4

Adopting Technology

Interviews with David Liddle, Mat Hunter, Rikako Sakai, David Kelley, and Paul Mercer


TV
ON/OFF
POWER
VOLUME
VOL
SATELLITE ON/OFF
CHANNELS UP/DOWN
OK
2
3
5
6
8
9
LOCAL CHANNELS
START AT 884

We interviewed some people with beautiful and very elaborate new media systems who were quite discouraged and quite unhappy with them. The solution from the manufacturers of consumer products was to produce the most dumbfounding, enormous remote controls. Thirty buttons was not a large number for those controls. There was a period of suppression of the adoption of the best of this technology simply because it was too complicated to use.

David Liddle, 2003[1]

■ Understanding how to use a TV remote is made easier by a friend

Photo
Nicolas Zurcher

David Liddle, project leader for the design of the Xerox Star, the founder of Metaphor Computer, and head of Interval Research, explains that a technology is adopted in three phases[2]—the enthusiast phase, the professional phase, and the consumer phase—and that these phases apply to the technology of a remote control just as much as to a computer. The maturity of a technology in this sequence has profound implications for designing interactions, as the nature of the design process changes as each phase is reached. Using the car and the camera as examples, David explains this process of adoption and its importance to an interaction designer.

The theme of camera is expanded to take a look at the way digital photography is being adopted, replacing previous technologies in processing and printing as well as in the camera itself. Mat Hunter was lead interaction designer in the team at IDEO that developed an "interaction architecture" for digital photography with Kodak, giving the company an early advantage in the market for digital cameras. He describes the development of the architecture and tells how "experience prototypes" were used to good effect.


Pocket Digital
Logitech


MINI
ANTENNE TP

Connecting images together to form a panorama is an attractive function of Canon digital cameras and software. The story behind the development of the interaction design for Canon's PhotoStitch technology is revealed in an interview with Rikako Sakai, a human factors specialist and interaction designer from the internal design department at Canon.

We look at the output side of digital photography with Epson "printables," a conceptual project to design printers for digital photographs, thinking of the qualities of photography that could influence the designs to be different from normal computer printers, yielding designs that fit naturally into everyday life.

David Kelley is both a practitioner and a teacher. Tracing the development of interaction design from screen graphics to complete experiences, he explains how the design disciplines have adopted technology and proposes a future where designers act as integrators.

The iPod from Apple is a success story of the adoption of a technology in the consumer phase, including interactive product, computer application, and Internet-based service. Paul Mercer, an inventive software engineer with a background at Apple, as well as founder of Pixo and Iventor, tells the story of the iPod development in the context of his more general contribution to the creation of software tools for the design of portable devices.

- DigiCamera from Logitech

  Photo Courtesy of Logitech

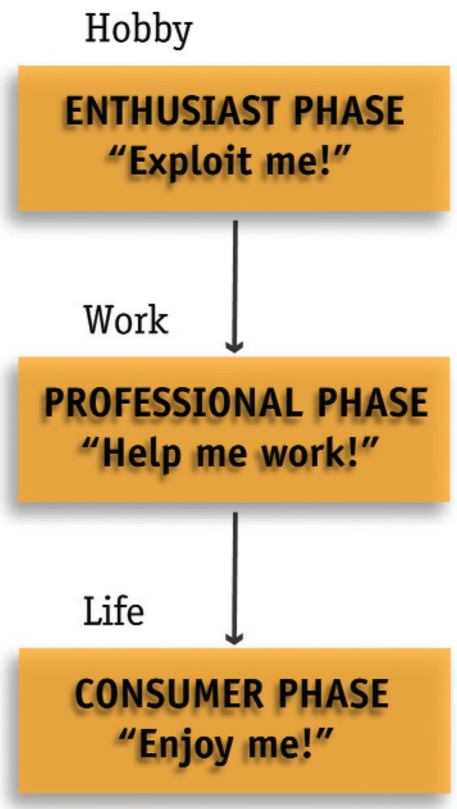
- Driving controls from Mini Cooper

  Photo Courtesy of BMW

Photo Author

David Liddle

At the age of ten David Liddle saw his first computer. It was two stories tall and belonged to the Burroughs Corporation[3] in Detroit, where his father was responding to an emergency request for components on a weekend. While he was waiting for his father, the technicians taught him to count to 1023 in binary on his fingers, and explained Boolean logic to him. On the way home in the car he said, "This software thing is going to be big, really big!" He went on to study electrical engineering at the University of Michigan, and received his PhD at the University of Toledo while working at the same time on the design of a plasma display that sparked his interest in the potential for designing graphical interactions. He arrived at Xerox PARC in time to help with the POLOS project and design the display controller for the Alto. He went on to become the project leader for the development of the Star, the precedent-setting design for graphical user interfaces, about which he says, "As far as I'm concerned, the Star GUI was a great improvement over all of its successors!" In 1982 he left Xerox to found Metaphor Computer and, using the advantages of graphical interaction design for database access and program development applications, built Metaphor into a successful company, until IBM acquired it in 1991. In 1992 he was asked to set up and lead a new research laboratory, Interval Research, to stir up some new thinking for commercial possibilities. He assembled a stellar team of researchers, including interaction, graphic, and product designers, and media and behavioral people, as well as computer scientists. He is now a venture capitalist.


Hobby
ENTHUSIAST PHASE
"Exploit me!"
Work
PROFESSIONAL PHASE
"Help me work!"
Life
CONSUMER PHASE
"Enjoy me!"

David Liddle

Three Phases of Adoption

David Liddle has a simple explanation of the development phases that a technology can be expected to go through, a process that has profound implications for designing interactions, as the nature of the design process changes as each phase is reached. Here is his explanation of the enthusiast phase, the professional phase and the consumer phase:

> In the twentieth century at least, the adoption of a fresh technology, ordinarily passed through thee stages, and seems to continue to do so now. Sometimes one of the stages may be very small.
>
> The normal progression is first to enthusiast users, who actually love and appreciate the technology in an aesthetic way, who enjoy exploiting it. The fact that it may be difficult to use actually adds to the fun, and it's certainly the case that competing variants of it will always be operated very differently. This was clearly true of automobiles, clearly true of cameras and all photographic equipment, true really of all the things like that that we might think of. The enthusiast phase is important because the enthusiasts take the

technology far beyond what the inventors and designers imagined could be done with it; they show the extremes of its potential. During this period there's always a great deal of ferment, quickly produced competing approaches. The controls for such a technology always vary a lot, because, for a while at least, people try to use them as the basis for competition. If you're an enthusiast, you're somewhat proud of you ability to manage all of the complexities and difficulties. Early automobiles broke down every four or five miles, and you had to stop and pump up the tires, or recrank the starter or something, but that was a good part of the fun. It was after all just a Sunday afternoon thing that you did.

Once enough enthusiasts have their hands on a technology, sooner or later one of them will say, "I can use this in my work!" They get a clever idea about how they're going to do something really practical with it. Notwithstanding that they enjoy its use, they decide to find a way to fit it into some practical part of their life, either literally their livelihood, or at least their home life in a practical way. As this begins to happen, there is a great change in the priorities of the developers of the technology. For one thing, they become more focused on costs and prices, not because it's going to become inexpensive, but because it will now be judged to some extent by how practical or useful it's going to be. The people who buy it, whether business people or consumers, are now saying, "Well, is that worth it for what I'm going to do?" There becomes a much more stabilized view about how much things are permitted to cost, and reliability and so on becomes important, but particularly we see the standardization of controls.

After a product has built up big enough volumes through this business phase, that's when suddenly one begins to reach a price point where it's practical for consumers to buy it. It goes from being the buy based on the aesthetic property for enthusiasts, to a practical return-on-investment kind of purchase by a professional, and now it becomes a very easy discriminatory purchase for a consumer, who feels it's practical and within their price range.

The enthusiast wants that product to say, "Exploit me! Look at my capabilities."

The business user wants the product to say, "Look at the productivity I can give you; here's how I'll change your activities."

The consumer wants the product to say, "Look at how I fit in with your style! Here's who you are. Use me and enjoy my capabilities."

In that stage the priorities for the product have dramatically changed, and one thing that we always see is that most of the important controls become automatic; for example, automobiles have automated safety functions, and cameras are automated to allow you to point and click. In this third stage we see prices that allow easy consumer decisions, the automation of the most subtle and important of the controls, and a great emphasis on the compatibility of the lifestyle of the purchaser with the image of the product.

Enthusiast, Professional, Consumer

Designing for these three phases requires different skills and processes. Inventors are often good at coming up with the first version of a technology and can find the "enthusiasts" to adopt the technology by creating nothing more than an innovative solution. A single designer is sometimes enough, although great inventors, like Thomas Edison, employed a team of experts to increase their output.

The inventor, even when supported by a band of technicians, cannot develop the technology once it enters the "professional" phase. New design values apply when people adopt the technology for practical purposes. Now the design must be reliable, it must perform consistently, it must be priced to offer reasonable value, and above all it must be both useful and usable. This is a much more demanding set of requirements than necessarily applied to the enthusiast product. Large and complex organizations in companies for development of new products and services have evolved to respond to these needs. Teams of developers must include people with scientific, technical, and engineering skills to provide performance and reliability; people with business and value engineering skills to create solutions with the right balance between price and value; and marketing and human factors expertise to ensure the combination of usefulness and usability.

A design for the professional phase does not need to be easy to use, as people take pride in acquiring skill in their work; their learned skill separates them from the unskilled and allows them to feel expert. The design does not have to be enjoyable, as people tend to take their work seriously and are willing to try hard to be productive, even if the experience is unpleasant. Education for human factors professionals has evolved around this need for professional productivity, with methods that focus on evaluation of the way people use technologies in their work, both civilian and military. The version of this contribution in digital technologies is called human-computer interaction (HCI).

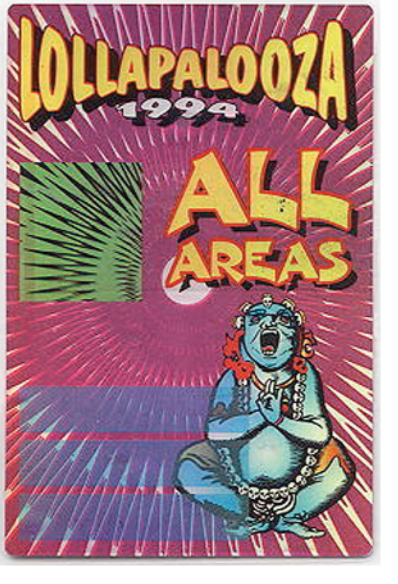
The makeup of a design team needs to change once again when a technology enters the "consumer" phase. Ordinary people will not buy products and services unless they like them and find them easy to use. There are plenty of examples of things that are not easy to use and cause frustration, but the basic value proposition must be there. The VCR is a good example of this, as the basic function of playing a tape is an excellent piece of interaction design: you push the tape cassette into a slot that is easy to see, the mechanism grabs it, and the tape automatically starts. Press the eject button, and the cassette is presented to you. By contrast, the functions needed for recording were badly designed for many years after the VCR was cheap enough to be accessible to consumers, with incomprehensible remote controllers, cryptic feedback, and little in the way of helpful automation. This classic example of bad interaction design persisted, resulting in all those examples of unprogrammed VCR displays flashing "12:00." Here the value proposition of playing the tape was enough to justify the purchase, but most people never used the device as a tape recorder because it was too difficult, until the designers eventually made use of the TV screen for output.

The design team has to be structured in a different way during the "consumer" phase in order to create solutions that are both easy to use and enjoyable. The same skills that were needed for the professional phase are still in demand, at an even more critical level, as the balance between price and performance is harder to achieve; but now it is also necessary to bring designers into the team who are capable of creating interaction design solutions that people want, that they find easy to use, that they enjoy, and that will give lasting satisfaction. In this situation interaction design fills the equivalent role for digital technologies that industrial design has filled for physical objects. Designs must work at every level but should be beautiful and delightful as well. Competitive advantage will accrue if they are behaviorally and aesthetically enjoyable.

Learning from Kids

David Liddle describes the techniques that the researchers at Interval developed to learn about the adoption of technology by studying the behavior of young people:

> One of the things we worked on very hard at Interval was developing techniques to try to understand—by age group and by background as well as simply by temperament—what kinds of products would be interesting, particularly information technology products, and how they spread from one such group or segment to another. We learned many interesting things about this, particularly among teenagers and young people of fourteen to twenty-four, but also in a lot of other age groups. We did a lot of work on this and found it to be part and parcel of how you thought about product design, human interaction design, graphical design, and so on.
>
> One very interesting thing that we learned came from our sending a touring tent along on the "Lollapalooza" alternative rock concert tour in the summer of 1994. This is a huge, multiday event, in each of forty cities or so. Besides the continuous performances, there are other little exhibits. We had a tent called "The Electric Circus" and we had a number of experiments in new media in this tent, and several hundred thousand people between fourteen and twenty-four passed through these exhibits. We interviewed a subset of them on video, or by questionnaires.
>
> We learned a number of interesting things, some of which are somewhat dated to that period of 1994, but let me give you a very



Backstage pass

interesting example. At this time, it was the practice of the Telecom industry to say, “We’ve got to start advertising things directly to kids in junior high school, or in high school, because then we’ll have them as long-time customers. We’ve got to show them the value, how much fun it is to use the telephone in various ways, to interest them in various new telephone technologies and accessories.”

That turned out to be completely wrong. At that time, if you were between twelve and seventeen, being on the phone a lot was a way of showing that you had a life, that you were in demand and interesting, and part of it. Your parents of course controlled what you did, but being on the phone was a clear way for you to be in contact with your friends. What we discovered was that once kids turned about eighteen, they were either in college or they were working for a living, but in any case they were not subject to anything like the same level of parental control. As a result, being on the phone a lot was a sign of NOT having a life. We got responses to the questionnaires saying things like, “Are you kidding? Why would I carry a cell phone around? Like, it would ring when I was at a rock concert or something!” or “Why do I need a message machine? Do you think I’m so lame that I can’t miss a phone call? They’ll call me back!”

In other words the idea that the telephone played a big role in their lives was exactly opposite to the image that they were now trying to construct. The question was, Is there an approach to telephony, to this kind of communication, that would be interesting to them, that wouldn’t speak to their being restricted or at home alone with nothing to do but talk on the phone, that seemed to work with the idea of being out in the big outside world?

In Japan and in Western Europe, the great take-off among teenagers in the use of cell phones didn’t come because they wanted to dial one another up; it came because they wanted to send messages, exchange ring tones, and do lots of other interesting things. In Japan you had this gigantic adoption of these surprising and strange techniques—capturing your own little photos; people standing round the block to have their photo taken such that it could be embedded in their cell phone—before there was easy data transport. The connection of telephony to messaging and games and other things like that was a part of what that age group saw themselves doing—a symbol of freedom and the completely fresh use of a technology rather than the earliest stages of “adult use” of a new communication device.

The insights from this kind of subjective research were used to seed projects at Interval, several of which developed into "spin-out" ventures, for example Purple Moon,[4] a company devoted to developing interactive media for girls.

David Liddle refers to the car and the camera as he explains the enthusiast, professional, and consumer phases of the adoption of technologies. This concept of phases helps to explain how interaction design has evolved and is a crucial idea to grasp if we are to be more successful in structuring both the education and the practice of designing interactions. Deeper looks at the examples of the car and photography follow.


Kommunikation
Klima
Navigation
Entertainment
12:23
BMW

The Car

Its cabin, too, is awash with innovation. A color screen is linked to the parking radar and shows how far you are away from a scrape. The iDrive switch—which controls the music, the navigation, the phone and the cabin's climate—is carried over from the 7. In the 5, though, its functions have been simplified. It is now an intuitive control, rather than a baffling one. It's still a lot more complicated than a few simple knobs, though. To change radio channels you need to push the "Menu" button, scroll down to "entertainment" with the iDrive knob, push the knob again to choose the correct radio frequency, and then swivel through to your desired station. Still, once you've read the owner's manual you're fine. Owning a modern BMW is like buying a new computer. Puzzling at first, but marvelous when you're up to speed.

From a description of the controls for BMW 5 series, new in 2004[5]

BMW iDrive

Photo
Downloaded from
www.bmwworld.com

The automobile has had plenty of time to move through the three phases of adoption. The enthusiast phase lasted from the day in 1886 when Gottlieb Daimler fitted one of his petrol engines to a horse-drawn carriage until the launch of the Model T Ford in 1908; indeed, you could argue that the enthusiast phase continued much longer, particularly with the design of European sports cars that were unreliable enough to demand careful nurturing. The Model T was an amazing leap forward in design, making the car accessible to consumers as well as professionals, and establishing a dominant design for the basic control functions that is still with us.

You have to look at the peripheral interactions of entertainment, communication, or navigation to see examples of the struggle to create new controls that are easy to use. This is well illustrated by the quote above, about the BMW 5 series, where even this sophisticated driving machine is offering interaction design solutions for the functions on the central console that are not just difficult to use but that also create a safety hazard by diverting the attention of the driver away from the road ahead for a complex sequence of interactions—when the driver is looking forward, the central console is located in the peripheral vision zone.

At first the car was called the horseless carriage, and there was some confusion about the best approach to the design of the interactions. There was the tradition of interacting with the horse-drawn carriage, with reins and verbal commands and steering by guiding the path of the horse. Take away the horse and this made no sense! At least the front axle could be angled to steer, so the first question was how to turn it right or left relative to the body of the vehicle. Some coachbuilders tried selective braking for the right and left side. Others tried to build on boating traditions by devising tillers. Still others turned to the tradition of larger ships by designing steering wheels, which soon became the dominant design, providing direct manipulation at the same time as mechanical advantage for adjusting the steering linkage.

Then there was the question of how to start and stop. At first a pedestrian with a red flag was required to walk in front of the car to warn people of the danger approaching, but faster progress was soon wanted, and hand levers and foot pads were adopted for engaging and disengaging the engine, applying the brakes, and adjusting the speed.

The pattern of steering wheel, foot pedals, and hand controls has stayed almost the same since. Some variations still persist, like the islands of Britain, Japan, and New Zealand, where you drive on the left side of the road, mirroring all of the controls. Persistence is the key here, in that once a set of interactions as complicated as driving is learned by a large population, an escalating resistance to change sets in. The dominant design is in place, infrastructure is built up to support it, and it is difficult to persuade people to adopt another solution. A variation in gear changing has survived with the invention of the automatic transmission, but it has still not replaced the stick shift. Otherwise everything is the same; we still steer with a wheel, and both accelerate and stop with foot pedals. Even the details of the controls for signals and windshield wipers are now standardized enough that you can get into an unfamiliar make of rental car and know how to drive without confusion, at least with the controls that obviously matter to safety, even if the interactions for comfort and entertainment in the car are still confusingly different with different designs.

Changes in technology have had surprisingly little effect on the basics of driving. The value of electronic technology in cars continues to grow and was already more than a third of the total cost of a typical vehicle by the 1990s. Computer systems control instrumentation, fuel economy, emissions, and emergency behaviors like airbag activation and antilock braking. Most people are unaware how much of their driving experience has been subtly altered by this technology, which has invaded the vehicle transparently. We usually think of technology as expressed in the design of the personal computer, with keyboards, mice, and screens. It would be strange and frightening to have to sit in the front seat of a car with a keyboard, and steer by typing commands like, "15 degrees right in 35 yards—enter." The direct manipulation of steering is simple, effective, and enjoyable. Learning to drive takes time and patience, but once we have acquired the skill, we take pride and pleasure in the power of controlling the movement of this huge object that travels so fast. The design has evolved slowly by iterative steps, with each step making it a little easier, or a little safer, or a little faster, but the interactions have stayed substantially the same.

Even the instrumentation on the dashboard has remained analog, with simple rotating pointers in round dials, resisting the temptation to look more high-tech and digital. The car is a remarkable example of interaction design that matured early in the history of the product category, developing an approach that worked well for consumers as well as professionals. There was no need to change from a dominant design that remains satisfying to most people. For several decades, technologists have been predicting more automated ways of controlling cars, for example drive-by-wire to allow you to surrender control to the computer and tuck into a fast moving stream of traffic on a specially equipped highway. This may eventually be the future, but there is a huge psychological barrier against giving up control. We prefer the risks of death and injury on the roads to the idea of letting a machine take over, partly because we mistrust the computer—and also because we love to exercise our driving skills.

Digital Photography

David Liddle points to the camera to illustrate the three phases of adoption of technology. He uses the cameras that were taken into space by the early astronauts as an example of the "enthusiast" phase, saying, "The 35mm cameras used by the astronauts in the fifties, nearly required a PhD in optics to operate them."

> In the case of the 35mm camera, the design suddenly stabilized when it went from very expensive exotica to being broadly used by professional photographers and serious photographers. The viewfinder was in one place, you exposed by pushing with your right index finger, you wound the film with a lever with your right thumb, and you focused in a particular way. There was a stabilization of the controls.
>
> Today when you buy a 35mm camera, even if it has film in it, it will read the film speed automatically, set the exposure automatically, set the flash automatically, and actually a chimpanzee can take pretty good photographs with today's highly automatic 35mm camera.

They are dubbed "idiot cameras" because anyone can take a pretty good photograph, even without expertise in photography. Point and shoot—the consumer-phase technology—takes care of all the details and gives you the best shot that is technically possible in the circumstances. When the consumer phase arrives,

- Nikon F3 NASA—Space Nikon—front
- Rear

it does not replace the market for professional solutions, but often expands it as more people realize the potential to improve over their amateur skills.

The experience of photography is much broader than the design of the camera on its own. There is a huge infrastructure supporting both professional and consumer photography in parallel. The professional side includes photographers who will take the pictures, processing and printing services, endless categories of equipment, and the gear to control lighting at every scale. On the consumer side there is an enormous network of processing facilities, most of them offering one-hour services to go from film to print.

Digital photography is replacing film and print photography in an industry that is already mature enough to have both professional and consumer versions of the technology, so we can see sweeping changes—and the opportunities for innovative solutions that accompany them—affecting the whole industry at once. In the interview that follows, Mat Hunter tells the story of developing an "interaction architecture" with Kodak for digital cameras and photography in the consumer market.

Photo Author

Mat Hunter

When Mat Hunter was studying industrial design in London, he chose to design a videocassette recorder for his major project. He turned the VCR on its back so that it could sit on the same shelf as the videocassettes in the library, and reshaped the remote control to avoid losing it in between the sofa cushions. He found a quote from George Bush Sr., president at the time, declaring his intention to make every American able to program their VCR: the obvious failure in meeting this goal made Mat wonder if the design of the VCR might be as much to blame as the educational system, so he decided to study interaction design at the Royal College of Art and to master the art of designing interfaces that would be easy and enjoyable to use. He joined the San Francisco office of IDEO in 1995 and was key to the development of an interaction architecture for Kodak. He transferred to IDEO London in 1999 and became head of interaction design. Both as a manager and designer, he has worked on a wide range of innovative fusions of hardware and software, including medical devices, interactive guitar tutors, Web-based education software, cable modems, and a system for reserving meeting rooms in offices. He has created exploratory concept projects for connected appliances, for example "social mobiles,"[6] an exploration that looks at the social impact of mobile phones. He has appeared in television programs on design and innovation on the BBC and Open University and is currently head of the London office of IDEO. He also plays guitar and drums in Amped Up, the office band.


PIER 28
11:08 AM 09/03/96
4/50

Mat Hunter

Interaction Architecture

Clockwise from top

- Screen for prototype
- Experience prototype
- Concept for future camera

Photos
Courtesy of IDEO

MAT HUNTER WAS a key member of the team that developed an interaction architecture for digital photography, helping Kodak retain their strength in the consumer market, while attempting to bridge the gap from film-based photography. This case study illustrates some of the issues facing interaction designers as they design for the adoption of digital technology in an area that already has a strong consumer tradition but where a conventional technology is being replaced by electronics.

In 1995 Kodak commissioned IDEO to look at the design of digital cameras, but fairly soon it became evident to the team that they would need to expand the vision a great deal further than just the camera itself, because digital technology would change photography as a whole. Mat Hunter[7] describes this epiphany.

> George Eastman himself had said right at the beginning, "You press the button; we do the rest."
>
> He realized that actually it wasn't just the Box Brownie they were making, but also the whole service, the back end, for turning this

- User observations—sorting
- User observations—displaying
- User observations—storing
- Ceiva digtal picture frame

> captured Kodak moment into a print, or whatever else was the final artifact. What we had to do was take the digital camera, step back, and reassess why people wanted to take pictures in the first place. Then we would have to build up some picture of the ecology, you could call it, of products and services surrounding the digital camera that would be the replacement for the film and film processing, product, and service that Kodak had become successful with.
>
> We went and looked at how people took images and understood how they used them to capture special moments and to share memories with others. Perhaps it was a form of self-expression, photography as a hobby; sometimes it was merely to record very basic information. Underlying motivations were slowly extracted and synthesized, and these were used to inspire a range of products and services.
>
> For instance, if you want to share images easily, perhaps we should have a digital picture frame, and in this way, something very simple, much simpler than a PC, would sit on someone's mantelpiece. I would take a picture and send it over the Internet, over some sort of network, to the picture frame. And that way, Grandma could get to see the picture.
>
> Or what if you have much smaller personal communicators, where you can send and receive images? What would it be like? What role would Kodak have in that world? Would they actually make the device, or would they merely provide the underlying engine within the device?

Those initial scenarios for the potential of digital photography have come true since then. Ceiva[8] is one of the companies making digital picture frames, and image communicators are a standard offering from most of the cell phone service providers, with digital cameras built into the phones. A broad range of products and services is emerging to support this digital infrastructure for imaging. You can email images to an Internet site, and they will be printed for you and mailed back to you, providing a service that is analogous to the traditional Kodak print service; you can store your images online in some kind of online photo album. You can buy dedicated printers now, so that you can connect your camera directly to it without needing to go through a PC.

> If we look back to 1995 and see this strategy that we were constructing and come forward to 2003, we see that this new ecology of digital imaging has been built, but it's been very slow. The old silver halide photography that grew up over a hundred years cannot be shifted overnight despite the fact that digital cameras have leapt off the shelves and are taking over more and more. Kodak realized that we wouldn't really be moving to digital imaging completely overnight, so once we had understood the larger ecology, they directed us to make our first priority to design excellent interactions for digital cameras.
>
> It was fairly clear from other digital cameras that were just appearing that they would have things like small displays on the back of them so you could review the images. This immediately changed the nature of the camera, because previously cameras were purely about capture: now they were also about review. This means that you want to do things like delete images, group them, send them off somewhere, perhaps email them to people or to a printing house. Maybe you'd want to capture motion as well as still images. Perhaps you'd want to record a sound annotation, a sort of "voice note" on it. Suddenly this essentially rather simple device was becoming much more complex. We already knew that cameras could get very complex just dealing with the fine nuances of how to capture things. How complex would they become if suddenly they had all this other functionality?

The challenge from Kodak was, "Find a way to make sense of digital camera interactions!" The design team had to take another step back, and look at the fundamental modes of use. They were looking for a really cohesive user experience, something that would put Kodak out in front again. Kodak had already lost their dominant position in camera sales and was being overtaken by Japanese competitors in many areas, but they still had powerful organizations in the US and European markets, potentially capable of launching a whole new range of digital cameras. They could build a strong brand with a great reputation for ease of use, creating cameras for everyone, not just for technologists. Mat came to the conclusion that something more than a single camera with well-designed interactions was needed.

> We said that we had to build an "interaction architecture" because we weren't looking to build any one particular camera; we were still really working with the research team. Therefore, what we had to do was make an extensible system of design principles, values, and elements, so that the development teams would be able to find the bits that they needed as they were making their products and bundle them cohesively into a camera. What we hoped was that this would happen over many years, and there would some consistency across the cameras that were made as they evolved, as features were added or taken away. So it was something that was much more extensible than merely the design of one single camera.

The insights from the observations and research were gradually coalescing into concepts that were clear enough to use for design. Jane Fulton Suri,[9] a psychologist, was leading the human factors studies and research for the team at that time, and she remembers some of the challenges:

> Some professional photographers were already using digital cameras at that time, but there were no consumers with experience in digital photography. We were trying to push forward into the future, based on observing professionals and early adopters of the new technology, combined with an understanding of the way ordinary people thought about photography, plus projections of trends for the spread of digital equipment.
>
> I remember giving a high-end digital camera to one of the families that we'd interviewed, so that they could try it out for themselves. When I explained how to use the camera, the mother said, "Where's the film?" so I had to talk about receptors, pixels, and digital memory; things that most people know about now.
>
> Her next question was, "How do you know how many pictures you've got left?" so I had to tell her about the tradeoff between the size of each photograph, what you wanted to use it for, and the number of pictures that you could take. This was also a difficult issue for Kodak, as they were proud of their tradition of high-quality images supported by lots of carefully evolved metrics. It was hard for them to see the benefits of trading quality for other advantages, for example, taking a picture of the baby that would be small enough to transmit immediately over a phone line to a little screen on granny's mantelpiece.

Jane prepared a framework that summed up the opportunities for consumer digital photography under five headings:

1. Readiness to capture

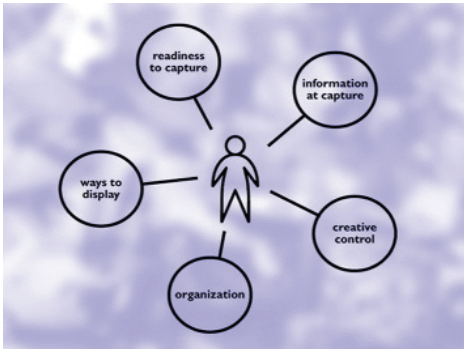
The professional photographer travels with cases full of lenses, camera bodies, tripods, and lighting equipment. The team predicted that when digital photography is adopted by the general public, cameras will include a broad range of devices, from the traditional professional kit at the top end, through simple cameras with built-in lenses, to devices such as cell phones, or wearable cameras that would look like jewelry.

2. Information at capture

The real-time feedback of the screen on the back of the camera would emerge as a highly valued feature of the digital camera, so you could see immediately if the shot looked promising or disappointing. Information about when a picture was taken could also be recorded, along with the technical details of the image, and perhaps a voice annotation. In a future where GPS is inexpensive enough to be integrated into the camera, the information about where the shot was taken could make browsing and sorting a lot easier.

3. Creative control

In traditional photography, the composition of the shot and the choice of lighting happens in the camera, but there is another set of opportunities for creative control that happens later in processing and printing. The team realized that digital photography is not so sequential. Any time during the process, you can apply filters, pixilate, choose sepia, add picture frames, text, and so on. This realization led to this framework, expressed as five balloons around the user, rather than a linear journey through the experience.



Framework for digital photography

4. Organization

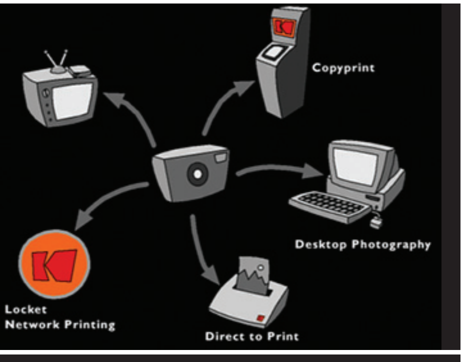
There are well-organized albums of photographs that people use to remember an event or a trip, or to recount the story to their friends, but there are also countless boxes full of unsorted photographs in almost every home. Digital photography offers the opportunity to sort using the "information at capture," but also to recognize images from small thumbnails. iPhoto from Apple has made excellent use of our ability to scan tiny versions of images to recognize the one we are looking for, leveraging the fact that we remember images best by a visual representation rather than by which shoe box we put it in.

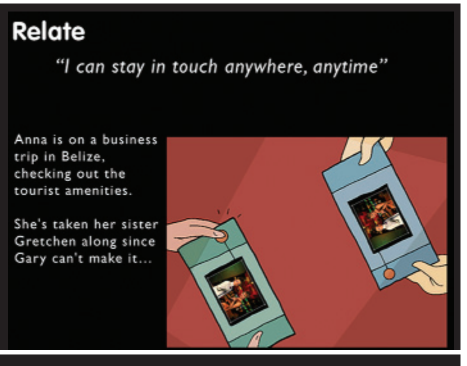
5. Ways to display

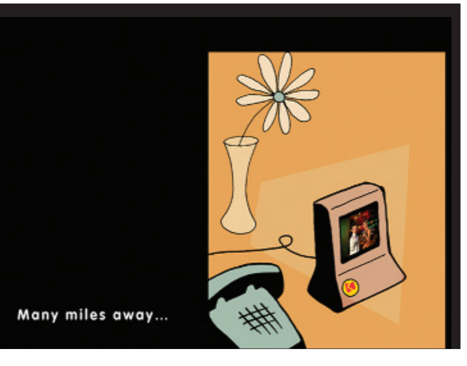
Pictures mean very little unless you can see them, so one of the great potentials of digital photography is to increase the diversity of means of display. Prints and transparencies are still valuable, and indeed a whole industry has grown up around printing, but there are also many new possibilities. Electronic displays include the television, computer screen, the electronic picture frame, the e-wallet, fridge door display, and cell phone, as well as others that will emerge in time. As digital images become more ubiquitous, it is interesting to see how we use them more habitually to illustrate a point in a conversation with a friend or to remember a piece of information. When the images are displayed electronically, they can be captured and shown at no incremental cost, so they spread into all sorts of unexpected places.

The system perspective

A system perspective was summarized by putting the camera at the center, with connections to the computer for editing, to the printer for output, to the television for display, to online resources and agencies for sending to other people and for remote printing, and to kiosks in public places for copying and printing.







- System diagram
- "Relate" scenario 1
- "Relate" scenario 2

Scenarios

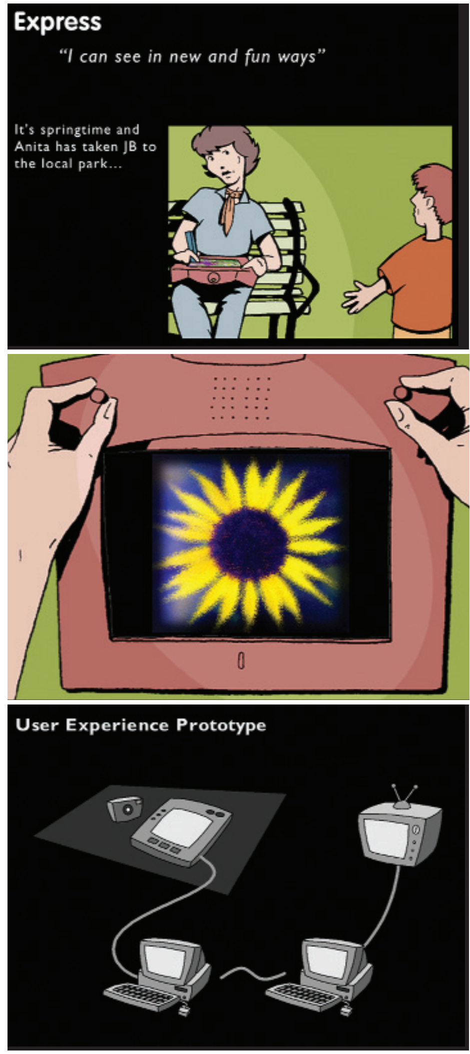
Scenarios were developed to bring the design opportunities to life. One was called "Relate" with a slogan "I can stay in touch anywhere, any time." The story was about a business trip to Belize, to check tourist amenities, yielding these highlights:

- Preview and review images
- Local image transmission
- Direct link to communication infrastructure
- Voice annotation
- Multiple display options
- Visibility of Kodak as the enabler

Another was called "Express" with a slogan "I can see in new and fun ways." The highlights from this one were:

- Emphasis on creativity at capture
- Stylus input for fine control
- Global effects pre-capture
- Limited manipulation post-capture
- Cards for loading effects
- Print output from service bureaus

These lists of highlights were used to develop patterns of value that would appeal to Kodak's customers and formed the basis of design for the interaction architecture for the system as a whole, as well as potential business cases for analysis. The elements of the interaction architecture for the cameras themselves were also starting to emerge, with the first creative synthesis of the concepts for the display screen, the mode dial, the filmstrip, and the navigational controls.



"Express" scenario 1
"Express" scenario 2
User experience prototype diagram

User Experience Prototype

Mat sums up another difficult challenge, caused by the separation of this research activity from the development teams that would want to use the "interaction architecture":

> Because we weren't actually designing directly with the development teams, we knew we had a few challenges. One was that we couldn't actually be sure of the technology we were using. We knew we had to make certain assumptions about the size of the display on the back of a camera and the processing speed that would affect the way in which it captured images and presented menus. We decided to think a little bit in the future and say, "This is the idealized camera experience, and we believe this technology will be obtainable within three to five years." Then we hoped that the development teams would be able to "down-sample" what we had done; they would take the essence of it, applied in practice to whatever screen display size or whatever processing speed they actually had.

They racked their brains for a way to communicate the interaction architecture that would engage the imagination of designers and engineers in the future. A book of guidelines would be the conventional approach, but however interesting and beautiful they could make a book, diagrams and text would not be a very compelling form of communication. They came up with the idea of creating a working prototype of an archetype for the interaction behavior, so that the members of the development teams could experience the behaviors directly.

■ Kodak DC 210—first production camera

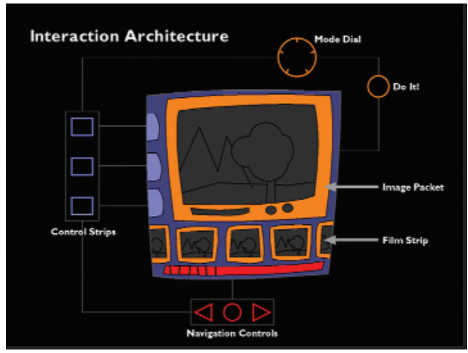
> It was a real live working specification. We called it a "User Experience Prototype." In physical form it was pretty big and ugly. It was a beige plastic box the size of a very large hardback book. We got a CCTV security camera and bolted it onto the front; we tethered it via a rather large cable to a tower Macintosh, the fastest we could get at the time, but achingly slow today, and built a working system.
>
> The whole point was that you shouldn't really look at the physical form; you should become immersed in what you could see on the display and what happens when you press the buttons and rotate the knobs. What we created in this device, therefore, was what we

believed to be the epitome of the user experience as you are interacting with a device. This is how it should behave, both logically and in terms of its animation and its time-based behavior. What was exciting was that when we finished this user-experience prototype and showed it to Kodak, they immediately got what we were talking about, both in terms of the logic and the spirit of the device. This was something that not only was logically easy to use, but it felt easy to use. It was responsive, it moved as if it were a very simple mechanical device. People could understand it.

The idea of the "user experience prototype" was working, and Mat was very gratified when Kodak said, "Make us four of them, please." They wanted to ship one out to Japan to the camera factory, so that the development teams based in Japan making the next generation could immediately immerse themselves in the specification. They used the other three to send to different parts of the Kodak organization. Only eight months later, the DC 210 was born, which was the first of this new generation of digital cameras, using the concepts in the interaction architecture.

The architecture had several basic elements. Firstly, there were three modes to make sure that the user wasn't overwhelmed with everything at once. These were "capture" for taking pictures, "review" for seeing what was already captured, and "send" for the possibilities of communicating the images. They decided on a rotary control for the prototype, and called it the "mode dial." It was big and prominent with icons printed on it, so that if you looked at it, you immediately had a good idea what the camera could do.

Second, there was the way in which the images that had already been captured could be shown on the display. In the previous designs from Kodak and the cameras from competitors, when the camera memory contained a lot of images, perhaps forty or fifty, it was very hard to navigate. It was important to be able to see, especially in the "review" mode, what was already captured, both the individual image and where it was relative to the other images:



Interaction architecture diagram ■
Design concept ■
Kodak DC 210 implementation ■

We created a little filmstrip at the bottom—something that everyone could understand, a bit like holding up a thirty-five millimeter negative, where you had little pictures, little thumbnails. The thumbnail in the center of that strip was the one that you would see bigger up above. At the bottom of the display, because you could only see five thumbnails on the display as they slid left and right, we added an "elevator," like the scrollbar you see on a Windows or Macintosh interface; it was a little dot that moved across the screen, showing the whole of camera memory. It showed you how many pictures remain to be taken and where you are in this big filmstrip, so you don't get lost.

What was very interesting, beyond this static representation of the filmstrip and this little scroll bar at the bottom, was the way in which it moved. The microprocessors in those days were quite slow, and what would happen on most contemporary cameras was that moving from one image to another would take perhaps a second. While a second doesn't perhaps sound like a lot of time, if I'm really trying to move through thirty images, that's thirty seconds, and it's achingly slow.

So what we'd done with the user experience prototype was say, "It should move this fast."

"This fast" was a really nice speed. It felt like it was a very competent device. And yet, when the development teams were developing the DC 210, they realized that their processing power wasn't able to move images that fast, so what they did instead was to design a little trick; they moved the thumbnails of the images as fast we wanted them to, and when you stopped, what you got is a big thumbnail. Then, a second later, the full resolution of the image materialized. What you got was the feeling of movement, which made the device seem really great, but it tricked you actually into thinking that it was more powerful than it was.

Mat and the other members of the team were delighted that the dynamic prototype successfully communicated the behavioral intention, and they were grateful to the designers in Japan for achieving the intent of the behavior with such limited resources. Several different versions of the controls were visualized in appearance models of concepts for cameras that were not yet feasible, so that the people working on the designs for the first real products could see beautifully designed options as well as the big and clumsy "experience prototype." This illustrated the potential for the interaction architecture to be applied for the full range of Kodak cameras in the future.

The Kodak DC 210 camera was very well received when it first came out and became the top-selling camera in Japan for a while. Reviews in photographic magazines are usually couched in very mechanical terms, itemizing the elements of the specification, such as the number of megapixels and the battery life, but even these reviewers appreciated the fact that the Kodak camera was so easy to use. They acknowledged that this was what consumers wanted and said, "Yes, Kodak's got a winner here. Something that really is fantastically easy to use. It really fits with the Kodak brand."

Kodak reaffirmed their positioning as the company that makes "the camera for the rest of us," something that you can just point and shoot.

In 2001 Mat and a group of his interaction design colleagues were asked to create a top-ten list of outstanding examples of interaction design to demonstrate excellence in any area of digital technology. One of the examples that they chose was the Stitch Assist mode in Canon cameras, which allows the photographer to easily align images in the camera when they want to create panoramas or large tiled images; the shots are then imported into the PhotoStitch program on a personal computer for integration into the large format. The combination of in-camera interactive guidance with on-screen processing and editing is made possible by the adoption of digital photography. An interview follows with Rikako Sakai, the Canon interaction designer responsible for Version 3 of the PhotoStitch implementation, and through her we trace the development of this outstanding design.

Photo Ivan Gasparini

Rikako Sakai with her work at Interaction Design Institute Ivrea

Including industrial and graphic designers, human factors people, and interaction designers, there are close to 180 people in the Canon Design Center in Tokyo. Each year, two or three outstanding engineers and researchers from key areas receive scholarships from Canon to study in any overseas postgraduate program that they wish. Rikako Sakai won a scholarship for 2002–2003 and elected to go to the Interaction Design Institute Ivrea (IDII)[10] for the master's program. Her thesis project was about "wearables," worn by objects such as tables and chairs rather than people, whose goal is to stimulate interpersonal communication. For example, a chair cover contains a display on the back showing how long the person has been sitting on the chair. A friend seeing two hours on the display might suggest to the sitter that it was time to interact with someone. Back at Canon, Rikako says, "My big dream is to be able to control all the design decisions in a development process. I want to design whole products; both the hardware and software." The closest that she had come to realizing that ambition before going to Ivrea was in the design of version 3 of the PhotoStitch software; for this project she had control of every aspect of the human factors, but she did not have the chance to design the screen graphics or the camera interface. In 1993 Rikako graduated from Tama Art University in Tokyo with a BA in industrial design. She has acquired her expertise in interaction design and human factors through her experience at Canon.


Canon
1 → 2
SET MENU DISP. FUNC.
CF OPEN
52

Rikako Sakai

I have been working for Canon for more than eight years. When I was involved in the development of PhotoStitch, I was working in the human factors department as a kind of human factors specialist. This was the first time I had worked on something related to digital cameras, since that engineering team hadn't asked us for help to improve their products, to make them user-friendly.

Rikako Sakai, 2002

Using Stitch Assist mode on a Canon Digital Elph camera

Photo
Nicolas Zurcher

In 1996 Canon launched their first version of software for stitching images together to form panoramas or large tiled photographs. In order to use it, you had to take overlapping shots in your camera as best as you could, download them to your PC, and then stitch them together using the program. Kenji Hatori was the software engineer at Canon in Japan who led the team that designed the program. He said that initially they

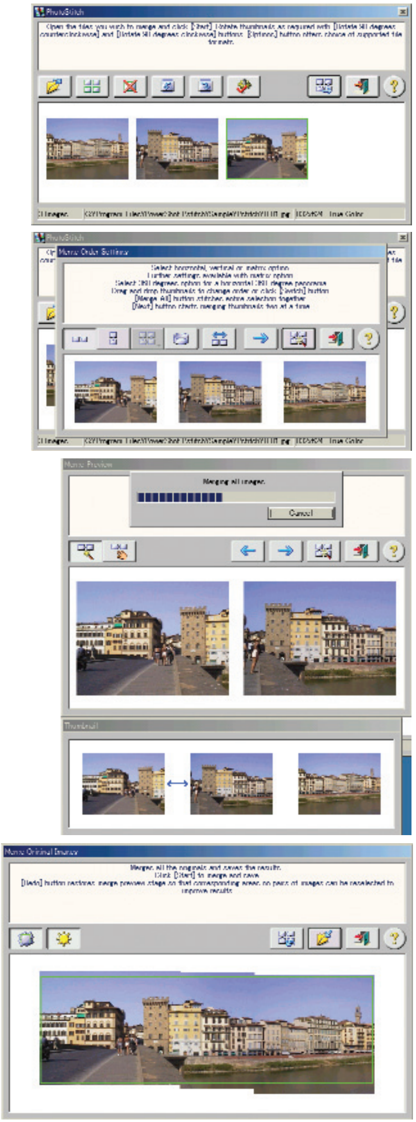
> were thinking of a big feature that can be done by digital camera, not the traditional camera. The first idea came from a development team that investigated the algorithm of image stitching; they were investigating for a few years. When the algorithm was almost fixed, the product planning division thought it could be put into a product, but there was no UI. There was only algorithm, and the product planning division found the member who could implement the UI. There were several ideas, but the most impressive and biggest feature was the PhotoStitch idea.

This idea of automating the connection between adjoining photographs was first developed as a software product to reside within Windows, so the user of a PC could import overlapping images from a digital camera, or scans from prints or slides, and the algorithm would help them stitch the images together. There was no connection to the camera, so the photographer had to guess how to juxtapose the images with the right amount of overlap. This meant that the first version of the software only performed well if the photographs were carefully positioned, using a tripod and rotating at a defined angle for each shot. It was hard to use, both in the camera and on the screen.

In the second version they added a Stitch Assist mode in the camera, helping the user to take the best shots as they record the images. This integrated the physical controls on the camera with the software that would enable the stitching and made it much easier to use. As Kenji Hatori recalls:

> With the Stitch Assist mode, the user can easily take the panorama image. First, the user takes one image, and then the first image will remain to the left side of the LCD. The user can adjust the camera image to match the remaining image. The remaining image and the current image is transparent, so you can see the difference. Then the user can match the current image with the previous image.
>
> The Stitch Assist mode in the camera I think came from the PPD [Product Planning Division] and the camera engineers. They knew that the version 1 was already shipped. They looked at it and tried to use it and found that it was a little difficult to take such photos because the images had some common area. They thought the camera could assist the software. The concrete idea came from the camera hardware development team.

When the mode is selected, a guide appears in the viewfinder display to help you as you take one image, showing the position of the current image relative to the next one in the series. When the first image is recorded, it is repositioned to show about a third of its area adjoining the next image, so that the viewfinder display can easily be used to help align the new photograph with the one before. This gives you a preview of what you are trying to do and guides you through the process in a very fluid way. The actual stitching is still done on the PC after the shots are recorded, but without this real-time interactive help, it is very difficult for people to position each shot on their cameras effectively for

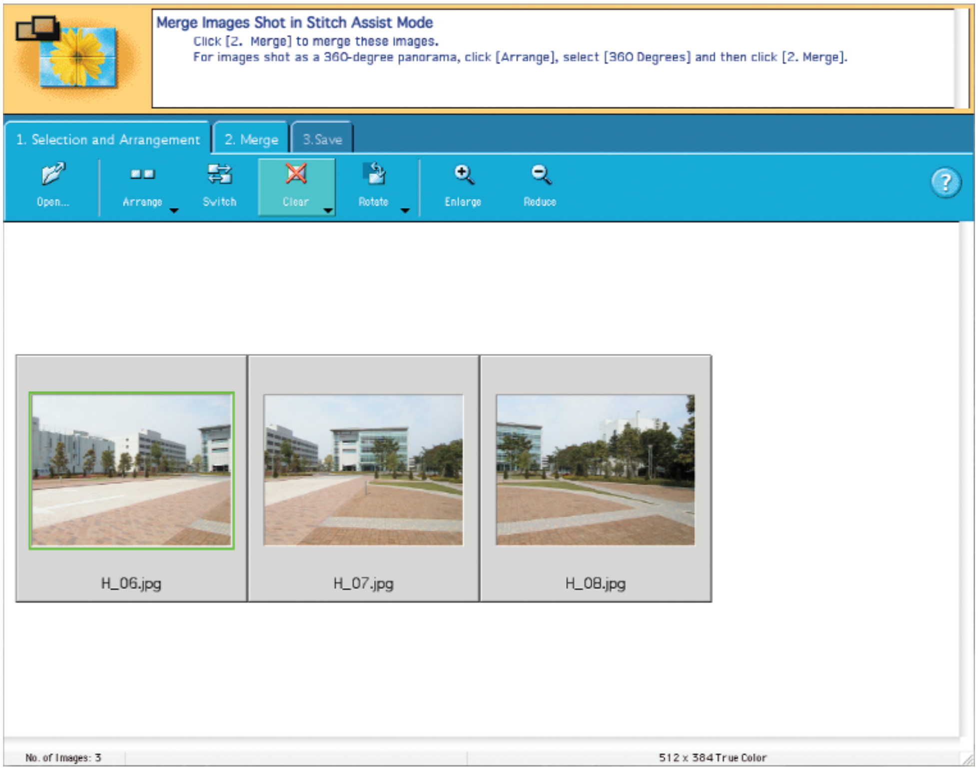


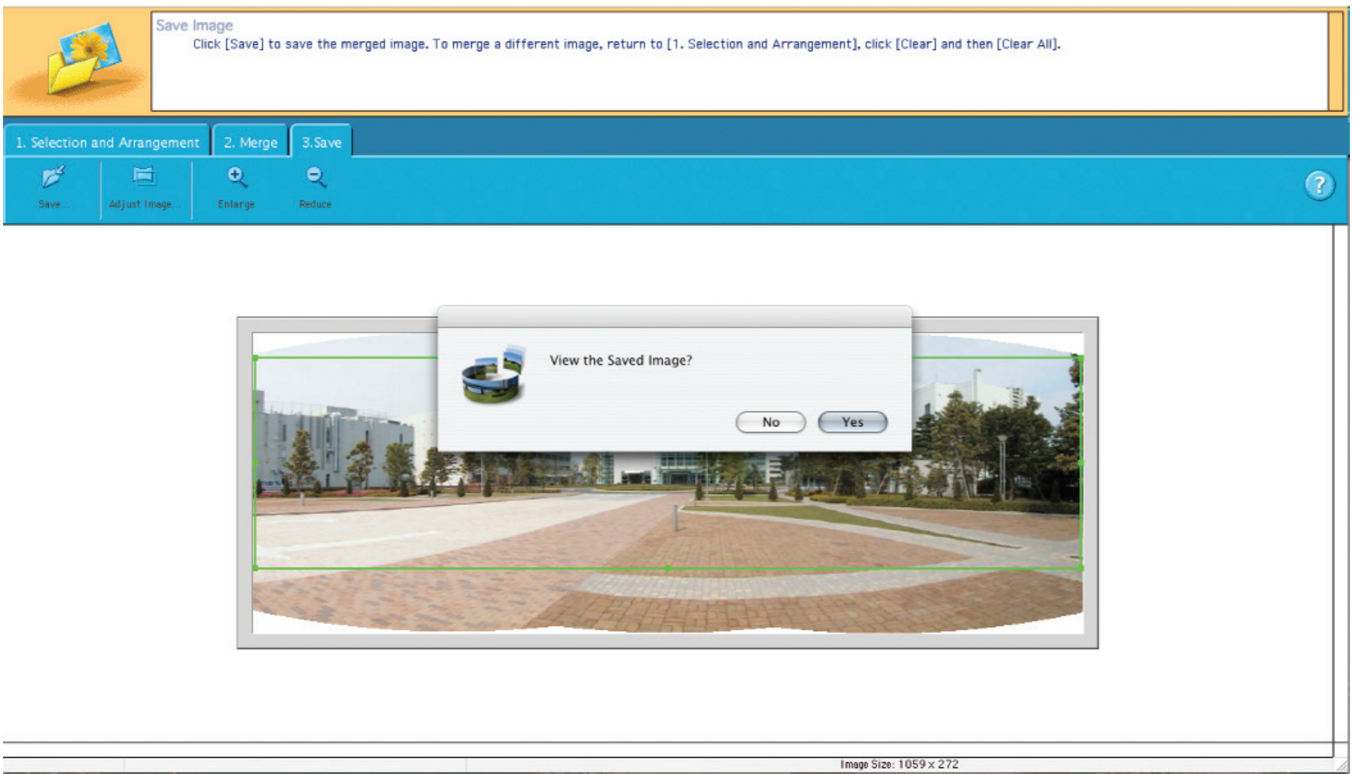
- PhotoStitch version 2—open files
- PhotoStitch version 2—merge order settings
- PhotoStitch version 2—merging all images
- PhotoStitch version 2—save result

completing a panorama. By building it into the camera itself, the experience is much more enjoyable both when you are shooting and afterward on the PC.

The second version also improved the merging process on the PC screen by adding a "merge all" feature, but people still found it difficult to make the final adjustments to the connected images. For version 3, Rikako Sakai joined the team from the Canon Design Department. She is an interaction designer and human factors expert, and she guided the engineering team through a process of user testing and the design of a much more intuitive user interface for the screen-based software. She added animated icons, tabs for the steps in the process, informative text messaging, and transparent overlays and indicators for the fine adjustments.

Stitch Assist—mode dial
Stitch Assist—first picture
Stitch Assist—second picture
Kenji Hatori, Canon software engineer


Merge Images Shot in Stitch Assist Mode
Click [2. Merge] to merge these images.
For images shot as a 360-degree panorama, click [Arrange], select [360 Degrees] and then click [2. Merge].
1. Selection and Arrangement
2. Merge
3.Save
Open...
Arrange
Switch
Clear
Rotate
Enlarge
Reduce
H_06.jpg
H_07.jpg
H_08.jpg
No. of Images: 3
512 x 384 True Color


Save Image
Click [Save] to save the merged image. To merge a different image, return to [1. Selection and Arrangement], click [Clear] and then [Clear All].
1. Selection and Arrangement
2. Merge
3.Save
Save...
Adjust Image...
Enlarge
Reduce
View the Saved Image?
No
Yes
Image Size: 1059 x 272

Canon PhotoStitch

After we shipped version 1, Product Planning Division and customers said it was not so good UI. It was not good looking. It had no color, only grey color because it was based on the Windows standard buttons.

Kenji Hatori

PhotoStitch version 3

- *Top* Selection and arrangement
- *Bottom* Saving

For the first two versions of PhotoStitch, the engineering team developed the design without working with interaction designers or human factors specialists, only requesting help from Canon's Corporate Design Center for the design of the screen graphics when the behavioral aspects of the product were already defined. When Rikako Sakai was brought in to help with the design of version 3, the first thing that she did was to set up thorough usability testing of the first two versions. She assembled the whole team who were involved with the project, including engineers and screen graphic designers, to observe how people used the products from behind a one-way mirror:

> Throughout the usability testing I found many problems with the old version, more than I imagined. In fact, before the usability testing, the engineers had already noticed some of the problems, and they told me what they had known before the usability testing. I added their findings and my findings on a checklist.
>
> After the tests, we realized that the biggest problem was its structure. The structure was not clear and visible to users at all, and there were too many operational steps to stitch downloaded photos. We also noticed some problems with the guidance messages. There

- PhotoStitch version 1—arrange image order
- PhotoStitch version 1—preview of results
- PhotoStitch version 1—merge two images
- PhotoStitch version 1—merge and save

> was navigational guidance on the user interface, but it didn't make sense to users at all, and it was not really helpful. The old version also had a help function, but it didn't make sense to users either, so once they had problems during the operation, they had no way to get out of their difficulties. I found some words were difficult to understand for users who didn't have much computer experience. With the guidance messages, there were some technical words. Also the surface design—the geographical user interface design—was not intuitive to beginners. Some icons didn't communicate distinct functions to the users. The entire GUI looked a little bit complicated.

Great strides had been made to improve the design in the second version, both by adding the Stitch Assist mode in the camera interface and by improving the basic algorithm for the stitching process. Rikako continued to work with the engineers who were writing the software and the designers of the screen graphics, but she was able to control all of the interactive elements for the new version of PhotoStitch, including graphical expression, text and help messages, as well as all the error messages and dialog boxes.

The navigation was improved by changing to tabs, so that the user could move backward and forward at will, rather than being forced into a linear step-by-step process:

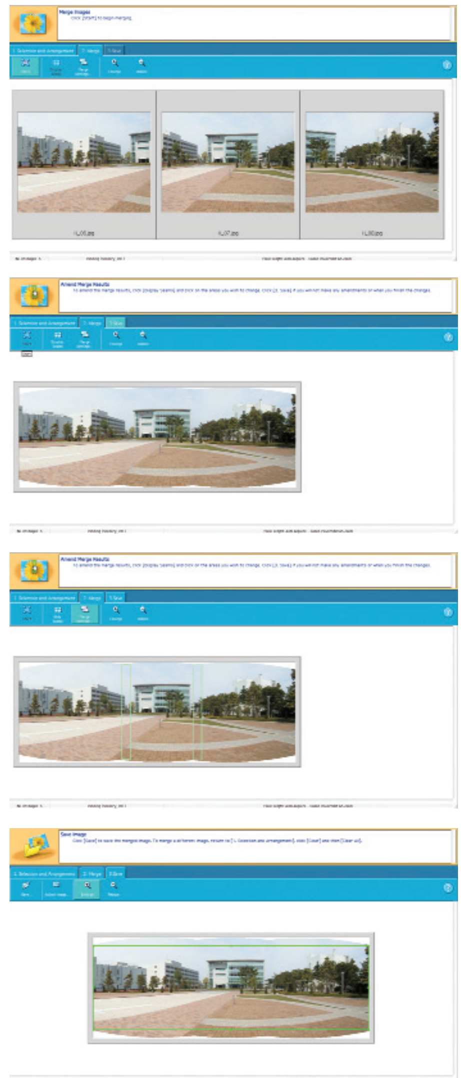
> I made a lot of changes from the old version. The biggest change was adopting the tab user interface instead of the wizard. We had lots of arguments about this. At the time we developed the software, it was not so common to adopt a tab user interface for step-by-step operations. We had to design each operational step to be very clear to users. We also had to revise navigational messages at the same time. With the new user interface, when you import all the photos and you are going to stitch, the second tab will be focused automatically, so you realize that the second step is available to choose. That's a part of the new user interface that we created. Also, the revised guidance messages inform you what you have to do next with plain words. So, if by any chance users get stuck, the guidance will help them overcome their problems.

Another navigational problem was in starting the application, as it was buried among a long list of utilities offered by Canon for digital cameras, so Rikako developed distinctive icons to represent the application and viewer. Importing folders and images was supported through an "Open" command, but during the observations Rikako noticed that people were trying to select them on the desktop, and drag and drop them into the open application, so she decided to support that alternative. She also noticed that when people had a lot of images to be stitched, they often lost track of the order, so she added file names and numbers to the images as they were presented and reduced the size of the thumbnails to give a better overview:

> I also made some suggestions about improving the scaling functions of the viewer. One of the concerns with the old viewer was difficulty with getting the overview of an entire image. I made questionnaires during the usability testing to figure out how many steps the user actually needed to take to get an overview.
>
> With the new version, you can go back whenever you want to the full montage. This idea also came from the usability testing. During the usability testing, we observed users trying to go back and forth. Especially when users got stuck, they tried to get back to the previous step, so we decided to support that movement.

The icons were also improved, with a combination of individual icons on the buttons under the tabs, plus an animated descriptive icon next to the text prompt box to reinforce the indication of the tab selection:

> We also modified each icon to be more intuitive to beginner users. With the old version, the icons didn't make sense to the beginners, so I decided to improve all the icons and add text labels. The first time I asked Osami Matsuda, our graphic designer, to revise the icons, he refused to put text labels together with the icons because it doesn't look cool. But after the usability testing, we realized again that displaying graphical icons is not enough to communicate distinct information to beginners, so in the end, we decided to display both icons and labels.
>
> We discussed every detail of the icons. For example, I really wanted these clickable icons to look like buttons, like square



PhotoStitch version 3—merge images
PhotoStitch version 3—amend merge results
PhotoStitch version 3—view seams
PhotoStitch version 3—save image

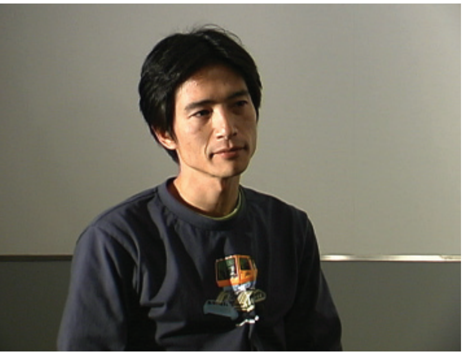
buttons or something like this, but the graphic designer didn't want to do that because it doesn't look cool. So, we fought for a while, and in the end I lost, but with the new version, when you put the cursor on a clickable icon, the icon changes its appearance into a button, and you can select it.

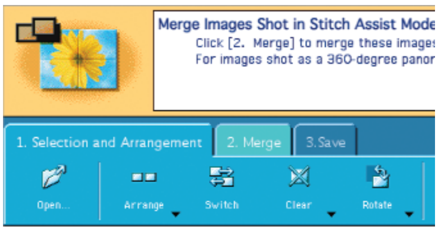
Linear green frames and borders are used to indicate active images, the overlapping areas, and to crop the merged image. Rikako was pleased that the engineers came up with these delicately designed indicators without prompting.

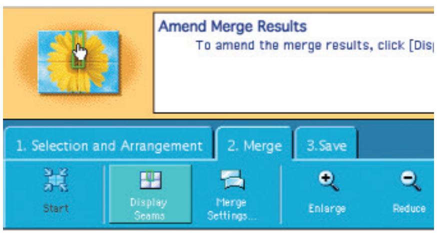
The success of the resultant version of PhotoStitch can be attributed to the combination of the Stitch Assist mode in the camera, integrated with the computer-based application:

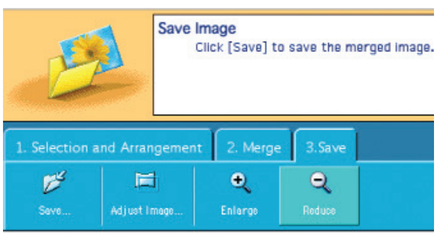
> If you use the Stitch Assist mode with a digital camera that Canon made, this process is going to be very easy, because the order and orientation of the photo is automatically set by the software, so users don't have to do that by themselves.
>
> If you want to make some changes with the automatic result, you can choose the parts that you want to modify. I made a change with the user interface too. Another problem with the old stitching software was lack of information on how to fix the automatic result, how to adjust the overlapping areas. A navigational message described the process, but with technical words like "drag" or "click" that don't make sense to beginners. We noticed that the text message was not enough to explain this process, because dragging one photo image onto another was not a common task for all users. This was the most challenging part to improve in terms of interaction design, so we decided to include animations to support the text messages. When we tested version 3, we were convinced that this kind of animation was helpful to novice users. The animation describes how to drag one photo onto another. Osami Matsuda designed the behavior. I just asked him to add animations to reinforce the text messages. He did a great job!

Masaaki Sakai, the senior general manager responsible for Canon's Corporate Design Center, says that "the role of design is to be the interpreter of technology." He declares that design can help people adopt new technology successfully. It seems surprising therefore that the skills of the people in his department









- Osami Matsuda, Canon GUI designer
- PhotoStitch version 3—merge icon
- PhotoStitch version 3—seams icon
- PhotoStitch version 3—file icon

were not engaged in the first two releases of the PhotoStitch software. Why was the algorithm applied to a standard Windows interface, in a foreign language, without reference to design and usability expertise? Perhaps the answer is typical of technology driven companies around the world, where so much effort is put into invention and patent applications, but there is an assumption that the inventors can design usable solutions themselves. In the earlier phases of adoption of the technology, this has often worked, as enthusiasts and professionals are patient and acquisitive. Once the adoption of the technology reaches the consumer phase, the skills of interaction designers and psychologists are much more likely to be essential for a successful outcome.

PhotoStitch version 3—completed three-image panorama

Printers for Digital Photography

"Drawer for drawings"

Photo
Hidetoyo Sasaki

COMPUTER PRINTERS HAVE a history of innocuous and self-effacing design, trying hard to fit into the background of the office environment as the undistinguished peripheral cousins of the computers that drive them. In the days of impact printers and needle printers, this attempt to hide was futile, as they generated enough noise to be agonizingly noticeable, and a lot of effort was put into designing sound reducing enclosures for them. Then the inkjet and laser technologies took over, and the designs could sit quietly on a shelf or in a corner—bland beige boxes performing their professional duties on demand.

Sometimes they needed attention from a person to clear a paper jam or replenish the ink supply, so design efforts were focused on making the diagnostics of maintenance comprehensible to people who had no training. Xerox pioneered this work in the early eighties, with a design strategy[11] for copier printers complete with color-coding for access, interactive touch screens, animated diagnostic diagrams, and simple sequential procedures for changing cartridges or finding paper jams. Since then, the replacement of consumables for printers by untrained

personnel has become commonplace, with diagnostics more on the PC display than the printer itself. Epson has done a particularly good job of designing these interactions.

As digital photography started to be adopted by consumers, printers took on new roles; now they were starting to live in the domestic environment, producing images that people cared about in a very intimate and personal way rather than just printing spreadsheets and pages of text. This raised the question of the design of the printers themselves; would they still be neutral designs, looking as if they had been borrowed from the office of yesterday, or would they start to evolve aesthetic characteristics that fit them to their new photographic tasks?

In 1998 Epson commissioned a conceptual design project called "Printables" to examine this question, to create designs that would be appropriate for printing photographs at home, and to build working prototypes of the ideas. The project was led by Naoto Fukasawa,[12] working with his team at IDEO Tokyo, in collaboration with seven of the Epson designers. The examples shown here demonstrate an aesthetic that fits the domestic environment of the consumer phase rather than the conventional appearance of printers as extensions of business computers.

- "Kinetic Deliverance"
- "Mysterious Thoughts"

Photos Hidetoyo Sasaki

"Drawer for Drawings," designed by Shoichi Ishizawa

The printer is built into a simple piece of wooden furniture, made of cherry veneer on particleboard. The top can be used as a general-purpose surface or to support a connected laptop. The paper is inserted into a slot on the top, and the print is delivered into a drawer, which can also be used to store supplies of paper for printing.

"Kinetic Deliverance," designed by Mugio Kawasaki

A delicate paper tray, made as a wire frame rolling on tiny wheels, supports the print as it emerges from the simple block form of the printer. The tray moves with the image, celebrating its arrival and enhancing the transition between start and finish.


EPSON

“Mysterious Thoughts,” designed by Hirokazu Yamano

Draped in a white cloth, the printer appears to be alive as the print head moves back and forth, gently disturbing the surface of the cloth, with the subtle sound of the mechanism enhancing the mystery. The image gradually appears from under the edge of the cloth, as if from a slow motion version of a magician’s handkerchief.

“Memory Developing,” designed by Sam Hecht[13]

This design harks back to the days of photographic development using liquid chemicals. The paper tray that receives the digital print is shaped like a traditional developing tray, lending a sentimental familiarity to the characteristics of the design.

“Imperfect Perfection,” designed by Naoto Fukasawa

In contrast to the predictions of the paperless office, digital printing has caused dramatic increases in the consumption of paper, mostly due to the fact that it is easy to print out a proof and then improve the on-screen original before trying another version. This printer is designed to support this pattern of use by mounting the printer on top of a recycling bin, so the user can inspect the proof and conveniently drop it in the bin if they are not satisfied with the result.

- “Memory Developing”
- “Imperfect Perfection”

Photos
Hidetoyo Sasaki

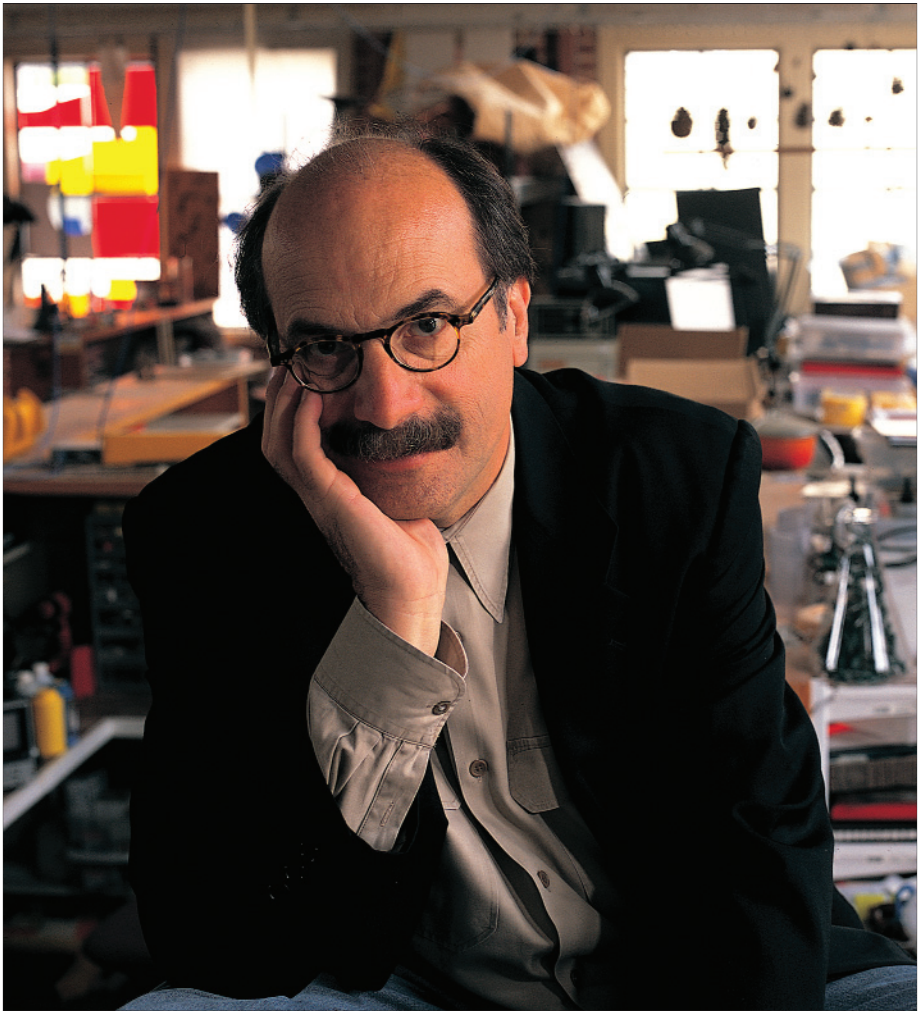


Photo Rod Searcey

David Kelley

Armed with an undergraduate degree in electrical engineering from Carnegie Mellon, David Kelley went to work at Boeing, in Seattle, where he was given the task of designing the lavatory "occupied" signs. He felt a little out of place in such a large engineering department, so when he discovered the product design program at Stanford University, he went there for his master's. The program was focused on innovation and understanding user needs, and what's more, they really built the products that they designed. It was a perfect fit for David, so much so that he has never really left, moving from student, to lecturer, and now tenured full professor in design. When he graduated in 1978, he started a consulting company with his friends from Stanford and soon became successful developing products, such as the first Apple mouse,[14] for Silicon Valley companies. David Kelley Design grew in experience and stature throughout the eighties, providing the ingenuity and technical expertise to bring many new product concepts to reality. In 1991 David founded IDEO, combining his engineering and innovation firm, which by now had offices in Chicago and Boston as well as Palo Alto, with the author's design firm, with offices in San Francisco and London, and Mike Nuttal's industrial design firm in Palo Alto. The combination of expertise on both the technical and human side of product development made IDEO immediately successful, and the company grew steadily in both size and influence. David handed the roles of president and CEO to Tim Brown in 2002 and now spends much of his time at Stanford, while staying connected to IDEO as chairman.

David Kelley

Interaction design started from two separate directions, with screen graphics for displays and separate input devices, but it got more interesting when the hardware and software came together in products. Then along came the information appliance, implying that technology would start to fit into our everyday lives, and when the Internet connected everything together, we found ourselves designing complete experiences.

David Kelley, 2004

Design Adopts Technology

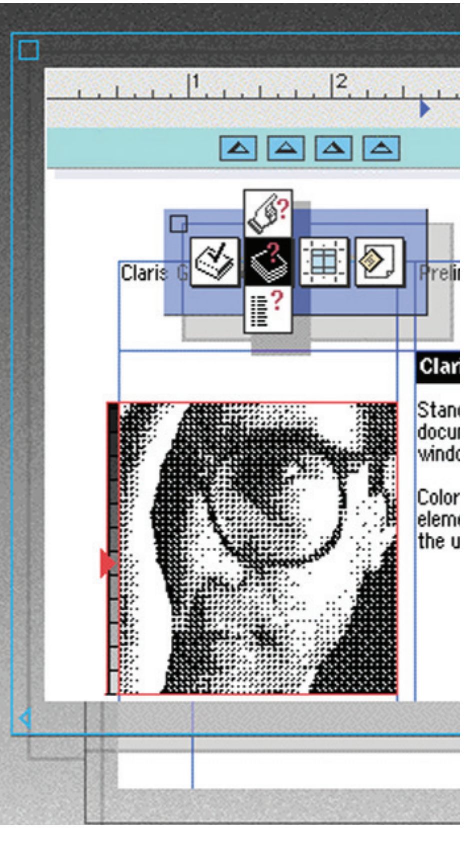
Roller Controller input device for young children for Philips

Photo Joe Watson

David Kelley has been involved with the development of interaction design as it is practiced at IDEO, and he has partnered with Terry Winograd[15] to teach classes in the subject at Stanford since 1995. Here is his summary of what it means to him.

> When I hear the words "designing interactions," I think of designing for people. When I was a student at Carnegie Mellon, I took Fortran. Fortran was a language for scientists, so you learned it and became an expert, but it really wasn't very interesting until you actually used it for something that resulted in work, something that made sense to someone. I see interaction design as the opposite of that. It's not something learned scientifically that you can then use for some technical purpose. Interaction design is using your technical knowledge in order to make it useful for people, to delight someone, to make someone get excited about the new technology they're using. I guess I would say that interaction design is making technology fit people.
>
> It started with screen graphics. Screen graphics are kind of like graphic design. Designers got all excited about the fact that they

could do things that moved. It used to be that you printed it and that was the way it was forever. Having something that would come alive and that you could actually change over time was exciting to them. It wasn't exciting to them, though, that the richness of the screen was so much worse than they were used to. They had all this paper and ink, and it was a real visceral feel to get the color just right, but then they took this big step backward to these pixilated graphics. It polarized the graphic design community into people who wanted to hold on to the richness of the inks and the paper, and people that said, "Hey, this is they way of the future!" It was more capable of doing interesting things, but it was less rich from a purely artist point of view.



Screen design for Claris, 1989, showing concepts for transparency and flying menus

Input devices were the first aspect of involvement in the design of interactive products for David Kelley Design.

We have designed a lot of input devices. To begin with, input devices were really exciting to us because they were new to the world; nobody had ever made these things in the same way before. The excitement was that we were designing things that were going to be very personal, like the bar of soap that you take a shower with in the morning is such a personal thing. We started out trying to see what kinds of behaviors were natural, coming up with an idea, and then testing it.

We originally thought that a trackball was the best idea. If you picture that the surface of the table has a ball coming out of it and you could spin the ball, that seemed like a much more natural thing than sliding this mouse thing on the table. It was interesting learning what you had to be mindful of. For example, "What were you dragging on the table? Were you dragging your palm on the table, or was your arm extended and it wasn't touching the table? Where did you get the friction in the system?" I had not been involved in anything quite that tactile before.

In the early days of designing Apple's mouse, we spent a lot of time trying to get it to be accurate, in a one-to-one relationship between the movement on the table and the movement on the screen; if you moved it one millimeter on the table, the cursor on the screen would move one millimeter. We were crazy about accuracy because we were engineers. At some point along the way, we had a breakthrough when we realized that your brain was in the loop. If you draw a circuit diagram, there's the mouse on the table, and then the wire goes through your brain and then goes into the computer, so you didn't have to be accurate at all! Your brain would send a signal to your hand to move it to the right place; you could hit a space on the screen very easily because you've got all this processing power of your brain to take out the little inconsistencies and inaccuracies. This meant that we didn't have to spend so much money on technical precision, which allowed the mouse to be quite a bit cheaper than we originally thought it could be. As long as it was consistent, it didn't have to be specifically one-to-one proportional to what was happening on the screen.

David and his team fulfilled their wish to develop a trackball, but in an unexpected way. They were asked to develop an input device for young children by a startup company in southern California. The company had a contract with Sesame Street to create a controller that would be easy for their audience of three-to five-year-old kids to use. The designers enjoyed playing with the kids of that age group and prototyping various options for them to try.

When we tested mice and joysticks with kids, we found out that they had trouble watching the screen while they moved the device. Whenever they wanted to do something they would look down at their hand, so they couldn't watch the screen and look; it just wasn't natural for them. We played with lots of things, but none of them worked. Someone had the idea that if we use big muscle groups instead of just little hand movements, that they would be more comfortable looking up, which actually turned out to be true. So we went to a really big trackball where they were using their arms and shoulders, and then they had no trouble looking at the screen while they were activating it.

A significant development was the integration of hardware and software into products in which the way people interact with the product could drive the whole solution.

In the early days of Silicon Valley, we had a lot of design of screens and input devices, and they were mostly general-purpose things. It got real exciting when the hardware and the software

Roller Controller—side
Roller Controller


ergo
year
month.day
datebook
Datebook: Audrey
Monday December 11, 2000
Audrey
December 2000
S M T W Th F S
1:00p
pick up Dry-Cleaning
2:00p
Doc. Appt.
3:00p
Jane's swimming practice
4:00p
5:00p
John's Soccer Match
6:00p
7:00p
go to today
schedule
list
December 30, 2000 7:06pm
3com

came together and we were really making products. Now you're talking about special-purpose things where you can design the controls, you don't have to use a general-purpose solution. This allowed things like the Palm Pilot, cell phones, and so forth to be quite exciting from a design point of view. You can make the displays of a size that were enough to do interesting graphics but were small enough to fit in your pocket. Things became more personal as they became more productlike, opening up a whole new explosion of possible products for us to design. Then behaviors really came into it.

Before, we had the approach of, "Word processing exists, spreadsheets exist, we're going to teach you that. You're going to conform to the way we, the software designers, expect it to happen."

As you got to products, we went the other way. We were looking at "What do people want to do? What's people's behavior like around telephones or their phone lists or other things that they normally use now?" so we could design the product to meet the behavior and the need rather than expecting that we're going to teach them a software program.

Then along came the information appliance, the idea being that technology-enabled devices would start to fit into the everyday lives of people in a similar way to appliances in the kitchen, like a toaster or a blender. This is the opposite concept to the general-purpose computer, in that the appliance is designed for a specific purpose and used only in that context. There are examples of the integration of technology in appliances, for example the toaster that uses "fuzzy logic" to brown more consistently. A digital camera can be thought of as an information appliance, in that it contains electronics to record the image as well as for automatic exposure and focusing, but people still think of it as a camera rather than a computer.

■ 3Com Audrey Internet Appliance

Photo Courtesy of IDEO

A heralded example of an information appliance that IDEO designed was 3Com's Audrey Internet Appliance. It was an Internet-access computer designed to be used in a communal area of the home, perhaps the kitchen, and the navigation was simplified to a single control akin to a radio tuning knob, with a push function to select. The simple controls, personalization for individual family members, and an "always on" approach to the

Internet connectivity, made it convenient to gain access to weather, traffic, sports, and a unified family calendar. Audrey periodically updated its information in the background, so it was available the moment you woke it up, without having to wait for the PC's long boot-up-and-connect process. It was launched with great fanfare at Comdex, just in time for the end of the dot-com boom, but was too expensive to survive the downturn.

David Kelley believes that the era of the information appliance is yet to come; he is not yet confident in the future of the idea, but he adds:

> The thing that will happen, I believe, is that the products will be smart enough, or integrated enough, that they will be able to react to us; that the product will know what's going on with us and will be able to do the right thing. I think that's different from the toaster and the blender; the toaster and the blender sit there, not knowing how we are, waiting for commands. I think information appliances, highly technological appliances, will know we're there and anticipate what we want just from the way we act.
>
> I think an automobile is a very good example of technology integrating into our lives to the point where we're not aware of it; it's seamless, and it works very well. The average person doesn't realize how many computers they are actually commanding when they drive a car. There are already examples where computing power can save your life. An airbag is a great example of that. It's up to the computer to know when to open that airbag, and your life is threatened if it doesn't, and we somehow accept that. But I don't think people sit around thinking, "Oh, I hope the computer works when the airbag . . ." The electronics, the mechanics, the fabric, the thermodynamics, and the physics are all kind of wrapped up together. It's a very sophisticated thing, but anybody who's been in enough of a fender bender that the bag goes off, just sees it as an ordinary machine that works quite well.

IDEO was steered steadily through the Internet boom, continuing with the same kinds of innovative development of products and places as before, rather than riding the wave of design for e-business. This proved to be a wise choice when the crash came, and the design organizations that had focused on the

Internet started to fade away. It was hard to find a U-Haul trailer to rent in San Francisco—people were leaving town in droves.

By 2004 Silicon Valley was returning to normal, and new people were starting to arrive, but at least the cost of living had returned from the dizzying heights of the boom to something that was merely outrageous. David remembers:

> Living in the valley through the Internet boom and bust was quite interesting, mostly for my students who became CEOs of companies overnight and then came back to chopping carrots in restaurants!
>
> The Internet is a really important technology and business tool. My belief is that, from a business point of view, it makes the market more perfect. You can find the people who want the products and services more easily. From a design point of view, it allows us to explain the features of our products more fully. Because the information is more complete, you can do more subtle things in products, and people have time to understand them. The Internet is disconnected from real time. If I'm interested in some special tulip bulb that only comes from Holland, I have a chance of being able to find out about it over the Internet.
>
> In my own life, I'm particularly interested in old vintage cars and trying to find them. It used to be, if you were looking for the right Alpha-Romeo, you had to stumble across it or hear about it from someone. Perhaps you would hear that it's in a garage in Argentina, and then you had to figure out if it's worth going to Argentina or not. Today, with the Internet, you know where all of those cars are, what condition they are in, and you can see pictures and so forth.

When you go to the hospital, you expect an unenjoyable experience. If it is a visit to the emergency room, it is traumatic by definition, but even for a routine checkup, there is uncertainty and confusion to contend with. David points to the design of the hospital experience as an example of products, environments, information display, and services all playing a part:

> You go into a hospital and nobody is around. There's a desk where there's supposed to be a person, and there's no one there, or

Place of Waiting

Urgency Cards
Color-coded and time-stamped, the cards allow patients to flag their symptoms and keep track of the time spent waiting.

Self Diagnostic Posters
A quick glance allows the patient to understand the urgency of his symptoms and, consequently, his place in the emergency room queue.

Patient Journey Punch Card
Carried by the patient and updated by staff, the punch card allows for a simple, graphic way for the patient to know where they've been and what is coming next.

Simple Self Treatment
Vending machines dispense aspirin, bandages, etc. and allow patients to give themselves simple remedies to alleviate discomfort while waiting.

Self Registration
Replacing unmanned registration windows with a mobile cart allows registration to occur based on place in the emergency room queue. This can be replaced with self registration.

Managed Views
Live feed into the Emergency Room allows for the waiting patient to get a sense of the degree of activity. Alternately, shrouded glimpses of activity with status thermometer can give a more controlled vision of activity levels.

Patient Hotline
Patient Hotline allows for instant connection to staff who can act as a systems translator. All mediated staff interactions happen through the phone because it is a familiar interface.

Valet Parking
This service allows for easy arrival when it really matters.

Local Hospital Tour
Tours while the patient is healthy can inform future patients of the kind of treatment and facilities they will have available at the hospital. It is an important step in allowing the patient to understand the appropriate way to enter the facility.

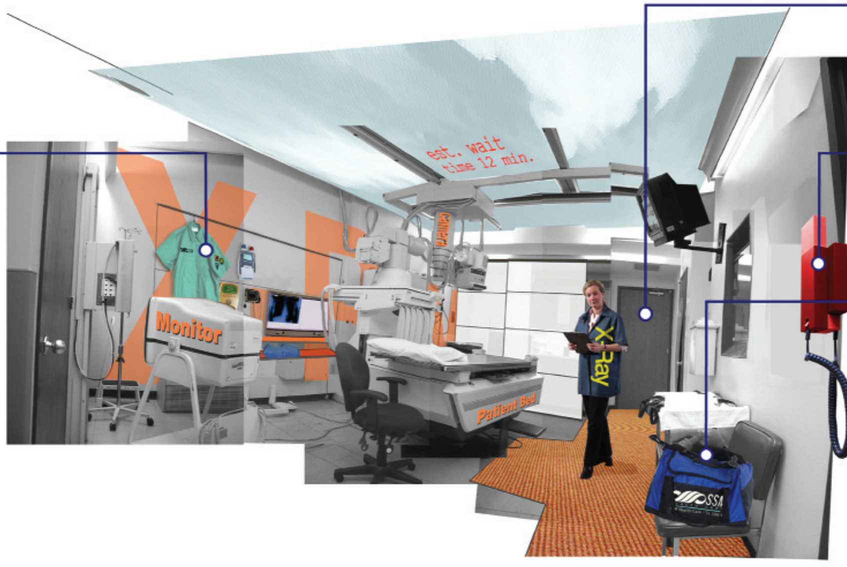
Frequent Flyer Cards
Return patients' needs, preferences and rewards are acknowledged automatically upon registration.

Place of Practice

Patient Journey Punch Card
Carried with the patient and updated by staff allows for a simple graphic way for the patient to know where they've been and what is coming next.

Patient Garment with Information Badges
Additive badges allows the staff to get a brief history of the patient visually.

Modal Diversion
Patient has automatic control over simple things in their environment everything from important information to personal soundtrack accompanies the patient on their journey.



Information Transfer Doorway
A place for a simple log of patient interactions, patient photos and useful tools facilitate the doorway as a translation zone. Eventually information transfer can happen wirelessly.

Patient Hotline
Patient Hotline allows for instant connection to staff who can act as a systems translator. All mediated staff interactions happen through the phone because it is a familiar interface.

Patient Manuals
Customized patient manuals which track the patient journey based on their specific illness can allow for the possibility of in depth information at any point.

Staff Cross-Training
Anyone coming in contact with the patient has the minimum knowledge necessary to answer simple questions and help in basic matters.

there's a big sign that says "triage nurse," and you don't know what "triage" means. You are thinking, "I just want someone to talk to, and to find out how I should be treated." It would help to just know where you are in the queue. You're the eighth person who's been in there, but are you going to go next? There's a tape running through your head saying, "I wonder what's going to happen next?" The more you can answer that question by giving people an expectation of what's going to happen, the better they feel about the experience.

David believes that the number of instances where it is important to design the complete experience is growing and represents a significant shift for design education and practice. He started his design consulting business in Silicon Valley in 1978, when the engineers and entrepreneurs were so focused on chips, printed circuits, and software that they used the word "packaging" to describe the physical objects that enclosed the technology. As products became more important, making those physical enclosures work well emerged as a key to success; mechanical and production engineering contributions were recognized.

David saw the opportunity to integrate the human design disciplines of industrial design, human factors, and interaction design with the technical disciplines of mechanical, electrical, and production engineering. He formed IDEO to combine experts in those areas into interdisciplinary teams for product development. This combination proved very successful for the development of products that included sophisticated user interfaces, and was key to the success that IDEO enjoys.

■ DePaul Health Center for SSM Health Care 2001 patient care delivery model

The Internet expanded with the amazing surge of the dot.com boom, and everything was suddenly connected to everything else. Technology was being employed to enable services, so businesses had to design the experiences in a more holistic way. They could no longer simply produce products and train employees to offer the related services. David saw this as the next significant change for design:

I would say the most interesting thing that's happened for me personally as a designer over the last twenty-five years is the move away from the design of objects. I was a kid who grew up and took apart radios and cars and put them back together, and it was kind of physical. And then, of course, with the advent of this kind of electronic stuff, as an electrical engineer I got into that, but that was still physical, although you couldn't see the electrons moving. I was always focused on a thing, not so focused on the people who use it, though accepting that they were a necessary evil.

The movement away from designing the object itself to designing the experience, which everybody talks about, is really profound in my opinion. Designing experiences and services takes you away from the nuts and bolts, for a while anyway, and gets you into understanding people and their needs. I personally believe that technologists and engineers are even more motivated by doing something that has value to people than they are by technology. They think, "We're going to design a service or an experience that people really want and need, and their lives are going to be changed by it." This motivates them to do even better work, to push the technology even further, to work even harder, or knock themselves out to be more innovative, because they have more empathy for the person they are designing for.

Teaching designers to be integrators

At the time of writing in 2004, David Kelley was establishing a new design school at Stanford, dubbed the "d" school. Its goal is to provide an interdisciplinary program for the development of products, experiences, services, and spaces, with design as the core skill. There will be a kernel of resident design faculty, with visiting professors from other departments in the university, including business, social science, and technical areas such as computer science. David describes the philosophy of the new school:

We used to design objects, and we had machine shops to prototype them. We still need machine shops, but if you're going to design experiences and services, you have to have new prototyping tools to explain to people what it would be like with this new service, or how different it would be doing something if we had this new technology. Now we're using more storytelling. My students are taking improv classes and acting things out. They all know how to use some kind of quick and dirty video process so they can tell a story; they are becoming cartoonists so that they can do quick storyboards. Design is moving from understanding technology and building devices toward understanding people and telling stories. The challenge for teaching design is about designing experiences instead of just objects, and prototyping with video and storyboards, and acting things out.

Another issue for teaching design in university now is the difference between depth and breadth. Designers are inherently people who are good at breadth. They have broad interests, and they apply their skills broadly. That is why we teach breadth. But there is also the issue of depth. What are they going to be good at? The answer for us at Stanford is that they are going to be good at the process of how you bring all the experts together.

They are the experts in the methodology, so they hold the other disciplines together. Because of the complexities of the projects, you need a whole room full of experts in order to get anything interesting in the way of innovation. We decided that designers are going to be the people who integrate the technology and the process and are the glue that holds these experts together. They are empathetic to other disciplines, which translates to having breadth.

In 2004 the Apple iPod was a dramatic success story of the adoption of a technology in the consumer phase, including the integration of interactive product, computer application, and Internet-based service. Next, Paul Mercer tells the story of the iPod development.

Segment of David Kelley's mind map for the "d-school"


menu

iPod

The original iPod

Photo
Courtesy of Apple

The iPod from Apple shows how a physical product with a screen and controls can be seamlessly integrated with a computer-based application and an Internet-based service. When the first iPod was launched, the beautiful design was captivating, but the integration with iTunes really made the interactivity irresistible. It was so convenient to be able to use your personal computer to download music from the Internet, or copy it from your CD collection, and then sort your play list before you copied it onto the iPod. The first version had separate switches as controls, with wonderful tactile feedback, so the feel of using the product was delightful. Reliability and cost advantages made the designers change over to a touchpad for input in the versions that followed, in spite of the loss in tactile quality.

When Apple added their online music store, the design of the service was complete, and the success of all of the components in the system accelerated. People loved the object, they valued the connectivity between portable device and PC, and they found it hard to resist adding music to their library through the store, doing so without the guilt associated with Internet piracy. Other examples of "designing services" are covered in more depth in chapter six.

Paul Mercer's[16] team, at software startup Pixo, was responsible for the implementation of the user interface software for the iPod. In the interview that follows he tells the story of the development, as well as talking about the adoption of mobile devices and the changing nature of the design process.

Photo Author

Paul Mercer

"I built my own Timex Sinclair ZX81 computer in high school; it was $100 if you built it yourself from a kit rather than paying the full $150 to buy it preassembled. I'm a software guy, but this is where I broke out the soldering iron and built it myself." Paul Mercer showed an amazing aptitude for writing code when he was still very young and went on to hone his skill at Syracuse University, before being lured out to the West Coast by Apple at the age of twenty in 1987. Within a few months he found himself working on the next generation of Finder, the core of the Macintosh user experience, maintaining system 6 Finder while the underpinnings of the Finder (NuFinder) were rewritten in the then newfangled C++. In 1991 he put together a team to explore concepts for handheld Macintoshes and the software to go with them. Apple CEO John Sculley was impressed by working prototypes. By that time, though, he was committed to Newton and asked Paul to help move the Newton design from research to product. He built the core software framework for Newton and along the way came up with the little "poof cloud" that evaporates when you erase something on the Newton screen. In 1994 Paul left Apple to start his own company, called Pixo, as he wanted to develop the building blocks for the next generation of devices and user interfaces. Pixo grew to employ 150 people, with clients including Nokia and Samsung, in mobile phones, and Apple, for the implementation of the iPod. After eight years, Paul saw a new opportunity as more and more devices became wireless. He has founded a new company, called Iventor, to create enabling structures for ubiquitous computing.


iPod
Music
Extras
Settings
Shuffle Songs
Backlight
MENU

Look at a culture like Sony, where they spent decades refining their capabilities in industrial design and miniaturization, and they really took that to its zenith and were world masters at it. But now as the world has changed to include display devices in just about every product you can imagine, we're moving into an era where software is the primary differentiator of products, dictating the usefulness and functionality of a device. Look at something like the iPod, which has a very simple interface, and compare it to the myriad of other music players that are out there that don't measure up, that are very difficult to use even though we're talking about a screen of very few pixels.

Paul Mercer, 2004

iPod 2

Photo
Courtesy of Apple

Pixo

Paul Mercer was always passionate about building portable electronic devices. He translated his early experience on the Apple Finder into an effort to develop handheld Macintoshes. He developed working prototypes that were just a little larger than the production Newton that shipped two years later but used the well-understood and low-cost Macintosh technology. His idea was to leverage the Macintosh installed base, using a technology that had already been enthusiastically adopted and even had fanatical evangelists supporting it. He added networking capability to share files with the Macintosh on your desk and a notepad application to allow you to bring your files with you, view them and annotate them. He spent a lot of time trying to promote the concept inside Apple:

> I think Mac users will remember, during the nineties there was very little going on for the Macintosh. There were just minor


iPod
Playlists
Browse
Extras
Settings
Backlight
MENU

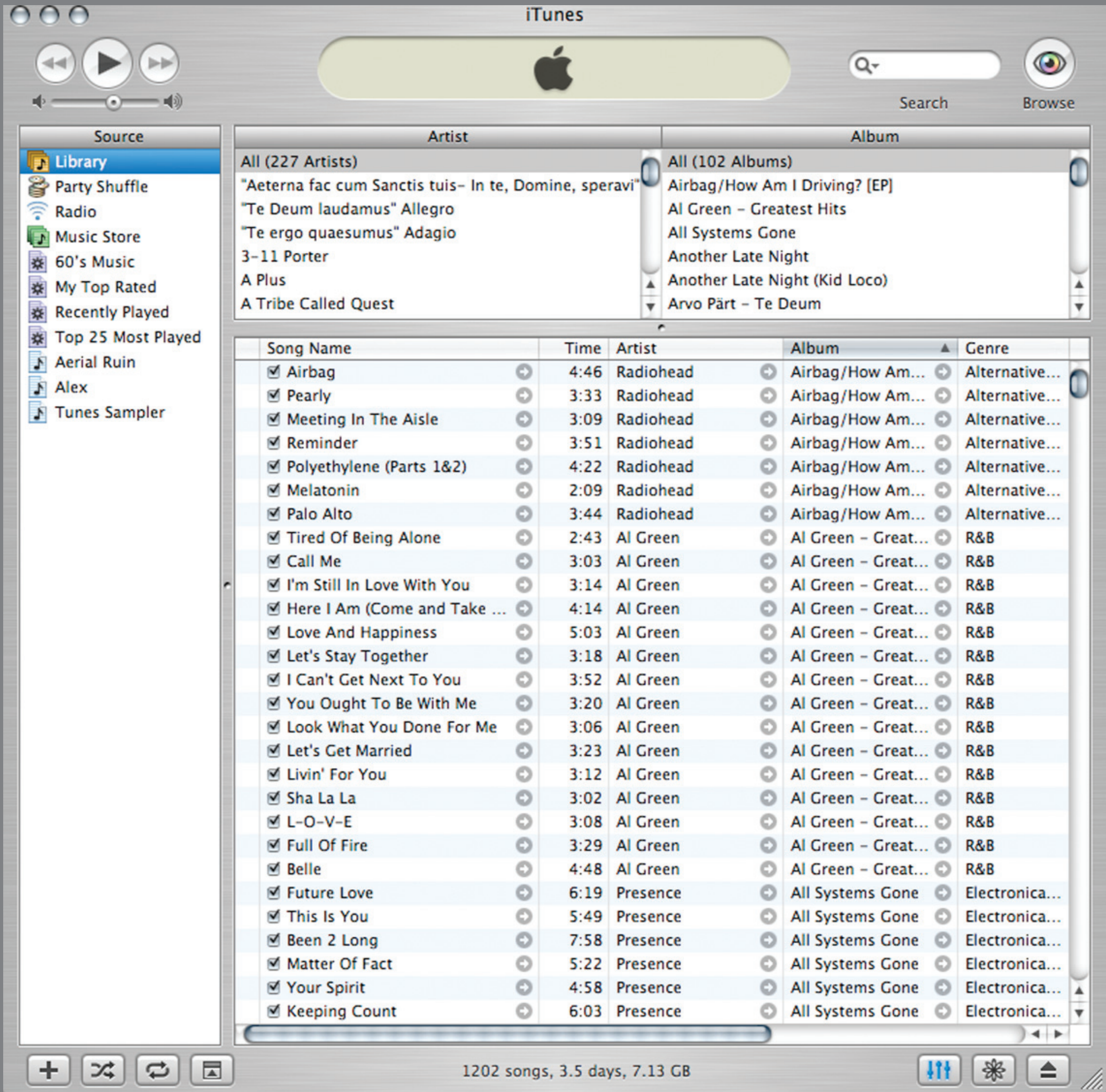
> enhancements. The fundamental work of reinventing the core OS and the underpinnings was really floundering, and Apple spent a lot of money and resources trying to get out of that mess. I had a small role in that, and after Newton, one of my jobs was to help move the Macintosh architecture forward. This was difficult to do at a low level, because the Macintosh was so chaotically built. At the same time, I was still in love with the idea of a handheld Macintosh that ran on batteries. I thought it would be a very good move for Apple, technologically and marketwise. So with every regime change at Apple, I would bring out my working prototypes of the handheld and try to get Apple interested in building it.
>
> There were several eras at Apple during which there was a lot of interest in building devices, so the hope of building little mini-Macintoshes lived on for a couple of more years before I finally gave up on the idea. Eventually I realized that the Macintosh technology, which was supposed to be a stepping-stone and accelerant to building devices, was getting old. The processor world was moving away from the Motorola MC68000 architecture, which the Macintosh was built on, toward the RISC-based architectures. So with that I realized, "Hey, that Macintosh magic, that capability at Apple is no longer necessarily a differentiating ingredient."
>
> In 1994 I left Apple and started up a company, called Pixo. My goal was to create the building blocks for the next generation of devices and to build user interfaces for those devices.

■ iPod Minis

Photo Courtesy of Apple

The idea behind Pixo was to write modules of code in C++ that would enable flexible user interfaces and functionality. In the mid nineties, it was a major achievement for a device maker to go from programming devices in assembly language to using C and eventually C++. They were typically working with very poor tools for software, as they came from a culture of building devices based on hardware capabilities, not building software and user experiences. Paul describes the offering that made his new company successful:

> Pixo was formed with the idea of creating fundamental building blocks at a low level, to enable more sophisticated software on portable devices. This meant that the building blocks had to be very low-cost, had to be easy to use, and had to be fairly powerful. On the one hand, we had to build that technology, and then in parallel we


iTunes
Search
Browse
Source
Library
Party Shuffle
Radio
Music Store
60's Music
My Top Rated
Recently Played
Top 25 Most Played
Aerial Ruin
Alex
Tunes Sampler
Artist
All (227 Artists)
"Aeterna fac cum Sanctis tuis- In te, Domine, speravi"
"Te Deum laudamus" Allegro
"Te ergo quaesumus" Adagio
3-11 Porter
A Plus
A Tribe Called Quest
Album
All (102 Albums)
Airbag/How Am I Driving? [EP]
Al Green - Greatest Hits
All Systems Gone
Another Late Night
Another Late Night (Kid Loco)
Arvo Pärt - Te Deum
Song Name
Time
Artist
Album
Genre
Airbag 4:46 Radiohead Airbag/How Am... Alternative...
Pearly 3:33 Radiohead Airbag/How Am... Alternative...
Meeting In The Aisle 3:09 Radiohead Airbag/How Am... Alternative...
Reminder 3:51 Radiohead Airbag/How Am... Alternative...
Polyethylene (Parts 1&2) 4:22 Radiohead Airbag/How Am... Alternative...
Melatonin 2:09 Radiohead Airbag/How Am... Alternative...
Palo Alto 3:44 Radiohead Airbag/How Am... Alternative...
Tired Of Being Alone 2:43 Al Green Al Green - Great... R&B
Call Me 3:03 Al Green Al Green - Great... R&B
I'm Still In Love With You 3:14 Al Green Al Green - Great... R&B
Here I Am (Come and Take ... 4:14 Al Green Al Green - Great... R&B
Love And Happiness 5:03 Al Green Al Green - Great... R&B
Let's Stay Together 3:18 Al Green Al Green - Great... R&B
I Can't Get Next To You 3:52 Al Green Al Green - Great... R&B
You Ought To Be With Me 3:20 Al Green Al Green - Great... R&B
Look What You Done For Me 3:06 Al Green Al Green - Great... R&B
Let's Get Married 3:23 Al Green Al Green - Great... R&B
Livin' For You 3:12 Al Green Al Green - Great... R&B
Sha La La 3:02 Al Green Al Green - Great... R&B
L-O-V-E 3:08 Al Green Al Green - Great... R&B
Full Of Fire 3:29 Al Green Al Green - Great... R&B
Belle 4:48 Al Green Al Green - Great... R&B
Future Love 6:19 Presence All Systems Gone Electronica...
This Is You 5:49 Presence All Systems Gone Electronica...
Been 2 Long 7:58 Presence All Systems Gone Electronica...
Matter Of Fact 5:22 Presence All Systems Gone Electronica...
Your Spirit 4:58 Presence All Systems Gone Electronica...
Keeping Count 6:03 Presence All Systems Gone Electronica...
1202 songs, 3.5 days, 7.13 GB

had to build examples of what the user experience might be. As the technology improved throughout the second half of the nineties, we steadily added things like color, audio support, and scripting capabilities.

At its heart, the idea was to democratize the level of capability that the Macintosh gave developers and to bring it to the world of devices. It was very easy for our customers to look at Pixo and say, "Ah, you have these great building blocks, and we can pick and choose, save some time, and save some money in building our product."

That really wasn't what Pixo was about. We created the building blocks to help build products cheaper and faster, but our ultimate aim was to help people build better products, such as the iPod.

iPod and iTunes

Apple was one of Paul's customers, coming to Pixo for help in creating the system software toolkit for iPod and implementing the user interface to Apple's specification. Paul has great respect for the culture and leadership at Apple:

■ iTunes

Photo
Author's screen capture

> I think if you look at the culture of Apple, it's a company that's had twenty-five years of building personal products, and in the last few years they've had a very astute individual at the top who can really dictate in many ways, or arbitrate, what's good and not good. I think that culture and management capability at Apple makes it unique in this industry.
>
> The answer to the question "Why is the iPod so great compared to the competition?" is really the same answer as to why the Macintosh is still different from Windows or anything else before it. I think it's really this culture of being able to build good products.
>
> Apple started working on music very slowly and built the business up deliberately over a period of years. They started by acquiring the technology. In September 2000 they acquired a company called SoundJam to build iTunes, and then turned that into the iTunes franchise, starting first on the Macintosh. The ability to generate music, to let the user rip that music, that's the root of the

application—you've got to have it. Apple invested in iTunes, built that up and improved it over the years.

In November 2001 Apple shipped the first device that synchronized with iTunes. The fact that the two were deeply integrated together, and the fact that the integration was through a high-speed bus like FireWire was another necessary ingredient of being able to get the content and then move it to the device. If it took ten times longer to move, the iPod would not have been a success in 2001.

Then in 2003 Apple launched the iTunes music store. This again was something that did not exist before at a consumer level that was mass-marketable. There were MP3 players before Apple shipped the first iPod, and there were music stores before the iTunes music store, but Apple was the first to make it mass-market capable. They applied the ineffable "Apple magic" to make that possible, whatever that is. If you could put it onto five bullet points on a piece of paper, you could probably make a fortune explaining it to people.

It took Apple a couple of years before iTunes for Windows was actually developed and shipped; they took their time in doing it. They were practically moving in slow motion if you look at it with hindsight, but the industry has not been able to match it. Now they are moving into color displays, and eventually they'll move into video, as Moore's law continues giving us its bounty.

Paul is very interested in the subtleties that make up a culture in an organization to allow creative design. He is a developer of what he calls "building blocks," or toolkits, that he intends should be used for the design of solutions that are easy to adopt, but he is aware that culture is based on a subtle synthesis of attributes, rather than just the use of the tools. It is not just what you do, but also how you do it:

Just because you have the building blocks, doesn't mean you'll build great products. You witness what Microsoft has done. They have the fastest computers in the world running their platform; they've got a scientific staff, a research staff that's really unparalleled today. Many of the great computer science and user experience researchers of the last twenty years are now working for Microsoft. And yet, what is missing there that does not allow them to be the first to build iPod and iTunes and the next generation of products?

iPod Shuffle

What is missing there? I think it goes back to the culture. Even at Pixo, where we were a very small company, where we didn't get beyond 150 people, we had that culture. If you don't set that tone in the beginning, it's very hard to alter those forces.

There have been some reports in the press crediting Pixo for the design of the iPod, but Paul is very clear that he only provided the building blocks; he did not invent or design the iPod. He does not want to take credit for what Apple accomplished, and waxes enthusiastic about the phenomenal job that they have done in carrying forward the promise of devices, but he is still puzzled by the competitive advantage that Apple has maintained:

Now, keep in mind, the iPod is very simple-minded, in terms of at least what the device does. It's very smooth in what it does, but the screen is low-resolution, and it really doesn't do much other than let you navigate your music. That tells you two things. It tells you first that the simplification that went into the design was very well thought through, and second that the capability to build it is not commoditized.

The fact that nobody has been able to build this thing, to duplicate the capabilities, seems at first sight surprising. It means that the building blocks may be difficult to come by, and that the design sense, to create a simple and easy-to-adopt solution, does not exist in most of these product development organizations worldwide.

It is very curious, because the iPod has been out there for about three years now, essentially unchanged in terms of its core software design. The dozens of products that have come out since have not been able to catch up to it. People always want to beat the champ—why are you in this business if you don't want to dethrone the champ? Nobody's really done that. It's very curious that over the years and all of the money that Apple has made, that the competition hasn't figured this out.

iPod Nano

The Interaction Design Challenge

Paul has recently started a new company, Iventor, with a mission to democratize the design of better user interfaces for portable devices. He wants to create tools for developers that will enable better interactive experiences for more people. He sees the shortage of good interaction designers as a worldwide problem. There is tremendous pressure on the consumer side, as people welcome devices that are well enough designed to be easy to adopt, but the culture and skills of design are the limiting factor. There is a real business need to solve some of these user interface problems:

> The world has changed a lot in the last few years. Everything is connected now; everything is wireless. We have P2P technologies; we have very high-speed wireless now, practically to the level of ubiquity. I think we're ready for another generation of technology and user interfaces for devices. I think in the years to come you're going to see portable devices take off in many more directions and be much more pleasant to use.
>
> Today, when you get a new cell phone, you might love the industrial design, you might love the radio on it, and you might love the battery life, but I don't know anybody who loves the software on their cell phone; it just does not exist. Getting something into your calendar, downloading a number from your address book is a nontrivial exercise in any cell phone today.
>
> One of the great frustrations of working in the device world, particularly in the wireless device world, is that the ability of a designer to develop and launch a good product is not fluid right now. In the PC industry, anybody with a good idea and some talent can go and build something, and put it up on the Web or launch it as shareware. That kind of opportunity does not yet exist in the mobile world, because the cell phone industry is built around state-sanctioned monopolies of the airwaves, which has resulted in each carrier building controlled economies around their network. This structure does not award innovation, particularly software innovation, and consequently, you have seen very stilted development of software on phone handsets. This will change soon since the software stack

> and the usability challenges are getting too difficult for the handset industry to address. You will see companies such as Apple and Microsoft, with deep system software and user interface experience, come to the fore.
>
> A core piece of technology that we're bringing to the puzzle is the idea of much more robust, managed code, or virtual machine-based environments. You don't need to write code that is specific to a particular piece of hardware any more. We've got enough horsepower now, even in the simplest of devices, to be able to develop in the abstract, driven by user needs rather than hardware capabilities. You will have much better building blocks and a better vocabulary for building devices, giving a new level of prototyping and development speed, and a level of robustness that does not yet exist in the industry, allowing very cool applications.

If Paul Mercer's Iventor is able to create a set of tools that will enable a level of interaction design that is only offered by a few elite products such as the iPod today, we may see a surge in the adoption of portable devices, starting with the already ubiquitous cell phone, and expanding into other devices that we can carry around, whether they are cameras, music players, or communicators.

Will you want a single device that can do everything—a cell phone that includes a PDA, camera, music player, video player, and communicator? Or will you prefer several devices with more limited functions that you can wear like jewelry or carry one or two at a time? Both options will be out there during the period of confusion that comes with the adoption of new technologies, until consumers vote with their pocketbooks.