From "Human-Computer Interaction" to "The Use and Design of Digital Artefacts"

Submission to the CHI 2006 Workshop "What is the Next Generation of Human-Computer Interaction?"

Lars Erik Holmquist
Future Applications Lab, Viktoria Institute
leh@viktoria.se
www.viktoria.se/fal

When seeing the title of this workshop, I realize that in academia, we are often fixated on the computer itself rather than on what it is supposed to accomplish. Our conferences are called things like "Human-Computer Interaction", "Ubiquitous Computing", or "Computer Graphics". Even supposedly alternative paradigms like "ambient" or "pervasive" computing places too much emphasis on *how* things are done, rather than *what* we should be doing.

In fact, the notion of "human-computer interaction" is a remnant of an increasingly outmoded way of thinking about digital technology. The term "computer" itself is now so loaded with meaning, that looking for new ways to "interact with computers" can be genuinely counterproductive. Instead of looking for new *human-computer interfaces,* what we should think about is *interaction models* and *applications* where humans and computing (rather than computers) co-exist. I propose that rather than "human-computer interaction", we should talk about "the use and design of digital artefacts."

Let's take a few steps back. During the last decade, HCI researchers have been developing a multitude of alternatives to the traditional desktop computer. In the hope of improving how we interact with digital information, we have seen a number of new paradigms, such as virtual reality, graspable interfaces, tangible interfaces, augmented reality, mixed reality and so on.

But at the same time, re-thinking the desktop computer has become a largely irrelevant pursuit. It is a battle that has already been both won and lost. Won, in the sense that the computer we have today is as successful an incarnation of the original vision of "intuitive interfaces" as we could hope for. Lost, in the sense that progress has already bypassed the "computer" – and a large part of the HCI community with it.

The modern computer interface came together at Xerox PARC in the mid-70's, building on research in the 60's on interactive graphics, input devices, networking and other fundamental technologies. After Xerox had failed to capitalize on their inventions, Apple licensed some of the patents and in 1984 launched the Macintosh. The rest is history, as windows, icons, mouse and pointers eventually became the standard for interacting with computers. So successful was this paradigm that the gigabyte and gigahertz workstations we all use today are – on the surface at least – hard to distinguish from a Xerox terminal of almost 30 years ago. But if capabilities like memory and speed have increased many thousand times, why are we still using the same tired setup of screen, keyboard and pointer to get data into and out of the computer?

The main reason is that while the idea was excellent to start with, technology took a while to catch up. The Xerox Star, launched in 1981, had just about everything we expect of a computer today; WYSIWYG editing, intuitive graphical interfaces, networking with laser printers and servers, and so on. It was also practically unusable. The graphics-intensive software and flexible networking capabilities coupled with a slow processor and limited memory led to a system that had no chance of running at reasonable speed. The engineers had been so concerned with making a "perfect" system that they failed to take into account if it was actually feasible given the technological limitations of the day.

Around the same time, engineers at a rival company were working within strict technical limitations on a project that turned out to be much more successful: the IBM Personal Computer, or PC. Compared to the Star, the PC was laughably primitive – no networking, no graphical interface, no WYSIWYG editing, not even a mouse. But it worked. And the PC standard turned out to remarkably flexible, so that over the years it has been able to accommodate all the Star's features and more – turning it into the computer that Xerox tried, and failed, to make.

Today's desktop computer is very close to what the engineers at Xerox tried to create, but couldn't with the technology of the day. The interface is by no means perfect, but for what it does it is pretty damn good – and it is not likely to change. For information processing, document creation, network management, and the hundreds of other tasks we perform at the desktop, it works just fine, thank you very much. And desktop computers with graphical user interfaces are used by hundreds of millions of people around the world. The battle has been won – the intuitive computer interface is here.

This remarkable success story is perhaps the best confirmation possible that a sufficiently strong vision of interactive technology can have a huge – if in this case substantially delayed – impact. The interface that the engineers and designers at Xerox envisioned just needed a few decades of increasing computing power to become viable, but the basic idea was good enough to still be in use 30 years later.

Today, there is no need to re-think the fundamental way in which we interact with desktop computers. Instead, if we want to progress we should be looking for applications that are completely *orthogonal* to the desktop. This means abandoning not only the idea of a "computer", but even that of an "interface" altogether!

The problem is more deeply rooted than many perhaps realize. The traditional (and technically correct) view of a computer is a programmable central processor with which humans communicate using a set of input and output devices. This view has been relevant ever since the first digital computers right up until the last decade or so, and is still correct when it comes to the desktop computer. But it is not the right way to think for future digital applications. As computing becomes diffused into the world, it is not meaningful anymore to separate "computer" and "interface" into distinct entities.

Yet, this is exactly the thinking behind many popular new interface paradigms. One example is "tangible interfaces". The vision described by Ishii and Ullmer (1997) was to turn "painted bits" (graphical user interfaces) into "tangible bits" (tangible user interfaces). This involved creating

new physical input and output devices, such as Phicons, Ambient displays, etc. These input and output devices were all clearly different from the mouse, keyboard and screen paradigm; yet, at heart they were fundamentally the same. Rather than being a new model for interacting with computation, tangible interfaces keep the same mental model of a computer – a central processor coupled with a set of inputs and outputs. This model is still obviously reflected in popular toolkits for building tangible interfaces, such as Phidgets (Greenberg and Fitchett 2001), where new interactions devices are always controlled by a central (desktop) computer.

A problem with this way of thinking is that it decouples the computation from the actual activity. The "computer" is on one side of a barrier, with the human on the other, and the interface sit in-between. It can lead to the mistake of thinking that it is enough to re-design the *interface,* whereas in fact the *whole activity* must be considered as a unit if a real change is to be accomplished. But the "computer" (or rather the program running on it) is not an abstract entity isolated from the user – because if we change what the computer does, we also change the interaction.

Therefore, creating novel input and output devices is simply not enough, if we keep the same mental model of a human and a computer separated by an interface. "Re-thinking the way we interact with computers" to find the interactive applications of the future is about as fruitful as making a new running shoe by "re-thinking the car".

If we continue the analogue of a car, we do not think of the gas pedal as a "human-motor interface", or the steering wheel as a "human-wheel interface" even though that is technically correct (in the same way that the mouse or GUI is a "human-computer interface"). Instead, we think of them as controlling the things a car should do – taking us from one place to the other. In the same way, we do not "interact with computers" anymore, we "do things that are supported by computations" such as write documents listen to music, etc. – just like we do not "interact with the motor and the wheels" in a car, we "drive from A to B".

An alternative way of thinking that I find attractive, but still not completely satisfactory, is the "augmented artefacts" approach (Beigl et al 2001). This was also the way of thinking that motivated the Smart-Its project, a system for digitally augmenting everyday artefacts with sensing and computation (Holmquist et al 2004). In this approach, the starting point is an everyday artefact that is augmented with sensing and computation. The original function of the artefact is unaffected by this augmentation but additional functionality is added.

In the case of the *MediaCup*, for instance, an ordinary coffee was augmented by adding a processor, heat and movement sensors, and an infrared transceiver. The original function remains – the cup is still good for drinking from – but the cup is also context-aware, and can for instance give a report on its position and usage. While the MediaCup still required a central computer for data processing (as did most, but not all, Smart-Its-based demonstrations) it was still a break from the human-computer interface mental model in that users never consciously "interacted" with a computer – they just used a coffee cup as they would have done anyway.

Of course, I am not the first one to try to drop the "computer" from the field of human-computer interaction. The terms "interaction design" and, more recently "user experience design" are commendable alternatives. "Information appliances" have been introduced to describe smaller,

more specialized digital devices. And the original vision of "ubiquitous computing" as proposed by Mark Weiser captured much of what I am looking for, in particular by explicitly substituting "computers" with "computing", thus emphasising what computers do rather than the actual computer (a fine point that is often lost!)

However, for the sake of this workshop, I wanted to find a term that focuses on the specific artefacts that will usher in the next era, not of human-computer interaction, but of human activities augmented by digital technology. Therefore, I propose to replace "human-computer interaction" with "the use of digital artefacts". This captures both the activity aspect ("use"), that we are doing something new ("design"), the reliance on computation ("digital"), and finally, that we do not consider interface and computer separate but the entire object as a whole ("artefact"). It is a slightly unwieldy term, but it will do for now.

I look forward to discussing this further at CHI 2006 – no matter what the workshop happens to be called!

References

Beigl, M., HW Gellersen, A Schmidt (2001). Mediacups: experience with design and use of computer-augmented everyday artefacts. *Computer Networks,* 2001.

Greenberg. S. and Fitchett, C. (2001) Phidgets: Easy development of physical interfaces through physical widgets. Proceedings of ACM UIST 2001 User Interface Software and Technology, 2001.

Holmquist, L.E. et. al. Building Intelligent Environments with Smart-Its. *IEEE Computer Graphics & Applications,* January/February 2004, pp 56-64.

Ishii, H. and Ullmer, B. (1997). Tangible Bits: Towards Seamless Interfaces between People, Bits and Atoms, in *Proceedings of Conference on Human Factors in Computing Systems (CHI '97),* (Atlanta, March 1997), ACM Press, pp. 234-241.