For a device whose fundamental properties have changed so radically over the past thirty years, the personal computer itself—the familiar beige box sitting by the desk—has changed remarkably little.

The personal computer (PC) as we currently know it has its origins in work carried out at Xerox’s Palo Alto Research Center in the early 1970s. The forerunner of the modern PC was, arguably, the Alto workstation developed by researchers there; it pioneered such now-common features as bitmapped displays with overlapping windows, graphical interfaces with multiple fonts and pop-up menus, and machines linked together over local-area networks. Although underpowered by today’s standards (it was clocked at 6 MHz rather than the many hundreds of today’s PCs), it nonetheless set the stage for what was to come, and its basic feature set, built around “the three ‘M’s’”—millions of pixels, a megabyte of memory, and a million instructions per second—is still with us today.

On the other hand, an Alto in those days cost around $16,000 to build, scarcely affordable enough to put “a computer on every desk,” as Microsoft would later set out to do. A more affordable option in 1977 (by which time the PARC researchers were working on the Dorado, a considerably faster and more powerful machine) was the Apple II, the device which, arguably, kick-started the personal computer industry. The Apple II was powered by a 6502 8-bit processor running at 1.5 MHz. It had 8 kilobytes of semiconductor memory and stored programs on cassette tape; optional floppy disk drives stored around 150 kilobytes each. Compare that to the modern personal computer. The laptop computer on which I’m writing this is certainly not top-of-the-line; it wasn’t even top-of-the-line when I bought it a year ago. It has a 166 MHz 32-bit...
processor, 64 megabytes of memory, and a 13-inch color display and can store up to 6 Gb on an internal hard disk; and it cost under $4,000.1

Imagine what it would be like if any other technology had undergone such rapid advances in price/performance. A car would cost a few dollars; airplanes would travel at hundreds of times the speed of sound; televisions would weigh a few ounces. More to the point, if cars, airplanes, and televisions had been so radically transformed, they would not be cars, airplanes, and televisions any more. They would have transformed themselves into something else altogether.

Computers, though, remain computers. As we enter the twenty-first century, today's PC still looks remarkably similar to that of the late 1970s (and perhaps even more like the Alto of the earlier part of that decade; see figure 2.1). This is not simply a matter of packaging and

![Figure 2.1](image.png)

Xerox's Alto (1974). This early personal computer is somewhat bulkier than today's, but is otherwise very recognizable in form. Reprinted by permission of Xerox Palo Alto Research Center.

industrial design, although it is certainly the case that with a few notable exceptions, we seem to be firmly stuck in an age of beige boxes. My concern is not so much about the boxes themselves as about the relationship of the user to the box. Despite the fact that computers are so radically different from the computers of twenty years ago, and that their capabilities are so vastly different, we interact with them in just the same way; we sit at a desk, watching the screen and typing on the keyboard. If you were to look at a photograph of people using computers some time over the last twenty years, their clothes and hairstyle might give you a clue to the date when the picture was taken, but the style of interaction with the computer certainly would not.

Similarly, the style of interaction concerns not simply the set of physical devices (keyboards, screens, and mice) or the set of virtual devices (dialog boxes, scroll bars, and menus) through which we interact, but also the ways in which the computer fits into our environments and our lives. Interaction with screen and keyboard, for instance, tends to demand our direct attention; we have to look at the screen to see what we're doing, which involves looking away from whatever other elements are in our environment, including other people. Interaction with the keyboard requires both of our hands. The computer sits by the desk and ties us to the desk, too. So, it is not simply the form of the computer that has changed remarkably little over the last thirty years; it is also the forms of computer-based activity and the roles that we imagine computers playing in our everyday lives.

Although this model of everyday computing might be conventional, it is not inevitable. The rise of the personal computer—and, more broadly, of personal computing—was an attempt to break away from the then-dominant paradigm of mainframe computing. Similarly, while personal computing may now be established as the dominant model, a variety of alternatives have been explored in the research community; departures from the world of the conventional PC as radical as the PC was from the world of the mainframe. In this chapter, I will take a brief tour through some of the research laboratories where these alternatives are being explored. In particular, I will focus on an approach that looks at the relationship between computers on the desktop and the world in which they (and we) operate. This is a model of interaction that I refer to as “tangible
computing.” Although it is only lately that the tangible computing paradigm has become broadly established, it has emerged from a research program that stretches back over a decade.

**Ubiquitous Computing**

We begin the tour, ironically enough, in the Computer Science Lab at Xerox PARC—the same place that gave us the desktop PC. In the 1970s, Xerox had set up PARC to explore “the architecture of information,” and the Computer Science Lab, under the guidance of former ARPA manager Bob Taylor, had delivered what was to become the basic elements of office information technology in the decades to follow—powerful personal workstations, laser printers, and shared servers, linked together on local area networks. Xerox, famously, had failed to recognize its own future in PARC’s vision, so today’s office technology generally doesn’t carry a Xerox label (Smith and Alexander 1988).

By the start of the 1990s, the situation was different. PARC’s vision of the architecture of information had, largely, come to pass; and, in the opinion of the new manager of the Computer Science Lab, Mark Weiser, it was time for a new and equally radical vision of the future of technology.

What Weiser proposed was a research program that he dubbed “Ubiquitous Computing.” Weiser saw that the development and diffusion of general-purpose computers, and in particular PC’s, had resulted in a focus on the computer rather than on the tasks that the computer was used to accomplish. He argued that ongoing technological developments, particularly in mobile and low-power devices, would transform the nature of computers and the way we interact with them. Why deal with a single, large, expensive computer when you could harness many tiny, low-cost devices spread throughout the environment? Instead of always taking work to the computer, why not put computation wherever it might be needed? Through the technical developments that supported this new model, he saw an opportunity to turn attention away from the dominating focus on the computer sitting on the desktop and back to the applications, and to the artifacts around which those applications were structured. Weiser’s vision of “ubiquitous computing” was one of computationally enhanced walls, floors, pens, and desks, in which the power of computation could be seamlessly integrated into the objects and activities of everyday life.

One analogy that Weiser proposed as a way of understanding his vision for the new role of computation was that of solenoids, the electronically actuated switches that are part of the fabric of many everyday technologies. For example, he observed, a modern car has a vast number of solenoids, invisibly controlling everything from the air conditioning to the fuel intake. Solenoids are a critical component of modern technological design and are used in all sorts of settings. And yet, we don’t deal directly with solenoids in the way we do with computers. We don’t have to think about the design of “human-solenoid interface”; we don’t have programs on “solenoid literacy” in schools; you can’t take a degree in “solenoid science,” and nobody has to upgrade to “Solenoids 2000.”

Why have computers and solenoids followed different paths? Various possibilities present themselves. Perhaps it is because of the nature of computers as multipurpose devices; or perhaps it is a historical accident, a feature of how computer technology was introduced into the home and work environments. And to be sure, there are all sorts of computer technologies surrounding us that are far more like solenoids than they are like PCs, such as the computer processors inside my television set, microwave oven, and car. The difference between my PC and those other devices is that those other devices are organized around human needs and functions.

Weiser’s model of ubiquitous computing was also, paradoxically, one of invisible computers. He argued for a vision of computers in which the computer had become so ubiquitous that it had, essentially, disappeared. He proposed that the computer of the twenty-first century would have proceeded further along the path from the mainframe to the processor in my microwave oven, and that the intermediate step—the desktop PC—would be all but gone. However, in this world, although there might be no more computers as we understand them today, there would certainly be computation. In fact, there might be a great deal more computation than there is now. Computational devices would be embedded in all sorts of technologies, Weiser argued, creating a variety of specialized devices augmented with computational power. Computers would
disappear into the woodwork; computers would be nowhere to be seen, but computation would be everywhere.

Computation by the Inch, Foot, and Yard
In the Computer Science Lab at Xerox PARC, Weiser initiated a wide-ranging research program around his vision of Ubiquitous Computing, fostering the development of new computational technologies, the infrastructure necessary to support them, and new application models. PARC’s ubiquitous computing strategy followed three tracks: they were known as computation by the inch, the foot and the yard (see figure 2.2).

“Computation by the inch” focused on the development of small devices, like electronic tags or computational “Post-It” notes. One focus of attention was the use of devices called “Active badges,” originally developed at the Olivetti Research Centre in Cambridge, England (Want et al. 1992). Active badges are devices measuring roughly 1.5 inches square that are intended to be worn like normal identity badges. However, they house some simple electronics and emit a fixed, coded infrared signal every thirty seconds or so (or whenever a button on the badge is pressed). These signals are detected by a network of infrared receivers located in the environment, and which are connected to a computational server process. Because each badge emits an individual code, and because its signal will generally only be received by the closest detector, the server can maintain a map of the location of each badge within the sensor network, which in turn can locate the badge’s wearer within the environment.

When people wear active badges, then applications can help make the environment responsive to their movements. The system can route telephone calls to the current location of the person being called, display relevant information on nearby monitors as they pass by, or customize the behavior of a computer system to the needs of the person sitting at it. In Weiser’s model, badges or similar tags could also be attached to books and other artifacts, so that their location and mutual proximity could become a resource to computer-based applications.

If computation “by the inch” sought a model of computationally enhanced Post-It Notes, the computation “by the foot” was concerned with computationally enhanced pads of paper. The primary focus of this

Figure 2.2
Computing by the inch, the foot, and the yard: (a) an active badge, (b) the PARC Tab, (c) the PARC Pad, and (d) a meeting at the Liveboard. Reprinted by permission of Xerox Palo Alto Research Center.
area of work was the development and use of computational devices of about the size and power of recent laptop computers. Laptop computers were, of course, already widely available at this point, but they tended (as they still do) to function simply as scaled-down versions of their desktop cousins. In contrast, the goal of ubiquitous computing research was not simply on the size and packaging of the devices, but of how they would fit into a world of everyday activities and interaction. As a result, research concentrated on other concerns. Examples included stylus-based interaction, which could eliminate keyboards as the primary source of interaction, and which could support note-taking and sketching, and mobile operation, so that devices could be moved from place to place without interfering with their operation.

Finally, investigations into computation “by the yard” introduced the opportunity to consider much larger devices. In particular, attention focussed on wall-sized devices such as the LiveBoard. LiveBoard was a large-scale display (approximately five feet by three feet) supporting multiple pens, a sort of computationally enhanced whiteboard. Researchers observed how the very physical form of this device was an important component in structuring interactions with it. On the one hand, the use of pen input meant that collaborative activities (such as brainstorming in a meeting) would be implicitly structured by the fact that the board was large enough for everyone to see at once, but that two people could not stand in front of the same part of the board or write in the same area at the same time. On the other hand, the board’s large size also meant that new interaction techniques would have to be developed; using a scroll bar or pull-down menu on a board a board five feet wide could be, quite literally, a pain in the neck.

Discussing each of these components of PARC’s ubiquitous computing strategy independently can mask the critical integration of the various facets of the program. None of these devices was intended to operate on its own. The focus, after all, was on a form of computation more deeply integrated with the everyday environment, and the everyday environment is filled with a variety of objects and devices. So it was with the ubiquitous computing vision. A single user might have, at his or her disposal, tens or more of the inch-sized devices, just as we might have many Post-It notes dotted around, stuck to computer screens, walls, books, and sheets of paper; at the same time, they might also have three or four foot-sized devices, just as I might have a number of notebooks for different topics or projects; but just as I probably only have one or maybe two whiteboards in my office, there will be fewer of the devices at the larger scale. What is more, information is expected to be able to move around between the different devices. Notes that I have prepared on an electronic pad might be beamed onto the board for group consideration in a meeting; while action items might be migrated off into a hand-held device that stores my calendar and to-do list. In the everyday environment, information continually undergoes transformations and translations, and we should expect the same in a computationally enhanced version of that environment such as might be delivered to us by ubiquitous computing.

The Digital Desk

At much the same time as Weiser and his PARC colleagues were developing the ubiquitous computing program, related activity was going on in another Xerox lab, in Cambridge, England. EuroPARC had been set up as a European satellite laboratory of PARC. It was a much smaller lab (with a research complement of around twenty) with a focus on interdisciplinary research into Human-Computer Interaction and Computer-Supported Cooperative Work.

EuroPARC was home to a variety of technological developments, but the particular technology that concerns us here is the Digital Desk, designed and developed by Pierre Wellner (Wellner 1991; Newman and Wellner 1992). In common with many people, Wellner had observed that the “paperless office” envisioned by many in the 1970s and early 1980s had manifestly failed to develop. However, that was not to say that the development of personal computers, and increasingly networked personal computers, had not caused an massive increase in the number of digital or online documents that we all have to deal with everyday. Wellner was concerned with how we could work with both paper and electronic documents in a much more fluid and seamless way than is normally the case. The traditional approach to these problems was either to scan in the paper documents to bring them into the
electronic realm, or to print out the electronic documents to bring them into the physical realm. By moving across the boundary from online documents to paper documents and back again, users could take exploit the advantages of each; the digital malleability and computational power of electronic documents with the portability, readability, and informal interaction of paper ones. As many studies have attested, paper has many properties that are hard to reproduce in the electronic world (Sellen and Harper 1997; Henderson 1998), while, at the same time, electronic documents increasingly exploit features (such as animation, hyperlinks, or interactive elements) that paper documents cannot capture. So, the move back and forth between electronic and paper forms is not only inconvenient but also impoverished, since some features always remain behind. Taking his cue from Weiser’s ubiquitous computing work, Wellner wondered if there wasn’t a way to combine the two worlds more effectively by augmenting the physical world with computational properties.

Wellner’s Digital Desk (figure 2.3) combines elements of each. The Digital Desk was a physical desktop, much like any other, holding papers, pens, coffee cups, and other traditional office accoutrements. However, it was also augmented with some distinctly nontraditional components. Above the desk were placed a video projector and a video camera. Both of these were pointed down toward the desktop; the projector would project images onto the desk, over whatever objects were lying there, and the camera could watch what happened on the desktop. These devices were connected to a nearby computer. Image processing software running on the computer could analyze the signal from the video camera to read documents on the desk and watch the user’s activity. At the same time, the computer could also make images appear on the desk by displaying them via the video projector.

The result was a computationally enhanced desktop supporting interaction with both paper and electronic documents (Wellner 1993). Electronic documents could be projected onto the desktop by the video projector, but then could be moved around the (physical) desktop by hand (using the video camera to track the user’s hand movements and then “moving” the displayed document in coordination). Similarly, physical documents could be given computational abilities on the same
For example, a paper document containing a list of numbers could be used as input to a virtual calculator; the computer could use the camera to “read” the numbers off the printed page, and then project the result of a calculation over those figures.

Two features of the Digital Desk were critical to its design. The first was its support for manipulation. In Wellner’s first prototype, one moved objects around on the desk with one’s fingers; in contrast with the prevailing approach to interface design, this was really direct manipulation. What’s more, of course, while our computer systems typically have only one mouse, we have two hands and ten fingers. By tracking the position and movements of both hands or of multiple fingers, the Digital Desk could naturally support other behaviors that were more complicated in traditional systems, such as using both hands at once to express scaling or rotation of objects. The second critical design feature was the way in which electronic and physical worlds were integrated. A document on the digital desk could consist of both physical content (printed on a page) and electronic content (projected onto it), and printers and cameras allowed material to move from one domain to the other fluidly so that objects created on paper could be manipulated electronically. The Digital Desk offered developers and researchers an opportunity to think about the boundary between the physical and virtual worlds as a permeable one.

While the work on ubiquitous computing had shown how computation could be brought out of the “box on the desk” and into the everyday world, Wellner’s work on the digital desk expanded on this by considering how, once the real world was a site of computational activity, the real and electronic worlds could actually work together.

Virtual Reality and Augmented Reality

Weiser and Wellner shared the goal of creating computationally augmented reality. They both attempted to take computation and embed it in the everyday world. This follows in the trend, outlined earlier, to expand the range of human skills and abilities on which interaction can draw. In this case, the abilities to be exploited are those familiar ways in which we interact with the everyday world; drawing on whiteboards, moving around our environments, shuffling pieces of paper, and so on. One of the interesting feature of these approaches, at the time, was the way in which they developed in opposition to another major trend—immersive virtual reality.

Virtual reality (VR) is, at least in the popular consciousness, a technology of recent times; it became particularly prominent in the 1990s. Immersive VR as we know it today came about through the increase in computer power, and particularly graphics processing, that became available in the late 1980s, as well as some radical sensor developments that gave us data gloves and body suits. The technical developments supporting immersive VR became widespread at around the same time as William Gibson’s notion of “cyberspace”—a technologically mediated hallucination in which people and technology interacted—also entered the popular consciousness. Virtual reality has been around a good deal longer than that, however. Ivan Sutherland, the father of interactive computer graphics, went on to investigate what we now recognize as virtual reality technology back in the 1960s, and the use of digital technology to create environments such as flight training simulators is well-known. Howard Rheingold’s book Virtual Reality (1992) documents some of the early history of this seemingly recent technology.

Virtual reality immerses the user in a computationally generated reality. Users don head-mounted displays, which present slightly different computer-generated images to each eye, giving the illusion of a three-dimensional space. By monitoring the user’s head movements and adjusting the image appropriately, this three-dimensional space can be extended beyond the immediate field of view; the user can move his head around, and the image moves to match. With appropriating sensing technologies, the user can enter the virtual space and act within it. A “dataglove” is a glove augmented with sensors that report the position and orientation of the hand and fingers to a computer; the hand of the user wearing the glove is projected as a virtual hand into the same computer-generated three-dimensional space that the virtual reality system generates, so that the user can pick up virtual objects, examine them, move them around, and act in the space.

The ubiquitous computing program was getting under way at about the point when virtual reality technology began to make its way out of
research laboratories and into newspaper articles. Both approaches to
the future of computing are based on similarly science-fiction notions;
immersion in a computer-generated reality, on the one hand, and com-
puters in doorknobs and pens on the other. They embody, however, fun-
damentally different approaches to the relationship between computers,
persons and the world. In the virtual reality approach, interaction takes
place in a fictional, computer-generated world; the user moves into
that world, either through immersion or, more commonly these days,
through a window onto the world on a computer screen. The world of
interaction is the world of the computer. The ubiquitous computing
approach to interaction—what Weiser dubbed “physical virtuality” and
would become known as augmented reality—does just the opposite. It
moves the computer into the real world. The site of interaction is the
world of the user, not that of the system. That world, in the augmented
reality vision, may be imbued with computation, but the computer itself
takes a back seat.

The Reactive Room

The ubiquitous computing model distributes computation throughout
the environment. All sorts of objects, from walls to pens, might have
computational power embedded in them. For someone concerned with
interaction, this raises one enormous question—how can all this com-
putation be controlled?

At the University of Toronto, Jeremy Cooperstock and colleagues
explored this question in an environment they called the Reactive Room
(Cooperstock et al. 1995). The Reactive Room was a meeting room sup-
porting a variety of physical and virtual encounters. It grew out of both
the ubiquitous computing perspective and the “media space” tradition,
an approach to supporting collaboration and interaction through a com-
bination of audio, video, and computational technology (Bly, Harrison,
and Irwin 1993). The room was designed to support not only normal,
face-to-face meetings, but also meetings distributed in space (where
some participants are in remote locations) and time (recording meeting
activity to be viewed later by someone else). To that end, it also featured
a shared computer display, for electronic presentations and application-
based work; a variety of video and audio recorders; and audio and video
units connected to a distributed analog A/V network that could be con-
ected to similar “nodes” in people’s offices, so they could remotely
“attend” meetings.

However, such a complex and highly configurable environment pre-
sented considerable challenges for control and management. To config-
ure the room for any given situation (such as a presentation to be
attended by remote participants), each device in the room would have to
be configured independently, and adjusting the configuration to support
the dynamics of the meeting was even more challenging. The design of
the Reactive Room sought to use ubiquitous computing technology as a
means to manage this problem. The critical move here was to see ubiq-
uitous computing as a technology of context; where traditional interactive
systems focus on what the user does, ubiquitous computing technologies
allow the system to explore who the user is, when and where they are
acting, and so on.

In the case of the reactive room, contextual information could be used
to disambiguate the potential forms of action in which a user might
engage. For example, by using an active badge or similar system, the
room’s control software can be informed of who is in the room and can
configure itself appropriately to them. Similarly, if the room “knows”
that there is a meeting in progress, then it can take that information into
account to generate an appropriate configuration. If a user presses the
“meeting record” button on a VCR, to record a meeting in progress, the
Reactive Room can determine whether or not there are any remote par-
ticipants connected to the audio/video nodes and, if so, ensure that it
adds those signals to the recording. When someone in the room makes
use of the document camera or the projected computer display, the room
software can detect these activities and automatically make the docu-
ment camera view or the computer display available to those people
attending the presentation, either locally or remotely.

In other words, the design of the Reactive Room attempts to exploit
the fact that the people’s activities happen in a context, which can be
made available to the software in order to disambiguate action. Clearly,
of course, the sort of context that can be gathered with current tech-
ology is limited; the Reactive Room would make use of motion in
particular parts of the room, presence and activity as detected using active badges or pressure sensors, and so on. The other, perhaps most important, piece of context it made use of was the fact that it was the Reactive Room. That is, the room was designed for meetings and presentations, and so much activity in the room could be interpreted as being appropriate to meetings and presentations. The same sorts of inferences would probably be inappropriate in other settings, such as a private office, or a home. The “meeting” context, then, also serves to disambiguate the user's goals.

The Reactive Room demonstrated the way that ubiquitous computing did not simply move out of the box on the desk and into the environment but, at the same time, also got involved in the relationship between the environment and the activities that took place there. The topic of “setting-ed” behavior will come back into focus in the next chapter; for the moment, however, we will continue to explore the development of tangible computing.

Design Trends

The systems that have been described—the vision of Ubiquitous Computing, and the Digital Desk and Reactive Room prototypes—have been firmly located in the domain of Computer Science research. However, “academic science” has by no means been the only contributor to the development of Tangible Computing. In fact, one striking aspect of the development of this line of investigation has been the contributions from the perspectives of art and design. Two pieces that have proved to be particularly inspirational to a number of researchers in this area were Durrell Bishop’s Marble Answering Machine, and Natalie Jeremijenko’s Live Wire.

The Marble Answering Machine was a design exercise undertaken by Bishop in the Computer-Related Design department at the Royal College of Art in London (Crampton-Smith 1995). It explored possible approaches to physical interaction for a telephone answering machine. Rather than the traditional array of lights and buttons, Bishop’s answering machine has a stock of marbles. Whenever a caller leaves a message on the answering machine, it associates that message with a marble from the stock, and the marble rolls down a track to the bottom, where it sits along with the marbles representing previous messages. When the owner of the machine comes home, a glance at the track shows, easily and distinctly, how many messages are waiting—the number of marbles arrayed at the bottom of the track. To play a message, the owner picks up one of the marbles and drops it in a depression at the top of the answering machine; because each marble is associated with a particular message, it knows which message to play. Once the message has been played, the owner can decide what to do; either return the marble to the common stock for reuse (so deleting the message), or returning it to the track (saving it to play again later).

The Marble Answering Machine uses physical reality to model the virtual or electronic world. In Bishop’s design, marbles act as physical proxies for digital audio messages. By introducing this equivalence, it also enriches the opportunities for interacting with the device. The problem of interacting with the virtual has been translated into interacting with the physical, and so we can rely on the natural structure of the everyday world and our casual familiarity with it. So, counting the number of messages is easy, because we can rapidly assess the visual scene; and operations such as playing messages out of order, deleting messages selectively, or storing them in a different sequence, all of which would require any number of buttons, dials, and controls on a normal digital answering machine, all become simple and straightforward because we can rely on the affordances of the everyday world.

Natalie Jeremijenko’s piece “Live Wire,” also sometimes known as “the Dangling String” and described by Weiser and Brown (1996), was developed and installed at Xerox PARC in 1994 and explored similar questions of the boundary between the virtual and physical worlds. Physically, Live Wire was a length of plastic “string” around eight feet long, hanging from the ceiling at the end of a corridor. Above the ceiling tiles, the wire was connected to a small stepper motor, which in turn was connected to a device on the local ethernet. Every time a data “packet” passed by on the ethernet, the stepper motor would move, and its movements would be passed on to the string. Ethernet, in its classic form, is a “shared medium” technology—all the traffic, no matter which machine sends it or which machine is to receive it, travels along the same cable.
The busier the network, the more data packets would pass by, and the more the stepper motor would move. The ethernet can carry thousands of packets per second, and so when the network was busy the motor would whir and the string would spin around at high speed, its loose end whipping against the wall nearby.

Others have followed in the footsteps of Bishop and Jeremijenko and continued to explore the design “space” around these issues of the borders between physical and virtual worlds. Feather, Scent, and Shaker (Strong and Gaver 1996) are devices for “simple intimacy.” “Feather” features a feather that is gently lifted on a column of air, to indicate to its owner that, perhaps, a photograph of them has been picked up somewhere else; it is designed to convey a sense of fondness across distance. Scent, similarly, releases a pleasant, sweet smell in similar circumstances providing an awareness of distant action.

The topic of “awareness” is one that has concerned the developers of technologies for group working, who want their systems to be able to support the casual and passive awareness of group activity that coworkers achieve in a shared physical space. Strong and Gaver turn this around, though, and give us technologies for supporting shared intimacy rather than shared work. Their pieces are designed to be evocative and emotive rather than “efficient.” What is particularly interesting about this group of devices is that they originate not from a technical or scientific perspective, but from a design perspective. The result of this shift in perspective is that they reflect a very different set of concerns. It is not simply that they reflect an aesthetic component where the scientific developments are marked more by engineering concerns. That is certainly one part of it, of course; the design examples certainly do reflect a different set of principles at work. However, there is more than this.

First, the design examples discussed here reflect a concern with communication. What is important is not simply what they do, but what they convey, and how they convey it; and the communicative function that they carry is very much on the surface. There is an “at-a-glance readability” to these artifacts that stands in marked contrast to the “invisibility” of ubiquitous computing. Second, they reflect a holistic approach that takes full account of their physicality. The physical nature of these pieces is not simply a consequence of their design; it is fundamental to it. While it was a tenet of ubiquitous computing, for example, that the technology would move out into the world, the design pieces reflect a recognition that the technology is the world, and so its physicality and its presence is a deeply important part of its nature. Third, they reflect a different perspective on the role of computation, in which computation is integrated much more directly with the artifacts themselves. In the other examples, while they have aimed to distribute computation throughout the environment, there has always been a distinct “seam” between the computational and the physical worlds at the points where they meet. In these examples, however, the computational and physical worlds are much more directly connected.

The result is an approach to tangible computing that sees computation within a wider context. Ubiquitous Computing pioneers saw that, in order to support human activity, computation needs to move into the environment in which that activity unfolds. These design explorations take the next step of considering how computation is to be manifest when it moves into the physical environment, and recognizing that this move makes the physicality of computation central.

**Tangible Bits**

Most recently, perhaps the most prominent site for development of these ideas has been the Tangible Media group at the MIT Media Lab. A group of researchers led by Hiroshi Ishii has been exploring what they call “Tangible Bits,” a program of research that incorporates aspects of both the Ubiquitous Computing program and the design perspective explored by people like Jeremijenko.

The term “Tangible Bits” reveals a direct focus on the interface between the physical and virtual worlds. The rhetoric of the computer revolution has, pretty consistently, focused on a transition from physical (the world of atoms) to the virtual (the world of bits). We talk of the future in terms of “electronic cash” to replace the paper bills and coins we carry about with us, or we speak of the “paperless office” in which paper documents have disappeared in favor of electronic documents stored on servers and displayed on screens. We envision a world in which we communicate by electronic mail and video conferencing, in
which we read from “e-books,” telecommute over great distances via digital communication lines, and play in virtual worlds. What these visions have in common is the triumph of the virtual over the physical. They suggest that we will overcome the inherent limitations of the everyday world (such as the need to be in the same place to see each other, or that a thousand books actually take up real shelf space) by separating the “information content” from the physical form, distilling the digital essence and decanting it into a virtual world.

The MIT Media Lab, where Ishii and his colleagues are based, is one of the most prominent proponents of this vision, especially, perhaps, in the writings of its founding director, Nicholas Negroponte. His collection of essays *Being Digital* (Negroponte 1995), explores the relationship between atoms and bits and how the development and deployment of Internet technologies is changing that relationship.

The work on Tangible Bits provides some balance to the idea that a transition from atoms to bits is inevitable and uniformly positive. It is certainly not defined in opposition to the gradual and ongoing movement of traditionally physical forms into digital media. However, it observes that while digital and physical media might be *informationally* equivalent, they are not *interactionally* equivalent. By building information artifacts based on physical manipulation, the Tangible Bits programme attempts to reinvest these distilled digital essences with some of the physical features that support natural interaction in the real world.

**metaDESK, Phicons, and Tangible Geospace**

Let’s take an example from the work of the Tangible Bits group. The metaDESK (Ullmer and Ishii 1997) is a platform for tangible interaction. It consists of a horizontal back-projected surface that serves as the top of the physical desk itself; an “active lens,” which is a small flat-panel display mounted on an arm; a “passive lens,” which is transparent, also digitally instrumented; and a variety of physical objects called *phicons* (for “physical icons”). The metaDESK is shown in figure 2.4.

The functions of the various components of the metaDESK platform are best seen in terms of an application running on the desk. Tangible Geospace is a geographical information system augmented with tangible UI features and running on the metaDESK. It allows users to explore a visualization of a geographical space, such as the area of Cambridge, Massachusetts, around MIT.

The geographical information, in the form of a two-dimensional map, is back-projected onto the desk, so that the user seated at the desk can see it. The user can move and orient the map using phicons. One of the phicons represents MIT’s Great Dome, and when it is placed on the desk, the map is adjusted so that the position of the Great Dome corresponds to that of the phicon. As the user moves the phicon, the system adjusts the map to ensure that the phicon is always aligned with the point on the map that it represents. By moving the phicon around on the desk, the user can cause the map to move too, “scrolling” around in the geographical space. By rotating the phicon on the desk, the user can cause the map to rotate.

If a second phicon is added to the desk, say one representing the Media Lab building itself, then another degree of freedom can be constrained. The two icons, together, can be used to control the scale of the
map display. If the metaDESK always ensures that the virtual Great Dome always co-occurs with the Great Dome phicon, and the virtual Media Lab always co-occurs with the Media Lab phicon, then the user can control the scale of the map by moving these two phicons closer together or further apart.

The active and passive lenses can be used to provide access to other sorts of information. In the Tangible Geospace example, the active lens is used to view a three dimensional model of the MIT Campus. The active lens is a computer display mounted on an arm over the desk. It is instrumented so that the metaDESK computer system can determine the position and orientation of the display. When this information is coordinated with the current position, scaling, and orientation of the map being displayed on the desk, the result is that the active lens can be used to control a “virtual camera” moving through the geographical space being displayed on the metaDESK. When this is combined with a three dimensional model of the campus, then the active lens can be used to give a three-dimensional viewport onto the two-dimensional map. The illusion is of “looking through” the lens and seeing a transformed view of the map underneath.

The passive lens works in a similar way, although it rests on the desk surface. The passive lens is simply a piece of transparent plastic. As it is moved around the desk, the computer system can track its current location. On the desk area directly underneath the lens, the metaDESK replaces the map with a view onto a photographic aerial record of the campus. As before, this is correlated with the current position, scaling, and orientation of the basic map, as well as the position of the lens. The effect is that it seems to the user that the lens reveals the photographic model underneath as it moves across the desk. This is similar to a user interface technique known as “magic lenses” (Bier et al. 1993), user interface components that selectively transform the content of interfaces as they are moved across the screen, although, of course, in the case of the metaDESK the lens has a physical manifestation.

The Ambient Room

Tangible interfaces such as the metaDESK explore interaction that is situated in the environment, rather than on a screen. This is even more clearly demonstrated by another of the MIT prototypes, called the Ambient Room (Wisneski et al. 1998).

The Ambient Room is a small office cubicle that has been augmented with a variety of “ambient displays,” designed to provide peripheral, background information to the occupant of the room without being overwhelming or distracting. Examples of ambient displays include projected light patterns, non-speech sounds, and objects that respond to changes in air flow.

The information that the Ambient Room conveys is typically information about activities in either physical or virtual space, such as the presence or activity of others, e-mail arriving, people logging in and out, and so forth. These can be mapped onto the displays available in the room. For instance, light patterns projected on the wall can respond to the activities of a networked computer system, conveying information about network traffic and hence activity in the virtual space; or movements in a shared project room can be mapped onto subtle sounds in the Ambient Room so that the occupant can be aware of comings and goings in the project space. Reminiscent of the Feather, Scent, and Shaker work of Strong and Gaver, these ambient displays can be used to project the actions in one space (either physical or virtual) into another; like the technologies of the Reactive Room, they can also respond to the activity of the room’s occupant, providing a display that is appropriate to the context in which they are working.

It is tempting to think of the metaDESK as exploring the potential for tangible media as input technologies, and the Ambient Room as exploring their potential for output. To do so, though, would be to miss an important point, which is that, in the everyday environment, “input” and “output” are fundamentally interconnected. This is a critical feature of the tangible media explorations. They should be characterized not in terms of “input” and “output,” but in terms of the coordination between phenomena; between activity in a space and the pattern of light on a wall, or between the movement of objects on the desk and the information presented there. This sort of coordination, or coupling, is fundamental to the explorations presented here; they depend upon it for the causal illusion they want to maintain.
Illuminating Light and Urp

Two other applications developed in the MIT group echo the Digital Desk in their creation of mixed physical/virtual environments for task-focused work. These are Illuminating Light and Urp, both developed principally by John Underkoffler (and illustrated in figure 2.5).

Illuminating Light (Underkoffler and Ishii 1998) is a simulation of an optics workbench, aimed particularly at students of laser holography. The interface is based on a combination of phicons and a camera/projector arrangement (which Underkoffler dubs the “I/O Bulb”) similar to that of the Digital Desk. The application allows users to experiment with and explore configurations of equipment for laser holography. Real laser holography is a complex business, conducted using delicate and expensive instruments. Setting up and fine-tuning an experimental configuration can be extremely time-consuming, especially for novices. Illuminating Light allows holographers to simulate the effects of particular configurations and to explore them so as to develop a better intuitive sense for the interaction of their elements. Phicons represent physical elements such as lasers, lenses, mirrors, and beam-splitters, while the system provides a simulation of light paths through the experimental equipment, showing light emitted by the laser, redirected by mirrors, and so on. As the phicons are moved around a physical surface, the system continually updates its projection of the simulated light paths to reflect the moment-by-moment physical configuration. In addition to the simulated light beams, the system can also provide numerical descriptions of the configuration: incidence angles, distances, and so forth. In this way, users can rapidly explore a variety of configurations and develop an understanding of the consequences of different changes on the set-up.

Urp (Underkoffler and Ishii 1999) is an urban planning workbench in which physical models of buildings are combined with electronic simulations of features such as air flow, cast shadows, reflectance, and so forth. The underlying technology is similar to that of Illuminating Light but applied to a different domain. There are two sorts of phicons used in Urp. The first represent building structures. By placing these on the surface, the user can obtain a visualization of the shadows that the buildings will cast, or the wind patterns around them. Combining multiple structures allows urban planners and architects to explore the

Figure 2.5
Illuminating Light (a) and Urp (b) apply tangible interaction techniques to the domains of optics and urban planning. Reprinted by permission of The MIT Media Lab.
interactions of wind, reflection, and shadow effects in an urban landscape. As with Illuminating Light, real-time tracking of the position and orientation of these phicons allows the system to update the display continuously, so that users can move the buildings around or rotate them until they find a satisfactory arrangement. The second set of phicons act as controls for the simulation. For example, a “wand” can be used to change the material of the buildings, so that the computed reflectance patterns will simulate buildings clad in brick or glass, another controls the direction of the simulated wind, while a “clock” has hands that can be moved to specify the time of day and hence the position of the sun for the shadow simulation. In this way, the simulator’s controls are introduced into the same space that is the focus of the system’s primary input and output.

Interacting with Tangible Computing

Tangible computing takes a wide range of forms. It might be used to address problems in highly focused and task-specific work, or in more passive awareness of activities in the real world or the electronic. It might attempt to take familiar objects and invest them with computation, or it might present us with entirely new artifacts that disclose something of the hidden world inside the software system. The bulk of this chapter has explored a range of tangible computing systems, but the survey has been far from comprehensive; indeed, I have said nothing about whole areas, such as wearable computing and context-based computing, that are clearly strongly related. My goal, however, was not to provide a catalogue of tangible computing technologies, but rather to introduce a sample of the systems that have been developed, and to begin to look for some common features of their design.

The first of these general issues that we see across a range of cases is that, in tangible computing, there is no single point of control or interaction. Traditional interactive systems have a single center of interaction, or at least a small number. Only one window has the “focus” at any given moment; the cursor is always in exactly one place, and that place defines where my actions will be carried out. Cursors and window focus insure that the system always maintains a distinguished component within the interface, which is the current locus of interaction. To do something else, one must move the focus elsewhere. When computation moves out into the environment, as in the tangible computing approach, this is lost. Not only is there not a single point of interaction, there is not even a single device that is the object of interaction. The same action might be distributed across multiple devices, or, more accurately, achieved through the coordinated use of those artifacts. Imagine sitting at your desk to write a note. The writing comes about through the coordinated use of pen, paper, and ink, not to mention the desk itself and the chair you sit in; you might write on the page with your dominant hand while your nondominant hand is used to orient the page appropriately. These are all brought together to achieve a task; you act at multiple points at once. In the same way, ubiquitous computing distributes computation through the environment, and, at one and the same time, distributes activity across many different computational devices, which have to be coordinated in order to achieve a unified effect.

A related issue is how tangible interaction transforms the sequential nature of interaction at the interface. The single point of control that traditional interfaces adopt leads naturally to a sequential organization for interaction—one thing at a time, with each step leading inevitably to the next. This ordering is used both to manage the interface and to simplify system development. For instance, “modal” dialog boxes—one that will stubbornly refuse to let you do anything else until you click “okay,” “cancel,” or whatever they need—both structure your interaction with the computer, and save the programmer from the need to handle the complexity of worrying about other actions that might transform the system’s state while the dialog box is displayed. When we move from traditional models to tangible computing, sequential ordering does not hold. It is not simply that interaction with the physical world is “parallel” (a poor mapping of a computational metaphor onto real life), but that there is no way to tell quite what I might do next, because there are many different ways in which I might map my task onto the features of the environment.

These two issues are particularly challenging from a technical perspective, because they address the programming models we use to develop systems, embedded in software toolkits and applications. The
third feature of tangible interaction may, however, provide some relief. This is the fact that, in tangible design, we use the physical properties of the interface to suggest its use. This is nothing new; arguably, it is what product design or other forms of physical design are all about. Kettles are designed so that we can tell how to safely pick them up; remote controls are designed to sit comfortably in the hand when oriented for correct use (at least when we’re lucky). What is more, this sort of design that recognizes the interaction between the physical configuration of the environment and the activities that take place within it can also be a way to manage the sequential issues raised earlier. For instance, Gaver (1991), in his discussion of “sequential affordances” (which will be presented in more detail in chapter 4), gives the example of a door handle, which, in its normal position, lends itself naturally to turning and then, in its turned position, lends itself naturally to pulling; the whole arrangement helps “guide” one through the sequential process of opening the door through careful management of the physical configuration of the artifact. Taking this approach, designers can create artifacts that lead users through the process of using them, with each stage leading naturally to the next through the ways in which the physical configuration at each moment suggests the appropriate action to take. The relationship between physical form and possible action can give designers some purchase on the problems of unbounded parallel action.

Interacting with tangible computing opens up a new set of challenges and a new set of design problems. Our understanding of the nature of these problems is, so far, quite limited, certainly in comparison to the more traditional interactional style that characterizes most interactive systems today. The theories that govern traditional interaction have only limited applicability to this new domain. At the same time, tangible computing has been explored, largely, as a practical exercise. Most prototypes have been developed opportunistically, driven as much by the availability of sensor technology and the emergence of new control devices as by a reasoned understanding of the role of physicality in interaction. We have various clues and pointers, but there is no theory of tangible interaction. Why does tangible interaction work? Which features are important, which are merely convenient and which are simply wrong? How does tangible computing mediate between the environment and the activity that unfolds in it?

This book is about developing answers to these questions. The interpretation that it will offer is one that is concerned not just with what kind of technology we use, or with what sorts of interactions we can engage in with that technology, but about what makes those interactions meaningful to us. From this perspective, the essence of tangible computing lies in the way in which it allows computation to be manifest for us in the everyday world; a world that is available for our interpretation, and one which is meaningful for us in the ways in which we can understand and act in it. That might seem to be quite far removed from looking at application prototypes, reactive rooms, and digital desks. The path from practice to theory will be easier to see after looking at the second aspect of embodied interaction—social computing.