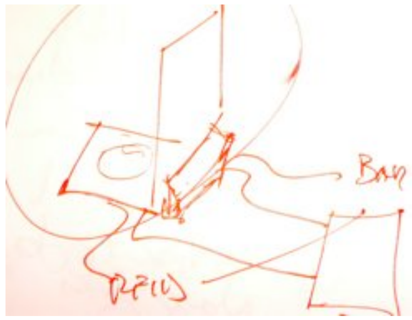
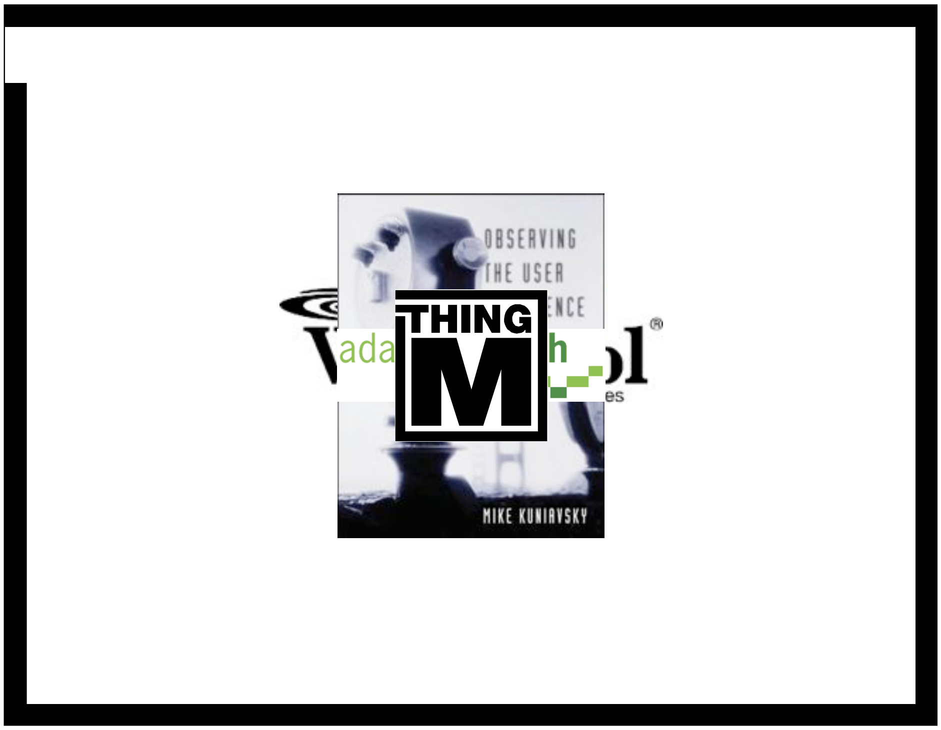
Sketching Smart Things



Mike Kuniavsky
BayCHI
February 12, 2008

Hi. Thank you, Rashmi and BayCHI for inviting me. It's really an honor to be talking about ubiquitous computing in ther its birthplace.

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.



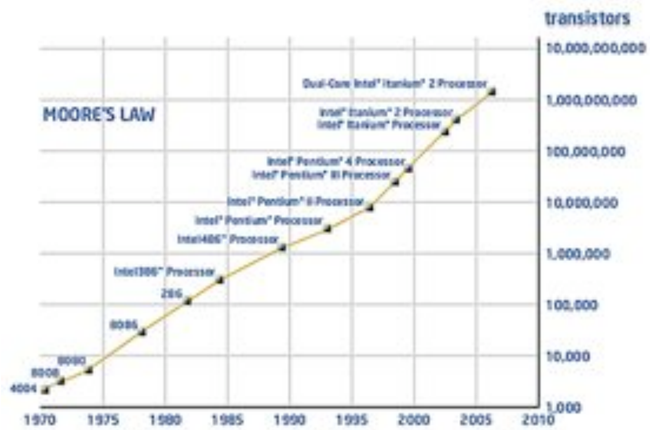
I'd like to start by telling you a bit about who I am.

I specialize in user experience design and user research. Over the years, I've worked with a wide variety of organizations to help them develop technology with people in mind.

[click]

As part of this practice, I wrote a book, a kind of cookbook covering a wide range of user research techniques. I co-founded a design company called Adaptive Path.

In 2006 I started a company called ThingM. We make ubiquitous computing consumer products.



Source: Intel

Let me start by talking a bit about ubiquitous computing and why I think it's really important.

Let's start with Moore's Law. You've seen it a thousand times, but let's look at it again. 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. Specifically, at introduction, CPUs cost between $500 and $1000.

Thus, the other way that you can read this chart is that the PRICE of older processor technology decreases proportionally to the increase in transistor density. People tend to concentrate on the right side of the curve, but I'd like to draw your attention a little to the left, to what I call the Hidden Middle of Moore's Law. I think it starts right around the 486. These processors can do an immense amount and correspond roughly to the beginning of the modern, internet-connected computer.

1989: $900

33MHz, 20MIPS

2007: $0.53

20MHz, 20MIPS

Sure enough, you can see that the actual chip prices reflect that price drop. At the time ubiquitous computing was invented here, that level of processing power cost about $900. Now, it's about 50 cents. And that's why I believe that we need to be looking at it very seriously now, because it's finally a practical reality.

Source: Flickr, jurvetson

What does this mean for design? As interaction designers, we just figured out how to make Web pages not totally suck; as industrial designers we're just starting to realize that functionality and psychology are part of the job, too.

Tod Kurt and I founded ThingM because we believe that ubiquitous computing holds amazing promise for making the world a better, happier and more interesting place. However, ubicomp is today where the Web was in 1992, and design for the field is still mostly full of unknowns.

To wrap our brain around it, we've bounced several concept to explain what ubiquitous computing means to experience design.

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. 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.

> ‘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 concept I’d like to introduce the idea of “applianceness.” This is a term coined by Bill Sharpe, a British interaction designer whose company designed Steelcase’s RoomWizard, a classic information appliance.

The core of the idea for me is that focus in functionality is more important than arbitrary flexibility. When information processing is cheap, we no longer have to make general purpose computers, you make appliances.

We also see applianceness as the augmentation of existing, familiar things. We can thus introduce new technology into interactions without changing the devices’ fundamental nature, like ice boxes became refrigerators and hand crank mixers became electric mixers in the early days of electric appliance design.

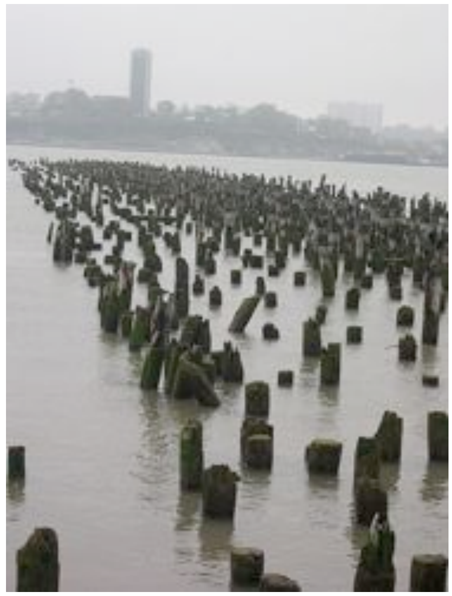
Source: Flickr, dailydog

3. PHYSICAL OBJECTS CAST INFORMATION SHADOWS

Now let me move on to another high-level concept we're working with. As we started looking at what it takes to redesign everyday objects using ubiquitous computing technology we realized that every material object casts what we call an "information shadow." In our modern world, everything exists simultaneously in the physical world and in the world of data. There is metadata about nearly every object and that information shadow can be examined and manipulated without having to touch the physical object. Information shadows have social lives of their own. Those lives can be as rich as the physical object's life, maybe richer.

What are some objects with rich information lives? Well, there is media: books, CDs, images, etc. I suspect more people manipulate images of the Mona Lisa than have ever seen it in person. Amazon is the great repository of a lot of this, and as Ulla-Maaria Mutanen identified Amazon's ASIN system allows people to reference and discuss virtually anything they sell it has a unique identifier that gives you fast access to that object's information shadow.

Take wine. I'll talk about our work with wine a little later, but wine has particularly active information shadows. Virtual Vineyard was the first ecommerce site. Bruce Sterling talks about wine extensively in Shaping Things, his book on ubiquitous computing, and wine has long been a favorite example for information architects. People talk about wine and collect wine information possibly more than they actually drink it.



Source: Flickr, urbanshoregirl

5. DEVICES ARE SERVICE AVATARS

The other key hardware component of ubiquitous computing is networking, which brings me to the next idea. Networks mean that the same information can be accessed and manipulated through a variety of devices. Most value rests in the information, rather in the device that's communicating it. The devices become secondary. A number of familiar information appliances--cell phones, ATMs, TiVos--are basically 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.

This means that when thinking about how to design 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 primarily 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, you can subscribe to use the latest fashionable purse only when you need it at Bag Borrow or Steal. That means our artifacts have to design have to increasingly reflect the service they represent.

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 of PARC's tab, pad and board as concepts for the scale of devices.

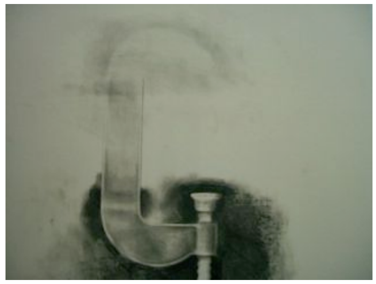
From Flickr: watch by funadium, box by ubermichael, phone booth by rastrus, room by bigpinkcookie

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 and meet people's expectations with appropriate behaviors. 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. Etc.



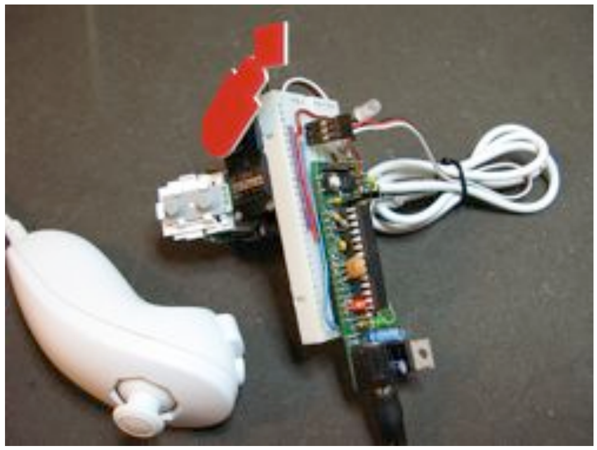
(Jim Dine, *Untitled* from *Ten Winter Tools*, 1973)

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 wrote in his book Sketching Interactions, 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 agile principles we embrace are

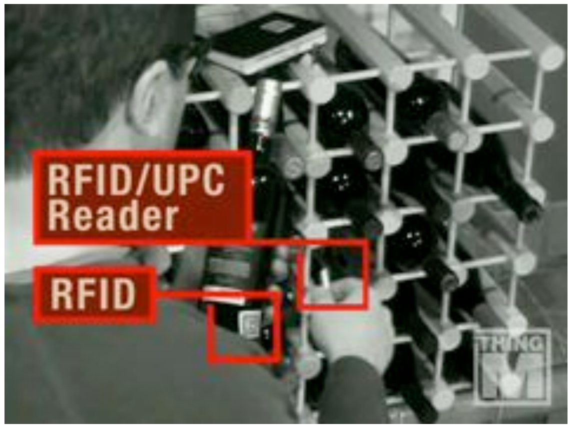
- Rapid iteration.
- Barely sufficient implementation, in other words, doing just enough to move on
- Always having something that works, even if It’s a paper mockup and fakes the functionality
- User research at every step
- Interdisciplinary collaboration from the start
- High adaptability, if our ideas are wrong, we move on and have other ideas

The first sketching technique we use is hacking. Which is kind of like using tracing paper, but with hardware. We take existing technology and attempt to extend its capabilities to see where things can go. What’s possible? That’s Tod, my business partner, on the left at Maker Faire last year with a Roomba that’s been hacked to be a giant plotter.



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.

Unfortunately, the tools for doing this kind of hacking, what we call sketching in hardware, are pretty primitive, but they’re coming up fast.



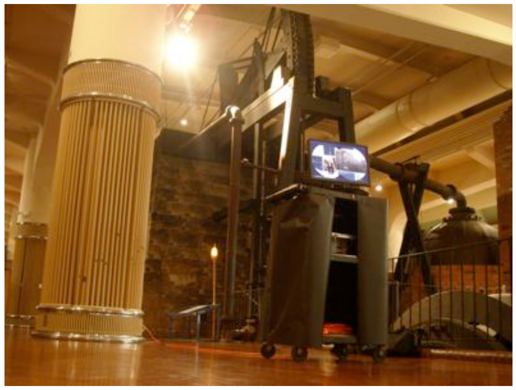
Another sketching technique we use is video prototyping, We're certainly not unique in this, but it helps us imagine how a technology could work without having to make it. We fake all of the technology using video so that we can concentrate on the interaction. Making the video is fun, but in making it 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]

We don’t consult much, but last year we got the opportunity to did an extensive sketching project for the Henry Ford Museum. 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. One set of wall text isn’t going to satisfy both of them.

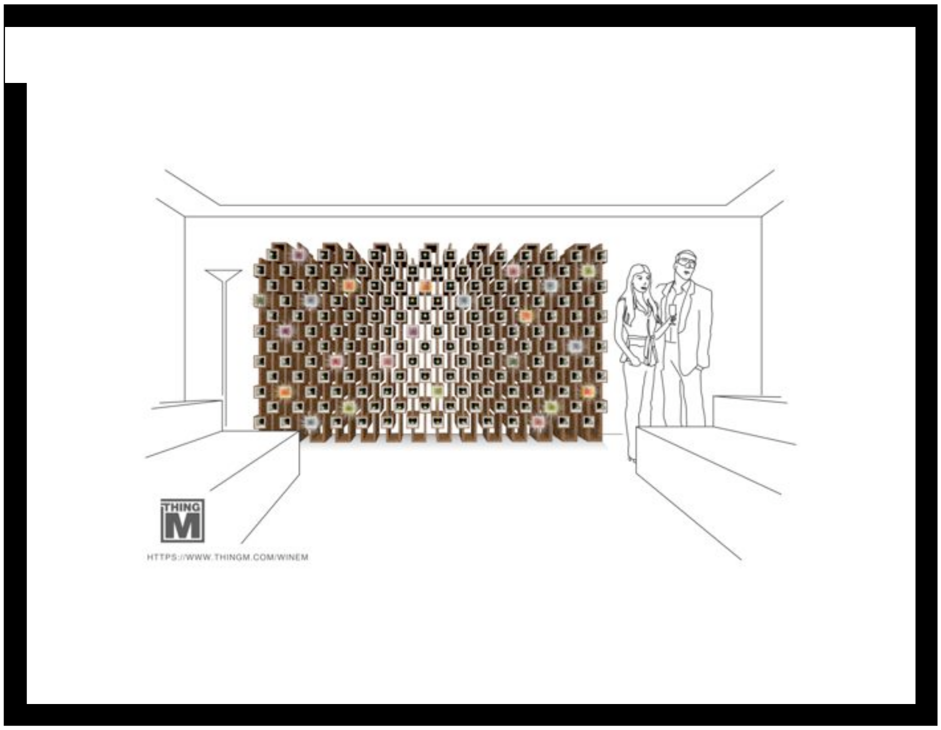
To understand the boundaries in the experience design space of the problem, we did a series of sketches. We began with an exhaustive literature search. Once we knew what others had done, we spent a week every month creating a new experience, with quasi-realistic content and semi-functional technology 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.



This is May's sketch. Our challenge here was to make sense of these enormous steam engines. These used to be in textile factories and moved water around giant locks in Birmingham, England. 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 as a kind of lightweight augmented reality. You see the scene in front of you, but there's additional information overlaid on it. You pivot the screen and the scene changes, just like binoculars. For example, if the engine was originally out in a field pumping water out of a mine, you can look at the countryside it was in and compare it to the setting of another that was in a mill. Or you can see a historical narrative with old photographs that float around each machine as you look at it. Yet another overlay turns each machine into its blueprint, with all the parts labeled.

Here’s another sketch. We made Wiimote magic wands that allows people to point at objects and get extra information about them. Depending on which wand they used, different information was projected around around each object.

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.



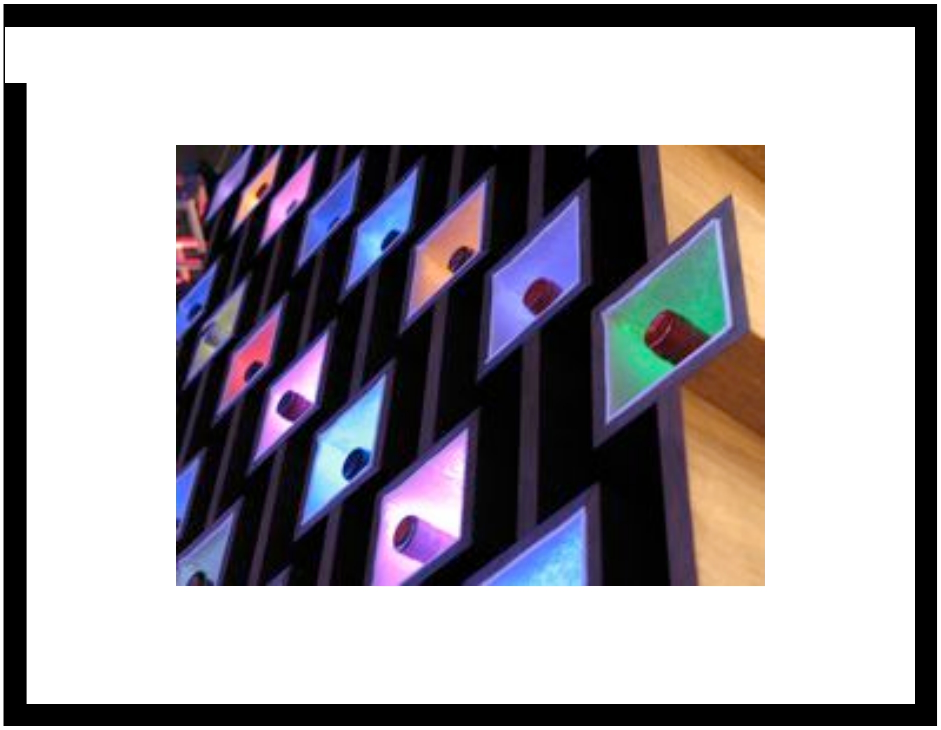
I’d like to show you another project we did last year.

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, the industrial designer we work with 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 screenless 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.



I think it’s at this point 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, by the way, is the 8th iteration of the interface. 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.

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 many consumer 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. We made 1000 of them, they went on sale two and a half weeks ago and we're happy to report we're nearly sold out.

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!