MITOCW | ocw_esd_57_2007-10-04.mp4

The following content is provided under a Creative Commons license. Your support will help MIT OpenCourseWare continue to offer high-quality educational resources for free. To make a donation or view additional materials from hundreds of MIT courses, visit MIT OpenCourseWare at ocw.mit.edu.

IRVING WLADAWSKY-BERGER: So, we've been talking for the last three weeks, about formulating a market strategy around a complex, disruptive technology formulating. And now we'll start on the next major subject, which is executing. But before that, let me just briefly go over it.

And remember, there are different kinds of reasons you do a strategy, some more incremental. We are after disruptive innovation in this class. That's what we are after here. Not because it's the best, but it's a big world out there, and we're focusing on disruptive innovation.

And we are using, as our overriding case study, the internet. We all know that. And then we looked at IBM in particular, and the decisions we went through in embracing the internet. And in the seminar too, we talked about the key market factors, which were this.

Was the technology ready? Yes, it was ready. How did you know it was ready? Because the marketplace was embracing it, dummy. That's often the most important reason why something is ready. And even if you say, I have no idea what those people are doing out there, well, then you better get them find out because they are doing something.

And now, you may decide it's not for you, but at least you better analyze it. And that's essentially what we did, is understand what's going on out there. What are the marketplaces doing? What are competitors doing? What are our clients doing?

And once we looked at that, it was obvious that we had no choice but to go into it. And of course, once you have no choice, then you get to the head of the parade and embrace it. And embrace it, not cynically, but you really embrace it.

And then the second major part are what I would call the organizational factors, which is now you have to pick a way of framing your strategy that fits you. And with disruptive innovations, there isn't just one way to do it. And each company has to fit what best fits their competencies, their products and install base, their organization and culture, and what they have brand permission to go after. And sometimes you cannot find a total fit, but the more of the things you can fit, that is, where you want to go with where you are, the easier it's going to be to move in those directions.

And as we said, with IBM, I think the breakthrough-- and it was a breakthrough we learned by watching what our customers were doing, like UPS, FedEx, and others. The breakthrough was to say-- OK, there are a lot of people out there that are saying this web stuff is brand new. It's a whole new infrastructure. It has nothing to do with existing IT. In, fact, all the existing businesses are going to die because the new economy will take over, and all the born-to-the-web businesses will replace all the existing businesses.

Trust me, people were saying that because that was a way of getting a lot of money from VCs, which most of them squandered. I don't know how, but they squandered lots of money. And what we did, and I thought this was very smart is, as you can imagine, our clients didn't like to hear that message.

Do you mean everything I've built? I'm a retailer, and I have physical stores. I was in Paris at a meeting in '97, and I was talking to the CIO-- it was a conference of a major Spanish retailer, El Corte Inglés. It's a huge retailer in Spain.

And they said, we just remodeled all these stores. And all these people are saying, it's over. Everybody will buy from home. I said, well, I don't think so. And we said, look, everything you have is a wonderful base to build on. What is critical is that you embrace the new technologies because the web now gives everything you have in your IT world, universal connectivity and reach.

And so all this stuff you have is far more valuable if you integrate it with the web, the internet, browsers, and so on. And that's what we call the business. And that was a strategy that resonated at least with our clients.

It wasn't a good strategy for everybody. If you were a brand new business, it made no sense for you. But for a company like IBM, that made a lot of sense. We did a business, which turned out to be pretty successful. And then let me spend a minute on this slide, which I went through very quickly last time.

I think that one of the hardest things to do with formulating disruptive, innovation strategies is this notion of striking a balance. Because it is as bad to ignore the innovation as it is to fall head over heels in love with the innovation and ignore your base. It's equally bad.

If you ignore your innovation, you'll die because over time, you won't be able to get any growth, and everybody else will move into the future. If you ignore your base, you'll die because you'll run out of cash. You may be having a ball in all this innovation for a while, but eventually, you'll notice that new businesses don't make money for a few years.

They take money for a few-- you have to invest in them, whether it's an entrepreneurial business or a new business within a company. And where does the money come from to feed the new business? Well, it comes from your legacy businesses, which is where you generate cash. They have high profit margins because they are more mature. You don't need to invest as much.

They are not growth businesses, but they are cash-generating businesses. And then you can take that cash and invest it in your new stuff, let alone, leverage your skills, your talents, start moving them over. So this notion of striking a balance is incredibly important and not easy to do. It's easy to fall in either way. And there is no recipe for balance. Each case will be a little bit different, but you really need a balance.

Before we go on to the next section, any comments or questions? Please.

AUDIENCE: [INAUDIBLE] with. You have to handle your customer relationships very, very delicately.

IRVING WLADAWSKY-BERGER: Absolutely, you have to handle your customer relations very delicately because-- in fact, the ideal is to help your customers strike a similar balance because you are trying to convince your customers to stay with you for the install base and stuff like that. But you also have to make the promise that you'll take them into the future.

And that's the ideal. And in fact, if you have a really good relation with your customer, one of the things that we love about the relationship is that you are making a promise to keep bringing them new technologies for the future, while helping them continue with their present businesses.

But all that takes tremendous nurturing, which is why-- don't forget the premise, I've said all along, is that you cannot be a leader of the business, no matter how great an engineer, and not have marketplace contact. You need to do great jobs in engineering things. But how do you learn this balance?

The biggest learning is in the marketplace because by watching your customers and what works for them, chances are that's what you should be doing inside the business because in the end, you want to serve them.

AUDIENCE: In this particular case, we're talking about a type of technology that was very complementary to VIT [INAUDIBLE] underlying.

IRVING WLADAWSKY-BERGER: Remember, we framed it that way. It didn't have to be.

AUDIENCE: So what if it's not complementary when you're replacing in a way you-- then I don't know how you would do that.

IRVING WLADAWSKY-BERGER: Well, can we take an example?

AUDIENCE: I was trying to think of one, and I was thinking, OK, if he had just taken a DVD player. That's not that good of one.

IRVING WLADAWSKY-BERGER: No, no, no, that is good.

AUDIENCE: [INAUDIBLE] like that one, I mean.

IRVING WLADAWSKY-BERGER: Well, token ring, remember, one strategy is we embraced ethernet. So sometimes we saw the light. We embraced the internet. We said that was good or screwed up or whatever.

Now, if we were only in the token-ring business, that would have been a problem. But no, no, but that's a very important thing. But for example, right now, let's stay with DVDs. Right now, as you know, there is this huge battle between Blu-ray and HD DVD.

Now, my expectation is at the end of the day, the customers could care less. They just want one to win. And whoever loses should quickly embrace the other and move on. They will lose some revenues because I think you get a license something or another. But you just have to move on.

AUDIENCE: So they continue to bail out until the market says who's the winner? Or should one of them just begin to embrace technology?

IRVING WLADAWSKY-BERGER: Yeah, and that's a very interesting mating dance, which is-- I hope Sony and Toshiba, which are the champions of each, have plan B's. In other words, at the same time they say, I hate it, in the back room, they have the people that have the Blu-ray strategy for Toshiba.

I know that while Microsoft hates Linux, there are people in Microsoft that could care less, that Linux is Linux. And so they would have strategies so that should Linux gain more and more acceptance, how about Office on Linux. So you do that.

And Lou Gerstner, on these cases, he always talked about the notion of sophisticated competition because at the extreme, sophisticated competition, an example is, so you're playing a tennis match, and you want to kill the person at the other side of the net, but after the match over, you go have a beer.

And in most businesses, let's say, IBM and Oracle, we compete like crazy in certain areas, relational databases. But Oracle and DB2 are the top leaders in relational databases. But we also cooperate a lot in other areas. And Oracle, especially as they've made more and more acquisitions PeopleSoft and JD Edwards, boy, it's critical to cooperate.

And we had lots of discussions, that's why Lou said that in senior management meetings, well, Lou, are they competitors or are they not? The answer is, well, it depends. Are they on the other side of the net, and you're playing a game against them, that is, you're competing for business with a client? Then you want to take the ball and shove it down their throat if you can.

But are you working with standard bodies on areas of mutual things? Well, then you hang out together. And it's really important to be able to do this. And it takes emotional maturity. I say emotional maturity because it's really professional maturity because you have to on the one hand, really want to kill them.

I mean, if you're competing for $100 million relational database deal and they win, it's really painful. But if you say, the bastards, I'm not going to do business with them again, now you're cutting your nose. So, that's why there is really no good answer.

That was not exactly your question, but framing a strategy to your advantage, in my mind, is a huge part of innovation. And again, most of you are either engineers or honorary engineers for the purpose of this course. As we have said, it takes as much hard thinking to know how to do that as it does to solve a big signal-processing problem with fast Fourier transforms.

You really need to put your brain and think through as if you were thinking about the toughest technical problem. And if you do that, and engineers are really good at solving complex problems, you usually will do very well. So yes, and then we go here.

AUDIENCE: So it sounds like if I'm a disruptive guy, it sounds like--

IRVING WLADAWSKY-BERGER: If you are a?

AUDIENCE: Disruptive technology, it sounds like my best philosophy should be to either fly under the radar until I get big enough or partner with or complement the existing thing, until I can take over.

IRVING WLADAWSKY-BERGER: Yeah, that's a good one.

AUDIENCE: Example would be the batteries. They had, these new HydraCell batteries get charged by whatever. But instead of saying, OK, you can charge your battery, you can run your laptop on it. They say, you can charge the battery that you have. So if you're on the road, and you need to charge your battery, you can just plug this in.

IRVING WLADAWSKY-BERGER: Yeah.

AUDIENCE: But eventually, they want to get rid of the door cells or whatever.

IRVING WLADAWSKY-BERGER: Yeah, and that's reasonable because-- and by the way, both sides can know it that this is a dating of convenience because you're learning from them, and I'm assuming they are getting something from you. But they know you can part companies, and that's totally fine, exactly.

And the maturity is that you need to be comfortable with that. It's just a very comfortable thing. And I'm saying that because often when people especially, are more junior, it's like, well, but how can I deal with the bastards? We have to-- and then, especially within IBM, where different groups hate versus cooperate, they send email messages at midnight attacking.

Trust me. I don't know if you've all gotten messages from some of your colleagues saying, how dare you do this. You're an idiot. These people are trying to kill me. And the answer is, well, that's it, I'm not playing tennis against them. We're actually doubles partners. [LAUGHS]

AUDIENCE: [INAUDIBLE]

IRVING WLADAWSKY-BERGER: Please.

AUDIENCE: So I'm taking up on the need to strike a balance between not falling in love with [INAUDIBLE] and doing that. So what kind of framework can we do, What. Kind of checks and balances can we do inside an organization, to make sure that happens?

IRVING WLADAWSKY-BERGER: Well, it really gets down to having a very sober analysis of what cards you are holding. There's no substitute for that. And the cards will come different kinds of cards. To be honest, the most important one, I think, is revenue and profit.

The reason it's the most important one is if you run out of money, you can have the most beautiful strategy in the world, but you're going to die. I mean, you just will. And as I said, I think that happened to the old AT&T. They called the right shots, but they took their eye off the ball of their legacy businesses.

And as a result, remember, it's my opinion, somebody else may see it differently, they didn't have enough time to see the new strategies through. And remember, it's much easier to predict what will happen than how quickly it will happen.

For example, let me take some concrete things and some different ones. Let's take virtual worlds. You know what I'm talking about?

AUDIENCE: Second Life.

IRVING WLADAWSKY-BERGER: Second Life and lots of things. I am convinced, and so are a lot of people, that this is going to be a very big deal. But if you ask me how quickly? When will-- I have no idea. Is it two years? Is it one year? Is it five years? I honestly have no idea.

IBM is very involved. And I worked hard to get us there. And I tell lots of companies to get on the learning curve, but I don't know when it will take off. That's an example. Most of what people said would happen during the height of the dotcom bubble, that we're building a digital economy and things like that, they're happening. It's just that those people said it would happen in two years. The reality is you don't build a digital economy for the world in two years. It takes a while to do that.

And because the new things are unpredictable, how quickly-- and remember, part of-- how quickly will they start generating serious profits and cash, you need to rely on what you have now. And you need to plan so that you keep living off this while this is nurtured and growing. So cash is one.

The other is talent. Do you have talented people that you can start moving into these areas, both technical people, management talent, and so on? And I've seen people do that. If you said no, our existing people will just put on the legacy crap and let them do the money, we'll hire all new people to do the exciting stuff, that is not a good idea.

You do want to acquire new talents, but the new talent usually has zero loyalty to the company. They don't even know-- you cannot show up on day one and say, I love this company. That would be suspect. It's like you go on the first date and say, let's get married. That takes a little while to fall in love. And so you want some people that can start bringing the culture. And if you don't have anybody who knows anything about this area, then you're out of balance.

Brand and history is extremely important. In fact, I just finished a conference call with Chris Wall, who is the person from O&M, Ogilvy & Mather, that will be the guest lecture on November 1, and I'm really looking forward to what he's going to say because he's an expert in brand and things like that. That is so important.

We talk quite a bit about major screw ups we've seen. Ford fixed Jaguar. Jaguars used to have horrible engineering. Now they are very reliable. You all know what I'm talking about?

Ford screwed up the Jaguar brand because Jaguar, it's a high-end brand. And they started developing low-end Jaguars on top of Ford chassis. Well, this is as if I tell you I'm going to give you Porsches on VW chassis. And VW is a very good car, but if you buy a Porsche, you want a goddamned Porsche. And if you start bringing Porsche to downscale, eventually you break the brand. People say, what the hell is a Porsche? I don't know anymore. And they did that to Jaguar.

So you have to be very careful. Is what you're trying to do, does it fit the brand? And if it doesn't fit the brand-- I don't know --then you're probably better off having a totally separate company and managing as a totally separate company. So these are all the myriad of strategic decisions you need to make.

And they take a lot of thinking. And let me keep repeating, sometimes people say, well, you must be an MBA. Well, you can be an MBA and have a lot of respect for MBAs. But people who are really good with complex systems, when they put their mind to it are superb at these because these are God-awful, complex problems. That's why I keep saying so much how we need more and more people with talented system skills, engineering skills to be in major management positions leading these kinds of areas.

So let's move and start talking about execution. And as a colleague of mine, Steve Mills, who runs IBM software business, says, often a strategy that you cannot execute is a fantasy. And fantasies are OK but not in the business world. Then you really have a bad strategy.

So let me start by talking about some of the questions that Clay Christensen and Michael Raynor have posed. And one of the major questions about executing a new strategy is to decide what are you going to do yourself, and what do you want to partner for?

Remember, this is really important for something new because you've built incredible skills here, and you want to bring some of these skills to execute the new strategy. But by definition of new and disruptive, you probably don't have all these skills. And if you say, well, I'm going to take my time, build it up, you don't have time. So you need to decide partnering and acquisition and so on.

A key question, especially now more and more is, how about proprietary versus standard? And as you know, we're making those decisions more and more. This is fairly new in the IT world. Actually in every engineering discipline goes through similar questions.

So what do you build using standard components versus what should you build making it more proprietary with automobiles, whatever you want to? And often it's very related to the first question. We'll talk about that. What's the dynamics? How do you gravitate between them? And once you decide one way, are you stuck, or do you often come back? So let's explore some of these questions that Christensen has posed.

And as I said, the decisions about what to insource and what to procure from others has a huge impact on the chance of success for any new venture. I'd love to hear your thoughts on why you think that is. I agree totally, but why would you think that is?

AUDIENCE: [INAUDIBLE] a very good idea.

IRVING WLADAWSKY-BERGER: Basically what?

AUDIENCE: I'm saying [INAUDIBLE]. It's a ripe area.

IRVING WLADAWSKY-BERGER: So where are you going to put effort? Yes?

AUDIENCE: It's really about keeping the knowledge and the IP within your own company and outsourcing all the complementary tasks. If you want to be-- as the next generation evolves, you need to have the knowledge in-house, rather than giving it out to your partners.

IRVING WLADAWSKY-BERGER: Mm-hmm, but now, what if your partners have the knowledge you want?

AUDIENCE: Then you are in trouble.

IRVING WLADAWSKY-BERGER: Or you make them your partner.

AUDIENCE: Yeah, but there's no guarantee that they won't work with your competitors.

IRVING WLADAWSKY-BERGER: Oh, no, no, you have to assume they will. But yeah, I'll get back with-- yes, go ahead.

AUDIENCE: I was thinking about Boeing. For Boeing 787, the person who was in charge of the program had come here and given a talk.

IRVING WLADAWSKY-BERGER: At MIT.

AUDIENCE: At MIT, and one of the things that they had mentioned is they made this massive decision for 787 that they are going to outsource a lot of things across the world. And one of the crown jewels, which is the wing development, they have given out. And one of the questions--

IRVING WLADAWSKY-BERGER: Who does that, Japan, am I correct?

AUDIENCE: I think so. It's Japanese, I think.

IRVING WLADAWSKY-BERGER: Yes.

AUDIENCE: Matsushita, I think.

IRVING WLADAWSKY-BERGER: Yeah.

AUDIENCE: And one of the questions was, China, Brazil, and a bunch of others are developing their own jets and would they learn? And he actually said the same thing, which is they would never have been able to make the capital investments and ramp as fast to beat Airbus as they did with this.

And they are already making an assumption, over a certain time frame, that people are going to absorb that knowledge. Is their skill really the heavy engineering aspect, or just the system integration aspect? It's a difficult balance.

IRVING WLADAWSKY-BERGER: Let me rephrase it and see if I got the gist, which is, you would say, everything else being equal, I want to do it all by myself, but what if that's impossible? Now, you have two choices. Either you say, then I'm not going to do it because I only do things I do by myself. And the second one is, I'm going to partner with other people.

Then you say, well, you idiot, but the Chinese just want to partner with you to learn from you. The answer is, well, let me repeat, if I don't partner with them-- and by the way, the Japanese, the Koreans, the Brazilians it's not like the Chinese are particularly nefarious. And everybody wants to learn and then go their own way. If you don't do that, then you don't play.

My feeling is that when you are in a leadership position, in a management position, if you keep thinking about the problem and thinking, and I don't just mean thinking, and you analyze and you ponder, once you get convinced this is the only way to get it done, then the decision is obvious because the alternative doesn't work.

Now, you don't know that at the beginning. So often what happened through this process of thinking is you start deciding, do I have alternatives? What would be the alternative? But once you've thought it through, and this is the only way you're able to do that, it's like it's over.

Somebody says yeah, but real men-- screw it. That was 50 years ago. Or do a product that is simpler. Don't do a 787. Do a Cessna. No, I want to do 78-- well, you cannot do that. And those are very interesting debates and discussions that you have to do.

But then your point is extremely well taken because once you say, well, I'm going to partner, then often what then happens is you start thinking of your business at the next level, which is you are the best in serving the market, serving the airlines. And, we'll talk about it throughout. And these people may learn to make wings, maybe airplanes, but I'm not sure they'll learn how to have loyal customer relationships, or it will take longer. So it's a very interesting thing.

AUDIENCE: I think this question is, at the very core of defining your company, because it defines how you're different from your competitors.

IRVING WLADAWSKY-BERGER: Exactly.

AUDIENCE: So basically, if you are designing-- if you're making compilers and you're buying lots of other things, you have to know what is your core business?

IRVING WLADAWSKY-BERGER: Yes.

AUDIENCE: That it is very different from everybody else.

IRVING WLADAWSKY-BERGER: Yes.

AUDIENCE: So it's very important to know.

IRVING WLADAWSKY-BERGER: Yeah, it's very important to know and remember, any changes over time. This is very important. Yes, I agree totally.

AUDIENCE: My question is, are we talking in terms of disruptive? So if you have something that's brand new in the market, it's hard. My guess is you may not be able to partner with the best. And so the only option you may--

IRVING WLADAWSKY-BERGER: Say that again. You may not be able to partner.

AUDIENCE: You get partners, you may be able to get supplies, you say, make something new. So what do you have-- I mean, would there be any other choice but to develop it yourself? Only when it comes to disruptive. I mean, there are other cases where it's too messy.

IRVING WLADAWSKY-BERGER: Well, but usually not the whole thing is disruptive, one part. And so part of the thinking is, OK, so keep to yourself the jewel. And then there is usually, again, in engineering projects there are a lot of, I was going to say crap, but that's not the right term. A lot of other important things that surround a system, a business, a product.

And the more you know where is the jewel in that crown, the more you keep that to yourself, that's where you protect the IP like crazy. You may not even share that with your partners. You may say, the partnership stops here. You don't go into my bedroom. You can come into the living room, but that's where you stay because there are certain things you're not willing to share, or there are partners, and there are partners.

And sometimes you say, but if you invest $1 billion here, now it's different. And by the way, for example, IBM, we decided the fabs in Fishkill got too expensive and we needed to share the cost, just got too expensive. Not unlike Boeing's decision. In principle Boeing could have done that, but now the risk is this high.

You all know what I mean by a fab where you build something? They are frightfully expensive. And so a number of companies, Samsung and Sony and others pay-- not just pay. They are partners. They co-own the facility. I don't know if they co-own it, but I don't know what the arrangement is.

AUDIENCE: When you share like that, you also share a lot of the other information with your partner. So aren't you giving up--

IRVING WLADAWSKY-BERGER: You are.

AUDIENCE: So I mean, for example--

IRVING WLADAWSKY-BERGER: Gets back to Boeing, so do you want the fab, or don't you want the fab? And the only way to have the fab is to partner.

AUDIENCE: And so cost is the only driver to go out, meaning--

IRVING WLADAWSKY-BERGER: No. no, because you now have to think of a business model that lets you stay in business with partners, versus a vertically integrated business model. You have to rethink your business model. And if you cannot make the business model close, then you'll get eaten up by your partner.

So you have to decide is there-- just one second. Let's assume that your partner starts undercutting you. They learn from you, and now they start undercutting you. Is there something better you can do from them?

By the way, sometimes you make a mistake. You think I can, and before you know it, you're dead because your partner is much better than you. But at least you shouldn't die quietly. You should say, well, I think I can do better than them.

This gets played out over and over. The Chinese cars, people said, well, sorry, nothing cars. And the Western companies-- sure the Japanese companies also, I don't know if Korean companies have business in China, but I'm assuming they do. To do that, you probably have to take Chinese partners. To do that you say, why did they do it, even though they knew the partners are learning from them?

Well, because China is a huge market. And if you don't partner, you don't get to play in the market. But just like the Japanese car companies, then the Korean car companies, is there any question that the Chinese car companies and the Indian car companies will start producing very good, high quality cars over time? How can I say? That's the mating dance of business. At some level, you're circling each other. Yes, please.

AUDIENCE: [INAUDIBLE] the two important factor is the speed to market and the cost advantage.

IRVING WLADAWSKY-BERGER: Yes.

AUDIENCE: I think, personally, from my experience-- I working in Siemens. But in 2000, 2002, 2003, we decided that for architecture with Siemens and we would outsource whatever is the best come first to get in the market and have the cost advantage.

And I've seen that if you develop everything in house, it might be you're lagging the market.

IRVING WLADAWSKY-BERGER: Yeah, that's a very good point.

AUDIENCE: In a way, the fast mover will always have some advantages, second mover. But if you have third mover--

IRVING WLADAWSKY-BERGER: Yeah, you're dead.

AUDIENCE: So it's not necessary that if you develop everything, you can be the first one.

IRVING WLADAWSKY-BERGER: No, that's right.

AUDIENCE: From that point if you have a big advantage. Second point, I thought that from the cost advantage, I have seen that the same system has [INAUDIBLE] business their own [INAUDIBLE] server. But if we find it, we are in an enormous cost ratio from IBM, from Oracle, from Sun. So what we decided that the way customer wants, we'll build it. If you choose this one, this is the price. If you [INAUDIBLE] this will be price if you sign the price.

So the customer has the option and you can play with your competitor, [INAUDIBLE]. So to speak to market then--

IRVING WLADAWSKY-BERGER: They're very good. I think that's a very good point. And we'll get to speak to market when we now talk about modular components. Because as you can imagine, that's another huge thing that comes. Yes, please.

AUDIENCE: I wanted to make some comments about the fab because I worked at the official fab before I came here.

IRVING WLADAWSKY-BERGER: OK.

AUDIENCE: It's interesting. Like you mentioned, everything is shared. It's a partnership in the true sense. We share defect density, everything. But you also get information from your partner. So you know, like, half the worth of fabs, where they're [INAUDIBLE]

IRVING WLADAWSKY-BERGER: No, that's very good.

AUDIENCE: So there is shared learning. That's one. The cost of developing a technology right now is $1 billion. It's going to $2 billion for every node. And the cost of a fab is doubling. It's $2 billion. Now, it's going to $4 billion. So there is no way we can even be in this market sustainably without that.

And thirdly, we can build on businesses which normally would not have chosen us, would have chosen TSMC, because we just don't have the capacity. So a lot of businesses, for example, Xbox with Christmas, you have nothing until November, then November, December, January, there is this big peak, and then there is nothing.

And so if you are delayed by a month or so, so you have to have multiple segments.

IRVING WLADAWSKY-BERGER: That's very good.

AUDIENCE: And the fact that we have, all these virtual fabs helps us compete with TSMC. So we got our foot in the door, and a lot of businesses which would not normally have considered us. And we have some similar cost structures. But when we go to market, a lot of times we have an order desk, which is one of these fabs.

And you go and typically the firm that wins the business has the right of first refusal. So if you cannot deal with it, then it can-- so a lot of times our partners will also win business, and they can't handle it because of some other order.

IRVING WLADAWSKY-BERGER: I see.

AUDIENCE: So there's a lot of synergy there.

IRVING WLADAWSKY-BERGER: Yeah, but you said a number of very good things, but the first one I should have said, which is you pick your partners very carefully. And let's say Samsung is a partner. Samsung is a wonderful company, as you know.

So you can say, well, Samsung will learn from you. Well, guess what? We learn from Samsung. Sony is a partner. Philips is a partner.

AUDIENCE: So we have [INAUDIBLE] and USTA, so IBM chartered, Samsung Infineon is one. And then AMD, Toshiba is [INAUDIBLE].

IRVING WLADAWSKY-BERGER: How about the Boeing, did they make that point also? So let's take the wing. The Japanese are superb engineers, the most incredibly disciplined engineers. So Boeing must be learning how the Japanese do these things.

AUDIENCE: I think their big thing was they were trying to hit this date. So it was primarily time to market and capital investment.

IRVING WLADAWSKY-BERGER: Yeah, but I should have mentioned that learning from your partners is critical. And in fact, the reason I'm sticking to that's so important is because the worst thing you can do is to have, let me use, a colonial attitude toward your partner. Say, well, we need their money, and they don't know anything. What a mistake. Look at Detroit.

So it's much better to say, oh my God, look what Samsung has done, which I mean, it's incredible what they have done. How did they do it? And now you learn from them and so on or now, more and more the Indian companies. Look at what they've been doing. So how do they do that?

So the learning is [INAUDIBLE], but I cannot say enough. And this is especially important in global markets because often what happens, like, everybody in the US will more or less, they hire the same MBAs from the same Sloan and Harvard and Stanford, Wharton. So pretty much they look the same.

And that's why often the real totally different things are being done by people someplace else, China and India and Japan. And they do the business differently. And you can say, well, but they're different. It's a lot they do that's much better than you forgot.

And so in a global world, the learning by partnering with companies from very different regions is particularly important because they often bring tremendous management techniques that you never had before because that wasn't the Sloan way or the Harvard way. And now it should.

So as Clay Christensen says, so if it fits your core, do it. Otherwise, do it outside. But if you read the chapter, he and Raynor make the point that don't assume you're making a decision forever. Because often you're making the decision now. But often markets will change again, and you may need to take it back.

So make a decision for sure, but keep your mind open because you need to decide strategically. What that really says is if you-- what's the term? I forgot. When you lose all your skills in an area, what's the term? Deskill yourself, but there is another term. If you lose all your skills in an area because you have decided to outsource, be very careful because if in the future you may decide to re-enter it, you lost all your skills.

And in fact, one of the things that often happens in competition is your competitor will want you to get rid of all your skills in an area, and they can even do that by selling you stuff very cheaply until you have no more skills. And then you're in trouble. So it's just something to watch out for. Please.

AUDIENCE: [INAUDIBLE] show me that by how well your competency has gone for several hours every single day for probably for whoever doesn't need to have [INAUDIBLE].

IRVING WLADAWSKY-BERGER: Yeah, but again, sometimes you totally outsource. Other times, like, let's say, BMW, I believe the X3 they don't manufacture, but the X5 they do. Those are SUVs. And so maybe there is a balance point.

Now, for some things, you said, well, we outsource the cafeteria. Should we retain some chefs? Sometimes you would say, that is not a core competency. No, I'm being very serious. Once upon a time, companies were so well integrated that they had their own cafeterias and their own people.

Most companies now have outsourced that, but they don't need to retain. They need people in procurement who understand how to negotiate with the best and so on, but otherwise, they don't need that. Manufacturing, marketing, you just have to be a little more careful because it's so core to so much of what you do that you have to be careful.

And again, this is the warning that Raynor and Christensen give, that if you want to go back, and you just get rid of it all together, it's going to be very difficult to go back. And sometimes you do want to go back. Whereas, if you've retained it, at a certain level, it becomes easier to now expand that.

Now, let's talk about-- OK, so we talked about this. The next major decisions, which it's really vertical integrated designs, proprietary vertically integrated designs versus modular designs. It's a huge, huge, huge decision. And then I thought chapter 5 had a very good discussion of vertical integration versus modular. And when do you use vertical integration, which interdependent architectures, and when do you use modular?

And as you know, he gives a very good definition of vertically integrated ones and what you're optimizing for. But by definition of proprietary, you're committing yourself to pretty much do it all versus when it's modular, which is much more you can distribute it more than you are relying on other people more, becomes easier to decompose it in time to market, please.

AUDIENCE: There's two things I would like to bring up. As I was reading this, the word performance jumped out at me.

IRVING WLADAWSKY-BERGER: Yes.

AUDIENCE: And the reason why, if you ask anyone in this room their definition of performance is going to be different. So as far as performance, how does Christensen define performance. Because performance to me, I mean, I do a lot of graphics. So I need a graphics card that has much more memory.

IRVING WLADAWSKY-BERGER: Mm-hmm.

AUDIENCE: But the new types of performances, for instance, let's look at the iPhone. Someone can look at the iPhone and say, OK, this performs well because it has an interface, and it has something that drives the aesthetic value.

IRVING WLADAWSKY-BERGER: Yeah, I don't think that's what he meant.

AUDIENCE: But I mean but it's--

IRVING WLADAWSKY-BERGER: I agree violently with--

AUDIENCE: Performance is being redefined.

IRVING WLADAWSKY-BERGER: That's right. That's right.

AUDIENCE: So what are we targeting our market? I mean, what markets are we targeting?

IRVING WLADAWSKY-BERGER: You're bringing up, and we'll discuss it much more. I think when Christensen is talking about performance, he's talking about classic product performance speed, reliability, and things like that. The kinds of performance where you need to carefully control different components.

For example, if the iPod in order to run, needed a custom micro and custom memory and custom storage, it could have been, then whoever is designing the iPod probably had to design all these components at the same time.

By the way, in the old days, that's how we did computers. The PC was probably the first time that those things got disaggregated. But before, you did minicomputers, you did it all. Everybody did it all, pretty much. Because the components were of a quality that you gave away too much performance and quality if you didn't do it yourself. And this stuff, because of the immaturity, you couldn't do that.

But you're bringing out an incredibly important point, and we'll talk about it in a second, that once you get to the level of good enough components, other criteria need to take over. Otherwise, you're spending money on something that your clients could care less about.

You can say, well, maps on the iPhone, take a second. And in my machine, take 0.2 seconds. But of course, you need to learn assembly language to deal with site maps, instead of whatever you do with the iPhone. That's a simple trade-off.

AUDIENCE: And in addition to that, I guess he never explains or talks about the design component.

IRVING WLADAWSKY-BERGER: No, I understand.

AUDIENCE: Because if you talk about interdependent architecture and modular architecture, compromising a lot when you talk about modular architecture. You're compromising design. Because if you take modular out of the context of technology, let us look at modular in the context of architecture, for instance.

When you talk about modular design in architecture, I mean, building architecture, you're compromising a lot as far as the aesthetics of it. So modular compromise--

IRVING WLADAWSKY-BERGER: Wait, wait, wait, wait, wait, and you're an architect, so you know a lot about this.

AUDIENCE: Mm-hmm.

IRVING WLADAWSKY-BERGER: I wonder if it depends what we mean by modular? Because--

AUDIENCE: It's component architecture

IRVING WLADAWSKY-BERGER: Yeah, yeah, yeah, but no, no, but I mean is let me go to an extreme. So let's take the Pru Tower. And if I ask you how much of what goes into the Pru Tower are standard things in the building ecosystem. Steel and bricks, all that is standard, am I correct?

AUDIENCE: Yes it is.

IRVING WLADAWSKY-BERGER: So no, no, but those are modular components. Don't underestimate-- so that's why I say it depends what we mean. Now, I think modular components is what we call a fractal concept, I don't know if you all know about fractal, which is at any level, you look at it, you will find a similar problem.

Remember, once upon a time, there may have been building contractors that had their own steel factories and did custom steal for their building. I bet you that happened once upon a time. Or they ask you a steel to do them a certain kind of steal. You no longer need to do that.

But what would you say in a building like the Pru Center is custom versus standard components?

AUDIENCE: As far as the design component of it?

IRVING WLADAWSKY-BERGER: Well, yeah, by design components,

AUDIENCE: I mean you talk about glass and steel. Back to what you talk about component issues, if you boil it down to components, steel is steel.

IRVING WLADAWSKY-BERGER: No, but that's important, and glass is glass, right?

AUDIENCE: Yes.

IRVING WLADAWSKY-BERGER: No, but that's very important.

AUDIENCE: But it's how you cut the deal--

IRVING WLADAWSKY-BERGER: I agree. No, no, I agree. My point, and this is incredibly important for businesses, I bet you, and I wish I had quantification of this, that 90% of what goes-- I don't know how much the Pru Center costs. Let's assume it costs $1 billion.

I bet you between 80% and 90% of that cost was based on standard components, and the amount that is custom is 10% to 20%. That's my feeling. That in mature engineering disciplines, and what I'm saying applies to automobiles and so on, the majority of the pieces are components. And the design, how you assemble them and unique things that you do and unique things that you hire expert people to do for the first time for you is a relatively smaller percentage. That's my expectation. Do you agree with that?

AUDIENCE: I guess my argument it still impacts the design result.

IRVING WLADAWSKY-BERGER: It totally.

AUDIENCE: And you compromise the design based on your components.

IRVING WLADAWSKY-BERGER: Well, yes, I agree, but remember, you also have a budget and a time to market. Just a second, just a second, we'll take everybody. Because no, this is an incredibly important discussion. Incredibly important because. I believe-- let's say, I believe that every discipline starts doing their own steel and glass. And then it takes forever to do that.

Cathedrals were pretty custom. But eventually it becomes an engineering discipline, and part of becoming an engineering discipline, one of the major pieces is the notion of modular components. I think the Industrial Revolution, a major step was the introduction of standard components.

You all know what I'm talking about with the, I think, it may have been the Winchester rifles that could be assembled? And if you look at mechanical engineering, civil engineering, they all start developing a discipline based on standards and standard components. And then the tremendous uniqueness is design at multiple levels.

So I'm agreeing violently with you that it's no longer in the components. It's in the design. That's what I'm saying. Immature disciplines spend too much time in the components and not enough in the design. That's my feeling.

And the reason I find this so fascinating, and then we'll get back to all the discussions, is that if you look at business as an engineering discipline, which I know, people, we haven't, it is perhaps the most immature form. But you notice that every goddamn company reinvents the processes for running the business.

And you say, well, why do you do that? Well, because that's a business. You cannot build this business using somebody else's steel. Now that's changing. That's where business process out. It's beginning to change. But boy, is it-- I view that to be, this is my opinion, the mother of all disruptive innovations in the 21st century is how business is transforming, becoming more distributed, more deconstructed, which then brings into business design, which is what you really want to have.

And business design and business strategy, in my mind, are totally intertwined. So that's why I was probing so much, because I think design is really, really, really critical. It's really, really critical. In fact, I would say it's everything. Yes, let me ask Soon-Yu to comment first.

AUDIENCE: I guess the first thing that you were trying to mention was that the trade-offs between modular design and interdependent design, which I think you guys are mostly aware of, you guys, from a system dynamics. And what [INAUDIBLE] brought about was the interaction or trade-off between the design, which is more like an aesthetic design, and then the architecture of it.

So I think, from my experience, I talked to a lot of people from Samsung and LG and these Korean electronics companies, which is a similar structure with Apple. So they're making this electronic stuff, which I think the core of the competition has shifted from engineering problem solving into more like a design--

IRVING WLADAWSKY-BERGER: I agree, violently.

AUDIENCE: --aspect. What they changed was that these industrial designers, who design the form factor, not the function factor, they are getting more and more and more importance in their organization.

IRVING WLADAWSKY-BERGER: Yes.

AUDIENCE: So what they do is they usually-- 10 years before, LG and Samsung didn't really concern about this aesthetic aspect. So what they did was their research centers and the product--

IRVING WLADAWSKY-BERGER: I agree totally.

AUDIENCE: They started the initiation of the product development.

IRVING WLADAWSKY-BERGER: I agree.

AUDIENCE: So they came up with nice functionalities and features and things like that. And then they thought about how to wrap it up. But now they are getting more of these industrial designers. They are getting more importance.

IRVING WLADAWSKY-BERGER: I couldn't agree with you more. In fact--

AUDIENCE: It's the same for Apple as well.

IRVING WLADAWSKY-BERGER: In fact, let me tell you, one of the projects that I've seen recently that I'm most excited about is a project in the UK called Design London that Gordon Brown has initiated. He initiated when he was Chancellor of the Exchequer. And I'm involved with Imperial College, and I was there in July and learned about it.

Now, Design London is a partnership of three institutions. Two are what you would expect Imperial College School of Engineering, [INAUDIBLE] College School of Management. That's what you would expect. The third is the RCA, the Royal College of Arts, which I'm sure you would say, it's one of the world's best design schools. Am I correct?

So the fact that they are an equal partner is what makes this so innovative that they have a school of design equal partner with engineering and management precisely for all the reasons we're discussing. Yeah, you had a comment in the back? Yes, please.

AUDIENCE: Yeah, I was going to find an analogy in civil engineering and software, excuse me. Civil engineering, about 20, 30 years back, bridges were created on site, in situ. And recently, you've seen that prefabricated and prestressed concrete arrangements and panels have increased the construction of bridge time to maybe less than a year, which is really fabulous. Now, even the aesthetics of prefabricated bridges are kind of better. That's one.

Second advantage is since the prefabricated panels are created in industries and not on site, you have better advanced composite materials that are used to create them.

IRVING WLADAWSKY-BERGER: Yes.

AUDIENCE: So from a reliability perspective, from--

IRVING WLADAWSKY-BERGER: Yeah, perfect. I understand.

AUDIENCE: It's a perfect scenario where things have changed drastically over decades.

IRVING WLADAWSKY-BERGER: Yes that's perfect.

AUDIENCE: If you compare this to software architecture, open source has come up a lot. And it has allowed people to work in smaller groups and create smaller products and then model products. And if you interface them and create a larger product, you have a quicker time to market. You have better reliable products because each of them has been tested and you have costs.

IRVING WLADAWSKY-BERGER: But you also brought up something very important, which is comparative advantage. Who did that, Schumpeter, who talked about comparative?

AUDIENCE: Porter.

IRVING WLADAWSKY-BERGER: Porter also, which is once you can deconstruct it-- oh, I'm sorry. The one who first probably brought it up was Adam Smith in *The Wealth of Nations,* which is that you start specializing.

So to your point, if you're a bridge builder and you are also building steel, or cables, whatever, you may be a superb bridge builder, but a medium-rate steel builder. But if you are working with another company that only does steel, they may be world class in steel. So now if you bring your design skills, your project management skills with their composite whatever, the world's quality goes up.

Adam Smith actually first formulated this principle in *The Wealth of Nations.* And then through the ages, Rowland, Coles, and Schumpeter and others, which is very, very important. And that's why if somebody said, well, I prefer to compete this way, the penalties eventually become extraordinary. So you have a good advantage initially, but eventually, as you pointed out, it's very difficult. You had a--

AUDIENCE: I guess it's a point about the architectural argument just then. We talked in our last class. This is our intention. We have an architect coming in. And he stresses a lot like how--

IRVING WLADAWSKY-BERGER: In the last class?

AUDIENCE: System architecture, and what he talks about is how--

IRVING WLADAWSKY-BERGER: You mean like a civil engineer?

AUDIENCE: Yeah it's a real architect who came in.

IRVING WLADAWSKY-BERGER: OK.

AUDIENCE: [INAUDIBLE]

IRVING WLADAWSKY-BERGER: OK, OK.

AUDIENCE: So he talked a lot about how traditional architecture gives a form follows function. And you as an architect need to understand what kind of functions it is. [INAUDIBLE]

IRVING WLADAWSKY-BERGER: Yes.

AUDIENCE: And that's the same for a lot of engineering stuff.

IRVING WLADAWSKY-BERGER: Yes, absolutely.

AUDIENCE: [INAUDIBLE] for in the [INAUDIBLE]. But right now, what he talks more about is there's an intertwined between form and function. You can't just do one.

IRVING WLADAWSKY-BERGER: I agree totally.

AUDIENCE: And in fact, when you look at the Christensen traditional Christensen, there's disruptive innovation and all this how performance keep going up. And then a lot of companies actually now start to have a surplus because now they reached a certain point.

IRVING WLADAWSKY-BERGER: Exactly.

AUDIENCE: And where do they compete and compete and fall.

IRVING WLADAWSKY-BERGER: Exactly.

AUDIENCE: Beauty rather than the functionality.

IRVING WLADAWSKY-BERGER: No, I agree violently. Perfect, I agree. Let me take a-- please.

AUDIENCE: I wanted to give another example of the trade-off that you talked about between performance and modular. So I used to work in chip design and specifically in application specific design. So different components that IBM designed, and we also designed our own memory designs. And it was all customized, so it was really high performance.

One of the things we did is we changed the architecture of our system to put all these things together to get third-party IP. And when we made that change, we had the option of making it modular, and accepting third-party drastically reduced performance.

But the fact was you could now have tens of companies making--

IRVING WLADAWSKY-BERGER: Exactly.

AUDIENCE: And a lot of times, each of these designs are optimized for various applications. So while the putting together of the thing reduced the performance, the fact that these memories were optimized increased it. And plus we were able to go after a much broader section of the market, which we otherwise wouldn't have been able to.

IRVING WLADAWSKY-BERGER: Yeah, yeah, please.

AUDIENCE: The last time, we talked about decision to enforce and outsource.

IRVING WLADAWSKY-BERGER: Yes.

AUDIENCE: In this slide, we talk about factors. [INAUDIBLE] we choose to outsource. So then we have to use modern architecture.

IRVING WLADAWSKY-BERGER: Yes, I would say that-- otherwise, it's really risky. Because if it's proprietary, but you're just contract-- otherwise, it's more contract hiring somebody. Whereas in fact, the modular architectures facilitate building an industry ecosystem because also most companies that you would outsource to, most companies don't want to be trapped by you.

And so if it's standard, then they have multiple clients themselves. And especially if they are the best companies in the world in that, then those are some of the trade-offs.

Let's talk a little bit about, again, just very quickly, I'm just summarizing some of what the chapter says that the proprietary vertical integrated systems or products, are clearly what you do in the early stages. You just don't know what you're doing. You haven't had the chance to deconstruct it yet. So that's what you need to do.

And usually that's what lets you do the product. But you have to invest a lot yourself. You're taking a lot of risk. And it takes awhile because you need to do all these other things. And at that point, you are vertically integrated, and you control all the pieces.

When you start moving to modular, open, flexible architecture, usually that requires a certain level of maturity because what you're saying is most of the components, not all, most are good enough. Which means, as you pointed out, if you keep making them-- if you say, well, no, no, screw the iPod. I'm going to keep doing a better micro.

Well, people who buy iPods, the micro is good enough. Or for most laptops, the micro is good enough. So just continuing the emphasis on the micro, you are overshooting the market. So that's the point when you say, I can go with standard components and then things begin to disaggregate, and you start building ecosystems.

And this is what most disciplines do. And now the question comes, which should you do? And it all depends. It's really the only answer, which is for the new things that you are innovating by yourself, the really new pieces, in architecture maybe an architect has an idea for an incredible new window, and they do the specs and get a company to do it for them. And for a while they are the only ones who have it. And that's a custom window. And that makes the whole building.

A few of those are totally fine. In fact, a few-- people do that. Am I correct that you may have some absolutely things? But that's a market excess, and my toilets are also-- eventually your client runs out of money. You say, no, I need special toilets.

So you have to pick. Am I not correct? There are a few things that may be the unique differentiators, and the rest, you get from your ecosystem. And I'm assuming if you're a really good contractor, or a really good, let's say, Toyota is famous for this, your partners are so good that they bring you their innovations. So you don't need to do new toilets. Whoever those toilets for you brings you the best toilets in the world. So that's what you're using now, please.

AUDIENCE: For example, I think actually Steve Jobs working on the new-- I mean, the store in Manhattan on Fifth Avenue redefined the windows system and patented it.

IRVING WLADAWSKY-BERGER: Redefined the--

AUDIENCE: The windows system that he had, the caulking was actually outside.

IRVING WLADAWSKY-BERGER: You don't mean redefine Windows?

AUDIENCE: No, the windows system. The window--

IRVING WLADAWSKY-BERGER: Wow, that would be--

AUDIENCE: No, the actual physical windows system.

IRVING WLADAWSKY-BERGER: Of the store?

AUDIENCE: Of the store.

IRVING WLADAWSKY-BERGER: I never heard that.

AUDIENCE: Yeah.

IRVING WLADAWSKY-BERGER: How so?

AUDIENCE: The caulking was actually outside and not concealed enough, and he did not want it to be seen from the outside. So he said, hey, you have to get rid of this caulking. You have to do something about it. Why don't you just put it inside? And he's like, OK, you cannot do that because it's a rain sealant. But he said, you have to come up with something.

So he got people to do it, and it redefined how it looked. And now you cannot see the sealant from outside. So it's pushing the envelope based on the design factor.

IRVING WLADAWSKY-BERGER: Yeah, yeah. and then other people concentrate on the architectural dimension. And this gets back to-- in my mind, it all gets back to design in my mind. That the reason this is so important is because I'm convinced that in the knowledge age we're in now, because the components we have to work with are so good, the incredible competition is design and its form.

But now this is what gets so interesting. Classically, we apply design to physical things, whether it's an iPod or a BMW 3 Series. But how about, let's say, a customer service in a business? Could you think of customer service from the point of view of design?

And apply similar criteria of-- I'm assuming one of the things in design have to be you put yourself in the shoes of the people looking at the design, correct?

AUDIENCE: Yes.

IRVING WLADAWSKY-BERGER: So in customer service, you put yourself in the shoes of the person receiving the customer service. And as we know, a lot of times when you do that, you say, it's amazing anybody does business with us because we do such a poor job.

Now, the reason this is so important is because the industry-- I mean, again, if you think of business as a system, we're building cathedrals. I don't know how else to say it. We are in the cathedral stage of businesses. And the biggest opportunity that I see is the ability to start taking this into the future by systematically doing certain processes far better than anybody else.

And now the reason we might have a chance to do that is because this notion of modularity are now beginning to show up in business processes. Before, you didn't have to-- remember, you needed the internet. You needed a SOA. There's a whole set of things that you needed to start getting a certain level of modular business components, but they are beginning to show up.

And part of them showing up, just like with the iPod and with the Boeing 787. if you can free up the people, the master architects, business architects from having to do every goddamn thing, then they can start focusing on a few unique advantages and partner for the other.

Remember how new this all is. This is barely happening now. So the opportunities for business to move from here to here, I think, are enormous. I don't know how you all feel about that, but I think that's-- in my mind, this is my feeling in the 21st century, that's the game.

And before we take the break, let me give you one example. I think I already mentioned it once, but let me give it again, which is Sunil Mittal, who is the chief of Bharti. And Bharti now is a conglomerate. They are a big conglomerate, but they are also the biggest private sector cellular company in India growing by leaps and bounds.

And Sunil Mittal said, I needed to grow fast. I didn't have enough energy to worry about all the other stuff, so he outsourced everything, except customer acquisition and retention. That's what he did. He said, that's it. That's my unique window. It's the customer acquisition and retention.

And to do that, he got, I believe, Eriksson and somebody else to do the network. And IBM is doing the computers, and other people are doing other stuff, which he said, I heard him personally give a talk at an IBM CEO conference, he said, when he did this, everybody else in the telecom industry thought he was crazy, that how can you be a telco without owning your network? They thought, you don't know what you're doing.

He said, well, the network is a commodity. Network is a commodity. Nobody had ever thought that you can be a telecommunications company and not own the network. Nobody ever thought of that. He reshaped the definition of a telecommunications company in terms of customers.

He just went out and did it in terms of the customers. And would you all agree Sunil Mittal is very, very successful? And I think that's a very good example of what I expect to see more and more in business innovation in the 21st century, precisely because we're moving from here to there.

Now, here is the problem, and here is where you all come in, which is don't forget that in order to design 787s, chips, buildings, bridges you use really sophisticated tools. You all know that EDA tools, CAD/CAM tools, all kinds of tools. And you do a lot of simulation, massive simulation.

Where are the tools for business design and simulation of different strategies? Where are they? And the answer is somebody has to invent them. It's a little bit of work in that. So none of this can happen because if what you say, oh, now that it's modular, you'll have people with drafting paper, architecting fidelity. It doesn't work that way.

We really need sophisticated tools. And that's all ahead of us. And, we'll talk more about it, but let's take a break now, and then we'll continue in 10 minutes.

Before we get to, so what did we do in IBM in operationalizing a business and go through some of the items, let me talk about how I've been-- when I'm talking about some of the biggest changes going on in innovation, what I tell people, which very much is the evolution of systems, it's the way I think about it, but remember, I have a very expanded view of systems that are pushing us to have to keep moving into this dimension, the architectural-design dimension more and more. And let me talk about this in two critical dimensions. And then I'll tell you the third dimension that I usually add that I didn't include here.

So one of the dimensions is breadth and scope. Now, let me tell you what I mean by that and illustrate it with IT systems. When I first started working with computers in the 1960s, I was first in college and then a graduate student in Chicago, a system was just the computer and the way it was connected to its I/O gear. That was the system.

So it was very much just a piece of technology that was isolated. We never thought of it as isolated. We thought of it as wonderful new stuff because you can do a lot of calculations, but you brought your deck of cards. Then I usually went off to have a pizza downtown at Uno's, came back, got the results, remember, oh, God, I forgot a comma in Fortran, so had to resubmit it. You're too young to remember.

AUDIENCE: I remember that day.

IRVING WLADAWSKY-BERGER: You used punch cards?

AUDIENCE: Yeah, 43.

IRVING WLADAWSKY-BERGER: Wait a second. When did you use?

[LAUGHTER]

AUDIENCE: At the Academy, we still had a [INAUDIBLE].

IRVING WLADAWSKY-BERGER: So it was a history course.

AUDIENCE: Yeah.

IRVING WLADAWSKY-BERGER: Well, in 1962--

AUDIENCE: [INAUDIBLE]

IRVING WLADAWSKY-BERGER: I'm sorry?

[LAUGHTER]

What did you say? I'm sorry.

AUDIENCE: I have seen five-inch disks.

[LAUGHTER]

IRVING WLADAWSKY-BERGER: But in 1962, when I was entering college--

AUDIENCE: [INAUDIBLE] every day.

IRVING WLADAWSKY-BERGER: Well, that was it. The interactive terminals didn't show up until late '60s, early '70s, text terminals. And then cathode ray tubes showed up in the '70s. But before that, it was cards.

But then the notion of a system changed radically over the next 20 years with the rise of networks, local area networks, wide area networks. It was a much more bigger view of systems.

And people could talk about, for example, banking systems with ATM terminals. Remember those showed up in the '70s were the first-- that I remember. You don't remember, but you must have read about it. And the first ATMs appeared in the '70s, connected over proprietary networks like IBM's systems network architecture, manufacturing systems. So people started looking more specialized systems.

Now, and this is very important, no question this is totally proprietary. I mean, this was almost the essence of functional performance guiding everything. And here, it was all pretty proprietary because networks were totally new. So they were proprietary.

Systems received an incredible boost when the internet and the web got adopted in the mid-90s because for the first time, for the very first time, you can connect your clients, to your systems. Remember, before the browser, you can only interact with your systems if you were inside the company, through very specialized applications like ATMs, or through some proprietary things that led you-- if you were a partner, you had maybe a special PC that let you in.

If in 1994, you wanted to find out where your FedEx package was, and if you were a normal person, as opposed to, let's say, a business that did a lot of business with them, you had to call a rep. And the rep typed into the proprietary network the number of the package. It went into its transaction system on the mainframe and got back the answer, and the rep told you.

For large customers, they would install a special PC with special software, proprietary, that link you directly. So that you essentially can be viewed by their system as if you were the rep.

The browser, the web revolutionized that. Because now everybody was using a browser, and everybody could connect to a web server. And now the web server, all it was doing initially was almost played the role of the operator that took your call.

So you typed it in the browser. It went to the web server. The web server said, oh, this is a request to the mainframe, a little bit of software went there got you back the answer. And you said, God, this is magic. But it was a massive expansion. And in fact, the reason the bubble became a bubble is because it felt like magic, that these standards could let you do so much.

I remember, in the mid '90s, maybe '96, '97, I was in Japan. I had flown in. I flew in, it must have been a Saturday out of here. Landed in Japan Sunday evening, and then woke up like 3:00 AM, 4:00 AM Monday morning in Japan. And I remember there was a baseball game back in New York. The Yankees or Mets? I don't remember. That I was very interested, but obviously, you wouldn't get it in Japan.

And then I remembered that oh, but my PC is connected to the internet. They were beginning to link the games over PCs so I could be drinking green tea at 3:00 AM in my room in Japan, listening to the baseball game back here. Now today, that is, like, why are you telling me this?

[LAUGHTER]

But when it first happened, it was magic. And it was this deconstruction of the system. That's the only way I can say that. Now, where we are heading is that we really want to take every single aspect of a business and integrate it together, every single aspect of a business.

We say, well, what do you mean by that? Well, a business has processes, information, people, and now we have all the work on standards, SOA Of course, the internet, we can begin to couple them together. So you can begin to look at the business as a system.

But now, once you start thinking that way, you realize, wait a second, there is no business that doesn't exist within an industry ecosystem. So if this is going to be useful, your definition of a system has to include the industry ecosystem because otherwise, if you say everybody inside the company can work with you, but the people making the wings of the Boeing have to make phone calls, it's all over. So you have to stretch what you're doing to all your suppliers and all your partners.

And because more and more industries like healthcare are a mix of different industries. There's the hospitals, the doctors, the insurance companies, and others. That's what's driving the digital economy that we're trying to look at all the things, it's a business.

Now, the technology to help us start doing that it's beginning to happen. I mentioned several times, more and more the internet, broadband, wireless, even software standards, SOA, the really hard question is, who the hell is designing this? Now, if you said to me, well, Irving, biology. This is not biology.

Remember in biology, yes, you designed this, but if you're the CEO of a company, you cannot go to the shareholders, and they say, how come you lost the records of all your clients? And you say, well, you know, shit happens.

[LAUGHTER]

We got a virus. Everybody knows in biology-- you cannot do that. Somebody has to design, an architect. And that gets back to the question, how do we do that? And by the way, there is no answer. Remember, for a long time, planes and cars, there were drafts, people drafting by hand. And I'm assuming for a long time, civil engineering architects were drafting by hand.

It's only later that the tools came in and the simulations. In business, who designs it? I don't know who designed. It's a fascinating question. I don't know. Soon-Yu, is there a secret course in Sloan on designing business processes?

And the reason, by the way, is that the whole notion of designing business systems, it's so new. I think, it's really pushing us. But why do we not have a choice? Because of the scope, the complexity here, is such that if you don't have some kind of design on top of that, and in fact, we'll get to this in a second, the notion of architecture that you apply to these incredibly complex systems has to be more flexible and adaptable than the notion of architecture that you use for a much simpler product.

But I guess you would say, the notion of architecture in an incredibly large microprocessor with a huge number of things has to be more flexible and adaptable than just a small one. It gets back to modularity. Soon-Yu?

AUDIENCE: Yes, just remember there's two professors who are related to this. So one is Steve Eppinger, who I'm pretty sure you guys are familiar with Steve Eppinger's work. What he's trying to do is he's trying to reduce the complexity of the architecture and use it as an organization framework.

And the other one is Tom Malone.

IRVING WLADAWSKY-BERGER: Exactly, exactly.

AUDIENCE: Make the primitives of these processes. It's a very small unit of processes. And trying to use it as how to exactly build more complex.

IRVING WLADAWSKY-BERGER: Yeah, I agree totally. Steve Eppinger and Tom Malone are leading the way. But how many professors are there in Sloan? These are two out of-- By the way, at some point, engineering started coming into the cathedral. I think we're seeing that. Yes.

AUDIENCE: I think it was Murphy who said that if you look out 20 years, 20 to 30 years on the macro economy, things will shift down into buckets, if you will. The first bucket will be that Americans will continue to be the innovators. The Chinese will be the blue-collar workers of America, and the Indians will be the white-collar workers here.

IRVING WLADAWSKY-BERGER: [LAUGHS]

AUDIENCE: And I think at the heart of all of that--

IRVING WLADAWSKY-BERGER: Who said that?

AUDIENCE: I think it was Murphy, I think who said it. I'm trying to remember, but someone said that.

IRVING WLADAWSKY-BERGER: I wonder if he really believes that or was trying to lull the Americans to sleep.

[LAUGHTER]

AUDIENCE: I think, though, do believe that. If you look at the innovation side of America, at the heart of all that, the principle is going to be mass customization.

IRVING WLADAWSKY-BERGER: No, no, I understand. I understand totally. I just want to say that, let's say, if you go to Tsinghua University, and you say you guys are the blue-collar workers, Tsinghua is a really superb university in Beijing or Peking or Xidian Actually, India needs to do much better at the high end, but I think you all know that.

But if you go to China and then India, hopefully, as universities-- I've been asking my Indian friends, why is it like that? And they say for whatever reason, the Indian universities still cannot pay faculty, top faculty, enough to attract them.

AUDIENCE: They were public. Most of the [INAUDIBLE] universities are public universities, funded by the government. But even then, they do not have a lot of control in the technical domain. Salary wise, they do have nothing.

IRVING WLADAWSKY-BERGER: But everybody knows India-- and that's why most Indian students go to the US to get PhDs. Am I correct?

AUDIENCE: The high end--

IRVING WLADAWSKY-BERGER: I'm talking about the high end. In China, Xidian, wouldn't you agree with me? They are becoming world class is my opinion. So I'm just saying, any US executive that really believes that is in trouble. That's all I'm saying.

AUDIENCE: Let's take it out of context.

IRVING WLADAWSKY-BERGER: By the way, I agree totally with you. And in fact, it's an interesting discussion to have. Let me go to the next section and then we'll talk about that. Because it's a very interesting discussion.

I've talked about breadth. Let me now go in the vertical dimension my view of systems. And I'll explain the shapes later. But if you look at a stack-- so we start with the technology component microprocessors, memories. And we know how to do those fairly well. We have tools, deterministic stuff, and so on. And then we use this to build products, hardware and software products. And again, we pretty much know how to do that.

And as you know, once upon a time, the technology and the products, I'm talking in IT now, were very proprietary, very unique. That was the competition. But as you know, more and mow, a lot of these are totally commoditized, like, DRAMs, HDD, a lot of microprocessors.

And even here, you can see standardization beginning to show up. And I think as SOA takes hold, that will happen more and more. But you can see the rise of Linux and other things. I think that what's happening is that the action is going up the stack.

And one of the major things is the design of applications using technologies and products, which keeps expanding, more and more and more and more, because as these things keep getting cheaper and cheaper and cheaper and more and more powerful and more reliable, there is more and more you can do in brand new areas. And then even more, is to now use all this stuff to help companies build business systems. Do it for companies, aggregate it.

What's interesting, when I look at these systems of the stack, what goes on down here, and what goes on up there, is very different. Systems, as you go down, they are deterministic. They are physical things. You know exactly what you're dealing with. It's like you're building a microprocessor. You're building a bridge. You're engineering, designing physical things.

As you go up the stack, you are designing market-facing systems that involve people, and that involve people dealing with each other, and that renders them unpredictable. And when I say unpredictable, I mean emergent in a biological sense of unpredictable. Things are very complicated and very fast changing. You don't have control over them. They are emergent.

Now, does that mean you cannot apply technology? No, of course, you can apply a lot of technology. But it's dynamic instead of static. When you're dealing with dynamic systems, you have to constantly be watching them. Otherwise, they'll get away from you. You have to constantly be monitoring them.

Let's take customer service. Somebody said, well, customer service is deterministic. When a client calls, I'll say, I'm programmed to answer in one of these five questions. What if the client has a sixth question? Well, I'm not programmed to answer that. Guess what, you have a very unhappy client. That's an example of a market-facing system that is unpredictable because you cannot control what your clients are going to want from you.

And think of a healthcare system. Think of systems to help nurses, doctors. So the design, and this is why I like the thinking of design, the architecture and design of systems at these higher levels with some predictability, have to be far more frameworks. I think it has much more. It's almost like urban planning rather than physical building.

Now, I don't know if they teach urban planning in schools of architecture. Do they do that?

AUDIENCE: Mm-hmm.

IRVING WLADAWSKY-BERGER: Did they teach--

AUDIENCE: At least in some schools. I remember in my undergrad days.

IRVING WLADAWSKY-BERGER: But you all have to agree, there is architecture of a different sort when you think of urban planning. And so when you start doing business systems-- now, what's so exciting is that before there was nothing we could do.

So for the most part, as you go up, you have to do it with labor. Because you say, well, if it's unpredictable, it's all people. It's labor, and guess what? As we get better and better and better, the industrial sector of the economy is smaller and smaller and smaller, and the services sector of the economy keeps growing. In most countries like the US, the UK, it's over 70% or 80% of the economy.

Also, more and more businesses, if you look at IBM, over 55% of IBM's revenue is now services. But that's true of almost every business, because in these areas they keep applying automation and technology, and those areas is labor. What is changing is with tools and collaborative systems, you can now begin to attack the labor part of those systems, the people-oriented part.

When I say attack, I mean make people more productive. Not by automation, but by giving them better and better tools. And in my opinion, the hardest part in all this gets back to design. Who is architecting, designing? How much knowledge do you need to do by industry vertical healthcare, investment banking, and so on? And the answer, I believe, is going to be very vertical for quite a while.

Let me explain this shape because the first time I did this was to go to Washington to talk at some government thing where people were saying, there were no jobs in technology, or there were, but they questioned all technology. It's bad to be a technologist because all jobs are going on.

And my hypothesis is everyone at these level, the requirement goes up by an order of magnitude. So for every one person doing this, you have 10 people doing that. And then you have 100 people doing that or 1,000 people doing that.

So if your definition of technology is down here, designing microprocessors, you better put up, what I tell people, David Ortiz kind of numbers. You have to be a really, really good microprocessor designed because we don't need that many. The competition is very, very strong.

And more and more, even at the product level, the more mature industries, the less people you need. Also, the more you learn, the more it gets automated. The bulk of the jobs are in application systems and then in business systems.

Just to give you a view, in IBM, I just came back from IBM Academy of Technology meeting, there are maybe 30,000 people doing this stuff in labs. And there are maybe 120,000, 140,000 people up there. And by the way, IBM has really good labs. So it's just the action is in the marketplace, and that's where the jobs are.

And this stuff is infinitely more complicated. I mean, you put together the previous slides I show, which is the scope of the systems that we try to design, with the fact that we can now attack systems at this much more market-facing, people-oriented level, and this is-- it's pretty interesting.

In ESD, MIT's engineering systems division, we're trying to put together some pretty nifty healthcare system initiatives because we can begin to try to attack healthcare as a systems problem because there is so much we can bring, although it's a very, very complicated problem.

So now, once you talk about things of this scope I'm talking about, the question of modularization versus vertically integrated disappear because you just cannot possibly do things of this scope if they are not standard based and modular. You just can't anymore.

You say, well, why is the internet standard? Well, it's like almost asking why is biology built out of standard components? Which it is cells, DNA, pretty standard. And it's because that's the only way you can scale to the complexity we're trying to scale. So we'll see where all this goes.

Now I'll move on to the specific implementations of a business in IBM. But any questions before that?

AUDIENCE: So you were saying that there are not many systems which actually do the business process management. But there have been right from [INAUDIBLE] we have CRM systems and business process engineering. BPM seems to be.

IRVING WLADAWSKY-BERGER: They're primitive. That's the way I would say. People are doing them, ERPs. We have ERP systems. But I think they work. If you give them a grade by form, you would give them a minus 10.

I don't know how many of you are familiar with CRP systems. Would you agree that most people are not very happy with them?

AUDIENCE: They work, yeah.

IRVING WLADAWSKY-BERGER: Yeah, they work. That would be as if we were talking about, well, look, I know this building looks horrendous, but it won't fall down. But it really looks horrendous. ERP systems, would you agree they have that quality?

AUDIENCE: They all have-- you have to know what they're capable of, and then you can ask them whatever you need to, and they'll give you a reasonable answer.

AUDIENCE: This is foundation, right?

AUDIENCE: Yeah.

IRVING WLADAWSKY-BERGER: That's it.

AUDIENCE: You have to be very critical.

IRVING WLADAWSKY-BERGER: Yeah, but, no, no, no, the foundation doesn't have to be pretty. But remember, you need to do interior decorating so where is that? And so I think what's happening is the end users are dealing with the foundation. As a foundation, it's wonderful. But there is nothing above the foundation. Nothing good about above the foundation.

That's OK. When I say that's OK, I mean that's an opportunity for all of you to go do some innovative stuff and invent new things and make systems that are much better, taking advantage of all the advances in technologies. But what we have today is very, very primitive because of the lack of form, the lack of-- again, they are not people oriented. They are foundational oriented, not people oriented.

Would you agree with that for the most part? Look at Windows. Windows works. We use it, but there aren't too many people that say, oh, my God, do I love my Windows system. I'm sure there are people who say that, Bill Gates, but it's just-- now, why is that? Well, when was Windows designed? Windows was designed in the late '80s.

Look at the advances in microprocessor technology, storage technology, everything else since the late '80s. Let's take user interfaces. The reason I'm so excited about virtual worlds, which comes from game playing and so on, is I think that's the biggest advance since the GUI.

You know when the GUI was designed?

AUDIENCE: Xerox did it in the '70s.

IRVING WLADAWSKY-BERGER: Yeah, so here we are with the GUI, which was designed in a world that was down here, and we're trying to use it to this-- so it's not surprising that it just doesn't work. And it's because of the lack of-- people haven't focused on form.

AUDIENCE: But also, with their models, maybe the other objection that people have as you go up, it's the service economy, and the kind of jobs you get are probably not high-paying jobs. And probably you are talking about customer-facing jobs, right? They probably don't require as much intellect as--

IRVING WLADAWSKY-BERGER: Remember, I'm talking about somebody designing hospital management systems for hospitals. Would you agree with me that is an intellectual--

AUDIENCE: People have been doing that for a while. I mean, you have CRM systems for healthcare. You have expert systems, rule-based systems for all of those things. And they do work. Now, I don't know how elegant the healthcare system is.

IRVING WLADAWSKY-BERGER: I think they work, but they are--

AUDIENCE: They are inefficient.

IRVING WLADAWSKY-BERGER: Not inefficient, no, no, this gets back to the conversation we had before. It's not the performance. It's that they are user hostile. So no, no, I'm serious. And by user, I don't mean programmers, or I don't mean graduates of SDM.

I mean nurses and doctors who are just trying to do their job or insurance people. I don't think they have properly taken them into account. That's my feeling. They are foundational. They were built bottoms up.

Well, let me ask you, do you think the iPod was designed bottoms-up or tops-down? What would you think?

AUDIENCE: Tops-down.

IRVING WLADAWSKY-BERGER: I'm sorry?

AUDIENCE: Tops-down.

IRVING WLADAWSKY-BERGER: Tops-down, I think that elegant design, this is my feeling, is tops-down. Now, obviously, you need to have a pretty good idea.

AUDIENCE: But most of these systems are designed for [INAUDIBLE] to use the computer. You get pervasive computing, maybe it will be much easier.

IRVING WLADAWSKY-BERGER: But we have. We're forcing the nurses and others to learn foundational stuff, and they don't want that. That's what I'm saying. By the way, I'm not saying this negatively. Remember, my point is incredible-- if the question is why is the internet and all these things disruptive innovations, it's because I believe that in the knowledge economy, they will change all this stuff here in profound ways.

But the key profound way is to totally transform the way human beings interface with technology by making form-- I mean, by making them much more appealing to humans. And appealing to humans is to say, well, but you mean artists?

Well, I don't know. Is industrial design an art? I mean, it feels at least at the RCA, that's-- I don't know if they're artists, but there's a lot of sophisticated stuff that goes into it. And if you have more and more tools, massive simulation, design, showing it to people, so that part, the design part, needs a massive boost. That's my feeling.

And more than my feeling, I also believe that's where the money is. DRAM is a commodity. It's hard to make a new innovation in DRAM or even in relational databases. Whereas up here, somebody reinvents it, I believe that's where the big-- it's a personal belief. Yes, somebody else had a question and then--

AUDIENCE: I like your ideas that it's a very interesting topic about how humans need to interact with the system.

IRVING WLADAWSKY-BERGER: Yes.

AUDIENCE: Because people always underestimate how much effort it takes to make that happen.

IRVING WLADAWSKY-BERGER: Exactly, that's why you celebrate good design. Because good design, I think, means somebody got it right.

AUDIENCE: Right, and usually at the end, the user can give you feedback, good or bad because they don't know why is it so simple because of all the engineering behind thinking about how to make it as simple as possible. In fact--

IRVING WLADAWSKY-BERGER: Yeah, well, that's why next week, we'll spend a lot of the class on Eric von Hippel's theory of end-user innovation, democratizing innovation because you really want as much as possible, the end users involved in this process.

AUDIENCE: And I just, I mean--

IRVING WLADAWSKY-BERGER: So it's precisely because of your point. The more you get the end users involved, the more you have a chance of developing something that will be appealing to the end user.

AUDIENCE: And I, guess maybe just one last point, I thought, is [INAUDIBLE] I just read an article. It's about this Mark Weiser. He said, the most profound technologies are the ones that disappear.

IRVING WLADAWSKY-BERGER: Yes, I agree totally.

AUDIENCE: So you don't even need a manual to operate.

IRVING WLADAWSKY-BERGER: That's right.

AUDIENCE: That's when you know it's so robust, so reliable, so easy to use.

IRVING WLADAWSKY-BERGER: You see, that's design. Would you agree? That's a conversation about design. That's exactly a conversation about design. I agree totally. That's exactly a conversation about design.

AUDIENCE: Do you see anything that requires expertise to run something that is not elegant?

IRVING WLADAWSKY-BERGER: Yeah, and remember, it's a gradation. You say, oh, well, so the most elegant design is any of us can do brain surgery. We're not taking it to that degree. And even let's take ERP systems for MIT. It's OK to say, you don't mean consumers need to be able to operate it.

So that's why it's so interesting. And we'll talk more about this next time. You have to have a pretty good idea who is your end user. Because let's say in a hospital management system, the doctors, nurses, administrators they have to be the end users. Would we agree with that? Maybe even the patients have to be the end user.

So you need to at least make sure they are usable by them. I would say at the very least, by them. But do they have to be usable by somebody in the street? Not necessarily. But when they become patients, those parts aimed at them, if you really want them to participate, then it has to be very natural.

AUDIENCE: And also, progressively, what technology is doing is it's replacing what experts used to do. Like pregnancy tests, 100 years ago, probably it required an expert to do that. Now, you can do it in five minutes or whatever.

IRVING WLADAWSKY-BERGER: I don't think you have to go 100 years.

[LAUGHTER]

I think it was much less than that. I agree, I agree totally. Please.

AUDIENCE: It's interesting that you're talking about design because on this month's issue of*Fast Company,* the front cover is actually [INAUDIBLE] designed.

IRVING WLADAWSKY-BERGER: Mm-hmm.

AUDIENCE: And [INAUDIBLE] there's an article with Steve Jobs saying that despite the fact that Apple product, they do crash, and they have their faults. And w don;t know anything about that because the design component is so elegant as far as aesthetic that people forget that. Apple crashes also.

They'll say, when you compare that with a normal PC, and that actually-- I mean, eliminates the fact of wow, it's the same.

IRVING WLADAWSKY-BERGER: That's right.

AUDIENCE: And crashing Apple.

IRVING WLADAWSKY-BERGER: Yeah, no, that's right.

AUDIENCE: A design component.

IRVING WLADAWSKY-BERGER: That's right.

AUDIENCE: [INAUDIBLE] Apple.

IRVING WLADAWSKY-BERGER: Well, I think, and then we'll go to this section, that-- remember, this is so-- the reason I showed those slides is because what we're talking about is a talk of technology becoming embedded into everything.

So look at the typical home. The typical home, yeah, typical home in a poor village in Africa. Typical, let's say, home in the US. Look at the number of devices in that home. And would you agree that in many ways, we may be in the red zone of complexity for a lot of those devices, and that breakthroughs may very well come calm from things for the home that make it much easier for most people to deal with that stuff, in whatever way?

That just because somebody steps through, put themselves in the shoes of people dealing with this stuff, who are not technology experts, and are now dealing with it. For example, I view TiVo as a wonderful example of design. Do you all know what TiVo is?

And what TiVo did is they-- boy, they really thought through the user interface. And it's just been wonderful what user interface they've had compared to most others. And that's what enabled them to stay in business. And God knows if they'll continue, but that's a very good example of design.

And again, the beauty is that we can have this discussion about almost retail store systems and telecommunications. The world is full of systems to go engineer to make them much prettier. But now we're talking about making them both excellent from a bottoms-up and then attack form and function at the same time. There is no end in sight, and the vast majority of them are marketplace facing. They're not lab-facing systems.

So let's talk about how we dealt with some of these key execution questions in the world of a business. And I'll talk about some more internal organizational questions. And then next week, we'll talk about the more market-facing question.

So first, the balance between open and proprietary, in-house versus partnerships and acquisitions. How did we focus and organize the offerings? And then, very important, how did we track progress? How do you track progress in disruptive innovations?

There is no question that IBM, before we embraced the internet, IBM was the bastion of proprietary technologies. In fact, IBM was the essence of a vertically integrated company. And that was the mainframes were. That's what they were because they were designed in the 1960s, where as we discussed before, you had to be vertically integrated to get the damn thing out the door and get it to work at all.

But as we talked in the first class, we didn't continue changing and adapting. So we stayed with the vertically integrated model until it was way past its usefulness. Lou Gerstner, I don't know how many of you have read the book, has some superb comments to make about what it means to stay beyond. And he explains the white shirt.

Do you all know that IBM used to pay known for white shirts? And the reason for that was when IBM started selling tabulating equipment, now we're talking the 1930s, 1940s, the people who dealt with the customers with equipment, they would have been mechanics or something like that. And IBM wanted an image that these were two notches above that.

And part of getting that image is to have the people who dealt with the customers wear clothes like what the customers wore. And if the customers were banks, guess what the bankers wore? They wore dark suits and blue shirts. So IBM told its people, so if they wore dark suits and blue shirts, you wore dark suits and blue shirts. This is in the '30s.

And then Lou goes on to explain, then everybody forgot why they did it to begin with. And so here you are in the 1980s, when everybody is going around in jeans and so on, except for the IBM people showing up in dark suits and blue shirts. So that's a very good example of an action that once upon a time, made a lot of sense, totally outlived its usefulness. But through bureaucracy, people forgot, and they thought, oh, that's what it means.

But what was really clear is that if you wanted to embrace the internet, you had to embrace open standards. You couldn't have it both ways. You had to embrace open standards to embrace the internet. You can say, well, how about Microsoft? I guess the answer is, well, there are multiple models, but Microsoft has embraced a lot of open standards, just not Linux and a few others.

And if you said, well, so how do you make money if you embrace open standards? The answer is, very carefully. So let's take browsers. And when we first started our internet division, we had an IBM browser. And we had an IBM HTTP stack, which is the basic technology on top of which you build web servers.

And we made a decision to get rid of them because the browsers-- no, but Netscape was giving away the browsers. When Microsoft entered with Internet Explorer, they were giving away the browser. So it was silly, we felt, to compete in something that you give away.

The HTTP stack, which we had one, we also got rid of it because we noticed that when our IBM people needed to run a really high-performance web server like the one for the Olympics, they used Apache, the open server browser. And we said, why are you doing that? And they said, well, it's much better than what IBM is building.

So the HTTP stack, not only did Apache had over 60% share and IBM's had 2%, but it was better to boot. That's the marketplace is sending you signals when that happens. So we got rid of IBM's proprietary and aggressively embraced Apache.

However, on top of Apache, we built web server products like WebSphere. And WebSphere is proprietary. It's built on a foundation that is totally open. But on top of it, what you built around it is proprietary.

So getting back to this question of balance, if you think carefully your business models, you can carefully decide that you're going to embrace open standards in a whole layer of the stack, in a whole set of components for your offerings. But then you see the money you're saving here and the skills you're saving here, you're now put to enhance it with your own proprietary offerings. And then WebSphere is a huge business for IBM.

Let me give you another example of this balance. Because once IBM got the open bug, we kept marching through. And as you know, IBM really embraced Linux. You can say, well, how much money do you get for Linux? For Linux, we get almost nothing, maybe nothing. It's open source. It's unloaded. There are companies that you can get Linux from Red Hat, Novell, but we're not a distributor.

However, IBM has a multi-billion dollar business around Linux. How so? Linux drives sales of servers. So servers running Linux, you obviously don't charge for Linux, you charge for the servers. All of IBM's proprietary software runs on the Linux operating system. So you sell the software, and you make money from that.

See, this gets back to business model innovation. If you're very careful about where the role of the standards and open source, even, and the role of proprietary you view, to the discussion we had earlier, the open source and standards as almost a way to outsource some of your work to this wonderful community, and then you can participate with that community and probably save 90% of what it would have cost if you had to do it all by yourself. So that's how we did it here.

So first point is it was a balance. And it's something we keep looking over and over and over and over and so do every other company. Any comments or questions about this point? It's a very, I think, profound point. Yes?

AUDIENCE: So when you're basing all your applications with open source--

IRVING WLADAWSKY-BERGER: Not all, I didn't say we're basing all of our-- no, no, this is very. I didn't say all.

AUDIENCE: OK, so you're basing a significant share of your systems on an open source, and that open source keeps changing because it's open, and everybody's adding stuff, it seems that creates a lot of work in order to keep up with.

IRVING WLADAWSKY-BERGER: Well, but the good open-source communities, like Apache or Linux, have governance that keep the interfaces standard. Exactly, you couldn't possibly keep coming out with releases of Linux that make all applications on top of Linux incompatible. You just couldn't do that. So there is a very good governance project.

And remember, you don't have to join every open-source community. You join the ones that you think make sense. And so you usually do due diligence. And you join those communities that have good governance. And part of good governance is exactly what you're saying. Please.

AUDIENCE: So what happens if somebody, another company, comes and releases an open-source version of something like WebSphere?

IRVING WLADAWSKY-BERGER: Well, they exist. No, no, they exist. Hey, what's the open--

AUDIENCE: [INAUDIBLE]

IRVING WLADAWSKY-BERGER: There was Jboss, and there were some others.

AUDIENCE: Yeah, so in such a situation, how would IBM still continue to make money out of WebSphere?

IRVING WLADAWSKY-BERGER: Excellent question because for the most part, the open-source communities attack a high-volume space. And so you then give up that market, and you go to the more complex sections where customers are willing to pay for the extra function, for the testing and the support that you don't get here. That's what it is.

Now, will that always be the case? Is it possible this will keep growing up? It's quite possible. It probably will happen. And then you have to keep adding-- or could it be that after a while, this is no longer a business? Well, sure, but you can make a lot of money here, and eventually, you have to move up to the next level.

AUDIENCE: Yeah, because stuff gets more and more commoditized, you have to keep on innovating.

IRVING WLADAWSKY-BERGER: Exactly, gets back to innovation. If you decide I'll make a stand and take them on, often that means you don't want to innovate. Now, if you're a monopolist, have 90% share, that's a hell of a stand because usually, when you have a huge share, you control the prices and margins, and you have them very high. So the new things are really disruptive, like happened to us in mainframes. Once you're playing in the marketplace, so screw it. If you have to eventually get off, you'll find another set of products to go after.

AUDIENCE: If somebody else comes up with a free version of WebSphere, that basically means you have to redefine your business.

IRVING WLADAWSKY-BERGER: Exactly.

AUDIENCE: Earlier it was developing WebSphere-like products. Now it is something else, like, giving services.

IRVING WLADAWSKY-BERGER: Exactly, exactly, exactly, or keep doing versions of WebSphere that go up to our business, providing more and more business services business performance management, and so on. How about the questions about in-house versus partnering and acquisitions? This is another gigantic cultural change.

One of the most interesting meetings I ever had with Lou Gerstner and companies, when we decided in mid-1996 that we would join Sun and embrace Java. Do you all know what Java is? It's a layer of software that works on all architectures. It was totally developed by Sun.

Sun was IBM's key competitor in the server space. They were kicking our ass. They were just killing us in servers. But Java really was getting a lot of traction, especially for in the internet space and especially for products like WebSphere that we wanted to do.

So we had to go and say, we need to do Java. We need to embrace Java. It's what we need. It cuts across all the platforms. Remember IBM had multiple architectures. So a layer like Java was particularly important to work on multiple architectures. And yeah, we realized we are working with perhaps our toughest competitor in a very important segment, but they have a good thing, so we joined them, and we worked very closely with them.

And that was a critical decision because it was embracing Java and something we did called J2EE, Java Enterprise Edition, is that what it is?

AUDIENCE: Yes.

IRVING WLADAWSKY-BERGER: That it allowed us to move forward and build things like WebSphere. Similarly, with SOA, IBM partnered with Microsoft, and IBM and Microsoft have been the primary companies that have pushed Service-Oriented Architectures, and XML and all this stuff around it.

And do we always agree? Hell no, there are huge battles because everybody is trying to find its own advantage. However, see this gets back-- remember these pictures, which is if you're going after this goal, and if you spend time in a skirmish here, that your clients could care less-- do you think a client cares whether IBM and Microsoft get along? Shut up and get your goddamn stuff to work together and let me run my business. I'm being very serious --that then drives your actions, as opposed to going to visit a client and say, let me tell you why that some people are bad.

You can do that if you are competing for a server sale because my servers are better than their servers, that's a different thing, and clients understand that. But if you're talking about thin clients, one integration, and the fact that you guys don't get along is in the way, the clients will punish you by not buying from you if you don't cooperate. So we cooperated.

At the same time, there were major areas we were lousy with our internal products, and we had to make acquisitions. And probably the two critical acquisitions we made were Lotus for office systems and collaboration and Tivoli for systems management. Those were critical acquisitions.

And prior to those acquisitions, the mood in IBM was that making an acquisition was a failure. That was like, you're not man enough to build it yourself. I'm not kidding. I mean it. If you are good, you would have built it yourself. So the fact that you keep failing for 10 years in a row on office systems, aren't you man enough to try the 11th time?

Now, usually if you fail for 10 years, maybe there is something in your DNA in the culture of the company that makes it very difficult to do certain kinds of systems. So as we talked about, I believe last week, you are not good at everything. And if you say, well, but I should be good at that. Well, maybe you should be good at that. You are not.

So if you are not, then you need an injection of culture and talent, obviously. You don't just buy culture. A McKinsey study that tells you how to change your culture. You can do that, too, give them a lot of money, and that and $5 will get you a nice cup of coffee. But when it comes to culture, there are many other things that will help you a lot.

But what you really need often, as we discussed last time, is an injection of new blood. And these acquisitions were new blood. And IBM culture has changed radically. So we've become more open to acquisitions like Cisco has and other companies.

So often there are companies in the marketplace that are very successful. They have done something. Could we have done it? Absolutely, we could have done it, but we didn't do it. It's like asking you, well, why didn't you buy this stock when it was low, and now it's so high? Couldn't you have bought it? Yes, you could have bought it. Why didn't you buy it? Well, sometimes you do, sometimes you don't.

But it sounds like I'm being glib, but it takes culturally you have to think objectively as opposed to feeling like you failed. And therefore, because you didn't do it, you're not going to make the acquisition. And the net result is you won't be successful in the marketplace. That's really stupid. But we had to get that cultural barrier this pushed us to get. Questions?

Now, when we were starting with the business and organizing our offerings, we had to pick certain things we would do, certain things we wouldn't do. I already told you we said browsers. First of all, Microsoft and Netscape were having what people call the browser wars. Do you all know what I'm talking about?

So could we have thrown our hat in the ring in those browser wars? We could have, but we all knew how that would have ended that we would have been crushed between these two people fighting for their lives.

Microsoft felt that if Netscape established dominance in their browser, it put their whole Windows franchise in peril. So they were going to do whatever it took. And Netscape was fighting for its life, which eventually it lost. And we wouldn't have done anything there. So we didn't. We got out of that.

But we found that people who were trying to do e-business, remember, integrate the web with IT, our enterprise customers needed a number of important things we were very good at. Security, remember now, for the first time, you were opening your jewels, applications, and data, to everybody. And I don't need to tell you, not everybody out there who came in was a good guy.

So all of a sudden, firewalls and security, that became a very big business, built security, hosting. Remember, before the internet, everybody knew exactly how big a machine they needed, how big everything they needed because they knew exactly how many people were going to access it. With the internet, you had no idea.

So a lot of people said, this is a pain in the ass. Why don't you host it for me, and you handle the variable scalability of hosting? So we developed hosting services. Backend integration, integrate the legacy backends, a lot of it was IBM systems, with the new web servers from IBM and others. So we built a whole set of new offerings from our services, from our products, that were very, very valuable in the internet and fit what we were good at.

And then we started to organize the internet offerings. These three categories, which we actually came up with in '96, they continue to make a lot of sense. And we said, when all is said and done, there are three classes of applications in the world of the web content, collaboration, and commerce.

And so we started counting everything, also, our market, our communications, our education, content, collaboration and commerce over and over and over and over. The reason this is very important is we learned to say, well, God, what the hell is any business application that is too amorphous a subject?

But if we say, OK, let's talk about content. What do you mean by content? Well, how about a website? Do you have a website? I don't have a goddamn website. Well, we can help you build a website. How do you help me build a website?

So now you can start doing offerings, customer self-service. By the way, I think customer self-service, in retrospect, was the killer app in the world of the web. Things like FedEx, UPS, and huge number of others that allow people to do for themselves things they couldn't do before, get access to information, those just those exploded. Those were, I think, the killer app. And those primarily required backend web integration and security. Remember, you open up the data center to everybody, you have to have very good security. So we developed that.

Collaboration, played very well because of our Lotus acquisition. We positioned Lotus Notes very much as a platform for collaboration and started building more and more offerings. Also, a lot of the intranet applications, intranet are applications inside the company to help employees share information and communicate, were collaboration in nature.

And then commerce, and remember, especially-- not at the beginning '96, '97 but as the internet got better, e-commerce just took off. And there was B2C and B2B, and you'll hear, in two weeks, from Chris McCormick of L.L.Bean, who was actually our launch partner for our e-commerce platform. And this kept going up more and more and more.

So we started imposing a structure that proved very beneficial, if nothing else, because it allowed the salespeople to know what the hell to talk about. Otherwise, they say, internet, well, good, what do you want me to do? Well, we have to have the internet. Well, you have to do more than that.

AUDIENCE: So you mentioned content. Did IBM ever think of doing a search engine portal?

IRVING WLADAWSKY-BERGER: Yes, and we did some, for patents, we did a superb-- but it never developed into a business. The first year I was the general manager of the internet division, I spent more time killing offerings we had started in the past than building new ones because we had started a lot of user-oriented activities that we were lousy at and substituting it for things we were much better at.

Because we just were not good at it, I don't know. It's almost like if you say, why is again, Michael Jordan not good at baseball, not as good in baseball as in basketball, I don't know, but we were just not as good in those things. So we tried a lot, and we tried and we killed lots of stuff. We killed lots and lots of stuff.

Now, the last one is very interesting because we needed to count something. Because if you didn't count anything, how the hell did you know if you were making progress? Counting revenue and profit didn't make any sense because this stuff was just ramping up.

So in the first few years, what we counted primarily, was marquee accounts in each area that we were interested. So when we had reviews with Gerstner and others-- so what new clients do we have in content, in collaboration, in commerce, in finance industry, distribution industry, transportation industry?

And so we were counting the really how were we winning in each major area that we were after. But we were defining winning as these clients were partnering with us. The deals were initially not very large. So from the point of view of the bottom line, they were minuscule.

But if you didn't do those deals, you didn't exist. When I say there were minuscule, if you were a brand-new company, you were very proud of those deals. Remember the discussions we had that if you're a larger company, the bar for what catches management attention is so high that you leave a lot of good ideas on the table.

However, if you count engagements by industry, by clients, and all of a sudden, you can start asking questions, well, Jesus, why did Citibank go with our competitor in this area instead of with us? I mean, what's wrong? Are we upset or not? Yes, we're very upset. Well, why did we lose? So you can start asking questions like that.

We had a few set of projects. We counted also financials. But how can I say that? It wasn't counting like a mature sector doing through quarterly earnings. What we were doing, in the early stages, was below the radar when it comes to quarterly earnings, rounding off, which is true any time you're starting something new.

However, there were a few key products we were counting. How are we doing in Lotus products, Domino, in particular? So we counted those. How are we doing in WebSphere when it started? So we were counting those. How about hosting deals? We were counting those.

And then we counted subjectively, how much revenue from the other parts of the company are being driven as a result of the internet initiative. So if a client bought extra mainframes because of this new e-business application, obviously the revenue for the mainframe went to the systems unit, but we counted that as indirect revenue.

And after awhile, we had over 20%, 25% of IBM's revenues were due to the internet. And by then, the game was over. When you get to that number, it's time to go away because now you're a pain to-- let them do the work themselves.

So we had to count because if you don't have the discipline of measuring, you don't have any discipline, and you don't know if you're winning or losing, but you cannot count what you count in a mature business that is material to the quarterly earnings. Any questions or comments? And we'll review this some more when we start next week. Thank you very much. And I'll see you on Thursday.