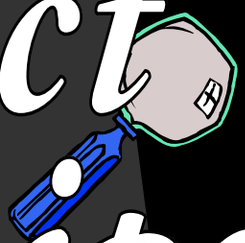
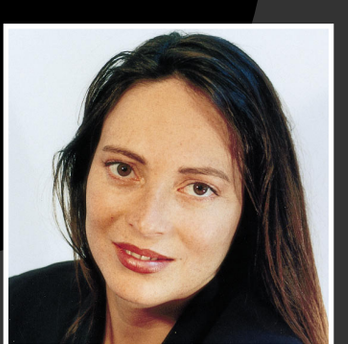
BEN SHNEIDERMAN

Direct Manipulation



Ben Shneiderman is a long-time proponent of direct manipulation for user interfaces. Direct manipulation affords the user control and predictability in their interfaces. Pattie Maes believes direct manipulation will have to give way to some form of delegation—namely software agents. Should users give up complete control of their interaction with interfaces? Will users want to risk depending on "agents" that learn their likes and dislikes and act on a user's behalf? Ben and Pattie debated these issues and more at both IUI 97 (Intelligent User Interfaces conference - January 6–9, 1997) and again at CHI 97 in Atlanta (March 22–27, 1997). Read on and decide for yourself where the future of interfaces should be headed—and why.

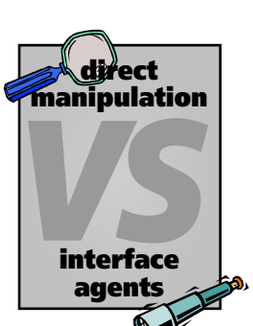
...s vs Interface Agents



PATTIE MAES

Excerpts from debates at IUI 97 and CHI 97

I think we would do best to focus on the remarkable human capabilities in the visual domain, which I think are largely underutilized by the current designs with 40 icons in 2-3 windows. I think we should have two or three orders of magnitude more: 4,000 or more items on the screen in an orderly way that enables people to see all of the possibilities and navigate among them.



Jim: Okay, welcome all to the afternoon session. My name is Jim Alty. I think I am supposed to be the moderator, whatever that means, for this session. I hope things don't get too rough. The debate topic this afternoon, as you all know, is direct manipulation versus intelligent agents. On my right I've got Ben Shneiderman—fresh from his triumphs this morning when he's been gnawing away at the agent people—from the University of Maryland. On my left, of course, fresh for a fight is Pattie Maes from the MIT Media Laboratory.

Let me just explain. It's 15 minutes from each speaker to place their position, and then 5 minutes allowed each for rebuttal. Then we open it up to the floor for about 30 minutes or so. Could you please use the microphones? If you want to line up behind the microphones, I will select you to make your comments. Then at the end, there will be two 5-minute summing ups. So, let us commence the debate.

Ben: First, my thanks to the organizers of this Intelligent User Interfaces workshop for daring to arrange this debate. It was Angel Puerta's careful attention to my concerns that made me feel comfortable in coming to speak here, and so I want to offer him a souvenir from our lab—the usual T-shirt. And to Jim Alty, a cup of tea for when he needs to relax, from our group at the lab.

I am pleased to represent the University of Maryland's Human-Computer Interaction Lab, which is now 14 years young. Over the years we've explored a host of user-interface design issues in an interdisciplinary way involving computer science, psychology, and the library school. Our goal is to create environments where users comprehend the display, where they feel in control, where the system is predictable, and where they are willing to take responsibility for their actions. To me, responsibility will be the central issues in this debate.

My pitch over 20 years has been to make a science out of user interface research. I want to get past the notion of "user-friendly," a vague, and misleading term, and to be really clear about specifying who the users are and what their tasks are. Then we can make a theory that allows us to predict the time it takes for a specific user to learn a specific task, the speed of performance on bench-mark tasks, the rate and distribution of errors, and the retention of those tasks over time. We look at subjective satisfaction as a secondary measure and have developed a standardized Questionnaire of User Interaction Satisfaction (QUIS) that the university has licensed to more than 100 organizations around the world. QUIS consists of 71 items, from low-level details to higher level concepts in the interface (Office of Technology Liaison, +1-301-405-4210).

We think accommodating individual differences is important—not just mentioning experts and novices, but understanding quantitatively what performance differences we anticipate. Do we expect experts to perform twice as fast or twenty times as fast as novices? Would men perform differently from women in the use of interfaces, or prefer different interfaces? And then we try to deal with broader cultural issues that are even more difficult to measure.

For me, the future is most clearly moving in the direction of "information visualization." I think we would do best to focus on the remarkable human capabilities in the visual domain, which I think are largely underutilized by the current designs with 40 icons in 2-3 windows. I think we should have two or three orders of magnitude more: 4,000 or more items on the screen in an orderly way that enables people to see all of the possibilities and navigate among them.

I will show you three brief videotaped examples. They all show applications and extensions of the strategy of "direct manipulation," a term I coined in 1982 to describe the existing successful systems. These systems all had rapid, incremental, and reversible actions, selection by pointing, and immediate feedback (100-millisecond updates for all actions). I believe that this strategy reduces errors and encourages exploration. The current manifestations of direct manipulation are the visual ways of searching in databases and on the Web accompanied by visual presentation of results.

Let's take a look at an example that goes back 4 years in the first videotape called the *FilmFinder.*

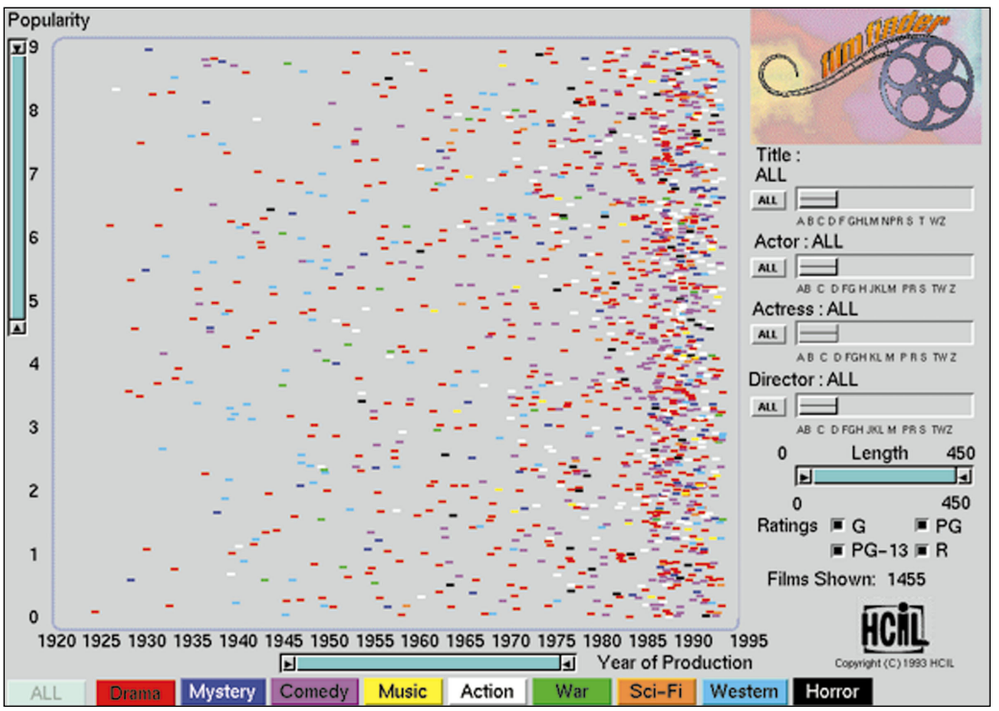


Figure 1(a): FilmFinder shows 1500 films in a starfield display where the location of each point is determined by the year of the film (x-axis) and its popularity in video store rentals (y-axis). The color encodes the film type (Ahlberg & Shneiderman, 1994). *ftp://www.cs.umd.edu/projects/hcil/Screen-dumps/Film/film-alldots.gif*

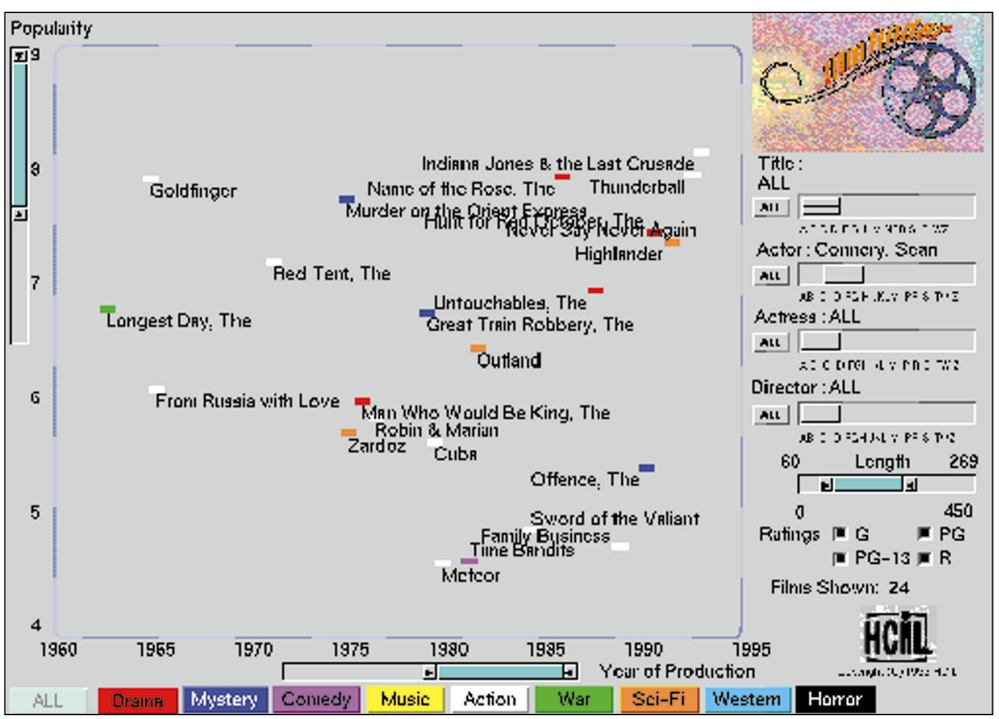


Figure 1(b): FilmFinder after zooming in on recent popular films. When fewer than 25 films remain, the titles appear automatically. *ftp://www.cs.umd.edu/projects/hcil/Screen-dumps/Film/titles.gif*

FilmFinder video (see Figures 1a-c, CHI94 videotape or HCIL 1995 Video tape reports): This display shows thousands of films arranged on the x-axis from 1920 to 1993 and on the y-axis from low to high popularity. We can use the length slider to trim the set of films by the number of minutes in the film so we do not have to see films that are too long, and then we can use the category button to show only drama or only action films. We can zoom in on more recent pictures and take only the more popular ones. And when there are fewer than 25, the titles will appear automatically. When we select one of them, we get a description of the film and information about the actress and actor. We can also hunt

Users have great control over the display and as they select items, the details appear in windows on the sides.

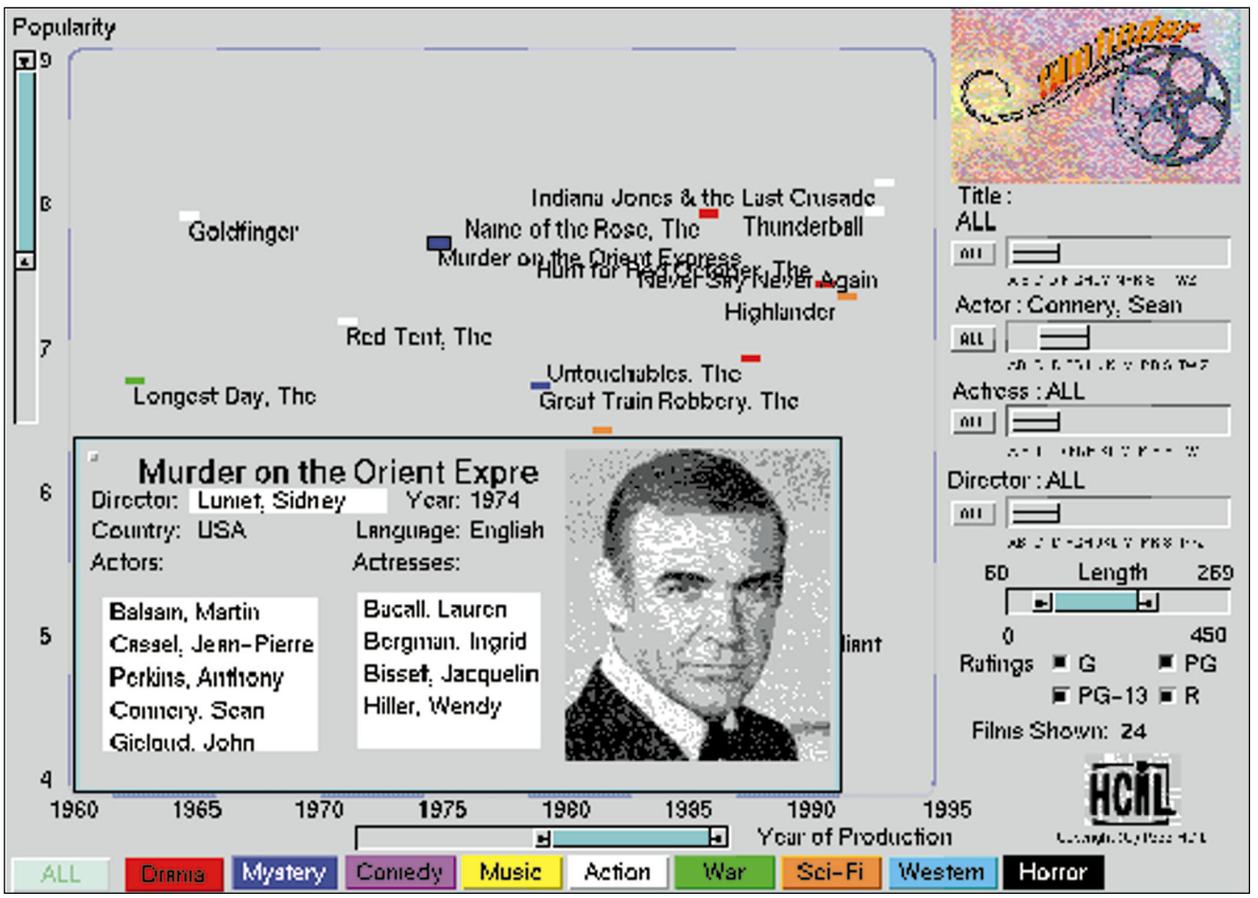
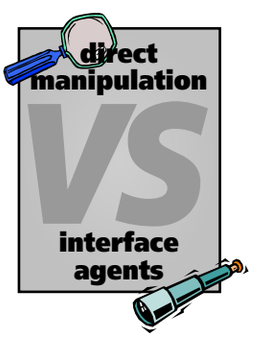


Figure 1(c): FilmFinder after selecting a single film. The info card pops up with details-on-demand.
ftp://www.cs.umd.edu/projects/hcil/Screen-dumps/Film/film-sean.gif

for films organized by actors. In this case, you might be interested in Sean Connery, and his films will appear on the screen.

Ben: Okay. I think you get an idea that the controls are visually apparent as you drag them. The updates occur in 100 milliseconds and users get a clear understanding of what the situation is. This work goes back to 1993, and the 1994 CHI conference has a couple of papers describing it [1, 2]. A general purpose version of that software was used for the Department of Juvenile Justice project, which you will hear about shortly.

Here is a *FilmFinder* done in the UNIX version of the product called SpotFire (Figure 2). Chris Ahlberg has made a commercial product out of this and you can download the demo version off of the Web (http://www.ivee.com).



It would be hard to see how to program any kind of agent tool to anticipate all of the possibilities that your eye would pick up. We show the age of the youthful offenders on the x-axis. There are 4,700 of them, from 10 to 19 years old. The number of days until a decision was made on their treatment plan is shown on the y-axis. The rules of this organization say that decisions must be made within 25 days, but you can see a lot more than they anticipated are well above the 25-day limit.

Interesting things pop up whenever you try a visualization. I hope you will spot these yellow lines—those were a surprise. We thought there was a bug in the program, but it turns out that if you start clicking on them to get the details-on-demand, you'll find out they all occur in a Hartford County. They were all brought in on a certain day. They all have the

same charge, which is narcotics possession. These were drug busts and they were all put on treatment plans at the same time. Those kind of patterns happen to anyone who tries visualization programs, and it would be hard to see how you could program an agent to anticipate all of the possibilities that your eye can pick up in 1/10th of a second.

Another problem we have dealt with in the second videotape has to do with visualizing personal histories. It is called *LifeLines* (Figure 3).

The placement line shows this youth is currently placed at Waxter, a detention center. We also see a placement in a drug-abuse program and a placement at Cheltenham. This one is thicker because the court found him guilty of auto theft. When I click on the line of the drug-abuse program, all of the related information is highlighted. I can tell that the placement was for a breaking and entering

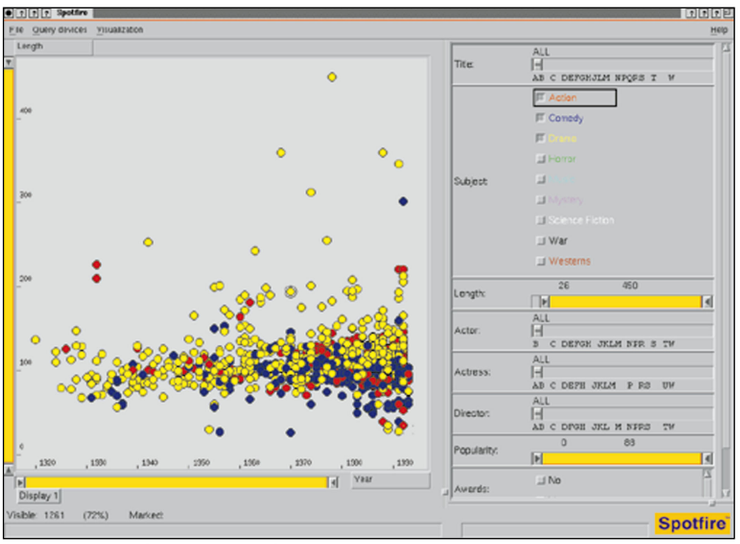


Figure 2: Spotfire version of FilmFinder provides increased user controls. Users can set axes (set to length in minues and year) and glyph attributes (color is set to subject and larger size indicates award winning film). Spotfire is available in Unix, Windows and Java versions *(http://www.ivee.com)*

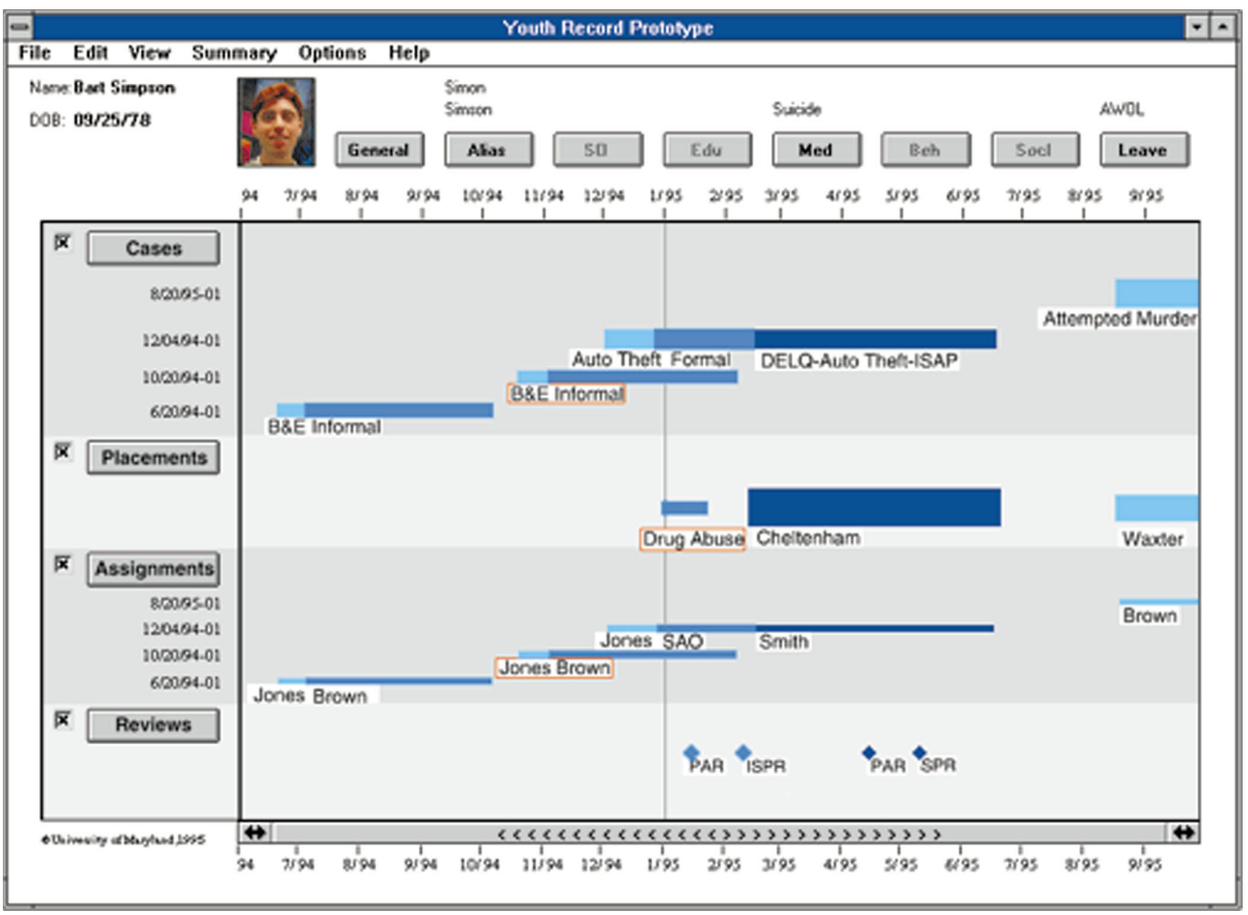


Figure 3: Youth Record prototype using the Lifelines display to show a case history for the Maryland Department of Juvenile Justice. (CHI96 videotape or HCIL 1996 Video Tape Reports; 4) *ftp://www.cs.umd.edu/projects/hcil/Research/1997/patientrecord.html*

Ben:
During the CHI97 debate, Pattie made a point about my use of autofocus cameras, suggesting that they were some sort of agent. As we were speaking Alan Wexelblat was taking pictures of the events using my camera. When I went through the color slides I found that most of the pictures he took were out of focus! As with most autofocus cameras, there is a narrow area in the center that is used for focusing and this must be placed on the intended subjects of the photo. Unfortunately Alan didn't know this and simply pointed at the stage area but the focus point was on the background poster. Almost all the photos were unusable.

We can all find this amusing and leave it at that, but I think there is a serious point which is that agents don't always do what we expect them to do, and it takes some knowledge to make effective use of agents (or auto-focus cameras). If we were to assess responsibility — I would take part of it in failing to give adequate instruction to Alan, he might take part because he was the direct user, and maybe the manufacturer has another part for a poor design which fails to provide appropriate feedback.

case and was requested by Brown. A click on Brown's name gives the contact information. Here are the reviews which were written. A click on the code brings the text of the review. In the same way, I can get more details about each case and placement by clicking on the labels. Those screens are all for the old system, showing that *LifeLines* can be used as a front end and acts as a large menu for all of the screens. The top buttons can access general information, but any critical information will appear on the overview screen. For example, here's a mention of suicide risk. Seeing the overview gives the user a better sense of how much information is available and what type of information it is. Of course, this implies that all of the information can be presented in one screen.

Ben: We think that *LifeLines* can also be applied for medical records, and we are now applying it in a project with IBM. We give an overview of patient histories that contains the consultations, conditions, documents, hospitalizations, and medications. Users have great control over the display and as they select items, the details appear in windows on the sides. We think this strategy has great power in providing convenient access to large and complex databases in a way that gives the users an overview of what is happening and an appreciation of where the details fit into the overall context. The visual presentation gives users enormous bandwidth and there are potentially thousands of selectable items on the screen at once offering rapid access to the item that you are seeking.

The third and final example, is a visual database of the human body called the *Visible Human.* Our role was to develop a browsing user interface to the 15 gigabytes of images at the National Library of Medicine (3; CHI96 videotape and HCIL 1996 Video Reports; free software available for Sun workstations from http://www.nlm.nih.gov and select the *Visible Human* links till you find our *Visible Human Explorer*).

This direct manipulation interface presents a thumbnail image of a coronal cross section of the body reconstructed directly from the axial cryosections (Figure 4). This coronal image acts as an overview, giving a visual representation of the entire body. The axial cryosections are a local view showing a thumbnail corresponding to the slider position on the overview. We can explore the body by simply dragging the slider. It updates in real time, giving an experience of flying through the human body. Here we see the brain, the shoulders, the torso, the abdomen, the thighs, the knees, and all of the way down to the toes. We press the retrieve button and the corresponding full-size image is retrieved over the network from the NLM archive in a couple of minutes. We provide several useful alternative overviews and also the ability to generate any coronal section overview, for example, near the back of the body or near the front.

Ben: Other labs are working on related ideas of information visualization. From the Pacific Northwest National Labs, this textual database that has been presented in a two-dimensional mountains-and-clusters visualization to give users an idea of the volume and interrelationship of items. Steve Eick at AT&T Labs has these wonderful visual overviews of large textual documents.

Here, the characters in a children's book are color coded so that you can see the progress of the story as it moves on to different characters. Departmental e-mail networks and richer information, such as 3D network representations, are part of the things he likes to show with a variety of user controls to filter the traffic and reveal patterns of usage that might be difficult to see with other data presentation strategies.

The closing slide says that the overview is the most important. It gives users a sense of context, of what to look at—the big picture. Then they zoom in on what they want, filter out what they don't want, and finally go for details-on-demand. My claim is that this gives users the feeling of being in control and therefore they can be responsible for the decisions they make. Thank you.

Jim: Okay, thanks very much, Ben. That was perfect timing. We now hand it over to Pattie.

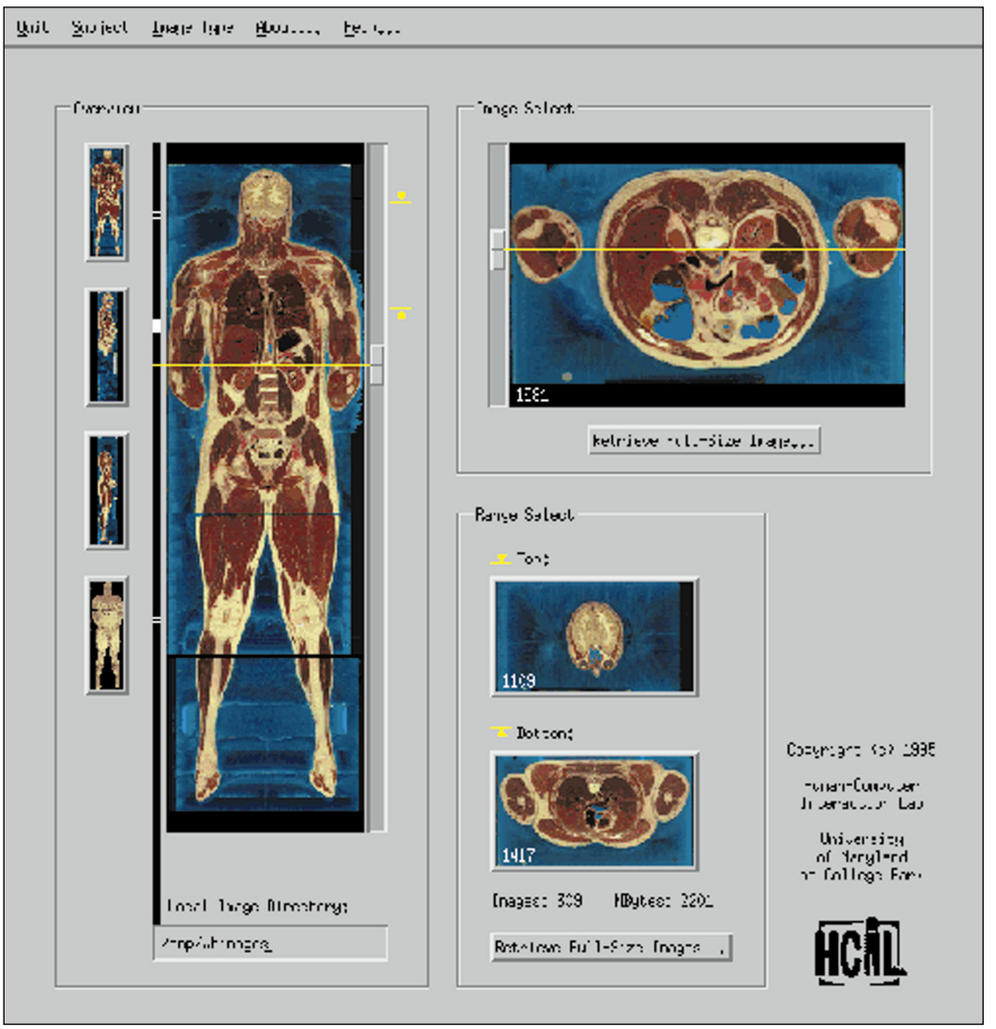
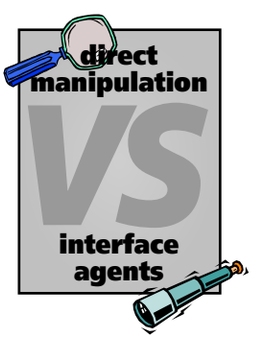


Figure 4: Visible Human Explorer user interface, showing a reconstructed coronal section overview (on the left) and an axial preview image of the upper abdominal region (on the upper right). Dragging the sliders animates the cross-sections through the body (North et al., 1996).
ftp://www.cs.umd.edu/projects/hcil/Research/1995/visible-human.html

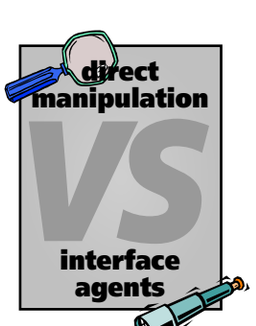
Pattie: I'm not going to bribe the moderator with tea or T-shirts or anything. I hope that the work will speak for itself. The word "agent" is used in a lot of different ways. I want to start this presentation by explaining what I mean by the word "agent," and in a particular, "software agent." Basically, software agents are a new approach to user software, a new way of thinking about software that end-users have to use. In particular, the way in which agents differ from the software that we use today is that a software agent is personalized.

A software agent knows the individual user's habits, preferences, and interests. Second, a software agent is proactive. It can take initiative because it knows what your interests are. It can, for example, tell you about something that you may want to know about based on the fact that you have particular interests. Current software, again, is not at all proactive. It doesn't take any initiative. All of the initiative has to come from the user. A third difference with current software is that software agents are more long-lived. They keep running, and they can run autonomously while the user goes about and does other things. Finally, software agents are adaptive in that they track the user's interests as they change over time.

So, you can ask, well, why do you call it an agent? Why call it an agent given that the term



Why do we need software agents? ... Take a look at the World Wide Web, for example. You couldn't possibly try to visualize the World Wide Web in any way because it is completely unstructured and because it has been built by so many different people and is continuously changing. I believe that the dominant metaphor that we have today is a mismatch for the computer environment we are dealing with tomorrow.



is already so overloaded, and given the fact that it's really software that is slightly different from existing software? Well, we call it an agent to emphasize the fact that agent software can act on your behalf while you are doing other things. We also want to emphasize that it does this based on its knowledge of your preferences, just like a travel agent will act on your behalf by buying you a travel ticket based on the information that the travel agent has about your preferences. Note that I prefer not to use the term "intelligent agents" nor "autonomous agents" because those terms have even more problems associated with them.

Now, why do we need software agents? Why does our software need to become more personalized? Why does our software need to take initiative to help us as a user? This needs to happen because our current computer environment is getting more and more complex, and the users are becoming more and more naive. Finally, the number of tasks to take care of, and the number of issues to keep track of, are continuously increasing. Let me tell you more about this. First of all, the nature of our computer environment is radically different today from 20 years ago, back when the current style of computer interaction was invented. Twenty years ago, one typically had one user using one computer, and everything in that computer, every file, every object, was in a particular place because the user put it there. There was a limit to the amount of information on that computer. It was completely static. Nothing changed unless the user made it change. It was completely structured and well organized. Today, our computer environments are completely different.

More and more the World Wide Web and our browser is becoming the one and only interface. It's not quite the case yet today, but it will be a year from now. In that situation, our computer is no longer this closed environment that we have complete control over. Instead, our computer or the screen is a window onto this vast network, this vast network of information and other people. That network is continuously changing. It is dynamic. Something may be in one place today, and the next day it may be in another place or may be gone. Continuously, new information is being created, new versions of software are being added. Also, that environment is completely unstructured. Take a look at the World Wide Web, for example. You couldn't possibly try to visualize the World Wide Web in any way because it is completely unstructured and because it has been built by so many different people and is continuously changing. I believe that the dominant metaphor that we have today is a mismatch for the computer environment we are dealing with tomorrow.

Second, the user 20 years ago was different from the typical user today. Twenty years ago we mostly dealt with professional users of computers. Today and tomorrow the consumer electronics market is going to be the one that dominates, and those users do not even know how to program their VCRs. How are they going to deal with user interfaces?

Third, the number of things that people have to keep track of and the number of tasks that they use their computers for is huge and is increasing all of the time. As we know from other domains, whenever workload or information load gets too high, there is a point where a person has to delegate. There is no other solution than to delegate. For example, many of you may have students that you delegate certain tasks to, or you may have personal assistants that you delegate certain tasks to, not because you can't deal with those tasks yourself, but because you are overloaded with work and information. I think the same will happen with our computer environments: that they become just so complex and we use them for so many different things that we need to be able to delegate.

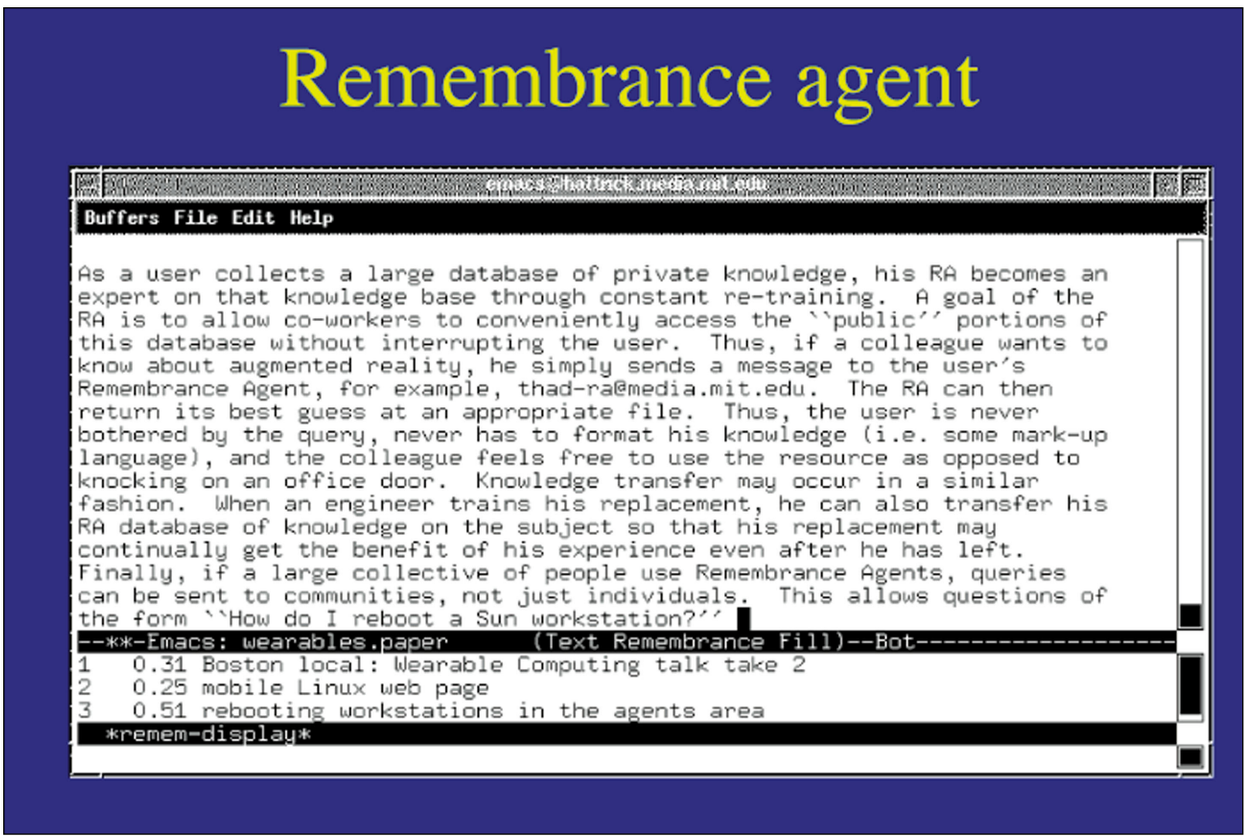
We need to be able to delegate to what could be thought of as "extra eyes or extra ears" that are on the lookout for things that you may be interested in. We also need "extra hands or extra brains," so to speak, because there will be tasks that we just cannot deal with because of our limited attention span or limited time, and we need other entities to be able to represent us and act on our behalf. Some examples to make this more concrete (I didn't bring any videos because of the limited amount of time), but most of you have seen at least one of these agents. These are some of

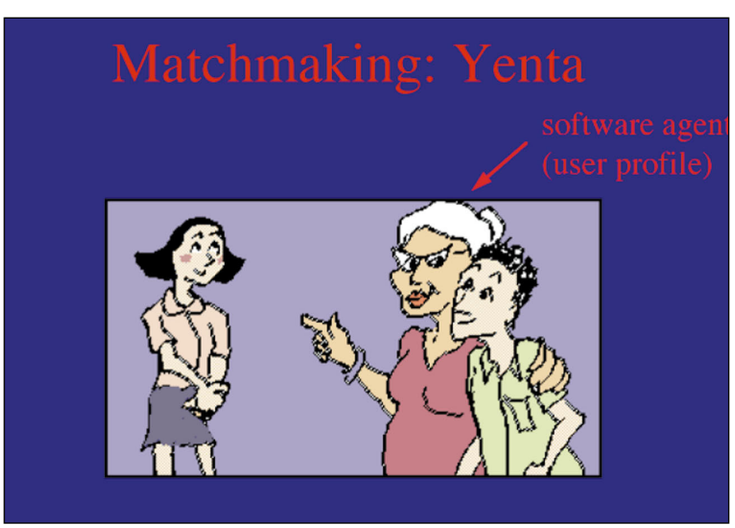
the ones we built in our lab. *Letizia,* built by Henry Lieberman, who is here at the conference, is an agent which continuously watches you as you browse the Web, analyzing and memorizing all of the pages that you visit. It extracts from those pages the common keywords.

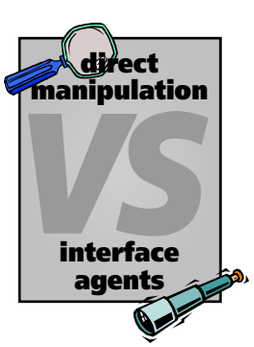
Whenever you are using your Web browser, *Letizia* always looks ahead and checks whether within a certain depth of the current page, there happen to be any pages that you may be interested in. So, for example, if I am interested in scuba diving, my agent may have picked it up because I look at a lot of pages about scuba diving. If I go to a particular entertainment site, it may look ahead and say, hey did you realize that if you follow that link that there are some pages about scuba diving in the Florida area? The *Remembrance Agent* is another agent that continuously tracks the behavior of the user. It helps you remember certain things. It helps you remember who sent e-mail or whether you already replied to a certain e-mail message. It may proactively remind you of information related to the information you are currently looking at. It works in EMACS. When I am, for example, looking at an e-mail message from a particular person, it proactively reminds me of the previous e-mail messages from that same person, which is very useful because I may have forgotten to reply to one of them.

Firefly, some of you may have tried that agent, is basically a personal filterer for entertainment, not unlike the movie application that Ben talked about, except that this agent will again keep track of your interests, your preferences, and proactively tell you about new movies that you may be interested in which you even forgot to ask about in the first place. *Yenta* is another agent that we built which tracks what the user's interests are by looking at your e-mail and files and extracting keywords. It talks to other *Yenta* agents belonging

We need to be able to delegate to what could be thought of as "extra eyes or extra ears" that are on the lookout for things that you may be interested in.







to other users, and if it notices that another user shares some of your interests, especially if those interests are very rare, then it introduces you to that other user. It may say "hey did you realize that at this IUI conference there is another person who is interested in going scuba diving in Florida" so that maybe then we can decide to go scuba diving together. Again, it's suggesting something that you wouldn't have thought of yourself. *Kasbah* is another set of agents that buy and sell on behalf of users. We are currently setting up this experiment MIT-wide, meaning for 15,000 people. We have already done tests with 200 people. It's basically a marketplace where you can create an agent who will buy or sell a second hand book or a second hand music CD for you. You just tell the agent, "I want to sell *The Joshua Tree* by U2. I want to ask $12 for it at first. You are allowed to go as low as $9. You have two months to sell this CD. You should be really tough and only change the price all the way at the end, near when the 2 months are over." That agent will represent you in that marketplace, negotiating on your behalf with other people or other agents who may be interested in buying that CD from you. Again, it is sort of acting on your behalf. You don't have to waste any time trying to make 10 bucks, but the agent will do this for you.

I think it's important to address some common misconceptions about agents: First of all, sorry to say so, but agents are not an alterna-

tive for direct manipulation. A lot of conferences and magazines pitch agents against direct manipulation. They are actually complementary metaphors. Whenever you have an agent interface, typically you also need a very good application interface because an agent is not a substitute for an interface. An agent basically interacts with the application just like you interact with the application. It's as if you had someone else looking over your shoulder as you are using the application, noticing some of your preferences and habits and then offering to automate some of the tasks for you. So you still need a very good direct manipulation interface—visualization—all of these wonderful tools so that the user can personally interact with the application. An agent can never predict all of the movies that I may possibly be interested in. It may be able to make some interesting suggestions to me, but I will still need to look up particular movies myself.

A second misconception is that some people think that agents are necessarily personified or anthropomorphized. In fact, most agents are not. Most of them don't even deal with a natural language interaction interface.

A third misconception is that agents necessarily rely on traditional AI (artificial intelligence) techniques, like knowledge representation and inferencing. In fact, most of the agents that are commercially available and have proven successful with large numbers of users rely on either user programming or on machine learning rather than traditional AI techniques.

I want to conclude by addressing some criticisms of software agents that Ben has come up with—as well as people like Jaron Lanier (see "A Conversation with Jaron Lanier," *interactions* ii.3 (July 1995), pp. 46-65), who is also very vocal about all of this. Opponents of agents typically argue that well-designed visualization interfaces are better. Like I said before, you still need a well-designed interface when incorporating agents in an application. However, some tasks I may just not want to do myself even if the interface was perfect. If my car had a perfect interface for fixing the engine, I still wouldn't fix it. I just don't want to bother with fixing cars. I want someone else to do it.

Pattie Maes is an Associate Professor at MIT's Media Laboratory, where she founded and directs the Software Agents Group. She currently holds the Sony Corporation Career Development Chair. Previously, she was a visiting Professor and a Research Scientist at the MIT Artificial Intelligence Laboratory. She holds a Bachelor's degree and Ph.D. degree in Computer Science from the Vrije Universiteit Brussel in Belgium. Her areas of expertise are Human Computer Interaction, Electronic Publishing and Electronic Commerce, Artificial Intelligence, and Artificial Life.

Pattie is one of the pioneers of a new research area called Software Agents, that is, semi-intelligent computer programs which assist a user with the overload of information and the complexity of the online world. She is one of the organizers for the leading conferences in this area such as the annual Autonomous Agents conference and the annual Practical Applications of Agents and Multi-Agent Systems conference. She is a founder and board member of the Agent Society, an international industry and professional organization established to assist in the widespread development and emergence of intelligent agent technologies and markets.

Pattie is a consultant in the area of Software Agents for several major companies such as Apple Computer, Hughes Research, etc. She is the editor of three books and is an editorial board member and reviewer for numerous professional journals and conferences, such as the User Modeling journal, the Personal Computing journal and the Artificial Life journal. Her work has achieved several prizes, including the IBM best Bachelor's thesis award (1984), the OOPSLA-1987 best paper award, the Ars Electronica 1995 award, the Interactive Media Festival Award 1995, the ArcTec 1995 award and so on.

Finally, she is a founder of Firefly Network, Inc. in Boston, Massachusetts—one of the first companies to commercialize software agent technology.

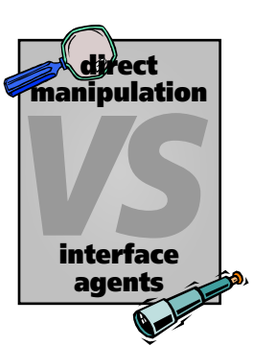
Opponents of agents typically argue that well-designed visualization interfaces are better. Like I said before, you still need a well-designed interface when incorporating agents in an application. However, some tasks I may just not want to do myself even if the interface was perfect. If my car had a perfect interface for fixing the engine, I still wouldn't fix it. I just don't want to bother with fixing cars. I want someone else to do it.

A second criticism is that agents make the user dumb. That's actually more one of Jaron's objections rather than Ben's, I think. To some extent it's true. If I don't fix my car then I'm not going to learn about fixing cars. However, this does not constitute a problem. As long as there's always an agent available or I can call one by a motor association like AAA, then that's fine. It's too bad that I will never learn about cars, but I want to learn about other things instead.

A third criticism expressed is that using agents implies giving up all control. That's incorrect. I think you do give up some control when you deal with an agent. I tell the car mechanic to fix my car or to fix this or that part of the car. I don't know how exactly he or she is going to do that. I don't mind giving up some control, actually, and giving up control over the details as long as the job is done in a more-or-less satisfactory way, and it saves me a lot of time.

Okay, just very briefly, I want to say that I think where the true challenge lies is in designing the right user-agent interface. In particular, we need to take care of these two issues: understanding and control. Understanding means that the agent-user collaboration can only be successful if the user can understand and trust the agent, and control means that users must be able to turn over control of tasks to agents but users must never feel out of control. I believe that this is a wonderful interface-design challenge, and we have come up with a lot of solutions to actually make sure that the agent's user interface has these two properties that the user feels in control or has control when he or she wants it, as well as that the user understands what the agent does and what its limitations are. Let me save that for later, maybe. Thanks.

Jim: Thanks very much. Okay, Ben's going to go up to 5 minutes to say whatever he likes.

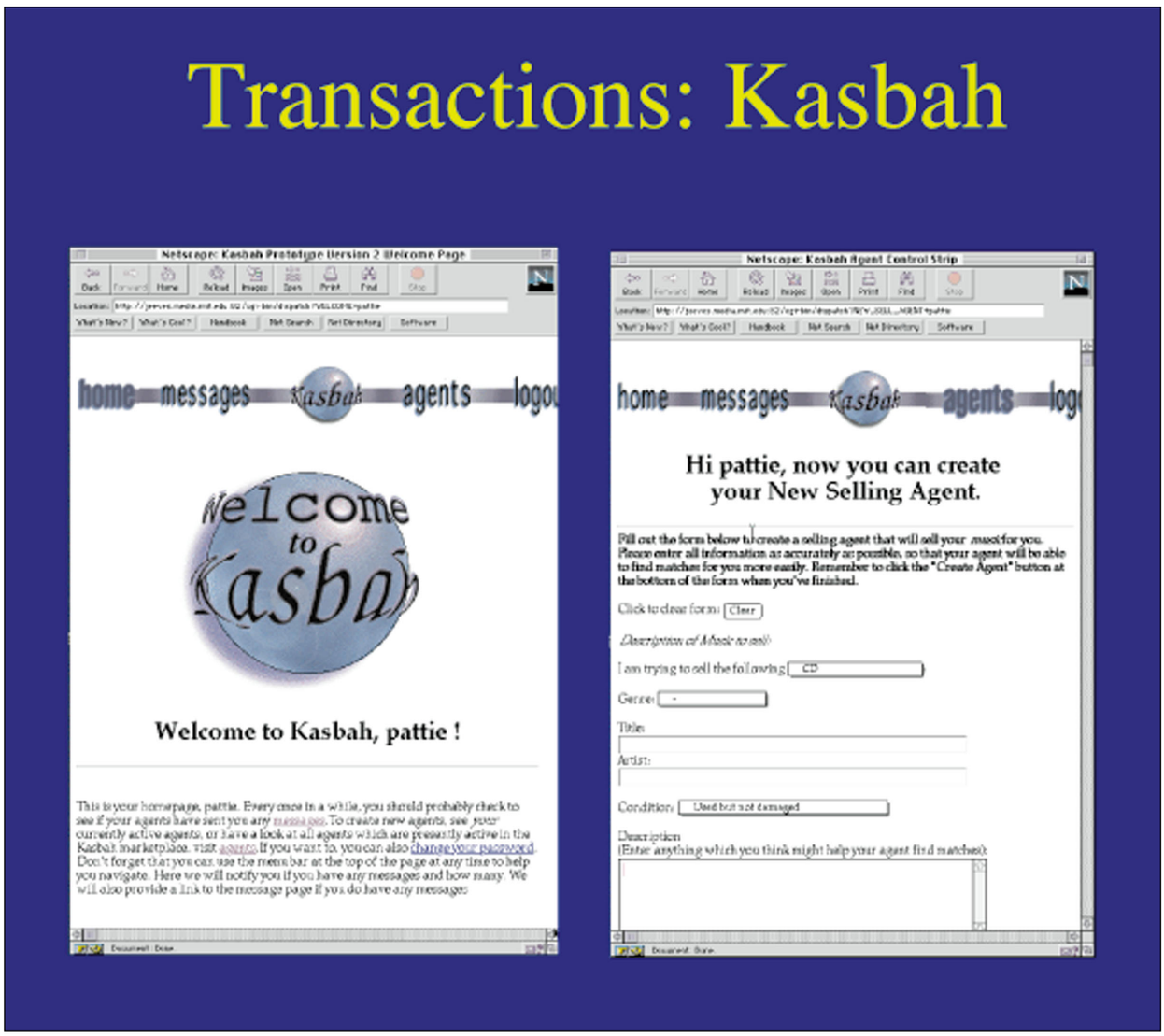


Ben: How interesting. We are debating, but part of me is drawn to the idea of celebrating Pattie Maes and encouraging you to follow her example. I want to draw the audience's attention to her transformation during the months we've had these discussions. As I go back to Pattie Maes's work and I read her earlier papers and her Web sites, she promotes autonomous agents and presents an anthropomorphic vision. Even in the current proceedings her article is titled "Intelligent Software," so I was delighted with her opening remarks that rejected intelligent and anthropomorphic designs. The old Pattie Maes wrote "agents will appear as living entities on the screen, conveying their current state of behavior with animated facial expression or body language rather than windows text, graphics, and figures." So we've got two Pattie Maes. I will choose the newer one that demonstrates movement in my direction including her last slide which might have been written by me: "User understanding is central, and user control is vital for people to be successful."

In fact, I have other ways of celebrating Pattie Maes. I encourage you to look at her Firefly Web site, which is an interesting application. Collaborative filtering, I think, will become an important approach for many domains. But as a user, I can't find the agents on the Firefly Web site. In fact, as I searched to find the agents, all I came up with was that the company had previously been called Agents, Inc. and is now called Firefly. If you read the Firefly Web site, you will not find the word "agents" in the description of this system. In fact, the interface is a quite lovely, direct manipulation environment, allowing users to make choices by clicking in a very direct manipulation way.

So, I think we've made progress in clarifying the issues in the past year of our ongoing discussions. For example I think we can separate out the issue of natural language interaction, which as far as I can see, has not been a success. The systems that were offered commercially even a few years ago, like Q&A from Symantec or Intellect from AI Corporation to do database query, and Lotus HAL for spreadsheets, are gone, and direct manipulation is the surviving technology.

A second issue is anthropomorphic interfaces such as chatty bank tellers and the Postal Buddy or the proposed Knowledge Navigator of Apple's 1988 video. Microsoft's playful attempt at a social interface in BOB is also a failed product. As far as I can see the anthropomorphic or social interface is not to be the future of computing.



A third issue of adaptive interfaces is quite interesting. I would concede half a point and say that we now see two levels: the user interface level, which users want to be predictable and controllable, as Pattie has stated, and the level below the table, where there may be some interesting algorithms such as collaborative filtering. If those can be adaptive there may be benefits in the way that Pattie describes. This is related to other adaptations such as when I save a file to disk, I see it saved, and it is retrievable by me. Under the table, there's a great deal of adaptation dealing with space allocation, disk fragmentation, and compression strategies, but from the user's point of view, there's no adaptation. It's quite predictable. The same goes for engines in

Kasbah example selling agent

Sell: Macintosh IIci

- Deadline: March 10th,1997
- Start price: $900.00
- Min. price: $700.00
- Strategy: tough bargainer
- Location: local
- Level of Autonomy: check before transaction
- Reporting Method: event driven

automobiles: I turn the key, I step on the gas, it goes. Underneath the hood, below the user's concern, there are adaptive algorithms that will set the engine speed based on many factors, such as the temperature, gas mixture, etc. That level of adaptivity is important as long as it does not interfere with the user's prediction of behavior.

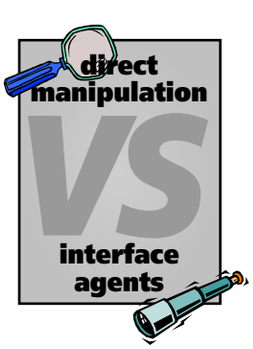
I am concerned about the anthropomorphic representation: it misleads the designers, it deceives the users; it increases anxiety about computer usage, interferes with predictability, reduces user control, and undermines users' responsibility—which I think is central. I think anthropomorphic representations destroy the users' sense of accomplishment; I think users want to have the feeling they did the job—not some magical agent.

I am concerned about the confusion of human and machine capabilities. I make the basic assertion that people are not machines and machines are not people. I do not think that human-to-human interaction is a good model for the design of user interfaces.

Finally I am concerned about the confusion of human and machine capabilities. I make the basic assertion that people are not machines and machines are not people. I do not think that human-to-human interaction is a good model for the design of user interfaces.

Jim: Okay, Pattie, would you like to respond to that?

Pattie: First of all, I should clarify that autonomous agents or the word "agents" has a much broader meaning than the words "software agent," and my group at the Media Lab does research on autonomous agents more generally, as well as software agents. So when Ben was quoting from our Web site, he's actually quoting other work that we do, for example, work on synthetic characters that can interact with people in a virtual environment, which doesn't have anything to do with the software agent's work. In fact, it has less and less to do with it than it may have at one point. So, if you go to autonomous agent conferences, for example, the Agents Conference in Marina del Rey in February (First International Conference on Autonomous Agents, see *interactions* iii.6, Conference Preview), you'll see work being discussed that relates to robots, autonomous robots. You'll see work about synthetic believable characters, and you'll see work about software agents. So that's one thing I wanted to respond to. It's important to distinguish these different types of agents and not lump them all together.



Second, I absolutely agree with Ben that so far the most successful interfaces are the ones where the agents are pretty much invisible. They are not visualized as a little anthropomorphic character. For example, in Firefly, that is the case. That doesn't mean that there isn't an agent operating. For example, in Firefly, the Firefly agent will proactively tell you about other users that you may want to talk to. It will warn you when there is something that has changed somewhere that you may be interested in. There is still an agent there monitoring all of your preferences and proactively making recommendations to you, but that doesn't mean that there has to be this little cute character on the screen.

Now, I think one of the reasons that Ben and I disagree is actually that we are focusing on completely different problem domains. In pretty much all of the problem domains that Ben looks at we are dealing with a user who is a professional user, and we are dealing with a task domain that is very well structured and an information domain that is very well organized, so that it lends itself to visualizing all of the different dimensions. The kind of problems that we have typically been dealing with are very different because we are dealing with end users who are not necessarily very trained. They may use the Web for a couple of hours per week, but that is about it. We are dealing with a very different information domain, an information domain that may be very ill structured and very dynamic. For example, the World Wide Web is actually sort of one of the key domains that we do all of our research on.

Finally, to illustrate that these approaches aren't necessarily incompatible, I could envision a version of Ben's movie finder which uses Ben's nice visualization interface, where an agent is continuously monitoring what movies you seem to be interested in. That agent may, for example, highlight particular areas in the interface which it thinks you will be specifically interested in. That kind of interface would actually combine an agent that learns about your preferences and proactively makes suggestions to you, with a nice visualization interface. The reason why you want that kind

of proactive software is that the user does not necessarily always want to have all of that control when searching for a movie. I believe that users sometimes want to be couch-potatoes and wait for an agent to suggest a movie to them to look at, rather than using 4,000 sliders, or however many it is, to come up with a movie that they may want to see.

•

Question: *Okay, I have a remark to Ben, which then transitions into a question for both speakers. Ben, I was a little bit irritated, I think Pattie was too, with your lovely presentation on information visualization, which seems to be entirely beside the point. As Pattie said, we will take the best direct manipulation and visualization as we can possibly get. It seems to me the contribution to this discussion would be some negative examples of where agentlike things are bad. At some level, I take the thrust of your position, Ben, as being reactionary, to put it in simpler words, sort of fear-driven. I would like to test my theory by asking the following question, which is, we have now a new medium on the interface of playing which is speech. Speech is now practical. IBM makes speech systems that are being used and so on.*

My question actually to both of you is to see what your reactions are to this new technology. I predict that if you are going to have speech with a computer, first of all, research at Stanford recently has shown that once you have a computer talking you cannot prevent people from anthropomorphizing the computer. I do not see how you are going to have a coherent speech interface without using human communication principles. So I predict that you will say just don't do it. I also want to hear Pattie's comments about speech technology.

Ben: Do we have another half an hour here? I thought I said very positive things about Firefly and its agent and adaptation, and I certainly like to see automaticity built into interfaces that amplify the user's capabilities. I have trouble with the words like "agents," and "expert," and "smart," and "intelligent" because, they mislead the designer, and designers wind-up leaving out important things. In fact, I love Pattie's slide up here. If the agent-oriented community would adopt

Ben Shneiderman is a Professor in the Department of Computer Science, Head of the Human-Computer Interaction Laboratory, and Member of the Institute for Systems Research, all at the University of Maryland at College Park. He has taught previously at the State University of New York and at Indiana University. He regularly teaches popular short courses and organizes an annual satellite television presentation on User Interface Strategies seen by thousands of professionals since 1987. The third edition of his book *Designing the User Interface:tion* (1987) has recently been published by Addison-Wesley Longman.

Ben is the author of *Software Psychology: Human Factors in Computer and Information Systems* (1980) Addison-Wesley Longman, Reading, MA and his 1989 book, co-authored with Greg Kearsley, *Hypertext Hands-On!,* contains a hypertext version on two disks. He is the originator of the Hyperties hypermedia system, now produced by Cognetics Corp., Princeton Junction, NJ. In addition he has co-authored two textbooks, edited three technical books, published more than 190 technical papers and book chapters. His 1993 edited book *Sparks of Innovation in Human-Computer Interaction* collects 25 papers from the past 10 years of research at the University of Maryland.

Ben has been on the Editorial Advisory Boards of nine journals including the *ACM Transactions on Computer- Human Interaction* and the ACM *interactions.* He edits the Ablex Publishing Co. book series on "Human-Computer Interaction." He has consulted and lectured for many organizations including Apple, AT&T, Citicorp, GE, Honeywell, IBM, Intel, Library of Congress, NASA, NCR, and university research groups.

these principles it would go a long way in making me sympathetic. For example Pattie writes "Make the user model available to the user." I don't see that being done in most of the work about agents. Explanations should be available and methods of operation should be understandable to the user. So much of the work in agentry goes against the principles. I like the new Pattie. I am ready to be partners and collaborate with the new Pattie.

Now, to focus on speech. We have heard for 25 years the great hope and dream that speech is going to solve our user interface problems. Dreamers prophecy that the Star Trek scenario is going to take over and we will talk to our computers. I do not believe that speech will be a generally usable tool. It has important niches: opportunities for disabled users, for certain hands-busy, eyes-busy, and mobility-required applications. In preparing the third edition of my book I worked hard to find speech applications that do recognition effectively. I am quite happy with speech store-and-forward applications by telephone, but the recognition paradigm is not being widely accepted, even for minor tasks such as voice dialing. Speech output, except by telephone, is also a problem because speech is very slow and disruptive of cognitive processing. I think what annoys me the most about the devotees of speech, is their failure to take in the scientific evidence that speaking commands is cognitively more demanding than pointing. Speech uses your short-term memory and working memory. By contrast, hand-eye coordination can be conducted in parallel with problem solving by another part of your brain and therefore does not degrade your performance as much as speaking.

User interfaces should be predictable, so that users trust them. User interfaces should be thoroughly tested, and users should be thoroughly trained for all emergencies. In emergency situations, people cannot solve problems. They can only do what's rehearsed and predictable.

•

Question: *You can do both at the same time.*

Ben: Yes, you can do hand-eye tasks in parallel with problem solving, more easily that you can speak while problem solving. This fact is not a barrier to use of speech, but it is a hurdle that designers of speech systems must recognize if they are to find ways to overcome it.
Pattie: I must admit, I actually agree with a lot of what Ben says. I haven't used speech at all in my research, the main reason being completely personal—that these systems often don't understand my accent, but apart from that, I do agree that it is not a very high bandwidth kind of connection. There is also a lot of ambiguity.

So I personally would like to see speech being used just for situations where the hands are not available, like in your car or as an additional channel, actually, for example, giving an agent some additional advice while you are also pointing at something. For example, Henry Lieberman, sitting here in the audience, did some interesting work where he taught an agent a particular procedure, and while he was performing actions with the mouse, he would give speech inputs to tell the agent what it had to pay attention to. For example, pay attention to this corner here that I am dragging or to this side of the rectangle. So in that situation it is very useful because your hands are already doing something else, and you need that additional channel to convey some more information in parallel.

•

Question: *This question is for both of you. Both of you seem to be concerned about protecting the user's control of the environment, but the one things studies have shown time and time again is that users are very good at making mistakes. So how do your positions relate to time-critical decision-support environments, such as medical systems or cockpit systems?*

Pattie: I have actually been focusing on a completely different kind of application, a type of application that is not as critical. For example, if your World Wide Web agent gives you a wrong Web page to look at—it assumes that you are interested in a Web page and you are not—that is not at all critical. It is not a big deal. I have been focusing on that kind of situation and those kinds of problems, the ones where if there is an error it is not very costly. I've been doing that because I believe that it will be very hard to make agents that always come up with the right answer, always do the right thing. I believe that there is a very large set of these kind of applications where things don't have to be completely precise or

100 percent correct in order for the agent to be very, very useful to a user.

Ben: I like your question. I think it is an extremely important research area. There is a long history of work often called supervisory control. Tom Sheridan is a key player in the area for nuclear-reactor control rooms, cockpits, and so on. I think the design of these systems is most effective when the users have a clear predictive model of what their actions will produce. If they do not know or are uncertain about what the results of their actions are, they will disengage the automatic system, as is the evidence with cockpit systems or nuclear-control rooms. So one danger is that in complex control-room environments when an emergency occurs, users are uncertain about its behavior. In these situations they are likely to disengage a potentially helpful system and do what makes sense to them. Therefore, as Jim Foley said, keep it simple, very simple.

User interfaces should be predictable, so that users trust them. User interfaces should be thoroughly tested, and users should be thoroughly trained for all emergencies. In emergency situations, people cannot solve problems. They can only do what's rehearsed and predictable. It is a good topic that I would love to see more attention to it by this community.

•

Question: *In the interest of brevity, I was going to bring up several human limitations or constraints on humans that make direct manipulation a little more interesting and wanted to ask you both for comments, not that I have all of the answers, but I think I would just limit it to your basic kind of law that the more things you have to scroll through or the more things you have to search through, the longer it takes you to search. The idea of how do you deal with this, particularly, for many people scrolling is not a particularly usable thing. I have particularly seen this with older people, and of course, if you want a good example just go to a fast food restaurant and watch a new person at the cash register try to find how to ring up your hamburger. It is pretty terrible. You will be there for days.*

That kind of idea, and particularly—we are talking about the wonderful visual processing and how important visualization is—this seems to be assuming that everybody has perfect vision. What if I am blind, if I am over sixty-five, and I have a very small useful field of view and I don't notice things so much on the periphery, or if I am, in my case, a person sitting right here who is having a problem with my contact lenses, how could you, Ben you were saying—well speech is okay for disabled users—how are you going to render that diagram into speech for a blind person, and what is the role of that?

Ben: That is a legitimate concern. Direct manipulation does benefit from and depends heavily on visual representations. For those who are vision challenged or blind, alternatives to visual displays are important. What surprises me is there are great supporters of direct manipulation in the visually-challenged community, because direct manipulation depends on spatial relationships. Blind users often are strong at spatial processing. If you can provide movement left, down, backward, forward, they can navigate fairly rich spaces in efficient ways.

I would say also you have been a little too quick about criticizing menu selection. The question is what would the alternative be, and how might those menus be better designed? I do believe that fast and vast menus are a great benefit in many applications.

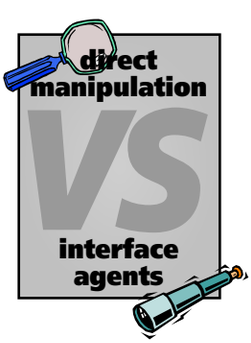
Jim: Okay. There are two more questions if you could be very brief, please.

•

Question: *This is directed more toward Ben than toward Pattie. Will this debate between direct manipulation and agency always exist in interface design or will it eventually be replaced by some kind of fusion of the two approaches? In other words, are we going to see new Bens and new Patties every day, or is there going to be some kind of "Shneider-Maes"?*

Ben: I think it has been interesting to see how the debates evolve. I certainly will point you to the new Pattie, whom I am ready to celebrate and be partners with, as I said, but I think the debate will move on. I think it has

"I have been focusing on that kind of situation and those kinds of problems, the ones where if there is an error it is not very costly. I've been doing that because I believe that it will be very hard to make agents that always come up with the right answer, always do the right thing. I believe that there is a very large set of these kind of applications where things don't have to be completely precise or 100 percent correct in order for the agent to be very, very useful to a user.



matured in interesting ways from where we were a year ago. I think we are all at the edge of looking at new interfaces, and so, as we push that envelope back, we are getting better understanding of the territory, of the strengths and weakness of direct manipulation, of the strengths and weaknesses of agents, and where they are appropriate. I am pleased by the progress in the discussion.

Pattie: I think we both have changed. Would you agree to that or not?

Jim: I think before they start kissing let's move on. Last question.

•

Question: *This starts out at least as a clarification question for Dr. Shneiderman, but it may go places from there. To what extent does direct manipulation, in your definition and in your view, admit autonomous system behavior? Because it seems to me that as soon as you admit anything unexpected, uncontrolled, potentially anthropomorphizable out of the system that you are interacting with, you have opened the door, you have taken a step down the slippery slope toward agentness.*

Ben: Yeah. I am in favor of increased automation that amplifies the productivity of users and gives them increased capabilities in carrying out their tasks, while preserving their sense of control and their responsibility, responsibility, responsibility. I am sort of not answering your question because I don't want to work in the language you are dealing with. We should be thinking about productivity improvement tools for users, whether they are graphical macros, dynamic queries, starfield displays or other things.

•

Question: *Then what is it about agents that you dislike?*

Ben: Can I go to my closing slide? I want to reassert the importance of scientific evaluations. We must get past the argumentation about my system being more friendly than yours or more natural or intuitive, and talk about user performance. We can deal with satisfaction also, but please focus on user performance and realistic tasks. Please, please, please do your studies—whether they are controlled scientific experiments, usability studies, or simply observations, and get past the wishful thinking and be a scientist and report on real users doing real tasks with these systems. That is my number one take-away message.

I am here to promote direct manipulation with comprehensible, predictable, and controllable actions. Direct manipulation designs promote rapid learning. It supports rapid performance and low error rates while supporting exploratory usage in positive ways.

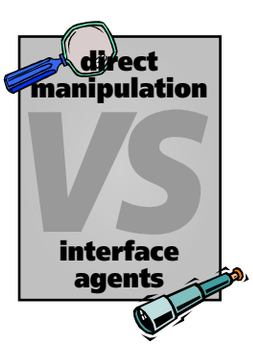
Direct manipulation is a youthful concept which is still emerging in wonderful ways. Our current work leans to information visualization with dynamic queries, but there are people doing fascinating things with enriched control panels style sheets, and end-user programming. Graphical macros would be my favorite project to advance the design of general computing tools. It is embarrassing that after 15 years of graphic user interface being widely available, we have no graphical macros tools. What is going on? This is the greatest opportunity for visual programming.

Third—and I am answering your question here—I think the intelligent agent notion limits the imagination of the designer, and it avoids dealing with interface issues. That's my view of the agent literature—there is insufficient attention to the interface. Maybe the way agents will mature is as Pattie is suggesting; that the agents take care of the processes below the table, and there is a nice direct manipulation interface that the user sees.

A leading AI researcher commented to me that the 30 years of planning work in AI is essentially down the tubes because of lack of attention to the user interface. The designers deliver a system and the first thing that the users say is, "This is great but what we really want to do is change these parameters." The designers say, "Well, you know, we didn't put that in the interface." They just haven't thought adequately about the interface, nor done testing early enough.

I believe that this language of "intelligent, autonomous agents" undermines human

I am here to promote direct manipulation with comprehensible, predictable, and controllable actions. Direct manipulation designs promote rapid learning. It supports rapid performance and low error rates while supporting exploratory usage in positive ways.



responsibility. I can show you numerous articles in the popular press which suggest the computer is the active and responsible party. We need to clarify that either programmers or operators are the cause of computer failures. Agent promoters might shift some attention to showing users what is happening so that they can monitor and supervise the performance of agents. I was disturbed that in the Autonomous Agents conference that Pattie is participating in, the organizers refused to include the topics of supervision of agents and user interfaces for programming agents. By contrast, I like Pattie's summary slide—I think her list is quite wonderful.

My closing comment is that I think there are exciting opportunities in these visual interfaces that give users greater control and therefore greater responsibility in the operation of computers.

Jim: Thanks. Okay, Pattie?

Pattie: I want to conclude by saying that I believe that there are real limits to what we can do with visualization and direct manipulation because our computer environments are becoming more and more complex. We cannot just add more and more sliders and buttons. Also, there are limitations because the users are not computer-trained. So, I believe that we will have to, to some extent, delegate certain tasks or certain parts of tasks to agents that can act on our behalf or that can at least make suggestions to us.

However, this is completely a complementary technique to well-designed interfaces—visualization, and direct manipulation—not a replacement. Users still need to be able to bypass the agent if they want to do that. Also I should say that what we have learned the hard way really is that we have to, when designing an agent, pay attention to user-interface issues, such as understanding and control. These are really very, very important if you want to build agents that really work and that users can trust. The user has to be able to understand what the agent does.

The user has to be able to control things if they desire or to the extent that they want to control things. I agree with Ben that the agent field definitely has grown a lot in the past 10 years or so. In fact, one of the ways I think in which a lot of this agent work distinguishes itself from traditional AI is that agent research focuses on building complete systems, systems that are tested, systems that really have to work, and those same principles and methodologies can be seen in all of the agents work, whether it be robots, synthetic characters, or software agents.

The field is maturing and paying more attention to building things that really work and paying attention to important UI issues. As to people taking responsibility for their agents, I think they indeed should. It is software that is running on your behalf and that you have delegated certain tasks to. So, personally, I don't see why that problem is specific to agents as opposed to software in general. Thanks.

Jim: Okay, I would like to thank the Shneiderman–Maes team for coming here today and talking to us, and thank you all for participating.

I believe that there are real limits to what we can do with visualization and direct manipulation because our computer environments are becoming more and more complex. We cannot just add more and more sliders and buttons. Also, there are limitations because the users are not computer-trained. So, I believe that we will have to, to some extent, delegate certain tasks or certain parts of tasks to agents that can act on our behalf or that can at least make suggestions to us.

References

[1] Ahlberg, C. and Shneiderman, B., Visual Information Seeking: Tight coupling of dynamic query filters with starfield displays, *Proceedings Of ACM CHI94 Conference* (April 1994), 313-317 + color plates.

[2] Ahlberg, C. and Shneiderman, B., AlphaSlider: A compact and rapid selector, *Proceedings of ACM CHI94 Conference,* (April 1994), 365-371.

[3] North, C., Shneiderman, B., and Plaisant, C., User controlled overviews of an image library: A case study of the Visible Human, *Proceedings 1st ACM International Conference on Digital Libraries,* (March 1996), 74-82.

[4] Plaisant, C., Rose, A., Milash, B., Widoff, S., and Shneiderman, B., LifeLines: Visualizing personal histories, *Proceedings of ACM CHI96 Conference* (April 1996), 221-227, 518.