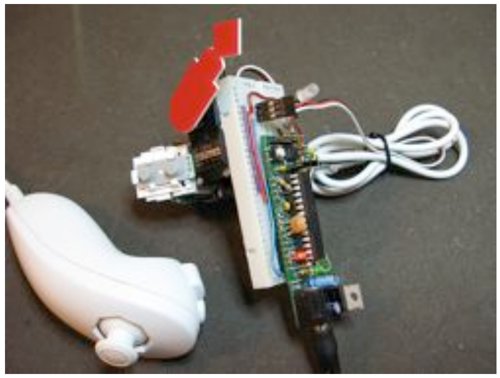
Sketching Smart Things



Mike Kuniavsky
UC Berkeley School of Information
March 6, 2008

Hi. Thank you, Liz and the School of Information for inviting me.

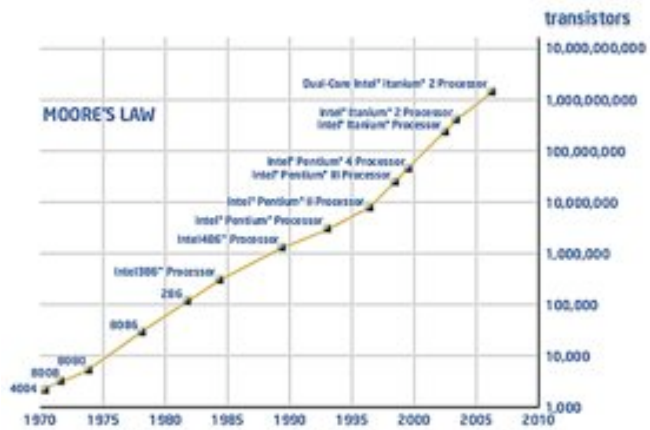
I'm dividing tonight's talk roughly in half. In the first half, I'm going to talk about some of the high level ideas we're developing and working with, in the second I'm going to talk about some specific techniques we're using and give you some examples of how we've used them in the last year.

I’d like to start by telling you a bit about who I am.

I’m a user experience researcher and designer. I spent a little more than 10 years doing design and research for the web. 4 years ago (almost to the day), I switched to thinking about devices and ubiuitous computing.

As part of this practice, I wrote a book. It’s a kind of cookbook covering a wide range of user research techniques that I understand is used as a textbook here at Berkeley. Thank you very much for that. I also co-founded a San Francisco design company called Adaptive Path.

In 2006 I started a company called ThingM with Tod Kurt. We design and manufacture ubiquitous computing consumer products. Well, in theory. We’re still in early startup mode.



Source: Intel

Let me start by talking a bit about ubiquitous computing and why I think it's really important. The term was coined by Mark Weiser in the early 1990s, but I don't think true ubiquitous computing was practical until just recently. Let's start with Moore's Law.

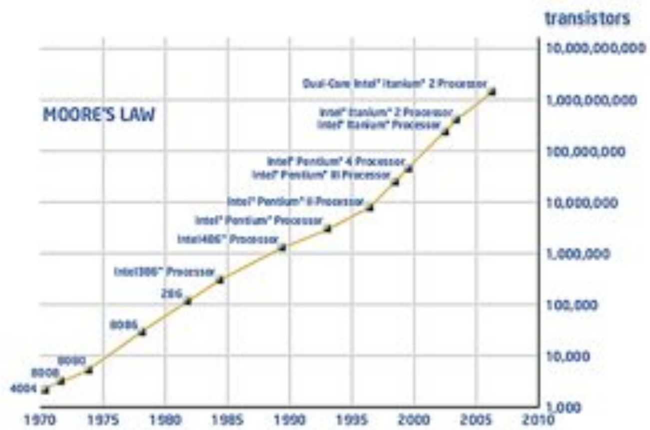
People typically read this chart as a trend focusing on the number of transistors.

What's implicit in this trend, however, is that this is happening within the context of a marketplace.

This is not just the theoretically largest number of transistors that's possible to put on a on a CPU die. It's the number of transistors that can be sold at a specific price point.

Data source: http://www.islandnet.com/~kpolsson/micropro/

The prices of new CPUs has stayed roughly the same over the last 25 years, generally between $500 and $1000 at the time of introduction.



Source: Intel

Thus, another way that to read this chart is that as transistor density increases, the price of older technology proportionally DECREASES.

I'd like to draw your attention to the middle of the chart. The 486 correspond roughly to the beginning of the modern, internet-connected computer and is a very powerful device. It was the state of the art when the promise of ubiquitous computing was first identified.

1989: $1500*

33MHz, 20MIPS

2008: $0.50

20MHz, 20MIPS

* CPI Adjusted to 2008, original price: $900

This price drop is why I believe that ubicomp has just become a practical reality.

I shifted from the Web to ubicomp because I wanted to answer the question of what happens to devices, to design, to communication and to society when devices with this kind of power become a commodity.

Source: Flickr, jurvetson

I believe that ubiquitous computing holds amazing promise for making the world a better, happier and more interesting place. Tod Kurt and I founded ThingM because we want to reinvent everyday objects in light of the capabilities of these new technologies.

However, ubiquitous computing is today where the Web was in 1992. We just figured out how to make Web pages not totally suck after 15 years. The capabilities of our technology outpace society’s ability to efficiently process it, so we can’t go out and look at best practices, because there are none.

To wrap our brain around what this means for design, we’ve been working with a handful of broad conceptal frameworks to guide us and then have been trying to distill those into design practices.

INFORMATION PROCESSING IS A MATERIAL

Source: Flickr, Kate_A

1. INFORMATION PROCESSING IS A MATERIAL

First, we need to think of embedded information processing and networking as a material.

Let me explain. When a designer can include information processing in a product for very little cost, the calculation becomes not one of engineering complexity, that's relatively cheap, but one of competitive advantage. Including a CPU to produce behaviors becomes a line item in the competitive analysis of making an object, just like the calculation about what to make it out of. What you do with that CPU becomes part of the design of the product and needs to be designed with the same attention to the other parts as any of the materials being used. And just like a material, it creates some new capabilities, and imposes new constraints.

I think the toy industry is leading the way here. All kinds of toys now depend not just on their physical appearance, but on behavior created by information processing, for their competitive advantage.

Information as material is the heart of ubiquitous computing. Any tool can now use information processing to help it do its job better.

Let me give you an example. 1998, V-Sync, a Japanese company, introduces their Internet Refrigerator. It has a computer with a touchscreen built in. 1999, Electrolux introduces their Screenfridge, it has a computer with a touchscreen built in. 2000, Whirlpool and Cisco introduce a fridge with a touchscreen. 2002, Whirlpool tries again with their Connected refrigerator. 2003, LG introduces their internet fridge. 2006, Electrolux tries again with a new Screenfridge.

Why am I showing you all of these refrigerators? Two reasons: first, to show you that although fridge computers have been for sale for nearly 10 years, the odds are that you’ve never seen one. Why? Because there’s no point. They don’t treat information processing as a design material in the development of the fridge, they treat it like something that has to be forcefully grafted on. They’re not its unique qualities to solve a problem or meet a basic emotional need. It only adds complexity.

APPLIANCENESS

> ‘Applianceness’ [is] the set of properties that guide the design process towards simple, helpful devices that exploit the potential of embedded information technology in everyday things.
>
> - Bill Sharpe,
> *Information Appliances: an introduction*, 2001

2. APPLIANCENESS

The second reason is that I’d like to introduce the idea of “applianceness.” This is a term coined by Bill Sharpe, a British interaction designer who has a consulting company called, appropriately, The Appliance Studio. They designed Steelcase’s RoomWizard scheduling device, a classic information appliance.

The core of the idea for me is that focus in functionality is more important than arbitrary flexibility. When computation is cheap, we no longer have to make general purpose computers, and this term reflects for me an important shift in emphasis. Computers are appliances, there is not the one-to-one relationship that terms like Human-Computer interaction. It’s one human to a multitude of appliances, some of which use information processing, others just electricity.

Source: Whirlpool

Let me show you a computer fridge that exhibits applianceness, in Sharpe's definition. This is Whirlpool's centralpark fridge, their third attempt at mating a computer with a refrigerator. I think it's the most successful so far. You can buy widgets that plug into the front of the fridge. There's a picture frame, an iPod dock, a DVD player, a calendar and a message board. They're all computers, but each has its own interface tuned that task. It's no longer about computation, it's about the experience you can create with it.

3. PHYSICAL OBJECTS CAST INFORMATION SHADOWS

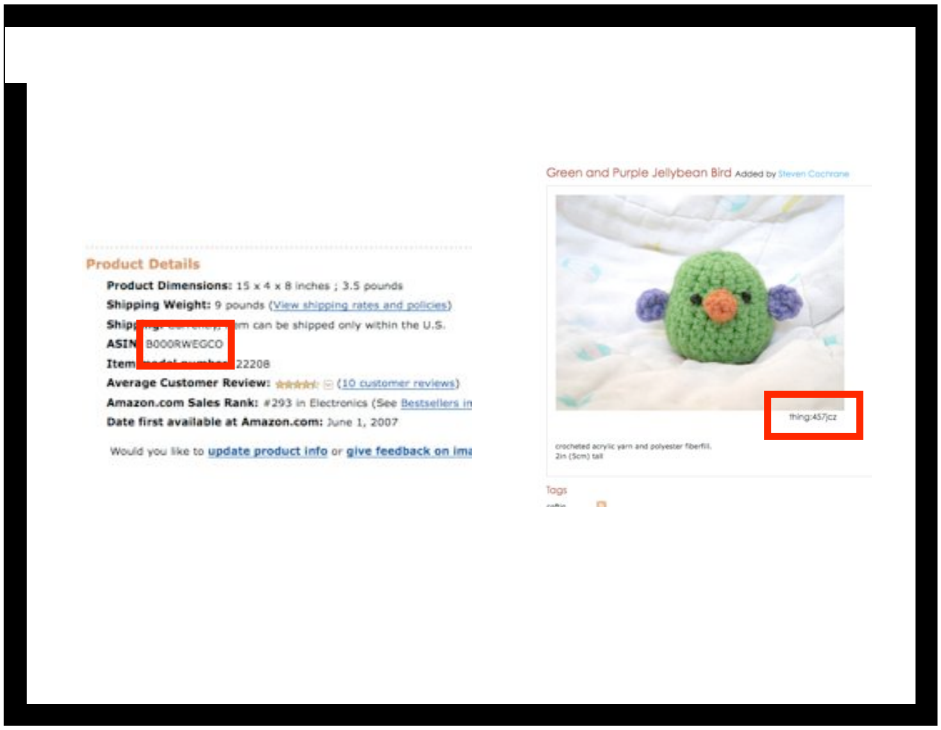
When we started looking at what it would take to redesign everyday objects using ubiquitous computing technology, we realized that nearly everything exists simultaneously in the physical world and in the world of data. We call that object's digital representation its "information shadow." Information shadow can be examined and manipulated without having to touch the physical object. Think of the Amazon and Google book APIs. Information shadows have lives of their own. Those lives can be as rich as the physical object's life, maybe richer.

Once you have decided what constitutes a programme episode then something really significant happens - you can give it a name, make it addressable, you can - for the first time point at it. Better still, you can move from pointing at something to glueing handles onto it. And once you have such a handle, then you can pick up the programme and throw it around and stick labels on it and join it together with other programmes with bits of semantic string.

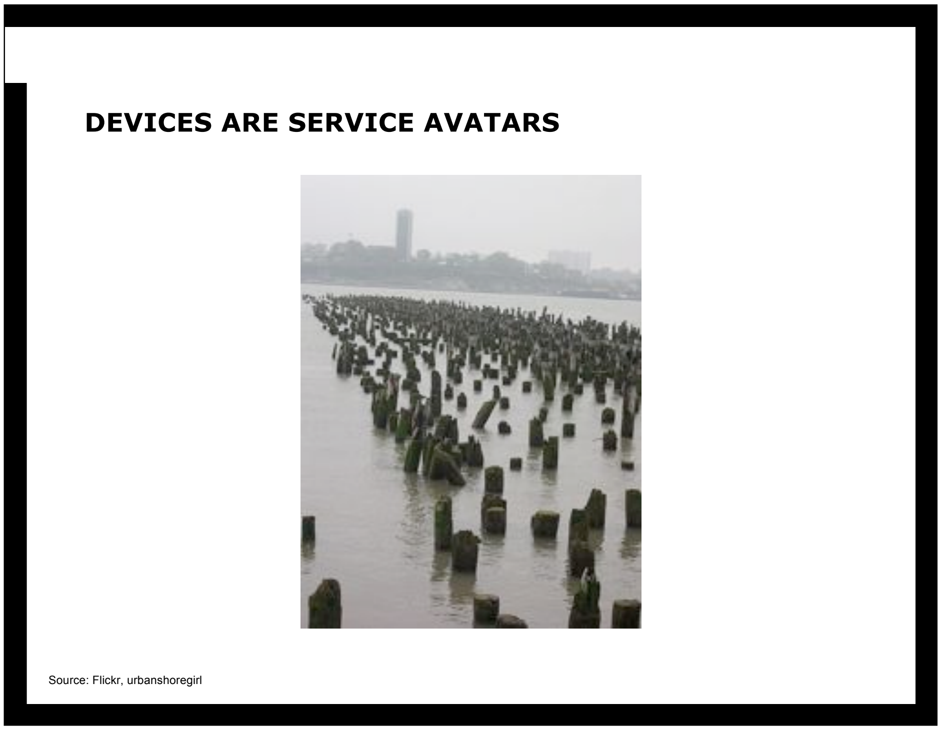
- Tom Coates,
The Age of Point-at-Things…, 2005

Source: http://tinyurl.com/yt9rlw

Tom Coates was designing Web program guides for the BBC when he realized the power that happens when you can uniquely identify a TV show in a digital networked environment. Now imagine what happens when you take that kind of power and apply it to the information shadows of physical objects.



Ulla-Maaria Mutanen realized that Amazon's ASIN system allows people to reference the information shadow of virtually anything they sell because it's a unique, global identifier. She created a system called Thinglink that creates such numbers for anything. Her system may not be the ultimate information shadow infrastructure, but I believe it points to a profoundly new way that digital technology can interact with the physical world.

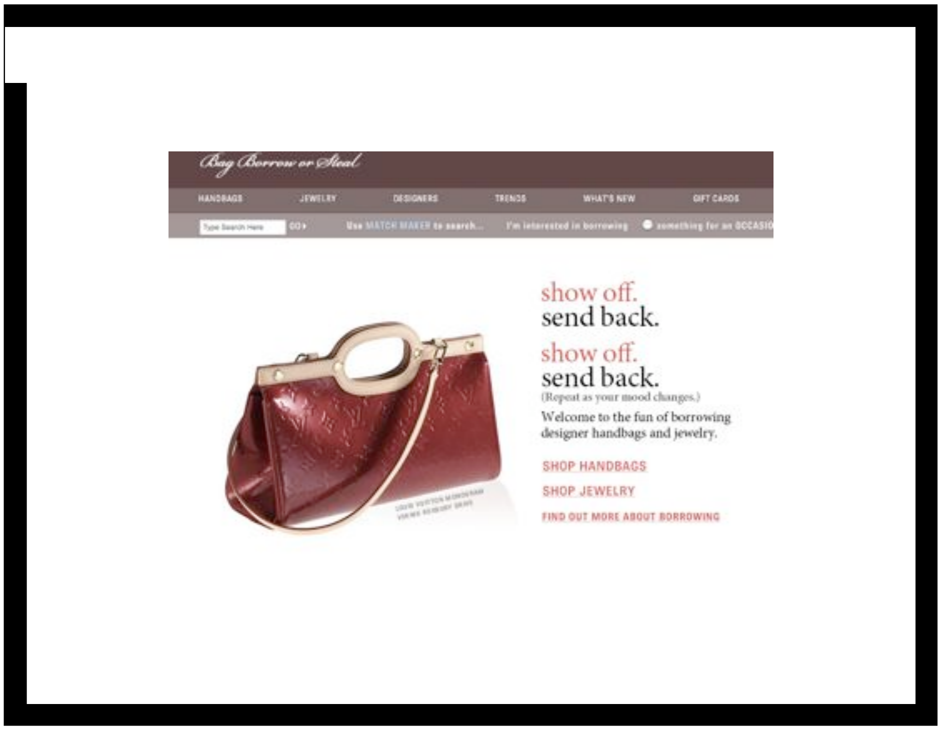


5. DEVICES ARE SERVICE AVATARS

Networking is a key hardware component of ubiquitous computing, which brings me to the next idea. Network connectivity mean that the same information can be accessed and manipulated through a variety of devices. This means that value can shift to the information, rather than the device that’s communicating it. Devices become secondary, they become temporary representations of information-based services.

A number of familiar appliances--cell phones, ATMs--are worthless without the networks they’re attached to. They are physical manifestations, avatars, projections into physical space of services, but are not services themselves. You really start to see this in purely information entities: what’s a plane ticket? what’s money? what’s a book? They become subscriptions and agreements, for which a device becomes a nearly disposable channel.

When designing user experiences for ubiquitous computing, the design of the service becomes as important as the design of the device. This is the genius of the iPod: it’s an avatar of the iTunes Music Store. The Amazon Kindle, as questionably designed as it is, is a physical manifestation of the Amazon Kindle Store. Right now most of these services are information or media related, but that’s changing. City Car Share means that your car is now a subscription.



I particularly like Bag Borrow or Steal. It's a subscription service for expensive purses that you only use occasionally.

The purses are not digital, but I think that it's what happens when digital and nondigital media intersect. Renting a suit or furniture used to be a bigger hassle than owning them outright. Digital technology means that more of our artifacts are becoming subscriptions and to me that means our designs must increasingly reflect the service they represent. Right now, you're getting a leather Vuitton handbag from Bag Borrow or Steal. What if you could, as my friend Ryan Duke proposed in a design project he did, subscribe to the latest Vuitton pattern and your purse's e-ink cover would automatically change once a month? What, then, do you own and what do you care about?

GRANULARITY CONSTRAINS EXPERIENCE

6. GRANULARITY DETERMINES KEY ASPECTS OF EXPERIENCE DESIGN

General purpose computers traditionally have interfaces that are person-scale. They're designed to be used in a wide variety of ways, and what typically makes sense is to make the input device about the size of your hands and the output about the size of your head.

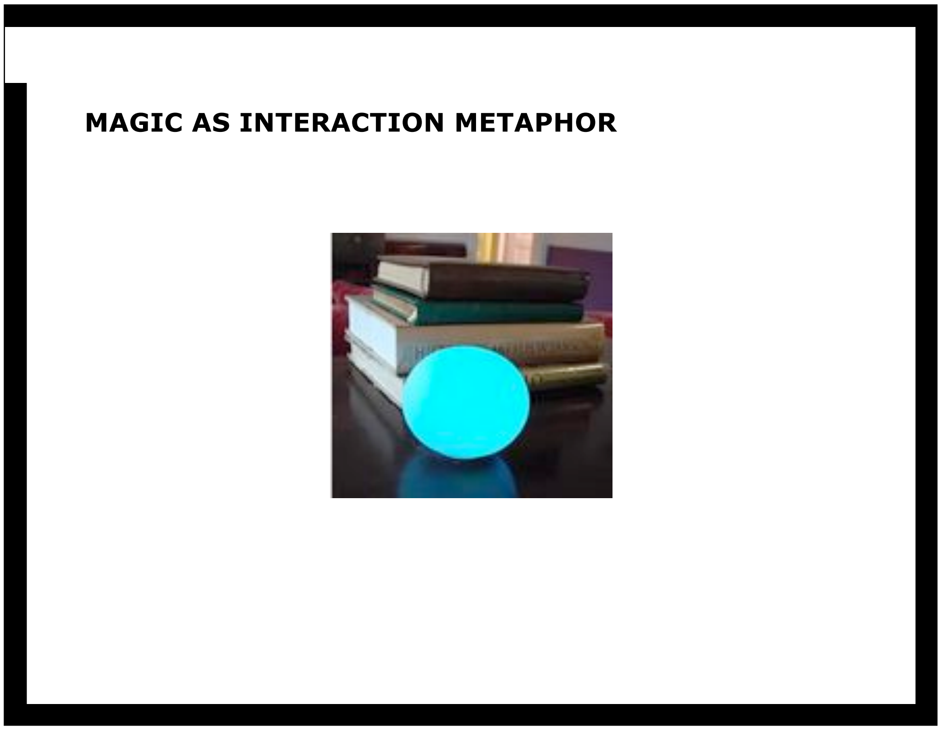
Ubiquitous computing devices can come in all sorts of sizes and the user experience design for them must take this into account. This has been true since the earliest days at PARC when Weiser defined the tab, pad and board as names for the scales of the devices they were developing.

That made sense in the document-centric world of Xerox, but I think it's too limiting when thinking about everything that's possible with ubicomp.

From Flickr: watch by funadium, box by ubermichael, phone booth by rastrus, room by bigpinkcookie

Scale	Label	Examples
1 cm	covert	RFID, nail polish, cochlear implant
10 cm	mobile	phone handset, portable media player, wallet
1 m	personal	chair, car, ATM, payphone, laptop
10 m	environmental	wall, door, chandelier
100 m	architectural	church clock, billboard, bus
1000 m	urban	street intersection, landmark, crowd

This is the scale I've been using. It's a set of definitions to talk about granularity and it helps us identify that works and doesn't work at various scales. Screens don't work when you approach the covert scale, which is why wrist TVs have never taken off. Buttons don't work well on the environmental scale and above, because they're too small relative to the object. You probably can't make anything that's designed to be immediately social at anything above the environmental level.



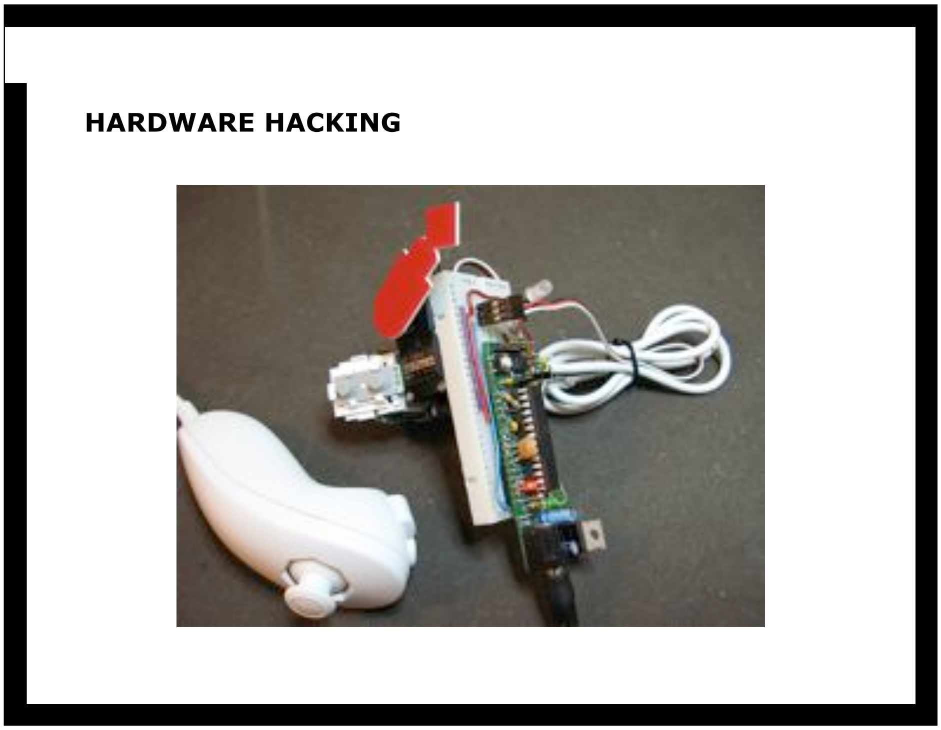
7. MAGIC IS A POWERFUL UBICOMP INTERACTION METAPHOR

The last general idea I'd like to talk about is magic. Not as hand-waving to obscure functionality, but as a design metaphor like the desktop metaphor. Specifically, I mean using the concept of enchanted objects to generate ideas about interaction and as a way to create user experiences that are easier to explain. People have a tendency to create animist explanations for the behavior of technologies that exhibit unpredictable behaviors. They treat their Roombas like pets, they get mad at their laptops, they think their iPod is obsessed with a band, etc. We can use these natural associations to design ubiquitous computing interactions. Many existing ubicomp products already reference magic implicitly. This is Ambient Devices Ambient Orb, which is a kind of digital crystal ball. The Wiimote acts very much like a magic wand, as do all of the phones that have accelerometers in them. Wearables often take the form of amulets. There's still a lot of useful space left in the metaphor.

SKETCHING

How does this all this theory work in practice? Well, since there are no best practices, you have to start somewhere. I went back to sketching. As Bill Buxton points out, sketching is not prototyping. It is not the first step in solving a problem, it is the process by which we understand the design space so we can define the problem in the first place.

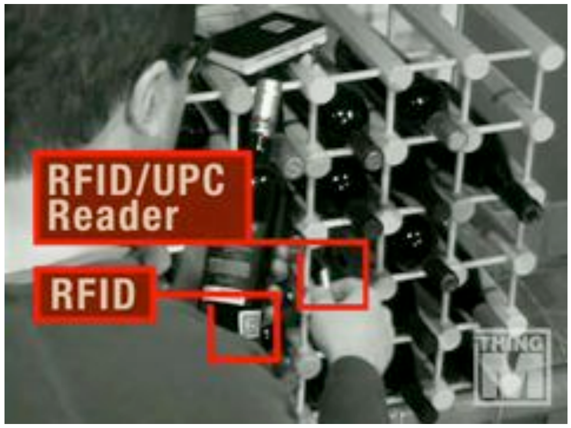
We see sketching as the application of agile software development principles to design.



The first sketching technique we use is hardware hacking, which is like using tracing paper, but with hardware. We take existing technology and attempt to extend its capabilities to see where things can go.

Here's a hacked Nintendo Wiimote nunchuck. Tod used the accelerometer in the Wiimote nunchuck, connected to an Arduino open source hardware I/O board and a small servo to create a kind of mini-Segway mechanism that always points up.

VIDEO PROTOTYPING



Another sketching technique we use is video prototyping, This again helps us imagine how a technology could work, without actually having to make it. We fake all of the technology using video so that we can concentrate on the interaction. Making the video is fun, but it's also incredibly valuable from a design perspective. We have to face a lot of our assumptions about the experience we're trying to create.

Last January, we made one about a smart wine rack.

[show video]

IDEATION TECHNIQUES: 10X, DEMOGRAPHIC SHIFT

Source: Flickr, Cory Doctorow

We start with a couple of ideation techniques.

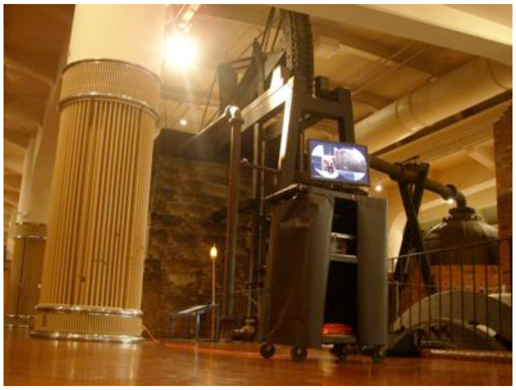
One that we use is the 10x technique. What if a technology that we have today was 10 times as cheap? 10 times a prevalent? 1/10th as small? Etc.

Another is a projection of current behavioral trends onto near-future situations. This is a way of doing user research without having to get helicopetered into Beijing by Nokia. For example, the folks whose formative late teen/early 20s years heavily featured Facebook and World of Warcraft are probably going to have a different attitudes toward revealing personal information online in a work context than people from earlier generations. We don’t know that, and we should research it, but when doing ideation, we can assume that some mapping will occur and use that as the basis for identifying problems people may be experiencing that can be solved with technology.

THE HENRY FORD CUSTOM CONTENT PROJECT

Let me tell you about a hardware sketching project we did last year. We worked with the Henry Ford museum in Dearborn to help them understand how to make their enormous collection more relevant. They're in an interesting position: they have a collection of millions of artifacts related to the history of technology from the 17th century until today. The place covers nearly a square mile and to give you a sense of scale: they have a large collection of buildings and airplanes. Making all those objects relevant is hard. A retired engineer is going to have a very different perspective than a 12 year-old on a class trip and one set of wall text isn't going to satisfy both of them.

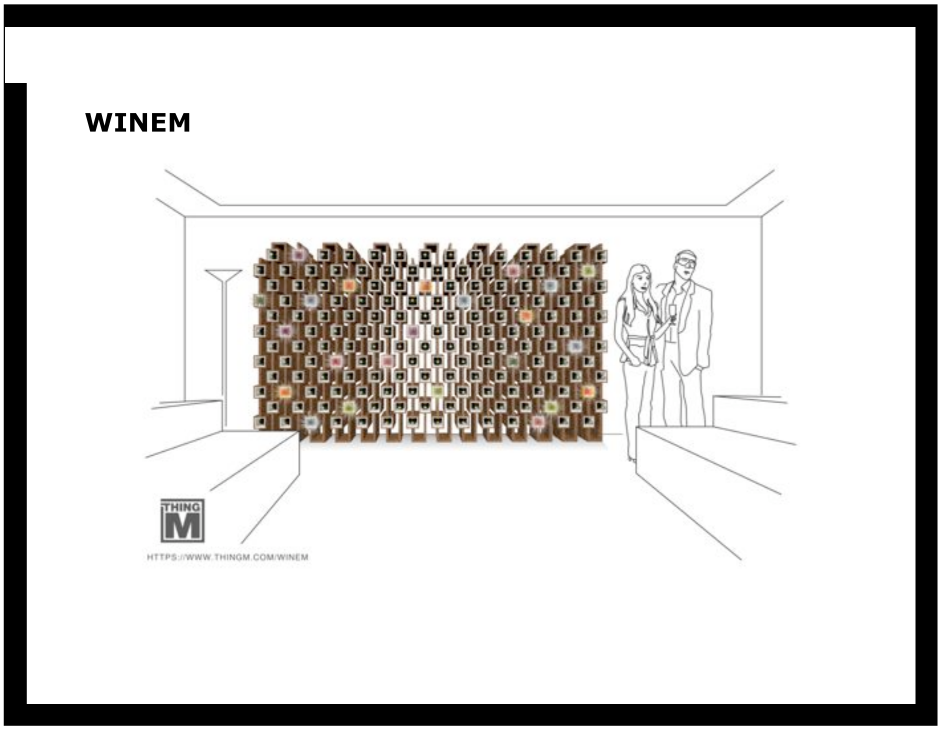
We did a series of sketches to see how ubiquitous computing technology could create different experiences for different audiences. Our goal was to understand the boundaries in the experience design space of the problem, rather than creating a finished solution. We began with an exhaustive literature search and once we knew what others had done, we started sketching. We spent about a week every month creating a completely new experience, with quasi-realistic content and semi-functional technology and then we tested it with real visitors. This is April's. We made RFID tickets that selected one of a set of videos which told a different narrative for each user experience persona. There was one that more engineering focused, one that more about history, one that was a game, etc. Using our previous definitions, we designed the ticket and reader on the mobile scale, while the video was on the environmental.



This is May's sketch. Our challenge here was to make sense of these enormous steam engines. These used to be in textile factories in England and moved water around giant locks in Birmingham. Few people even know how a steam engine works, and the relevance of the differences between different engines is difficult to explain. Since the machines were so large, we took our inspiration from coin-op binoculars and sketched out a set of magic binoculars, an enchanted spyglass, that are a kind of lightweight augmented reality. We made this by putting an optical mouse under a lazy susan and then moving images inside a vignette that looked like binoculars.

Here’s another sketch. We used a pair of Wiimotes to create magic wands that allows people to point at objects and get extra information about them. Depending on which wand they use, different information is projected around around each object. They can also pull the trigger and get additional in-depth audio description of that artifact.

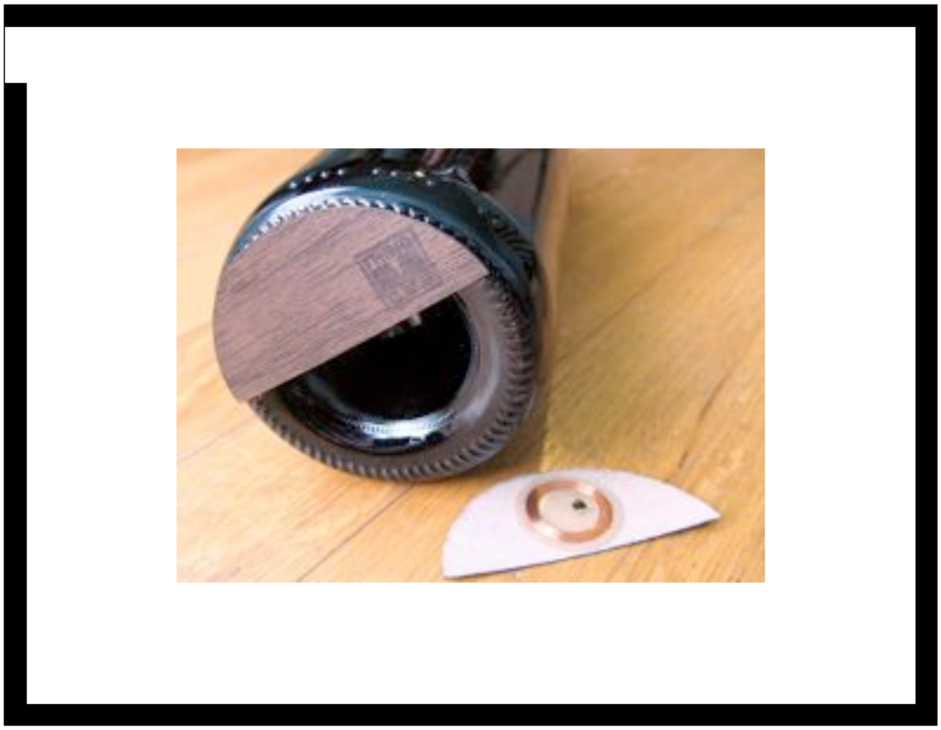
We tested each of these prototypes with end users and used that to distill a set of design and development guidelines for how to think about systems like this in general.



I’d like to show you another project we did last year.

We made that video I showed you about a year ago, blogged about it, and moved on. But the blogosphere didn’t let us get away that easily. For the next couple of months people kept asking us where they could buy it. Then, two months after we put the video out, Wired invited us to exhibit it at NextFest and we agreed. The problem was that it didn’t exist, it was a video. This is the problem of faking your technology.

However, we didn’t let that stop us and decided to take it to the next level and rather than just sketch it, we decided to make a working prototype. Tod focused on solving the RFID engineering problems while I designed the user experience. I started by developing several personas with Ryan Duke and then sketching designs for each persona. This is design 38 out of 60 for Vince, the Wired NextFest visitor. Its goal was to draw Vince’s attention in a crowded space and instantly communicate that this is a fundamental rethink of what a wine rack is. Here you can see how traditional industrial design and interaction design start bleeding into each other.



This is our RFID bottle tag. We wanted every piece of the design to communicate a set of values with the way it looked and worked. It’s made of the same walnut veneer as the rack, sits flat, isn’t visible when the bottle is on a table, and leaves enough space so that you can put your thumb in the back to pour the wine.

We tried to really treat the rack as an appliance and avoid using screens entirely and design only for the environmental scale, but we couldn't come up with a workable solution, so we decided to make a control panel that detached from the main unit and used as little of the visual language of software as possible. We used Nokia wifi touch tablets as control panels. We love these tablets for prototyping. They have beautiful high resolution touchscreens, connect to the Net over Wifi, they have decent Web browsers, and they run Flash. The problem is that there's not that much screen real estate…

Source: Qliktech

…and there's a lot of wine information. To deal with this I decided to use a faceted classification system. I felt that facets would allow people to explore the information they needed to choose a bottle of wine given a limited amount of screen real estate, and only using their index finger.

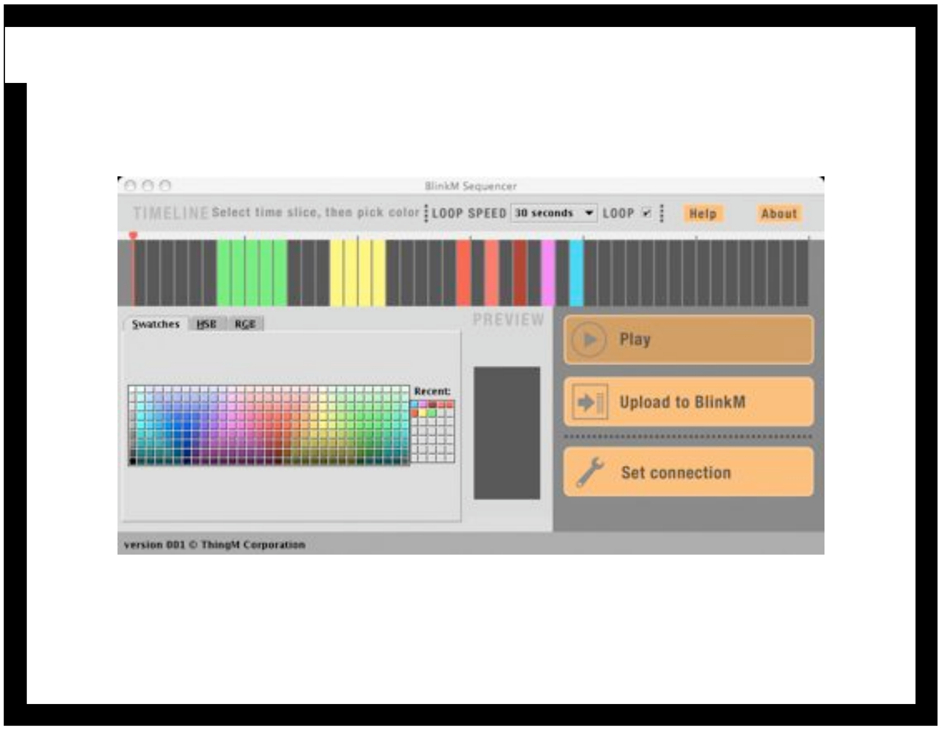
This was a pretty big challenge. But this is where what we do diverges from traditional UI design, and I think to our advantage. This is not our appliance. [click] This is. [click again] This is a control panel, so it doesn't have to show all the relevant information all the time and can focus on navigation.
This is a faceted classification browser; the tabs are facet categories. When you click on one you are simultaneously navigating an information space on the control panel and project corresponding information on the bottles in the rack. When you click a tab it lights up the bottles in different colors to identify the intersection of the currently selected contraints. As an appliance, the interface does not assume that the goals is to narrow down to one answer. It's a decision support tool at every action.
This is the 8th version of this UI. We did some testing in between each one and my intuition about the core functionality was wrong five of the 8 times.

Now let's put all of the pieces together. Let me show you the system in action.
[show video]

So that's how we backed into our first major product. We're actually making this a product now, and we prices by it by the end of March.

BLINKM

I’d like to wrap up by talking about something different. Not completely different, but different. We’re a consumer product company, but we recognize that there are other markets. Our work doing the Sketching in Hardware conferences and Maker Faire showed us that there's a need for better user experience design around basic electronic components and that ubicomp technology can satisfy that need. Making an LEDs light up is easy, but making an RGB LED a specific color or blink in a specific pattern is hard. You have to know about color theory and pulse-width modulation and power. We decided to make it easy and to see if we could push the concept of ubiquitous computing all the way down to the level of individual components. BlinkM is a smart LED. You give it an RGB triplet, it knows how to make it glow that color. You give it two, it’ll do a smooth fade between them through color space. You give it a pattern, it’ll blink in that pattern. Once programmed, it works just like an ordinary LED. It’s a slightly enchanted LED. The component as a tiny appliance, an atomic unit of ubicomp: a one bit output and a cpu.



Here’s the sequencer software for it. It’s a combination of a color picker interface and drum machine sequencer. The BlinkMs also network together and take input, so you can make a string of smart Christmas lights that react to sound or that generate new patterns based on the time color of the sky.

We’re looking forward to sketching with them, to designing with our new toys. But mostly, we look forward to what this means. If two guys can hack together all of the stuff that we’ve done over the last year, the future of design is going to be an exciting and challenging one.

Mike Kuniavsky
mikek@thingm.com

Thanks!