An Overview of
Human Computer Interaction Patterns
in Pervasive Systems

Adriana Wilde

DIUF – Department of Informatics
University of Fribourg, Switzerland
adrianagabriela.wilde@unifr.ch

Abstract
Despite a growing interest in design patterns by the Human-Computer Interaction (HCI) community, an interaction pattern collection that is universally accepted is yet to be found. Such a collection would aid the design of pervasive computing systems, in which interactions occur in both the material and logical worlds. This paper first seeks to define interaction patterns by adopting a model that captures such duality and then considers the challenges and opportunities that non-traditional interaction brings. Some patterns are then reviewed, with examples relating to motion-aware systems.

Keywords: Interaction, HCI, interaction design, patterns, pervasive systems, ubicomp, interaction patterns.

1. Introduction
The HCI field emerged 25 years ago with the aim of improving the interaction between people and computers. However, with the computer “becoming liberated from the desktop” [32], such interaction presents other challenges and opportunities. Computer systems have become ubiquitous in every aspect of our lives while becoming less visible as separate and distinct artifacts.

This paper reviews the current literature in pervasive systems in order to identify some patterns of interaction, considering non-traditional modalities of interaction, like interaction with mobile devices, intelligent environments, and other users that are equipped with mobile devices. In order to limit the scope of this overview, we are particularly interested in those that are relevant to motion-aware systems.

This paper is organised as follows: In section 2 we offer a definition of interaction, followed by a review of how traditional human-computer interaction (HCI) came into being, and then a discussion of the challenges that ubiquitous computing brings to HCI. Section 3 is about the pattern approach, and the evolution of IT community’s interest in this, since it was first formulated in architecture, up to Jonas Löwgren’s “inspirational” patterns. In section 4, we turn our interest towards interaction patterns, most especially those emerging while designing pervasive systems. Here we offer examples in motion-aware systems in which some of these patterns can be used. Finally, in section 5 we present some conclusions to be drawn from this review.

2. Human-Computer Interaction

Before talking about Human Computer Interaction, we need to define interaction in a way that it is not only accurate, but flexible enough to be used in the context of HCI in pervasive systems.

2.1. Interaction

The term interaction, as with many concepts that are used in everyday language, has many working definitions. In particular, Dix et al. (in [8], p.4) define it as “any communication between a user and a computer”, where computer means any technology, process or system, which in turn could have non-computerized parts including other users. Furthermore users could be people or other systems. From this definition we can establish that any interaction is a form of communication between given entities.

Nonetheless, it is necessary to consider a more specific definition that includes the necessary conditions under which an interaction can take place. From the definition above it is apparent that there need to be at least two communicating entities and, for such communication to be effective there must be a medium for the communication as well as a shared protocol of exchange. Here the word “exchange” suggests a two-way communication, therefore only interactive communication is considered under this definition (i.e. a one-way communication does not constitute an interaction, as the broadcast of a message does not require an explicit response from any receivers). Therefore, all interactions are a form of communication, but not all communications are interactive.

Arguably, there are forms of interactions for which there is no explicit exchange of information, for exam-
The HCI field is, and has always been, multidisciplinary, particularly because of the intrinsically disparate nature of the two interacting elements. Sharp et al. name as many as fifteen disciplines which are concerned with “the theory, research and practice of designing user experiences for all manner of technologies, systems and focus” [28] (p.10). Others, like Löwgren [17] (p.189), believe that this is not sufficient from the epistemological point of view, as HCI is distinct from ID in the same way that engineering is distinct from design. According to Löwgren, ID is far beyond HCI, and authors should be careful in using that term to name the movement in HCI towards design.

Design is a mixture of applied art and engineering, and is innovative and multifaceted, that is, it has many aspects, often conflicting, such as aesthetics and usability2. For this reason, it makes sense to try to support the design process with aids, such as by the application of patterns (explored in section 3).

The HCI field is, and has always been, multidisciplinary, particularly because of the intrinsically disparate nature of the two interacting elements. Sharp et al. name as many as fifteen disciplines which are concerned with

---

1This approach of studying systems is not new. Ontology is a concept borrowed from philosophy used increasingly in informatics. It “defines the basic terms and relations comprising the vocabulary of a topic area as well as the rules for combining terms and relations to define extensions to the vocabulary” [22] (p.5).

2see discussions in [20], pp.151–158 and in [16], pp. 156–158)
The underlying idea, though, remains to be “designing interactive products to support the way people communicate and interact in their everyday and working lives” [28] (p.8). This is a great challenge, and it is a challenge that is here to stay. As Beaudouin-Lafon claims, “interaction is the future of computing” [1] (p.263). As computing becomes ubiquitous, increasingly more users are interacting through these systems, and this shift is considered in the next subsection.

2.3. Interaction in Ubicomp

In the 1980s, the prevailing paradigm in HCI was for the design of user-centered applications for the desktop computer. Questions about what and how to design were framed in terms of specifying requirements for a single user interacting with a screen-based interface [28] (p.218).

The premise of ubiquitous computing (ubicomp for short), or pervasive computing, is that computing systems can be everywhere the user needs them to be, “weaving themselves into the fabric of everyday life until they are indistinguishable from it”. (paraphrasing Weiser, [32], p.3). In other words, ubicomp has the potential to simplify people’s lives through digital environments that sense, adapt, and respond to people’s needs. A device can be a portal into an application-data space, not just a repository of software a user must manage. An application can be a means by which a user performs a task, not just software written to exploit a device’s capabilities. And a computing environment can be an information-enhanced physical space, not just a virtual environment that exists to store and run software [25]. This phenomenon is encapsulated under the term embodiment [11], and it is the embodiment of digital artifacts into physical ones that is key to unlocking the power of affordances. In terms of interactive computation, it means that the environment becomes symbiotic with the computer system [2]. This suggests a different interaction model, one that allows for both traditional interaction, which is purposeful and explicit, as well as what Dix calls incidental interaction [7]. From a spectrum of interactions with varying purpose, traditional interactions would be at one end, and incidental interactions at the other. Here interactions emerge almost as a collateral effect. Dix’s illustrates this with shopping websites that keep track of customers’ purchases, which can “incidentally” infer users’ tastes in order to suggest additional purchases.

Incidental interaction goes against most of what is “proved and tested” in traditional HCI, where the interaction model is some form of intentional cycle such as the Norman execution-evaluation loop [20]. In traditional HCI, the intentional cycle is seen as starting with the user (who initiates the interactions), while in more contextual accounts of interaction the focus shifts to a cycle of activity starting with the state of the world and system ‘responses’ to the user actions. When the interaction is incidental, the user might not even have any awareness of the interaction taking place, and many of the traditional HCI design principles (most notably: feedback, visibility and consistency [28], p.29-33) need to be scrapped or at least redefined. A truly pervasive system fades into the background: it is only visible when something requires the user’s attention and therefore feedback is kept to the minimum. As a result, the interfaces have become more natural in terms of the affordances they provide to the user. Furthermore, computing devices tend to support the user by executing the task in hand in such way that there often occurs a transfer of control (even if unconsciously, such as in Norman’s example of the driver reliant on the correct operation of the cruise control feature in his car [21]).

As ubicomp is fast becoming a reality for wider audiences, in no small part due to the recent advances in miniaturisation of computing devices and affordability of these by a wider community, HCI research now falls in either of the two following general approaches: either to focus on “creating powerful yet not always totally reliable interfaces, such as speech or gesture input”, or on “creating less complex, more reliable input techniques” [14].

Chris Harrison, talking to Kroeker [14], predicts that eventually computers will be able to recognize nuanced human communication and interpret a complex range of gestures, eye movement, touch, and other cues. He states “if we ever hope for human-computer interaction to achieve the fluidity and expressiveness of human communication, we need to be equally diverse in how we approach interface design” (p.15). One valid approach to face such diversity is the same that designers and architects have been using long before the emergence of HCI: the pattern approach.

3. Patterns

As just hinted, the field of architecture was the origin of the pattern movement. In the seventies, Christopher Alexander wrote a series of books, A Pattern Language being one of the most cited in the subject. There he characterizes a pattern as follows

Each pattern describes a problem which occurs over and over again in our environment and then describes the core of the solution to that problem, in such a way that you can use this solution a million times over without ever doing it the same way twice [4].

A set of patterns which can be combined to create a variety of solutions is referred to by Alexander as a pattern.
Gang of Four (GoF), as Gamma et al. clearly specified, for example in the influential book by Gamma et al. [12], and others, such as [5, 6, 23]. The Gang of Four (GoF), as Gamma et al. are colloquially known, were the first to use Alexander’s pattern language as a model to create a form of documentation in software design. In the fifteen years since the first publication of their influential book, some patterns defined in it have been used and reused by developers as templates of the solution for their design problems, thanks to the immense detail of their specification[4]. This level of detail clearly indicates a purpose different from Alexander’s, and as opposed to facilitate diversity, it aimed to articulate and disseminate knowledge among programmers. Furthermore, Gardner et al. [13] (p.35), who created patterns for cognitive modelling of business systems and processes, state that “each pattern is reflected by a problem-solving template”, subscribing to the idea of patterns needing to be well specified to be useful. Derntl agrees with this notion and highlights that a problem is dependent on its context (what Alexander referred to as “environment”), and that the solution of such a problem can be a configuration of the discipline’s design options in the form of a template that can be applied in solving similar problems [6] (p.41).

Alexander’s works have also influenced researchers in the field of HCI (for example Norman admits so in [20], p.238), and nowadays the community is increasingly interested in patterns. Some authors return to Alexander’s original intentions, like Jan Borchers [3], who formalised a pattern language for interactive music exhibitions. Also Douglas Van Duyne et al. [10], who offer a pattern collection for web development, and Martijn Welie [30] who maintains another one for interaction design. What all of these have in common is the idea that a pattern represents “a proven and successful design solution” [18] (p.169).

Other pattern literature suggests that it is acceptable for the solutions offered in patterns to be less fully specified and, instead, a verbose outline of the user expectations is offered, such as in [24], and [29]. That is why distinctions between patterns, universal patterns, pattern catalogue and pattern language are used to indicate the degree of detail in the solution specified, and whether the collection of patterns is interrelated in a meaningful way. From the extensive literature Jenifer Tidwell’s “behavioural patterns” [29] are notable since, instead of presenting such patterns in the way of a template (such as the rest of the patterns in her book, no doubt inspired by the GoF), she presents them as “small essays”, describing human behaviours rather than interface elements (pp.10-17). Tidwell describes these behaviours in terms of needs, which are to be catered for if a good interface design is intended[5]. Jonas Löwgren’s inspirational patterns, or i-patterns, are similar to Tidwell’s in that they are abstract and only described in sentences and verbose examples. For Löwgren, it is a collection of “successful” patterns, where his notion of success differs from the traditional HCI view where success is measured in terms of user acceptance and performance. His nine patterns aim towards the innovative and inspirational: into “new parts of the design space of embodied interaction” [18]. They are as follows:

1. Virtual information is tied to positions in the material world;
2. Virtual bookmarks are tokens of positions in the material world;
3. Material objects are tokens of virtual information;
4. Virtual information “has” material properties;
5. Virtual information forms objects in the material world;
6. Material objects’ qualities influence interaction qualities;
7. Heterogeneous virtual information fuses into a few sensory parameters;
8. Interactive and broadcast media combine to form a positive spiral of participation; and
9. Virtual information and functions are limited at certain times.

Such patterns capture the relationship between the logical world of information and the physical world of objects described in section 2.1, and we will show in the next section how some of them are applied.

---


[5] For example: humans interacting with devices need to be allowed to explore safely, knowing that no dire consequences will arise as a consequence of trying out unfamiliar features. They also have a need for instant gratification, i.e. a positive result should arise within a few seconds of the interaction. Users need to be offered “good enough” as well as “best” if the latter involves extra cost. Humans need to be allowed to change their minds during an interaction and perhaps to return later without having to restart the whole process.
4. Interaction Patterns in Ubicomp

According to Michel Beaudouin-Lafon [2] (p.230) there are three interaction paradigms: first-person interfaces, in which users interact in a dedicated manner with a computer, as within traditional HCI research, and can follow the rules of direct manipulation; then second-person interfaces, in which users delegate some tasks to the system, who is seen as a “partner” assisting in reaching the user’s goal as in the cruise control example of section 2.3; and finally, third-person interfaces, where the system mediates the interaction among various users, such as in the case of computer-supported cooperative work and social networks.

Non-traditional modalities of interactions, like those to be considered in the design of pervasive systems, fall into all three paradigms, and we will comment on some patterns of each kind: interaction with mobile systems (first person interface), intelligent environments (second person interface), and other users that are equipped with such devices (third person interface).

4.1. Interactions with mobile systems

Let us consider the following problem: mobile devices used in a variety of locations and situations might interrupt or distract the user from performing a primary task or disturb a nearby group of people. This corresponds with Löwgren’s i-pattern “virtual information and functions are limited at certain times”.

Landay and Borriello [15] apply the design-pattern concept to this problem proposing that “input and output modalities should adapt to the user’s current context” (p.94). For example, relying on speech or audio output is not a good idea when the user is participating in a meeting, attending a lecture, or in a movie theatre. A context-sensitive mobile phone should know when its owner is in a meeting and switch automatically to a vibration alert. In addition, speech might be desired over direct manipulation input to a handheld device when the user is driving a car. Likewise, when the driver places or receives a call, the car stereo volume should lower automatically.

Erik Nilsson [19] developed a pattern collection for interacting with mobile applications, aiming to solve over 60 specific problems (which he groups in problem areas, and these further into three main problem areas), all aiming to improve the quality of the user interaction with handheld devices. Notable examples of these problems are those arising from interaction mechanisms (such as use/lack of a stylus, multimodal input, etc.), and utilising screen space (such as horizontal scrolling, switching from portrait to landscape, etc.).

Van Duyne et al., in their book “The design of sites” [10], propose three patterns for “the mobile web” in a manner not dissimilar to Nilsson: mobile screen sizing (M1), mobile input control (M2) and location-based services (M3). However, they present the relationships to other patterns in their vast collection.

Jörg Roth addressed some of these problems too, proposing separately Browse-it as a pattern that allows a handheld user to browse the web without struggling with device limitations such as screen resolution [24]. The proxy pre-processes web pages, downscales graphics and pre-computes the appropriate layout. As a result, the amount of data transferred to the handheld device is drastically reduced, and the devices are relieved from heavy rendering tasks.

4.2. Intelligent environments

Intelligent environments and their occupants interact in a natural yet adaptive way [9] (p.2). Monekosso et al. explain that in such environments the interaction occurs in a way that is natural for a human (for example, speech, motion, gestures), and the environment learns to recognise and change itself depending on the identity and activity taken by its occupants. Intelligent environments are made possible by permeating spaces with intelligent technology that enhances the quality of life, ranging from private to public spaces. The user information is gathered via wearable devices (such as RFID tags in badges) and/or via pervasive sensors.

A further example provided by Monekosso (p.6) is the Intelligent Classroom. In it, gestures and movement of the lecturer are interpreted by the environment and trigger actions, such as dimming the lights, opening blinds or switching on the overhead projector. In this way, the lecturer can concentrate on his lecture instead of on the technology, minimising unwanted breaks and therefore maintaining the attention of his audience. This corresponds to the i-pattern “interactive and broadcast media combine to form a positive spiral of participation”.

Another example is the intelligent environment for supporting independent living [9] (p.35–50), in which the environment learns to recognise the activities an elderly occupant is performing in order to detect anomalous behaviour and alert their caregivers if needed. Pervasive sensors are used to gather information about the occupant (in particular, motion detection) and the activity performed is recognised using classifying algorithms, so the system would record whether the occupant is out, asleep, watching television, cooking, eating or bathing, to cite a few. Taking this research a step further would be not just alerting a caregiver in case of the detection of anomalous behaviour, but supporting the occupant’s lifestyle.

---

6 also available online at www.flaminco.net

---

7 Example: Location-based services (M3) are a class of applications that allows the creation of Personalized content (D4) based on a visitors current physical location. M3 also allows customization of the customer experience for Stimulating arts and entertainment (A9) sites, emphasize nearby places for Organized search results (J3), and enable new ways of doing Personal e-commerce (A1).
by switching off lights and other appliances when the occupant is away, or asleep, for example, if forgetfulness were a problem. The pattern solving such a problem could be named “Assisted living”.

Another application of intelligent environments is in workplaces where staff are required or encouraged to work in a variety of locations, for example in colleges or multi-site corporations. In these cases, locating a given individual at a given time becomes a problem. Intelligent environments could allow for “Continuous tracking”, which not only solves the problem of locating an individual at a given time, but could trigger other actions in their environment as well, just as “hovering the pointer over a desktop in graphical user interfaces” does [31]. Therefore, this may well be a pattern of interaction for pervasive systems, corresponding to the i-pattern “virtual information is tied to positions in the material world”, or vice versa, since it is the location of the user that is the trigger of the actions in the environment.

4.3. Other users

When people get together to collaborate in some way, they should not have to spend a lot of time configuring their devices. For example, at a meeting, the appropriate files, including contact information, should appear automatically on each person’s laptop or PDA. Landay and Borriello [15] propose as a solution: “when users are near one another, it should be easy for their devices to connect and create an association that lets them share information over the life of a session” (p.95). A context-sensitive I/O device provides the appropriate output for making a Physical-Virtual Association. Some associations can occur automatically when two known devices come into physical proximity, but creating other associations might require direct user action (to grant permission for the interaction to occur). One example is Bump™, an application developed for iPhones and Android phones, which permits the swapping contact information and photos by bumping two phones together. This corresponds with Löwgren’s i-patterns “material objects are tokens of virtual information” and “material objects qualities influence interaction qualities”, as the phones are held in the hand as tokens of personal information, and, instead of a more personal handshake, people wishing to perform the exchange bump knuckles (being “restricted” by the physical dimensions of their phones).

5. Conclusions

The application of design patterns to pervasive computing systems opens up interesting research questions. The diversity and abundance of the research in this new area is an indication of how difficult it is to create a pattern collection that is universally accepted. Questions remain about how to validate such patterns given the time and expense required to test each one; and how to evaluate the process of using patterns, since conducting controlled studies is prohibitive because of the creativity and skill involved in the act of design. As Landay and Borriello point out, the process of developing design patterns is still fairly ad hoc. Despite this apparent disadvantage (compared to the field of software engineering, which count on the GoF patterns to aid developers), patterns are also a very useful aid to inspire designers of pervasive systems, letting them focus on how to improve the quality of the diverse interactions that occur is such systems.

6. References


8Bump™ is a registered brand by Bump Tecnologies http://bu.mp


[12] E. Gamma, R. Helm, R. Johnson, and J. Vlissides. Design Patterns: Elements of Reusable Object-Oriented Software. 1994.


