A CONVERSATION WITH TED SELKER

Ted Selker is an IBM Fellow and Manager of the User System Ergonomics Research (USER) Department at IBM Almaden Research Center. His group specializes in understanding the physical, cognitive, graphical and interpersonal needs of users.

The TrackPoint II in-keyboard pointing device typifies this on-going physical interface work. The COgnitive Adaptive Computer Help (COACH) system is an example of the group's agent research, exploring the use and value of adaptive user model based agents to change the way we use computers.

Since 1992 Ted and his group at USER have created products such as the Two-Handed Keyboard, TrackPoint Mouse, and Tactile TrackPoint, as well as such software as Web Browser Intelligence (WBI) and Comet Cursor (cursor visibility for LCD displays).

In this conversation, Ted and *interactions* Design column editor Kate Ehrlich discuss Ted's work before IBM and his eye toward the future, as well as the physiology, ergonomics, experimentation, and testing involved in the development of the TrackPoint.

Kate Ehrlich

I started paging back and forth wildly and thinking about this stuff and said, "I've got to integrate the pointing device into that qwerty space, and I can get rid of this 1¾ seconds of down time."

Kate Ehrlich: You're probably best known for your involvement in the IBM ThinkPad, which you carefully brought with you. Can you tell me a little bit about that project?

Ted Selker: Well, to be honest, I started being interested in integrating a pointing device into the qwerty keyboard in 1984 when I first saw some data that showed that it takes time to go over to a mouse and time to come back. Also, I noticed that novices in these first studies by English and his colleagues at PARC were actually outperforming the users of the mouse with a knee bar—a knee bar like you have on a sewing machine. When I realized it, I saw those two things together, nobody else saw this, but I saw this in one book. The data were a few pages apart from each other.

I started paging back and forth wildly and thinking about this stuff and said, "I've got to integrate the pointing device into that qwerty space, and I can get rid of this 1 3/4 seconds of down time." And maybe that's the reason these knee bars were working for these novices because even though the knee—there's physiological research on this—even though we have almost no representation in the brain for the knee, so it's a crappy pointing device, but so is the mouse when you are not used to using it. Maybe as soon as you got to be good at it, the mouse would outperform the knee bar, but in the meantime, the advantage of not having to take your hands away from the keyboard won. So then I started thinking about in-keyboard pointing devices, and that was the whole goal.

It was not to put it into a notebook computer. I've always thought that these dynabook kind of things were wonderful and exciting, but in fact, I was discouraged that the first people that would use it were using it for one of the really auxiliary reasons—that there isn't space for a pointing device on a notebook computer. I have been crazy about notebook computers ever since the beginning. I wanted to get a Grid. Ever since '85, I've been using, you know, the latest and simplest ones I could find.

Anyway, what happened is that about 1990 I got to the point where we had learned enough about pointing that we had an in-key-board pointing device that just outperformed a mouse for any integrated task, like doing mail, like doing text editing, like doing Lotus Notes. At that point, I went off and made 100 keyboards and was very excited about that, but the process of getting it into product was a much longer one. It involved working with the people that made the notebook and refining their idea of why they needed it. And that took time.

We did get a keyboard made with Track-Point, but actually we never got the buy-in to make it part of the mainstream products, the PC products. In spite of the fact that we have always made it with an auxiliary connector. So you could use the TrackPoint in addition to a mouse, and we did a special microcode thing where if somebody is using that mouse another person can't bump whatever they are doing. There's this social amount of pacing that makes it so that if you let go a 1/3 of a second later, they both are active—whoever takes control then gets it. Unlike other companies' machines where if there are two mice then if either person seizes one they ruin the other person's work, ours didn't.

Kate: Tell us about the connection between the idea of the TrackPoint and the connection to the ThinkPad product. Was there a version of the ThinkPad that came out without the TrackPoint?

Ted: Yeah, there was, first there was the L40SX. It was a tan machine that did not have a Track-Point in it. They thought of putting it in. They made six of them that had them in it. They just didn't get it, and it flopped. The next product flopped. The next product after that flopped.

When we finally got it into a notebook, which was October 1992, it was the ThinkPad 700. It was the first IBM ThinkPad notebook that even sold as many as they had planned to sell. At that point, they tripled their plan three times in three months and finally weren't able to find enough parts to make them, and they couldn't produce any more. It took them two years before they caught up with people's interest in it. That was the first time. It was the second product that had the TFT display in it. The TFT display, people say, was the other reason that that product worked out.

Kate: What does TFT stand for?

Ted: Thin Film Transistor. It's the LCD displays where they can turn on and off the LCD fast enough that if you move a cursor across the screen it doesn't disappear—we call that sub-marining, the cursor just disappears. So we had this very large, nice display in the product pre-ceding the 700 and then in the 700 we had both TFT and the TrackPoint. I just kind of gloat about the fact that I may have just hit it just right. For one reason or another, that product did extremely well.

And there was this wonderful Vice President at the time that took me into his office and said what's the next TrackPoint? I gave him five ideas, and he said do them all. The biggest one that you know of that went into product is the projection idea—that of taking the display and separating it from the base so that it becomes a projector, the panel display. I am extremely proud of that. IBM made a small number of them. They all sold. They had backorders. They did well financially from it, but they really felt they would have to hunker down and work on the standard line of ThinkPads now. I hope we get around to doing a job where we really pro-mote it.

Kate: Going back to the TrackPoint for a minute. You started to talk about having come up with the idea many years before based on data. After you developed the TrackPoint, was there any data that you had collected or other people had collected that talks about how it compares with other devices, such as a mouse, the Macintosh Trackpad or the conventional trackballs?

Ted: Basically, the hard part for the TrackPoint was that there are 100 years of data saying that great control devices, like a gas pedal for instance, are bad at doing position control or at least are worse than position-control devices. There are a bunch of issues here. A tablet is a position-control device. A mouse isn't really quite as much one.

All mice have transfer functions that make them have acceleration—go faster—and that's because people can't stand going off the edge of the pad. There are all sorts of other reasons,

but all in all, those algorithms actually make them worse at pointing, except they avoid a catastrophic event, which is picking up the mouse—you have to avoid that. You notice you don't pick up the mouse very often. Remember at the beginning we used to, and that was bad.

That's why people like trackballs. Track-balls are maybe 20% worse than a mouse. Some people like track-balls just because they are always in the same place. They take up a space, and they are always there. If they are always there, you can reach for them without having to look.

Kate: How did you arrive at the particular design that you had where the pointer is between the G, B, H, and N?

Ted: There are two issues. One is how to make a joystick be a good pointing device. That's the hardest story. This part was what I was really going after in the beginning. The thumb posi-tioning—we couldn't figure out how to get the buttons to work.

Kate: So you then had the idea that there was going to be separate buttons and that the but-tons were going to be where they are?

Ted: There were going to be buttons. With the joystick, we used the keyboard for the but-tons, but it was confusing; so we started going away from the joystick and from the keyboard at the same time. For a while, we used the pointing device with the keyboard buttons; then we separated them. We tried putting the joystick over to the right. We tried putting it on the top. We tried to put it down in the thumb position, and in fact, I designed the thumb position one for TI. TI used the thumb
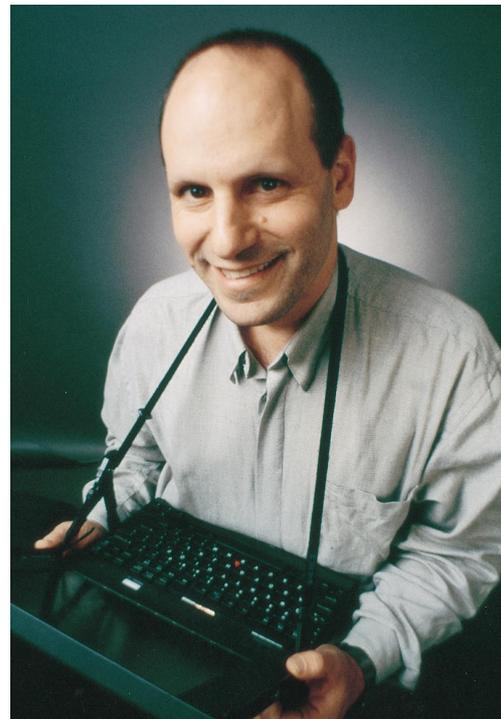
Ted Selker
IBM Almaden Research Center
650 Harry Road
San Jose, CA 95120-6099
selker@almaden.ibm.com

Kate Ehrlich is Design Column Editor of *interactions* and can be reached at Kate_Ehrlich@lotus.com

position for a while, and that was based on the design of mine.

At one point, I designed a wrist rest that has a pointing device that is cute because I tried to figure out how to fix this problem of the buttons getting in the way of the pointing devices. I did it. It took me forever to come up with this, I don't know why. The pointing device is on an angle, and the buttons are above that angle so that you can do the buttons without getting in the way of the thumb by having them be at a different angle. Also, this angle thing, there's a problem with using the thumb as a pointing device.

We don't understand everything, but even though the thumb has a bigger representation in the brain than the forefinger, it seems to be worse at doing noncollaborative activities. It tends to be that your thumb is always working with other things, gripping and grasping and stuff. And your forefinger, for some reason, is always in charge. So, anyway, that was an idea—the thumb thing. I was more excited about the forefinger, and so we tried it between the G and the H. Finally, we came to making those 100 of them right between the G and the H, and that was a great idea, I thought, because your fingers don't ever cross that line. Unfortunately, your fingers get as fat in the middle of the button, and it hit the post. We used to make the post about a certain size, and it kind of became painful. We didn't care—we're scientists right? Other people started bothering us about that, so we went away from that.

In the end, we decided that basically, your right index finger goes down to the B, and your left index finger goes to the V. And then it comes close to the TrackPoint spot, but you never go across to that place.

Let's get back to how that TrackPoint ended up being any good because the joysticks were 20% or 30% slower than a mouse. So what in retrospect turned out to be the big deal, was that I had this guy who started working for me, Joe Rutlage.

Joe Rutlage is about 68 years old now. This happened 10 years ago, so he was maybe 59. He's 6 feet 6, and he's got a white beard that's about this long. A gigantic logician, and he decided that he was sick of doing logician work. And paper and pencil was driving him

crazy. He didn't have anything else to do, so he started working on this. It was wonderful teaching this guy who had programmed the ENIAC, teaching him about electronics.

Joe and I designed an experiment where we would allow ourselves to change the transfer function. That's how hard you press on it and how fast it goes. Then run a little experiment, and then plot the results all in this little circle; we could do this whole thing on ourselves where the experiment took about 30 seconds, and the rest of the whole process took about 60 seconds, except for looking at the graph.

By doing that, we learned something very interesting. What happened was, one day, we made a transfer function that was so hard to press that it hurt your finger, but it had a major improvement in performance over all of the others that we had done before. We didn't understand it. First, we thought, well, it was the dynamic range. With all of that pressure it's got to be more dynamic range with more pressure; therefore, you can have more control. One thing I should mention is that we were making our own joysticks.

Kate: These experiments were done on joysticks or on things that were things that looked like the TrackPoint?

Ted: They were done on things that were like a 2 1/2 millimeter diameter rod that was sticking up because it was trying to be between the G and the H.

Kate: Was it embedded in a keyboard?

Ted: Yes.

The first thing that would happen is that we were getting bad results from these. I had invented this wonderful technology where you take a piece of conductive foam, like the kind you keep chips on, and you put copper above it and below it. You put this sandwich together, and you've got a pointing device. You've got a resistive pointing device, but it turns out it was a bad sensor. So we went out and bought a $500 sensor, so we had this sensor that was so accurate and sensitive that when you put your finger on it, you could see your heartbeat. And it kept reading up until 20

pounds. That's when we started collecting the data that was really disturbing, which was that we couldn't point with this thing even though it was so accurate.

There was this feeling that we noticed that we actually traced in other people's literature back to the 1980's, where with these joysticks, you would make a selection, but then you'd go zoom and then you'd have to go backtrack to get to it. So there's this overshoot problem. Well, we found ourselves not overshooting when we pressed so hard. So we did this study. The study went like this: We made a pointing device study to test how much force-accuracy a person can have.

Kate: Now, how were you able to make this device so you could change the force resistance in it so quickly?

Ted: We got a computer and put it in a data acquisition board.

Kate: What's that?

Ted: When we started out, we thought all we

had to do was make us some electronics to make the joystick's signal into a mouse output. We did that with a big breadboard of the TTL logic and analog devices and stuff. It was really fun because I got to teach it. Joe hadn't done electronics, when he worked with hardware it was tubes, relays and tubes. Anyway, as we went forward, finally, we made this thing and it didn't work quite the way we thought. So we said well, let's get a computer in here so we can change the relationship between the input and the output. So we got a data acquisition board and put it into a PC. In fact, that whole project, up until we got the $500 joystick, there was zero funding for it, so we had gotten an old luggable PC out of the hallway that someone was throwing out and a microscope, a microscope that somebody had dropped, and the mirrors had broken off inside. We glued them back together. It was great. It was great.

Kate: What were the acceptance issues, both within IBM and then with users? Because it's not obvious from looking at it, initially, what the function is, whereas with the trackball there's an affordance. There's a ball, you know

TrackPoint Mouse

Traditional GUIs have been based on a single pointing device for on-screen cursor manipulation. TrackPoint Mouse is a proposed Dual Pointing Device that works with existing GUIs. The primary pointing device remains as the cursor controller. The secondary TrackPoint can serve multiple purposes. Here are some software applications that take advantage of an additional TrackPoint:

SCROLLING
Instead of moving the cursor to use the scrollbars of a document, the user can now scroll any window using the additional Track-Point.

TWO CURSOR
With two cursors on a large desktop screen, the user will be able to increase productivity by using the primary cursor for editing, while leaving the secondary cursor at menubars or toolbars for selection purposes.

ZOOM WINDOW

This is a software simulation of a magnifying glass. The zoom window will magnify anything on the screen, directly over the target area. The secondary TrackPoint will be used to move this zoom window around the desktop.

One day the lead industrial designer of all of the industrial designers at IBM gave us 30 seconds of his time. He walks in and he wiggles the thing and he says, "I can't draw a circle." And he walked out.

that it moves. With a pad, there's a certain affordance that indicates that you should do things. But with the TrackPoint, until you start using it, you are not aware that it actually has so many degrees of freedom. What were the acceptance issues within IBM on this?

Ted: Well, you know, we tackled them with data. We had people in Yamato do the experiments. We had people in Austin, and people in Boca Raton, do experiments. The pad had been used in the Apollo in 1980. It was not a good pointing device. It was always worse than a mouse. The pad cannot be used as an integrated pointing device. By the time that had happened, the pad wasn't being used by anybody yet because you cannot use the pad with your thumb. (Women with small hands can, sometimes.) Because you can't use it with your thumb, you have to take your hands off the keyboard. It's as bad as going over to the mouse.

So, you see, because my whole orientation was about this integrated pointing device, and I was running around with data showing that it was valuable, that was the whole story. The acceptance issues were difficult because the first ones had drift problems, and it's such a strange thing to change the keyboard. And the cost—the cost seems to be the biggest issue.

Kate: In developing any product, there are always tradeoffs. Tell me about some of those and how they got resolved.

Ted: The tradeoffs thing is interesting because I think that sometimes people think they have to have tradeoffs when, in fact, if you think harder about it and you really take the constraints seriously, you can design win-win situations.

For example, we used to think we had to have a cup-shaped top on top of the Track-Point. This cup-shaped top was cool because it kept your finger from wanting to go on the side of TrackPoint. If it went to the side, then you had to move it to the other side to push the other direction. That's not good, which is one of the reasons a track pad isn't as fast. Anyway, your finger goes on top, but the problem is that it was this very firm controller and we thought because we had done a test with a

moving TrackPoint that it was that firm interface between your finger and the TrackPoint that gave you this control. If you had a mushy interface, you wouldn't have the control. We had a little bit of data about that. Unfortunately, when we went to Japan, the guy said look this cup-shaped thing hurts my finger. It makes a little ring in my finger. You've got to get rid of this.

Kate: There were competing products.

Ted: Oh, yes. Oh, yes. In fact, TrackPoint I was a trackball but turned over and was a mouse. It was designed for the L40SX, in competition with what is now TrackPoint. Very big people involved going into that product, and we did not get our pointing device, which was called the pointing stick at the time, to be evaluated as well, I think. In any case, it wasn't as mature.

One of the things that I kind of am embarrassed about—though I think is very natural—is that once the product people actually decided to work with us we learned more about our technology from that point to when it hit the product than I think we ever imagined was possible.

There was a guy that was assigned to looking for bugs. He found bugs like two or three a day. Before that, it had been two of us working on this alone, and we couldn't believe they put a whole other person finding bugs. It wasn't always bugs like there was something that was broken. It was just all sorts of little, funny things.

One day, for example, the lead industrial designer of all of the industrial designers at IBM gave us 30 seconds of his time. He walks in and he wiggles the thing and he says, "I can't draw a circle." And he walked out. This was very depressing to us because we thought we had this wonderful pointing device. We thought that by having these X and Y axis orientation proclivities that we could do text better because you would stay on the line and go between lines better, and we thought we had some data about that. It's very easy to get yourself confused, even by data. So we went home and we tried to make one that they could draw a circle with. In fact, we were able to make one that would make a circle without any of these orientation things for text, and it

didn't make the thing worse at pointing. That was a big surprise to us.

By being pushed from the outside we learned things that we never imagined we would learn. We had already taken an ego stance on it. We believed we knew something about it. For example, with that hard rubber thing—remember, after we had made those 100 keyboards we thought we knew every-thing now—this engineer sat with me and we got every material I could ever think of and I built dozens of different tops—out of cork, out of sandpaper, out of sorbithane.

Slowly we came to realize that if you made the top surface kind of mushy, it would make a cup for the finger, and if you made just the right stiffness on the sides, you'd get a little feedback, which makes the muscles better at being accurate. And we actually got a perfor-mance improvement to get that first 3 minutes better than a trackball.

Kate: Were there any tradeoffs around the placement of this TrackPoint in the middle of the keyboard? Was that going to make the construction of the keyboard more expensive? Did it have implications for the positioning of the electronics underneath?

Ted: Well, I used to do this trick. The party trick was that people would say it can't be put into this keyboard. I'd pull a Dremmel tool and elec-tronics and a TrackPoint sensor out of my brief-case and I'd say would you mind if I do it right here. My favorite moment of that sort was there was this rosewood table, and everyone was wearing suits. We were having this argument, and I said let me just end it. And I did it there. And they were all impressed because I took a random computer—this was one we were going to procure from someone—and did this.

Then they said, "Well, but it can't be manu-factured because the people that are making the keyboards are making them by robot." It turns out I was going close to where they were going to be making those keyboards. So I went there. They said, "Look, we're using robots, which is an automated process, we can't use it." Well, they were wrong. They weren't using robots. They were using fixtures. Fixtures aren't programmable and robots are.

USER Lab at IBM Almaden
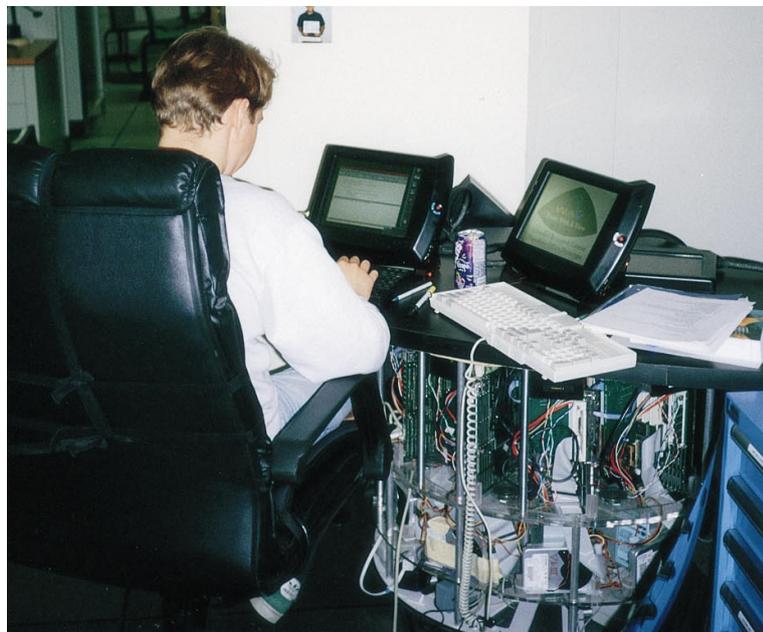
I went and looked at the process, and they said, "See, we would have to make the key-board; then we would have to take it off the end of the line, take it apart, put the thing in, and it would double the cost." And I said, "Well, excuse me if I am wrong, I'm not a man-ufacturing engineer. You guys are some of the world's best. I know that, but at step 7," or whatever the step was, "it seems like right before you put the keycaps on, if you took it out of the manufacturing line, put the Track-Point in, and put it back in right there, you'd be able to do it." They fell into Japanese. After a moment or two they said, "Yes, actually, that's a fine idea."

And again, when we're going to put the TrackPoint into the keyboard, they said, "It can't be produced because the sensors are too expensive." I got a lead on a procurement story where somebody had strain gauges that were $1.40 a piece instead of $15. I went and got that vendor, and I found them. I went to these guys and said, what if you use this sensor? So that stopped that argument. Then they said you can't have analog electronics inside of a computer, and by putting them in various of these things and making them work over and over again, it kind of became a pride issue.

Basically, when somebody is doing some-thing, they are the shepherd of it, and to change directions is always difficult because they are scared because they won't keep being

Two-Handed Keyboard

Most GUI applications are centered on using only the dominant hand for pointing and manipulation of objects on the desktop. However, in real life, many actions are more easily accomplished with both hands. The Two-Handed TrackPoint gives you two pointing devices embedded in your keyboard. Some applications of Two-Handed pointing are:

DRAWING
Detailed work is done on a drawing with the dominant hand; selecting tools with the other. Manipulating pixels with the dominant hand; selecting colors with the other. Drawing rectangles and other shapes by manipulating both corners at once.

TEXT
Selecting text with the dominant hand; modifying its style with the other. Positioning the cursor with the dominant hand; scrolling around the document with the other. Windowing: Rapidly selecting between two windows by keeping one cursor in each. Marking text in one window and pasting it in another.

This simple extension of TrackPoint technology has the possibility of speeding many user-interface tasks.

able to do it. You know. So, in this case anyway, we had to solve every single barrier.

One of the best moments was when some people didn't want there to be a press release because this was proprietary IBM information. Finally, when they weren't going to put it into product, I gave the guys a deadlock. If you don't make a decision by this date, I'll do a press release. And the press release led to an article that was very lovely about this wonderful thing. They gave it some space in their magazine and that really helped us through a couple of hurdles

So, the shepherds have to have this feeling that customers would want it, that they could do it physically, that the process wouldn't be destroyed, that it wouldn't cost them out of house and home, and we found ourselves having to solve lots of very tiny questions... the rubber top... the sensor design….

I remember very distinctly the product manager clamping his hand over his telephone when he talked to the part of the company that would have been responsible for the microcode and them giving him a price that was beyond his budget. It would have meant that it wouldn't have happened. He asked me, do I have to work with these people and I said no, we'll do it. I didn't realize what I was sign-

ing up for. It was just months and months of 12, 14, 15, 16 hour days and lots of airplane trips, but the point is, at that point just realize that it allowed us to improve things, to improve the product because we were close enough to doing it. So I hope this isn't how all product transfer works, but the product transfer where the champion stays involved to the end sometimes has advantages. We were able to keep the vision.

Kate: Tell me more about how the connection with the ThinkPad came about.

Ted: Well, again, I think it's so important to go out into the community and get reactions. We wrote a paper in 1990, which got published in Interact, and we had already tried to get people at IBM interested. They hadn't caught the fever or something. When we went out into the community and we had this demo that we gave at Interact, everybody was crazy about it. There was this videotape also. Gosh, that may have been the next year. What happened was that there were people from IBM Boca that were there were saw how excited the interaction community was about this thing and saw that this idea was making some waves, and they

went home and invited us to come visit them.

Kate: And Boca was where the ThinkPad...

Ted: Boca was one of the projects where IBM was making the notebook computer. In fact, the next one that was coming out was going to come from there. This design team tried to take it seriously. So we taught them how to evaluate each pointing device. They evaluated it, thought about it, almost went for it, decided not to for schedule and other reasons that I don't know. It got some people focused on it.

From there, we went over and we tried to get people in the RS6000 interested in it. They did a very nice study that shows that if you gave a person a desktop keyboard with a TrackPoint in it and a mouse, and you gave them a choice of either adjusting the Track-Point to improve it or not, the people that had a chance to adjust it to improve it always said, "Oh, yeah, it was kind of sluggish. Then I fixed it up and now it's a lot nicer, but I use the mouse." The other group—60% of the other group—had quit using their mouse and were using a TrackPoint on a regular basis. A full 30% of those people actually took their mouse and put it in the drawer because of the space that they could save.

The story, at that point, again, was extra cost. It was difficult to justify for the desktop. So it really became because clear that there were so many reasons to put it into the notebook computer that that was what we focused on.

Kate: How far down your exploration path did you feel like there was commitment from the ThinkPad team to do this? It sounds like there was a long period where you were doing some experiments and some exploration, you were gathering data, you were going back, you were doing some more changes.

Ted: Nine months of the 7 years was with them having a commitment.

Kate: You mean 9 months before they shipped they finally committed?

Ted: Right. They said we're going to put this in the product. We have to work it in a synchro-nized way, and I didn't realize just how big of a challenge it was for them because typically products tend to be made out of subsystems that are all known technologies from vendors or from internal vendors. This was, we didn't realize it, but it was quite immature technology. That 9 months was spent making it into a mature technology.

We had 100 users out there, but we didn't realize all of the implications of having to work with OS/2 and Windows and AIX and DOS and having to have it work inside a machine that has 100MHz clocks and to work in the cold and work in the hot and work after you've dropped something on the keyboard. We learned and learned and learned... for example about electromagnetic radiation problems. Do you know that when you take a probe with a 30,000 volt charge on it, and run it over the outside of an electronic device, it has to be able to stand that and not destroy it? Think about 30,000 volts in a sensor that is measuring millivolts. It had to work.

Kate: You have been talking about your involvement with this project, but I know there are lots of other things that you have done. Actually, you have a much broader, a much more eclectic background than one might suspect from the discussion about this TrackPoint. Can you tell me a little bit about the range of things that you have done, where you have started from...

T.S. I've always loved tools and making things, but typically after I get a tool I am more interested in the tool than using it. I was thinking about a ham radio as I said that. From very early on, when I was 2 years old, I used to drag a lamp around and plug it in. I loved that mechanism, that technology. Early on, I was very excited about electronics. As an 8-year-old, I was trying to make motors and trying to understand how to make microphones and speakers and stuff like that.

I got a ham radio license when I was 12 or so. I have always been interested in electronics. As a child, it was my hobby. Then I have this spatial, physical thing. I've always liked to build physical things. When I went to college, I got very excited about industrial design with my

I did a lot of sculpture. I was always very interested in the creative process and I did a lot of photography. The spatial-visual thing was one interest. The electronics and building things were another. Then computers were happening.

interest in creating tools, but discouraged by the industrial designs—interested in form not function—I found myself more interested in the function part. So I backed away from that and did more engineering stuff.

Kate: So was industrial design what you went to study?

Ted: No. I went to study applied math, actually. Rhode Island School of Design was right next to Brown. I was very excited about the industrial design program and thinking about it. But I went down there, and they were showing off this A+ project, which was a hair drying thing that had the intake right next to the ear where it would pull in the hair. I didn't care about the shape after I saw that. I was so pissed that someone got an A+ for a hair dryer that would be totally noisy and maybe dangerous. I just thought, "I'm going to care about this stuff, but I'm not going to study with these people." I'm sure they are great people, but it was just a kid's decision.

So what happened to me then was I thought that psychology is fascinating. How do people work and care about people? I went off to study neurophysiology. My undergraduate thesis is a model of how our eyes focus and then making an actual automatic-focusing, automatic-imaging system. I had this tendency to invent. I have a question that's kind of a scientific question, and I wrote it up as a scientific thing. But I built this thing. Maybe I should have gone off and sold it to the camera companies to make focused cameras. That's kind of the way I tend to act.

When I got to graduate school and started doing more brain modeling stuff, I started getting more discouraged about the fact that we can't understand the brain in my lifetime, and what is it doing for people anyway? I wanted to make more tools. Simultaneously, I was spending a lot of time doing photography and sculpture. I build a 6-ton sculpture around that time that was really gorgeous.

Kate: Do you still have it?

Ted: Well, it was on a college campus. It was a memorial for my mom. It was out of oak, and it was between a bunch of buildings in the wet

world of the Northwest, where all of that open-grained oak from all of the carving got wet, got mossy. It's gone now. I did a lot of sculpture. I was always very interested in the creative process and I did a lot of photography. The spatial-visual thing was one interest. The electronics and building things were another. Then computers were happening. I was getting very excited. Andy van Dam was a wonderful professor at Brown, so I learned about computer science and this idea that you could take something and turn it into a concept and get rid of the thing. That's what computer programs were great about.

My interest in understanding the brain led me, naturally, into artificial intelligence. In fact, in my Ph.D. thesis there was this idea of trying to understand how they make a computer that would act in a smart way with the person by mimicking or recording something to learn and improve the relationship. That sort of thing, of course, has turned into agents. The first agent I wrote was in '79. I made this adaptive kaleidoscope to try to make something.

All of the people trying to make these smart computer programs were always thinking about how to make them do things that are hard for people cognitively—like chess. They all turned out to be representation problems. I thought, "Well, if I really want to understand what it is to make a computer act intelligently, I had better not take on something like this great big hard problem on top of a hard problem. Why don't I just make it so the interaction feels smart and try to work in a domain that is not one that humans are good at."

So I made this kaleidoscope. Who's an expert at looking at kaleidoscopes? I used a kaleidoscope because I had lots of feedback. People are good at getting feedback from their eyes. I made it so that the interaction was just a yes or a no, so it was a simple interface to use. You notice I was getting interested in user interface. People loved it. This thing would just model whether you liked or didn't like it by watching which level of the kaleidoscope, like the rectangles being big or small and colored or not, and people loved it and played with it a lot.

That was really the beginning of me having this idea of an adaptive user model, which is what ended up being my Ph.D. thesis. They

(Left): Various experimental versions of: mice and other input devices; crazy plugs; clay mock-ups of remote control devices.

told me that it was no good to work on aesthetics, which is very depressing to me because I care about aesthetics. I always had this kind of amateur arts streak in me. I went off and I did this help system that had this learning model. I wanted to show that you can do learning and reasoning in real time for a person. The only way I'd know that it really worked is if people did better with it than not. That was COACH. COACH coached that, it achieved that.

We had this adaptive help system that actually gave people more of an affordance to be able to write LISP in our studies, LISP programs faster, learning the language faster, enjoying the experience more, writing better code than if they had the same interface with the same stuff available, but didn't have it being splashed up on their screen without them asking for it and not having to change the way it presented without them asking for it. So this really, I think, is the center of the Ben Shneiderman and Pattie Maes debate because, you know, he says it could never happen and she says it will happen someday. I've got some data that shows that actually I can have something that's doing something proactively without being asked to and actually improves people's perception, a feeling of control, and their ability to get it done and give you papers on it. I'm really proud of it.

Kate: There were some interesting companies that you worked for at interesting places.

Ted: Yeah. Well, I actually was part of and solved a major problem. The first digital dial indicator that was made, I was at Brown and Sharp. They are the people that invented the

machine that makes gears. One summer in college I almost left school to go be part of that company and what I did was I spent the summer being part of a group inventing the first digital dial indicator.

Basically, I went to graduate school at U Mass and then I got disgusted with graduate school when I found out that my seminar series that I was teaching at Stanford as a visiting graduate school student from U Mass was attended mostly by industry people and industrial research houses, not professors and students. It wasn't completely true, but that's kind of how I had come to feel. When Alan Kay came and saw some work I was doing called Dreamlighter, which was an idea of how to make a pen that is better than a pen, and offered me a job at—to my point of my view with the best researchers and user interface—at Atari, I jumped and just quit graduate school. I went to work for Alan Kay. That was really, really fun. Then when that collapsed, I was devastated.

I did all of this work and I did a bunch of user interface consulting. That was really fun, but then I got a job at Parc working on Colab.

The way I got the job working at Colab was while I was at graduate school, I built a very interesting piece of hardware for Stanford that allowed all of the faculty and students in the building to do collaborative computing by sharing a frame buffer so that everyone's terminal could, in a thirtieth of a second be changed to be looking at another person's terminal. That was really exciting work because it was just amazing how by actually sharing the physical real estate of the user interface, people would find themselves collaborating on a paper by grunting and talking into a phone piece while

they were typing on each other's stuff.

They would be sitting next to each other not even talking but just typing together. I was so excited about that that I actually stopped working on my thesis work for nine months to improve that and build a digital version of it and to add in other multimedia capabilities. I hooked it up to a bunch of televisions and television cameras and then an audio synthesizer and then a microphone. It just really made this rich environment.

Kate: Let's close this up by having you describe some of the other projects that are going on in your group. I know that you have quite a large group and there are a number of projects going on that maybe you could say a little about.

Ted: Yeah. Okay. Well, let's see. There's this theme in the agent work based on COACH that continues in WBI, a system that Rob Barrett really is responsible for, where we put an agent handling the stream of information going through the browser through to http and back again so that it can record interesting things about the web and about the person. It's a fascinating project where we are really able to actually change and improve, augment, annotate the user interface of the web based on things that we learn about you and your needs. That's a very exciting project.

Another project, this COACH project, for the last few years, especially one application that Ron Barber is responsible for, is essential for anyone to write this new kind of help; it required an authoring tool that would make it a wysiwyg-simple process. That was kind of interesting. Some visualization work that we've done as a follow-up to my visual representation and epistomology of presentation reading group that I ran for years is with Shumin Zhai working on visualization for system management and also for information presentation.

We've been doing some research showing that masking, putting a shroud over user interfaces, will get people's attention very well. That 3-dimensional user interfaces can actually outperform 2-dimensional interfaces. We've got data that shows that. We are very excited about that. Also, we have been working on alternative pointing devices. TrackPoints have been

used in everything from surgical instruments to 3-dimensional pointing devices and air traffic control devices and remote-control devices. Tom Zimmerman is working on a new sensor that will be cheap. Barton Smith has been working on some of these things with me.

Now, also, in the hardware area, Tom Zimmerman, the inventor of the Glove, is now working very hard on communication that will run on the surface of your skin, called personal area network. We are also always working on industrial design to make tangible affordances that really are improvements. We have been working on a virtual office, a virtual-reality office that is actually a physical office with a bunch of tablets. You point at things and those tablets magically become those things. That's been a fun scenario to explore.

Lots of wearable scenarios—ones for the school, ones for business, ones for vertical markets. We've been playing around with those for a couple of years trying to figure out what value wearable computers will have. I am really enjoying that exploration. Finally, we have been kind of thinking about broader goals of industrial design and other techniques for really reducing the amount of hassle it is to install and upgrade and maintain your computer. Some of that stuff has actually found its way into the ThinkPad and other places as well.

Kate: Do you care about whether or not things find their way into product?

Ted: Oh, I really wouldn't have come to IBM if I wasn't excited about the possibility of having an idea and having it have a legacy that's not just paper. That's really something that I live for is trying to get those things to happen. Nowadays, now that I am a fellow, I also am really enjoying it. I always have enjoyed this process of trying to help other people. There are some ways in which I have been quite involved in the formation of some user-interface strategy across the company and also other people that are doing user-interface projects other places.

I tend to, just because my spirit is in the spatial visual physical thing and brain model thing, not be as involved in anything that's linguistic in nature. It's just not my skills. I know it's something you care about deeply. I understand

Tactile TrackPoint—
Touch Feedback

The Tactile TrackPoint has an actuator beneath the TrackPoint device that can stimulate the user's finger. It actually bumps up against your finger to give you tactile feedback which indicates the meaning of the cursor location.

With Tactile TrackPoint, you can "feel" icons, title bars, menu items, and window borders. These responses allow more accurate movement and selection by complementing the visual feedback with a sense of touch.

its value. Maybe I'd be presumptuous to try to be competent in that area.

Kate: Prognosis for the future. One of the things people talk a lot about is ubiquitous computing. Given the range of projects that you just described, one might infer that you might have something to say about it.

Ted: Well, in fact, there's a guy that's just joined my group that I am very excited about working with. It's going to further some of my relationships with the sociologists at Stanford. Chris Dryer is a sociologist, and we are actively pursuing thinking about how do you use technology in social situations. As we move away from the desk as being the place that we do all of our work, we are more and more involved in more standard social situations that aren't about a table and something that somebody is focusing on. How can we use computing in those situations?

My view is that if we really want to have nonabsorbent interfaces, in other words, if you glance at your watch it absorbs some of your energy, everyone knows you might be bored. It might be time for you to go or one thing or another, but really you don't want to be having interfaces in those social situations that would take 30 seconds to interact with when you want to put in an address. It would take up both of your hands so that somebody could pick-pocket you! What are those issues? We are really kind of excited. Actually, at the moment,

we are going to start doing some Wizard of Oz experiments about how the social and technological can support each other.

Kate: Terrific. Thank you.