Projections into the world

Service avatars as Ambient Intelligence objects

Mike Kuniavsky (for ACM <interactions>, July/August 2005)

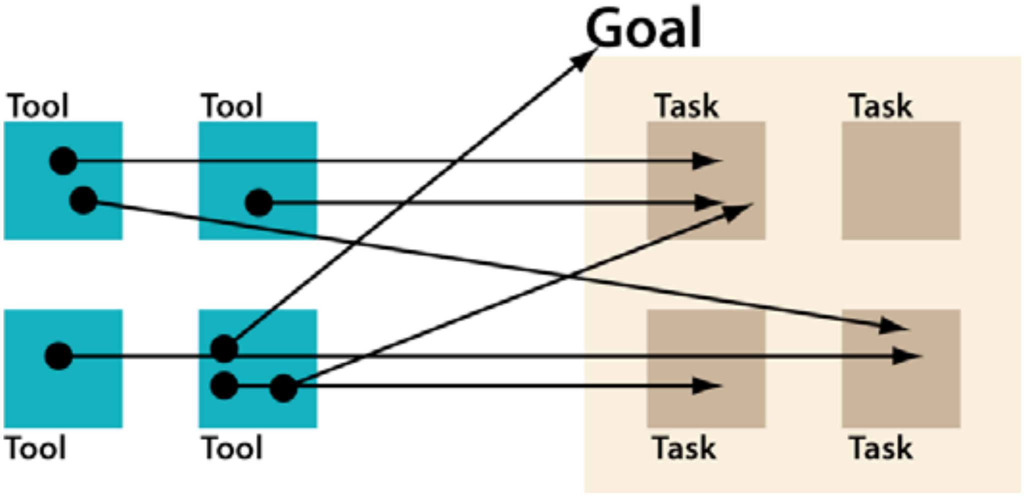
I'm a user experience designer and researcher. When I look at a technology, or an idea about technology, my first thought is of the end-user's perspective: why would they want to buy this? How would they use it? Why would it be interesting to them? For a while, Ambient Intelligence (and its twin sisters, Ubiquitous Computing and Pervasive Computing) seemed like a solution in search of a problem, in terms of the user experience. The basic idea was nice, but examples were often driven by the capabilities of the technology, rather than by user needs.

I like the basic idea and I'm convinced that small, task-focused computers embedded in everyday objects are the future, and it's a better future for it. The question is how we're going to get there--not just how we're going to create the technology, but how we're going to design the technology so that people want to use it. In other words, what incentive are we going to create for people to adopt this technology?

For most of the 90s, consumers and manufacturers of consumer electronics generally stayed away from products that embodied the AmI/Ubicomp/Pervasive space. All the talk of "convergence" between computers and domestic objects generally led to a lot of press, few commercial products and lackluster sales. Why? To my eye, the costs of making these new tools work together and learning new modes of use outweighed the benefits they provided. I know from personal experience that momentum rules the day as far as work and life practice goes, and the costs of using AmI technology kept it mostly in the lab. In the 00s, wireless communication standards, more powerful mobile phones and a shift away from engineering-driven product development has created several popular products that exhibit some nascent AmI features (obviously Apple's iPod, the Linksys NSLU2 home file server, the TiVo, the Adidas 1 shoe). Now that there's momentum, how do we keep it going?

Historically, the object that you interacted with for a service (and I'm defining service fairly broadly here) was where all the information about that service was stored. It was difficult to share that information other than through human labor. A cash register sitting in one shop had no way of communicating to one sitting in another shop, or a centralized accounting office, until recently. Photos had to be processed, printed and mailed. Synchronizing all the clocks in a building was a complex technological trick.

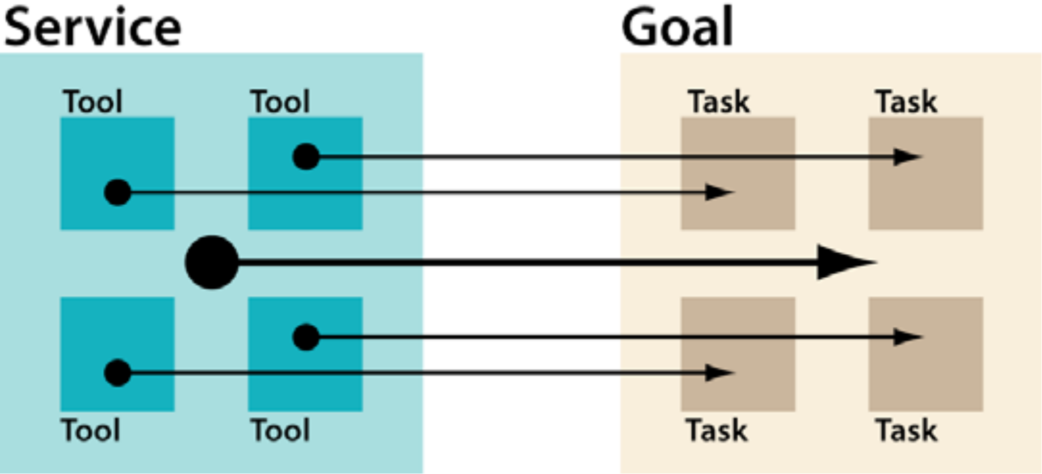
For the most part, the services of our lives are still very tightly bound to specific tools. Even the latest high-tech objects are still focused on embedding all functionality--often as much functionality as can be imagined--in individual objects without thought to the interrelationship between those objects and other objects in the immediate environment. Digital photos, for example, can be transferred between the tool that makes them (my camera), the one that organizes them (my laptop), the one that stores them (the large hard disk at home), the one that shares them (Flickr), and the one that's used to view them at my Mom's house (her desktop PC). Each of these tools is good at what it does, but it's a lot of work to transfer the same data between a half-dozen devices.



Now there's no reason that services must be coupled to specific objects or places, and of all of the possibilities AmI provides, the one I think may be most compelling is its ability to decouple services from individual objects.

People have already enthusiastically embraced services that span multiple physical objects when they use phone networks, ATMs, and Web-based email, without worrying about how to transfer information between them. Each ATM is an individual tool, but we don't treat it that way: it provides easy access to a service for which there are other tools (bank tellers, Web sites, phone call trees, etc.).

Treating objects as representatives of a service, rather than the service itself, is a fundamental change in our relationship with the objects of our lives, but in a way that feels like a natural extension. The service becomes the focus, and the objects its avatars: the projections of a single idea into the world, rather than each embodying a different idea. This sounds lofty, but matching service to goals may lead to better overall user experiences than trying to match tools to goals. Traditional tool design focuses on enabling concrete tasks used to fulfill abstract goals. In many cases the tool becomes a necessary burden on the way to satisfying the goal, and the design of the tool is assumed to be unable to address the goal directly. Designing a service to satisfy a goal, and then designing tools that use the service to support tasks that satisfy the goal is a potentially "cleaner" way of thinking about creating a user experience than trying to enable the goal by way of designing task-specific tools.



Like Thing on the *Addams Family*, a service can be available wherever I am, without me having to bring it there. A human-based version of this exists in luxury hotel chains. These organizations maintain Guest Historians on staff, whose job it is to collect and share knowledge about guests. When a repeat guest make a reservation at any Four Seasons, for example, that hotel knows that guest's preferences in advance, based on that guest's requests and information the Historian has collected from the staff. The corner room on the third floor will be waiting for one person, while a spa appointment timed for the anticipated arrival of another, without those people having ask for it. Sharing information between objects makes it possible to create tools that are focused on supporting specific user work practices, needs and desires, while maintaining a common context and using that knowledge in support of a variety of tasks.

Design from the view of a whole system, even if not designing the whole system, changes the way that we create tools, allowing more transparent interaction between objects and shifting attention from the objects back to tasks and desires. The emerging discipline of service design has started to codify these ideas, and many people working in the mobile phone world wrestle every day with the consequences of systems designed from the tool view, rather than the service view. The decoupling of services from their implementation is already happening in other fields, too. Open APIs, such as Amazon's or Flickr's, allow for the sharing of infrastructure and people are building task-specific applications with them. Mappr is a map product that uses Flickr's API. It would probably be inappropriate feature creep for its functionality to be rolled into Flickr proper, but it's a good standalone tool. With many of the technologies emerging right now, this kind of infrastructure sharing can extend to physical objects--a digital picture frame that speaks the Flickr API, for example.

I believe that mapping user goals to services, and then projecting tasks into tools provides a consistent way to conceptualize how to move away from general-purpose computers to clusters of task-specific technological assistants that share information, while each providing a straightforward interface. Ultimately, that kind of contextual support of everyday activities is what defines both ambience and, maybe from the user's perspective, intelligence.