dollar failures, and I learned a lot in that period.

[Laughter]

JL Gray:

So that actually brings up a very good point and I wanted to ask Andy about. I asked all the panelists to tell me - as part of a survey I asked them, "Can you please tell me what is it that engineers don't understand about how to make great products? What is it that we really, truly don't understand about this?" And I think, Andy, you mentioned that people need to understand that they need to be doing something that's more advanced than they did before and you can't just keep doing what you were doing because then you'll - you're gonna fall behind. I'm paraphrasing a little bit with that.

Andy Wright:

Yeah, you got it pretty close, so I called it Moore's Law for engineers and everybody in the room, I take it, is very familiar with Moore's Law and concept being you go from an individual design generation to the next design generation and over a period of time you expect to get some level of performance improvement from your tools, some level of performance improvement from abstraction, but you also expect as an engineering manager and as an engineering company to be able to get improvement from your engineers, for them to be able to grow in the abstraction level that they can deal with personally and the concepts and complexity of a product that they understand how to integrate. In any given year, you take an engineering team; you benchmark that as some level of performance. If you take that same engineering team a year and a half or two years later and they're able to deal with only exactly the same level of complexity in any of our businesses or in almost all of our businesses in the room, you're falling off the curve very rapidly versus your competitors because they have figured a way collectively - that's you in the room - have figured a way to grow rapidly in the level of abstraction that an individual engineer can deal with, not just inside the tools or in the capability of the EDA industry but the engineers themselves, so rapid development of the people.

JL Gray:

All right. So but now I guess when you're actually trying to push the cutting edge, there's always a risk that you can push the cutting edge and you may be going in the wrong direction, right? You can push the cutting edge and you may spectacularly fail, right? So, Mike Frazier, I wonder if you could - what is it - in your view, what does it mean to push the cutting edge? What does it mean - what kind of decision process, what kinda calculus is going on in your head when you all are deciding different - how we're gonna proceed with new product development between this tension of, "Well, we wanna do something great, but we also wanna have something we can sell"? And where does the tension lie in there?

Mike Frazier:

Right. So that's certainly one of the challenges that my team faces within Xilinx and just a little bit of background what we do, we actually build, as I mentioned, IP products that our customers can use, whether they're building wireless or wired communications infrastructure, whether they're building the latest and greatest navigation systems for automobiles or any of the broad set of applications that FPGAs can be used. It's important that we balance the current portfolio that we're trying to get to market today, in this case the seven - six series and seven series from Xilinx, with what's coming two or three years down the road, and we have to make sure that we balance the - not only the - ensuring that our IP is gonna be optimal for that next generation architecture but also from the software perspective. As you - most of you probably know, within Xilinx we have our place and route tools and mapping tools and algorithmic tools, so our organization is also helping the software team in Xilinx generate that new set of tools for the next generation products as well as helping define the next generation architecture, what were some of the gaps we saw and why our particular IP core couldn't be optimized very well for the FPGA.

So we have to strike that balance between getting the product out today versus not losing sight of the future, both from the IC design perspective, helping Xilinx build the best, latest, and greatest FPGA architecture as well as making sure that our software team can deliver the latest and greatest software that will allow that FPGA to be optimally utilized by the customer, so I think that's one of the things that we really, really have to struggle with within our team is making sure that we innovate looking forward, looking towards the future but also not losing sight of the fact that we've gotta make sure we take care of our current products and make sure that customers can realize the time to market advantage that FPGAs provide.

JL Gray:

Go ahead, Mike.

Mike Leary:

If I could chime in there, I mean failure is not necessarily bad. Failure is bad if it reaches the final product, but that's how you take engineers to the next level is you learn we're all products of our failures and the learning that we get from our failures, so if you find out what's stifling innovation in your company typically it is fear of failure, so people have to understand that it is okay to fail. It's just not okay to fail in the final product.

Kevin Thompson:

If I can -

JL Gray:

So - oh, go ahead. Go ahead.

Kevin Thompson:

- add to that real quick. I - in addition to all the projects I did externally I started three major initiatives within this company that was purely engineers and in retrospect my idea is all three times ended up many man years of development. They all failed when they hit the street, every single one of them, and in each case the president walked in the next day and said, "Well, this is a failure, but what if we do this?" And six months later we had a very successful product, and so it's the idea of if you don't have time to figure out what it's good for but you have a good idea and you've got the commitment of the company, get it put together and then put it out there and see what people wanna do with it. You can't always predict what's gonna happen to it, but don't walk away from it the day somebody says, "Well, this isn't what it - this isn't any good for me." Find out who it is good for.

JL Gray:

Now, Yu-Chin, I was actually reading an article earlier today, as I am prone to do while I'm waiting for simulations to finish running, although that was - today I was procrastinating not finishing working on the panel, talking about how companies like such as Microsoft, and they were specifically talking about Apple, and what they were saying was that actually it's easy for a company to make a decision that they feel is making their - so they've got a product that they really love, and they wanna pursue that product, but that product has some limitations. Maybe they have another product that is - sort of addresses that limitation but maybe it's not as lucrative or it's not - you're not sure that it's gonna be successful, so like in the Apple's case they're talking about how the app store

may stifle innovation because it's - Apple has a very clear goal of what they want but they may then push out all of the third-party developers which actually make the whole platform a success. I guess the question is how do you deal with the tension between various product groups in the company who may actually have different ideas for what it means to be successful?

Yu-Chin Hsu:

I think that's a good idea. We have like twice a product planning meeting and for a new product in SpringSoft typically we have - we start with sometimes the idea coming from technology, from R&D and sometimes the idea coming from market - from product marketing, okay? So the first - no matter whether it's product marketing initiate or it's technology driven, the first step we did is have the marketing people, the product marketing people do the MRD. Software MRD is so for the market size analysis. What is the customer needs? Why customer needs this? What - how big is the size, okay? And then we have the meeting that goes through the review of the idea and then decide what to do with the - go ahead.

JL Gray:

Well, now if I can interrupt you, what happens - so the customers may tell you what they want, but I think we touched on this earlier. They may not know what they want, right? How - who's to say - I mean I have a preference for how I like to view my waves, but who's to say that's actually any good? Maybe there's something better. How - so if you - it seems like if you only just judge by what the customers are saying, how do you deal with that? Somebody said, "This is - look at this fantastic idea, quantum wave form viewer. It's great. Everybody's gonna love it 20 years from now. We should invest in this technology now before it's too late." How do you decide?

Yu-Chin Hsu:

Yeah, so the product - some is like more short-term customer knows what they want and the requirement coming from the customers. Some other is coming from maybe R&D, coming with some innovated idea about some technology they can build, okay? And the general rule of thumb is if the idea is - you - really intriguing, okay, then we will think about it and then there is Chinese-like strategies called Sun Tzu, okay, and he did a book - he write a book called The Art of War. I believe a lotta people probably heard of the name of the work, and he's talking about like five factors to be assessed for - to win in a war, okay, and if you turn that into building a product, we can probably say equivalently there are several things that you need to look at that before you think about whether even that this is not something customer want, but you decide to go ahead to build it.

And first it's trend. I mean that criteria is a standard rule. The product would be used by customers, okay, and then look at the three C's like whether you have the core competence to build it, what's the competition, and what are those customer need and then the last one is the process, whether you can build a quality product for that, okay? So the product is not coming from the customer. I mean basically you should look at what's the industry trend, how are we - internal role in a couple of years or three years will this product be used.

JL Gray:

Okay. Go ahead.

Mike Frazier:

I also add, of course, I think it was Mr. Packard of Hewlett-Packard fame who said, "Marketing is too important to trust to marketing." So I joke about that but at Xilinx we have a long history of being an engineering-centric, engineering-driven company, and one of our original IC design leads who actually passed away several years, quite the innovator, held hundreds of patents, actually his philosophy was, "If you build it they will come." And so a lot of the features that you see in Xilinx FPGAs are driven not just by marketing, and I'm being a little bit disrespectful to marketing 'cause obviously they play a critical role to make sure we - engineer don't over-engineer things 'cause we have a tendency to do that, but it certainly is important that the engineering community and the marketing community work side by side to balance what marketing believes we think is important and the engineering team to bring that innovation forward that we were talking about earlier to make sure that we do deliver products that are gonna solve the customers' problems two to three years from now, not what they're trying to solve today.

Mike Frazier:

Gotcha.

Andy Wrights:

I was gonna chime in. I think myself, Mike, and Mike are very privileged in that we live in product markets where we're isolated somewhat from the end customer, the customer's customer. At AMD is the beauty of having to have their chips programmed by somebody to suit an end application. Xilinx has the wonderful VHDL or Verilog interface that allows them to satisfy any number of end customers and PSoC is a similarly programmable product, so we have the advantage of being sufficiently abstracted from and individual end customer that we are trying to satisfy a very broad spectrum of customers and in reality, certainly for Xilinx and ourselves, we're really trying to satisfy a

roadmap of what we think end customers are trying to achieve, not even what they're currently doing or what they tell us they're going to do.

It's looking at curves of performance over time and expectation for how those curves are going to change over time and prediction of a point that you can hit on that curve rather than individual customer satisfaction. Customer satisfaction helps. You got some validation that you're going in the right general direction from a few key customers, but that's not the defining factor for a product and certainly in the case of PSoC if we'd gone with the couple of individual customers we'd used to define the original architecture on PSoC 1 or on PSoC 3 we wouldn't be a billion units in. We'd be a few hundred thousand units in if we'd picked just those customers.

Mike Leary:

I guess I would say customers don't know what they want until they see it and so you really need to show the customer what they want and what the next generation needs to look like.

JL Gray:

How do you convince - so I'll ask you how you convince your management, but now also thinking about it from the perspective how do the engineers in the audience convince you. How do you convince someone? How do you make a case for doing these things? How do you make a case for a product in the first place? How do you get started? Let's see. Who wants to take that one now? Kevin, do you wanna go for it?

Kevin Thompson:

I mean I see it from both sides because I see people coming in to us who have sold their product to their own company, but from my point of view and, again, I'm with a smaller company that originally was entirely 100 percent engineers. We hired our first marketing person in 1994 and we've diversified since then to even have an accounting person, but at least in that environment a majority of it was personal enthusiasm combined with getting a collection of your colleagues to at least buy in with you and go in as a group to whatever the team who is between you and being able to use your time and your energy on what you're passionate about. If you go in on your own it's tough, but if you can get a small critical mass, even three or four people who of like minds and like passion, at least my experience both externally and internally is that that's pretty successful, and it's purely your enthusiasm because often the people listening don't actually know what you're talking about. They may know what you're trying - your goal is, but they may not understand how you're planning to get there.

[Crosstalk]

JL Gray:

And that must be a lot _____.

[Laughter]

JL Gray:

People don't know what I'm talking about when I'm speaking, so I totally understand what you're saying. Very - well, so I wanted to - and I wanna interrupt just a second because if you now - if you wanna ask questions, the mics are here and here. We're gonna keep going. Feel free to stand up at the mic and then what I'll do is I'll just pause it at an appropriate point and I'll - we'll go for the questions. So we're talk - but we're talking about how you get - convince someone of a new product idea . In what form - I mean is it better - do you need to have a - I mean I'm thinking like an engineer. I may be thinking wrong.

Do we need to have a prototype? Do I need to have a good story to tell? Do I need to have numbers to back it up? Do I need to have sheer luck that I bump into somebody in the hallway and I can - I bump into the CEO in the hallway and I - they somehow are in a happy mood and they take it. What is - what do you have to do? I mean what is required?

Kevin Thompson:

All those things, preferably. Yeah.

Andy Wright:

I would - I'd go with an example from Cypress' past 'cause I think it's worthwhile and CMS, for those of you who don't know, is an internally funded startup at Cypress. The origin of PSoC is not the core Cypress business model. It wasn't something we were good at or what we intended to do. It wasn't the direction the company intended to go ten years ago and a very small group of people came to the CEO and had a proposal for a business and when I say "small," it was really three people, and they said, "This is our idea. We think it's very worthwhile. Nobody's

serving this space. Cypress has the core competency to drive it and to be able to sell this product. This is what we think we should do. Are you willing to fund it?"

And T.J.'s kind of a serial entrepreneur. Our CEO is a serial entrepreneur, so he basically built a model where he funds this organization and he gives them options in an internally funded startup and he will have the privilege of buying them back at some fixed cost if they become successful. If, of course, they're very successful they get to sell themselves outside and they do a real IPO or they get sold to an external venture capitalist company and CMS for several years, and this is tied back to the failure thing, was highly unsuccessful. They had created brilliant silicon. The silicon worked wonderfully, but if you try to take a programmable product to market, as any of the many startups that have gone up against Xilinx and Altera probably know by now, and it's very challenging.

It's not the silicon that's the difficult part. It's building the sales model and the infrastructure associated with selling programmable silicon in the real world. Trying to get Cypress as a commodity product sale-type organization that happened to have some expertise in the PLD space to be able to sell a broad market programmable product was very challenging. It took years. It started with three people, an idea, and getting the audience with the CEO.

We can pitch this. We think it's valuable. We haven't got any external investment but we've got an idea and some virtual product - power point. No prototype, just the idea. I think it went through all of the phases Kevin said or JL said we had to have on the path to being successful, and it didn't turn into the core direction of the company for six or seven years. It took a long time before it became a thing that the company was really willing to say it was gonna make the core of what we do.

JL Gray:

So there's an element of patience involved, it sounds like, and having the tenacity to keep going even though things don't maybe appear that they're going so well. Is that true to say?

Andy Wright:

I would say in reaction to failure two things. Everybody fails and the people who do very well fail a whole lot. It's a - that's a numbers game. The people who are extremely successful have failed more than the people who are very unsuccessful. They've just failed more times and sometimes more spectacularly, often more spectacularly, and getting up quickly would be my primary piece of feedback, right? Everybody gets knocked down. It's how fast you get back up.

I told this story to a group of engineers at Cypress several years ago. It's how fast you get back up after the knock-down and how productively you're able to react to having had a huge problem. Silicon comes back and it's a copper penny. I don't know if any of you - have any of you in the room ever had a copper penny come back? You got a whole wafer and it's got shorts. All right, there's three guys in the room. Okay, so it's a spectacularly - okay, four. It's a spectacularly disappointing day. This is about as bad as it gets.

You have spent years of your life building something. You've made commitments to a number of first past customers and who are expecting your samples on a really short schedule always 'cause it's something important you're working on, and it comes back and if you're lucky it's only been misprocessed in one of the backend layers and if you're unfortunate it was the frontend layers or it was a defect in your mask or it was a real design problem. We had a copper penny on a tape-out that was a couple of years ago and it was a .2 ohm short. I mean this thing just sank current, right? It was spectacular.

Mike Leary:

Did it light up?

Andy Wright:

It didn't light up but it did destroy the first powers of play and the first probe card on the sorter and - but how fast do you bounce back from that? We actually had a group of guys in the lab less than a day after that figure out how to get the .2 ohm short to do register rights, all right, by being brutal, mechanical in their - the applied methods. They took the probe tips off the picoprober and they took the end of the mechanical holder of the probe and they shorted it onto the wafer. They stuffed it through a whole set of pads to get to the power pad, and it had destroyed all kinds of connections on the wafer, but they'd connected it to power. They'd connected to power - the ground and they could get enough amps into the thing to get the voltage across the part up high enough to be able to get it to do reads and writes, all right? Just getting up from the problem is the fundamental thing that makes a difference, how fast you recover from the defect.

JL Gray:

Now I just wanna ask Yu-Chin, so you're in a bit of a different situation because you're doing software, so if it fails maybe it's not a big deal?

Mike Frazier:

[Laughter] It's software, after all.

Yu-Chin Hsu:

You always can get a patch out, right? Well, in a software development team that we have, the development team that's supporting the coding software to a customer and we have R&D not doing more so-called like events, research, and it's a small group that are doing - they have some idea and then try to build some prototype and usually the people, in fact, three to five people, is not a big team. Our experience in 2002 we start looking beyond Verdi. We are looking into so-called formal verification. That seems like interesting area, so we have a group of three people, start with three people, and all Ph.D., research in this area and trying to see how we can - if we can go into this area and after like two or - two years' study and feel like we don't have that kind of a resource to build a team and a system for a formal.

But to now it's always interesting is that the technology can be applied in some other area, so some of the research is turning to the product that we have later in like Siloti or - and part of the Verdi. So, yeah, so in a way it is software development. Probably you fail if - it's okay and - but I want to say something about the product is sometimes you need a luck, but if you are not there you have no opportunity, okay? So...

JL Gray:

So that's - now that's an interesting point and, Mike, you wanted to jump in.

Mike Frazier:

I just wanna talk - and getting back to the - an initial question about how do you sell an idea to your management in order to get it to be a product, and I think the key thing there is really being able to communicate your vision 'cause the truly great products are usually disruptive products. They don't exist in the industry. It's a completely different idea and getting your upper management to really take a risk on that disruptive technology really requires you to be able to communicate that vision on how that product is gonna be used, what is this gonna be different, why is it gonna differate you as a company, and that is the key thing, I think, into really selling an idea to your CEO.

JL Gray:

And they have to be in a good mood, right? I think we've established that already.

Andy Wright:

And good luck.

Mike Frazier:

Actually, just one other comment to add to Mike is that I think it also depends on the scope, and I think well-managed companies - and I think Xilinx is a well-managed company and has been for a long time - they try to push the decision levels down to the appropriate level of the organization, so you can imagine what it would take to get something like diffusing a dual core arm processor into the FPGA. It obviously needs to go all the way up to E staff to make sure you've done the due diligence, do the - talk to the customers, make sure you identify the right processor for your applications that you're trying to target, so that takes a lot more persistence, a lot more data, a lot more analysis to make sure you're doing the right thing.

On the other hand, we have this little soft 32-bit processor called Microblaze that actually was a skunk works project by three engineers out in our Sweden office, which has now grown to four engineers in our Sweden office, and they came up with this idea of building this small 32-bit processor to help just the onboard control of the customers' designs as they're trying to build their FPGA designs. I didn't have to go very far up the management chain. We heard about the idea. They told us their plan. We said go do it and today Microblaze is incredibly popular in probably all FPGA designs, so it depends on the scope again and you need to manage it appropriately.

JL Gray:

Sure. All right, I see Ambar's here with a question.

Ambar:

Yeah, actually I like the bouncing back up, how fast to, and I totally agree with that. I have one concern, though,

like how do you know whether you're failing spectacularly or just taking a huge risk? If you come to me and my chip then failed and so I was just failing spectacularly. I mean who would buy that?

Andy Wright:

I think Mike said it best. You can fail in the silicon world as long as you don't fail in the end product and you don't lose the market window. All right? People make mistakes. People get it wrong. They recover fast. They get their product out fast enough to really hit the market window. You're still perceived as incredibly successful by your management team. Now if you -

JL Gray:

Just - well, wait. Hold on. I gotta stop you, though, because if - so the market window is - what I'm always told is that the market window is so urgent that we cannot be late. Now what you're saying is there is these failures and we can be late. I feel like I've been deceived. What's going on?

Andy Wright:

There are -

JL Gray:

Is there like a secret schedule that I don't know about?

Andy Wright:

There are beautifully different markets, right? And if you -

JL Gray:

Well, I don't know. I really wanna know. Is there a secret project schedule that the engineering team has not - I think that's what - there must be. Now I'm convinced now. We've just had a - it's a crack in the wall here. Is it true?

Andy Wright:

At Cypress there's not actually. We have a very, very transparent management organization and we see the same schedule the whole way through the organization and so there's not some secret expectation versus reality. There's very clear communication of the same schedule, both in the business plan as well as what the engineers are looking at for the end expectation. We manage to a tighter one than that, so we expect to have problems on the one we're managing to, and we expect to be able to recover from those defects fast enough to hit the business plan, so when I say you fail but still succeed, you fail on the one you're managing to. You still succeed in time to hit the business plan.

Mike Leary:

Yeah. I mean I think the key thing is having the processes in place to actually find when you're gonna have a problem and then the real hard part of - in the management space is how do you recover from that, right? And I think you're - that's the point you're getting to is it's not always easy, right? And if you look over a course of a project cycle there's also it's the little fire drills that happen because of these little failure, and it's your ability to manage those things that really make the success.

Andy Wright:

Thank you.

JL Gray:

So, Kevin, is there a secret schedule? You've been on government projects so we've all seen "Contact", right? Why build one if you can build two for twice the price? Is it - how does this work? Is this -?

Kevin Thompson:

I think I'll put a little different spin on it. I get the 100 companies coming basically 2 a week, and they're coming from everywhere with their little idea, and one thing I've learned is that typically they come in and their idea has some goal for performance that's roughly ten times beyond the money that they've brought to the table and that's fine, but you - they don't wanna know that on Day 1, so for the first couple of weeks my job is actually to slowly educate them on whether they have enough money to actually take a position in their market that's financially viable or whether they probably should go back home and keep working on this before they really commit, and so for about five years my job - personal job was to minimize the number of customers that went bankrupt while working with us.

[Laughter]

JL Gray:

Was that on your Web page _____ like -?

Kevin Thompson:

And I did pretty good. There was about one a year. One out of a hundred I would not and to be honest, about five percent of the people I would quit. I would say, "You're not gonna be successful. You're not listening. I'm not gonna be affiliated with your failure." And -

JL Gray:

But what did you do if your - now you had that flexibility being in the position you were in in your company and being able to have a lot of possible clients.

Kevin Thompson:

Yeah, that does help.

JL Gray:

Some other folks are not so lucky. I know I've worked with clients and I hear - we all know what the schedule is and then they come back. We say, "Well, why don't you go and tell the management, the senior management, the owner? Why don't you go and - you're gonna go tell them, right? You're gonna tell them."

Kevin Thompson:

Well, one thing -

JL Gray:

And then they come back with their tail between their legs and say, "Well, it's better just not to tell them."

Kevin Thompson:

No, I - the way - I do have a lotta _____ with scheduling and I use it as a management tool, and that is you can change the schedule exactly once, so you come in and you kind of know what the guy wants and you make a big layout. We rarely work on something for more than four months at a stretch 'cause reality is hard to predict for more than - even that's quite a ways, but what I found is that let's say you get three months in and it's really not gonna happen. You get one shot to go in and say, "Look, here's what's really gonna happen." And as long as you make the second shot work you'll be okay 90 - literally over 90 percent of the time. They'll buy the second. If you miss the second one, you've lost all credibility and then you've got a real problem.

JL Gray:

Is it - you guys, is this agreeable statement or - I don't know.

Andy Wright:

Varied from my perspective. I mean the comments I made when you were asking the question when we were submitting the original bio, I said one of the things I think that engineers don't know is that delivery to promise is probably even more important than being fast and after you've not delivered to your promise two or three times, especially to the same customer, once you get beyond twice it's starting to get hard. Once you get beyond three times, you're losing the relationship with the customer very rapidly. If you get to four or five times, you have zero credibility with them. They'll pay attention to you when you show up at their door with silicon that's production qualified and run a few million units. The importance of being there when you promised is higher in many cases than the importance of being there a month earlier or a week earlier.

JL Gray:

Do your engineering teams - so, Mike, for you do your engineering teams understand this when they're presenting you schedules?

Mike Frazier:

Well, I was gonna say it - first is you got three knobs to turn, the obvious things, schedule, scope, and resources, and so you just have to pick one of those. You get those very expensive consultants from Verilab to help you out. That's one way to - which we have used you guys before. That's one way to keep your schedules, of course, but do - to your second question - "Do the engineers understand?" - I think that there is certainly transmission loss from the top down into the - to - down to the leaf level engineers, and it's up to us as managers to make sure the engineers understand the simple things that they're doing. I mean even whether it's falling behind schedule

because you under-scope your verification effort or you under-scope how much design effort it was gonna take or you made some wrong architectural decisions is important that the engineers understand that they can make those mistakes, but they need to have - ideally have upfront a contingency plan of what are you gonna do when you start running into those barriers 'cause you're gonna run into them, so the best way is do it, and this is all of course not always the case in the real world, but put plans in place.

Try to identify your risk upfront. Identify where your contingency plans and your mitigation plans are upfront so that when you do hit those barriers you can trigger those recovery plans as quickly as possible and not spend weeks and weeks and weeks trying to go to management getting approval. In theory, if you're done that upfront then you should be better off. Now I think that's one of the things that we need to do a better job in my opinion is making sure that engineers do understand the implications of you just can't slip the schedule because it could be impacting not only a particular set of customers but at Xilinx' cases we tape multiple devices in a new family one right after the other over the first year of the node when we come out on a new node, so there's that cycle that if you screw up or you get off track on that cycle you're not only impacting that one tape-out. You're impacting potentially seven or eight or nine downstream, so you gotta make sure you identify the challenges and the risks and deal with them upfront.

JL Gray:

Very good. And we have a question, so I wanna get to the question here. Go ahead.

Male:

Yes. You - we've talked a lot about how there's an idea or a vision that you sometimes have to push for, but we also talked about there are short schedule times and various things that are ultra complex. I mean and we know the phrase, "there's the devil in the details." How do you take that great idea and how do you actually break it down into those little pieces that engineers can agree on and they'll implement and say, "Let go," instead of running with their chickens - with - like chickens with their heads cut off?

JL Gray:

Go.

Andy Wright:

Okay.

JL Gray:

Kevin, do you wanna go or -?

Andy Wright:

Kevin? Go ahead.

JL Gray:

Go for it.

Kevin Thompson:

What we do, and I mentioned earlier, is we will only present the customer what we're at least 80 percent sure we understand, so we often will quote things for only the next two or three weeks and the way communicate is we have a table of specifications, and we start with one that may have 40 items in it. Only 15 may be relevant, but we bring to the meeting that table, and one of the strictest rules in the company is no communication goes either way without that table going out, and it communicates at all times the problem we're trying to solve, and the reason it's important here is as important details emerge from whatever party they fall into the table and that way they don't pull off the table. One of the best ways to lose again and again and again is to solve 19 out of 20 of the problems and save that last one that's - that you don't quite know what you're gonna do with yet and do that last because it erases the first 19 and that'll end the project more often than not.

You can't go back six weeks. You can go back six days, but you can't go back six weeks, so you've gotta have a rolling vision for moving them all forward, but you've gotta have this document, and it's a big flaw I've seen, at least in our industry. It's extremely rare for people to write down what they're doing, everything, not kind of here's 3 things and there's 20 I'm not telling you. Have them all on that piece of paper and it has to go with every communication.

JL Gray:

So writing down what you're doing is important. Yu-Chin, do you agree?

Yu-Chin Hsu:

Yeah, I agree. I think when a new idea come in usually in a software site, you have to do a feasibility study and then decide whether this is doable, okay, and if it's not doable we have to be very honest to our customers and talking about the product because in a software product different from hardware is there are some tolerance, right, like new features. You can tolerate with some failure, and the way we did we learned from a lesson is, okay, we have to tell the customer these feature are in what stage, like we typically divide it into this is the engineering review stage. This is the engineering release stage or this is general beta and general release, so it's very clear for those beta release features has to be in very high quality, especially the industry. The product we provide is for people to debug and you cannot afford that in the middle that tool will fail.

Andy Wright:

Is the question you're asking, the systematical process question about is it oriented towards an established group or towards a startup-type company?

Male:

More towards maybe just a startup maybe not type company but just a product. I -

Andy Wright:

Startup group.

Male:

Startup group really, yes.

Andy Wright:

Okay. So the one thing I'd start with is acknowledge your ignorance, so you've got an idea you're trying to push. The first thing you should do is look at the set of people who are trying to push it and say, "Of the things we have to do broadly, not even details, broadly, where are we experienced and capable and what fraction of that space do we cover?" And for the fraction that we don't cover, if it's big or you think it's significant for the initial estimate, go get somebody. That'd be the very first thing I'd say. Find how to cover the major gaps in your skills. Biggest single piece of advice after failing that myself several times.

Mike Leary:

I mean I think the other piece of that is really being able to take that idea and review it with your peers that are - if it's a startup you started the company with and that you trust, right, and really venting that idea. I mean you're never gonna know all the details until you actually start doing it, but if you have enough smart people that can sit around and really just brainstorm how does this idea play out and get it to at least a level of detail that states, yeah, there's a few things that are unknown but we think we can do it and then you go.

Male:

So do you recommend like bringing most of the disciplines that would be part of this project together at the get-go and just to hammer these things out?

Andy Wright:

If it's very early on in concept you take the three or four or five or if it's a very complex idea ten people who you think are gonna get it mostly right on intuition, very experienced, broad, capable people and do it with the smallest group of those people you can to as far as you can before you get it to a big group 'cause it's much easier to get a group like that to function together than it is to get 40 people to sit in a room and go through the details.

JL Gray:

Did you have something to add, Kevin, or -?

Kevin Thompson:

Kinda one more point that my boss taught me on this. Find one company in the world or one product that's the closest to what you're about to do. Make sure there's at least one because if you're doing something you can find no precursor for, you're way out on the limb and - but if you can find anyone then you can focus on, okay, how are we different or better than them. Do we - have we covered all the bases that they've covered? What holes do we have that they don't have? Why did they get where they are and are we gonna be able to surpass them? Have a focus for the context of your idea, at least one.

JL Gray:

So are you saying you should let someone else be the sucker and go first? Is that -?

[Laughter]

Kevin Thompson:

No, but it says the really creative ideas, the number that have no predecessor are extremely rare and you probably don't have one. You probably just haven't looked hard enough for somebody that's either doing or done it.

JL Gray:

So like the guy on Saturday Night Live, motivational speaker. Like under a bridge down by the river. You'll not survive. You'll be no good. Tommy, did you have a question?

Tommy:

Yeah. So when my very, very expensive Verilab consultants come back in from the field so we see a lot of different clients, big, small, all over the world, and there's a correlation between those that make great products fast and consistently and others that don't, and it's kinda hard to pin down always what the factors are, but I'd be interested on the panel's views on specifically the factors to do with maybe organization, the management, style, the culture, the peopeware, the other - if you had top tip - a thing you've learned about that softer side of things that impact positively making great products what would you say it was?

JL Gray:

Mike, do you wanna go?

Mike Frazier:

So that's a really tough one, really, really tough. My organization actually has - I can't count anymore - seven or eight sites around the world: San Jose, Colorado, Albuquerque, Edinburgh, Hajiabad, Singapore. We're just all over the place and of course we have telecommuters as well, so to me it's - in that kind of environment or even in smaller environments, even if you're in the same room, it's a simple thing to say, but the communication and the relationships between the team members has really got to be strong, especially during the tough times. I think it's easy, knee-jerk reaction. Something goes wrong, escalate, escalate, get your management involved and go around, and you have lots of organizational boundaries that you need to overcome, so don't try to overcome those through your normal managerial go up the chain back down the chain.

Try to figure out how you can solve the problem with your peer who's helping you work on the problem and then come forward with the proposal for what you - how you expect to solve the problem, so I think it's very difficult today and becoming more difficult today to ensure that when you have these global organizations that you have sound communication and sound relationships. Get people on airplanes. Make sure they go out and visit with each other across the world. Make sure you have good plans written down upfront. It's all basic blocking and tackling relative to engineering, but if the people relationship is not there then I think that's gonna create a barrier that's not necessary that it be there if you have the relationship with the teams.

JL Gray:

Now I wanted to ask you about that because I know in some organizations that I - I'll give somebody in the audience here a hard time without naming them specifically. Sorry. Yeah. That'll be the last time I'll do this panel. So within a big company sometimes I think there are boundaries that are set and there are rules to be followed, and I'm not in a big company and I don't have any of those rules to follow so I typically don't, but I can - engineers in a company would probably feel a pretty strong push from all of the different managers swirling about them that maybe, hey, don't - why are you talking to this engineer over here? They're not supposed to be involved in your effort or you just - you were supposed to go through me because that's my - I wanna keep power over the situation so I don't wanna let you go across to the other team. Is that not a normal situation?

Andy Wright:

I'd like to try to address both of them a tiny bit, okay?

JL Gray:

Okay.

Andy Wright:

First of all, to your question, the softer side of people and you probably remember Pete Postlethwaite in *Brassed Off*. You remember the "I thought music mattered. Doesn't beep people matter?" I put that really high on the list of things that make things successful, and I actually said in answer to your point - the first thing I said was, "Acknowledge your ignorance, the thing that's wrong, the gap in your people and find the one or two people who are gonna make you get it mostly right the first time way up there on the list." The management style thing, Cypress is a small company with a big company infrastructure. Grew very fast. Grew to a big revenue stream.

Subsequently changed business models, switched between different places, and stayed with the same infrastructure as its business unit shrank, and so we've got rules.

In fact, we've got rules out the wazoo if anybody knows T.J.'s background and has read No-Excuses Management, and all of the rules in the world are there to try to help you build products better or faster or more appropriately and nobody wrote a rule that they put in a management book or in a spec or somewhere else for an organization that they expected you to look at and say that's driven or written by a deity, right? You're supposed to look at those and act with intelligence and when you come across the rule that is not intelligent to follow you're supposed to question it, and you're supposed to say to your boss or to your boss' boss, "This is not appropriate. How do I go get the right things done for the company?" And the people who care enough - again, back to the people - the people who care enough will challenge the rules that are stopping them from doing their job properly. Hopefully they'll change them, and the people who don't care will live inside the rules all the time.

JL Gray:

Now I did wanna point out Andy told me before this panel - I asked all of the panelists to invite some folks from their staffs and he told me he explicitly forbid members of his staff from joining today, so they were - they're in correlation. Were they not supposed to know this piece of information? I just was -

Andy Wright:

There are three guys who would have heckled a little bit too much.

JL Gray:

All right. Shankar, did you have a question?

Shankar:

Yes. This is really a very stimulating discussion, and I thought I might bring up something different. We've talked about engineering. We've talked about management, but a lot of these disruptive technologies seem to come from artists who are completely crazy sometimes. How do you deal with such people and how do you nurture them or do you just let them loose and go off on their own?

Andy Wright:

Keep them in the basement.

[Laughter]

JL Gray:

Yu-Chin, do you have any comments on that?

Yu-Chin Hsu:

From artists? _____ -

JL Gray:

You guys effect - you guys have the whole artsy - the - all the cartoons and the - so I guess you guys let your artists out every so often.

Yu-Chin Hsu:

We do have artists who help us to promote, do marketing for our products, okay, so it's hard to link the artist to the product technology itself but it's important. Like when we sell the product people when you see the cartoon of Novas or SpringSoft products you can link to our products, so I think there's - it's important to promoting or sell product.

Mike Leary:

I mean I think one thing I would comment 'cause Kevin and I were chatting a little bit before the panel and I was sympathizing with him having to manage a bunch of Ph.D.'s because I realize that I have to manage a bunch of analog designers and it's almost as bad, but I mean I think to first sort of answer your question, I think you really have to understand where they're coming from and kinda nurture what they do and value what they do but at the same time directing them to get the job done, and I think that is the key 'cause a lotta times the artists and analog designers in that scenario don't like to finish . Their painting or their work of art is never done, so it's actually trying to leverage that work and that headset and that creativity and actually gear them towards a schedule and a product development.

Kevin Thompson:

I think in my experience where I have all these people coming and probably 5 percent of the calls we get are artists and about 1 in 100 of those are truly visionary people and so 99 out of 100 I decline to work with because they don't seem to be able to relay what technology can accomplish and how it connects to their vision, but every now and then you get someone who kinda gets it, but what I've found is - and I probably only worked with 5 in 25 years and the outcome was never financially successful. Every now and then we did something with technology that had really never been seen as kinda cool and stuff that you see in the cinema. We work with Hollywood every now - and some of the special effects stuff, but it's basically not possible, at least in my experience, to get them to compromise with the technology. They lock on to their artistic vision and until you catch up with every aspect of that they keep pushing. They won't say, "This is good enough. Let's go see how the world will react." And there were some points we had some great stuff to take to market and they just wouldn't budge. They wanted to go all the way to the top and that was really _____.

[Crosstalk]

JL Gray:

So now - so it's important to know when to stop.

Kevin Thompson:

Yeah, yeah.

JL Gray:

Yeah. And, again, also important to know that you're probably not special. I think that's what I heard you say is - unless nobody is -

Kevin Thompson:

Every now and then but it's rare.

JL Gray:

It's rare. So take note, words of encouragement for all of you here. Anybody else have any comments on this? Mike?

Mike Frazier:

I was just gonna say that that's - you have the CTO office for a reason. You have architects for a reason. You have RTL designers for a reason. You have circuit designers for a reason. You have verification engineers, which I believe are the most important ones. I honestly do. I'm not just saying that. That's one - we actually have - in our organization we don't really have very many dedicated verification engineers, but we encourage our designers to rotate into the verification role and we have a lotta people now once they get in the verification that's where they wanna stay.

There's as much innovation, as much impact to the product you have and verification that you have as doing design, so I'm not really saying that because of the audience, but it's something we do to keep our engineers inspired by having them work in verification, so the point being is that you have different types of people. Put them into groups where it makes the most sense. The artists go into the architecture group. The implementers go in the RTL design and the guys that wanna break stuff go on the verification team.

JL Gray:

Adam, did you wanna jump in with a question?

Adam Donlin:

Yeah, I just wanted to ask what about closing the loop? What kinda challenges do you have in measuring and attributing ROI because a lot of times we're talking about big solutions that involve a lot of different things and methodology contributions? How do you go about that part of the process once you have a great idea that has turned into a great product? How do you attribute that back to great ideas that have fed into that?

Kevin Thompson:

I can comment but I don't want to... Do you want me to -?

JL Gray:

Go for it. Jump in.

Kevin Thompson:

Okay.

JL Gray:

Yeah, guys don't - I mean, well, it's at the end now, so go for it.

Kevin Thompson:

I mean what I've found is you've gotta keep - the people involved with the product, they want it to be both financially and technically successful. They have to take some ownership of the financial model that goes with it. They can't let the corporate accounting people decide the rules under which they'll be evaluated for their financial success because there are a lotta nuances to that that can make you look bad or not bad, so at the product level I've found you really have to take financial ownership also, which is unfortunate. You really don't wanna do that. I mean it's not inspiring, but the fact is it's the only way you can make sure the company appreciates where you fell.

Now in our company, as I mentioned, we were literally 100 percent engineers until '94. That was 31 years without a single person who was not an optics engineer, and my model - I joined in '85 - was at that point the end of the year they look at the checkbook and they'd go, "Yeah, still money. Guess we did good. Here. Here it is." Then they'd start the year over. They just kinda passed it out and that was a great time in a lotta ways.

Now the problem with that way was that that was bad because they didn't reinvest in research. They didn't really understand how to go forward with strategies that were very much - how do we make the world better today, and so we invested, for example, in a product that didn't create a profit for 17 years, okay? Now it created a really nice run rate for about 12 of those 17 years, but as a business it was extremely successful, but how would you sell that today? I don't know. I mean it - I don't - it's a very tough problem. You can't sell the best ideas on an ROI basis. Nobody will believe it.

JL Gray

Very good. And I am watching the time here and we're just a little bit past. Do you think, Karen, do we have time for one more questions or do we need to wrap up?

Karen:

One more quick question.

JL Gray:

Quick - one quick question and then I promise we are done, so can you go ahead with your question?

Male:

All right. I have a quick question regarding in today's environment there is a lot of competition there. The expectations for the customers is pretty high to get great products in their hands. In that scenario, what are the panelists' idea about under-promising and over-delivering, if you guys could put some idea on that?

JL Gray:

Yu-Chin, do you wanna jump in?

Yu-Chin Hsu:

That's a tough question. Oftentimes like we overpromise to customers, okay? And this is really hard to - we don't know. I'm never - I don't - I forget how we resolved those issues, but we have to say, "Okay, we really cannot resolve - we cannot deliver the feature." But a lotta time is that we need to prepare, like, for example, develops in the product with - there are several angles, the automation, the language support, and there - the performance capacities that you have to look ahead, okay? And sometimes we need to do - divert something. You have to wait for a couple of years before the market reach the status of what you already - originally predict.

Mike Leary:

And I guess the answer I would have is customer - managing the customer is critically important, and you can argue as what's better to under-commit and over-deliver, but oftentimes that customer has to put your product into his system and that system has to be designed in parallel with your product development, so if you come in way early to your schedule he's not gonna be ready and he's not gonna be happy, so customer management is really pretty important and then having that close tie with the customers that you can have that conversation about where you are in your program and be honest with them I think is critical is having a very good customer relationship.

Male:

Very good. And I'm watching the time and I wanna be sensitive to everybody's schedules this evening, so with that I'd - if you have - so if you have continue - would like to continue discussion, I recommend coming up after the panel, I guess. Hopefully you guys can stick around for a few minutes. I really wanna thank all of you for joining us here. I was - as I was - we were talking I was jotting down some lessons learned for me, so things I can take away from what we did here, so I've got a few things. I've got managing the customer is critical. That's what we just heard.

I wrote my list in upside down order so we'll go from the early - latest to the last. Question rules with Cypress - from Cypress. Andy Wright from Cypress says question the rules. This will be on tape. Acknowledge your ignorance. Heard that. You're allowed to slip schedule once. I didn't hear any strong disagreement from folks here, so here's the company as you see them: AMD, Cypress, Synopsys, Xilinx, SpringSoft. There you go. You guys are okay. Everyone else, you're in trouble.

Of course, the corollary to that, I believe there is a secret schedule which we didn't quite get to explore. It's okay to fail. Write down what you're doing, which I really strongly agree with that, and understand exactly where you're coming from. There was a comment about defensive engineering, have a backup plan. I think that's also something that people sometimes overlook and be able to work across boundaries in your organization. Break down the barriers. And this is - I think for me this is what I've taken away. Hopefully you all have also taken away some useful tips.

So let me just go through my notes and make sure I've covered everything I want here. So I'd like to thank Karen Bartleson, the DVCon Steering Committee, Kathy from MP Associates, and several of you who helped put together this tremendous group of panelists for me. I'm truly appreciative. We are going - everybody needs to stay for just a minute though if you would like to know the results of the DVCon best paper. That is actually coming up right now. Again, thanks everybody. Thank you all very much and give the panelists a round of applause. Thank you.

[Applause]