*Vortrag gehalten beim Workshop "Blended Interaction : Envisioning Future Collaborative Interactive Spaces" im Rahmen der CHI '13 2013 ACM SIGCHI Conference on Human Factors in Computing Systems ; April 27 - 2 May, 2013, Paris*

# Self-Organizing User Interfaces: Envisioning the Future of Ubicomp UIs

**Hans-Christian Jetter**
Human-Computer Interaction Group
University of Konstanz
Universitaetsstrasse 10
78457 Konstanz
Germany

hans-christian.jetter@uni-konstanz.de

**Harald Reiterer**
Human-Computer Interaction Group
University of Konstanz
Universitaetsstrasse 10
78457 Konstanz
Germany

harald.reiterer@uni-konstanz.de

*Konstanzer Online-Publikations-System (KOPS)*
*URL: http://nbn-resolving.de/urn:nbn:de:bsz:352-250974*

## Abstract
This workshop paper envisions a future of *self-organizing* UIs for ubiquitous computing. It proposes using self-organization to join many co-located interactive devices into an *ad-hoc community of devices* that serves users as a single usable and seamless UI. All devices of the community are aware of each other's presence and contribute their individual input/output capabilities for the common goal of providing users with a seamless, usable, and accessible interface that spans across device boundaries. This is achieved by letting the UI's behavior emerge from simple rules that react to changes in presence, location, distance, orientation, and movement of neighboring devices and users. By using simple rules of *proxemic interactions* between devices, deterministic preciseness of classic *top-down* design and modeling is traded in against less controllable, but more adaptable, robust, and scalable *bottom-up* designs that automatically react to the dynamics of ad-hoc real-world usage.

## Author Keywords
vision; ubiquitous computing; post-WIMP; self-organization; distributed user interfaces.

## ACM Classification Keywords
H.5.2. Information interfaces and presentation (e.g., HCI): User Interfaces.

## Introduction

The last decade created a multitude of novel computing devices with many different user interfaces (UIs) and form factors, e.g., smart phones, mobile tablets, tabletop computers, large multi-touch and pen-enabled screens, or wearable and tangible displays. Today, these devices have become a part of our everyday work practices and are woven into the fabric of our physical and social environment. In the light of this current shift to an "era of ubiquity" with several or even "thousands of computers per user" [5], the key challenge of HCI is to design and implement UIs for a natural and efficient interaction that is not provided by only a single device or gadget, but by entire *ad-hoc communities* of many co-located devices. Ideally all devices of such a community contribute their individual input/output capabilities for the common goal of providing users with a seamless, usable, and accessible UI that spans across the devices' boundaries. Some of these devices will be carried by us, while others will be embedded in our physical environment. Therefore, in future, "[…] the notion of an interface is no longer easily defined, stable or fixed" [5]. We believe that HCI must actively address this instability of future UIs that are not bound to single devices anymore but that must adapt to and make use of the always changing set of available devices and interaction resources present in the environment.

To date, HCI has only made little progress in achieving a seamless interplay of and interaction with ad-hoc communities of devices. Sharing interaction resources (e.g., display space, input modalities, tools, content) across device boundaries remains tedious. Often the upfront system design is too inflexible and optimized only for a predefined set of devices or tasks. If possible at all, the manual integration and configuration of devices is challenging for the users. Greenberg et al. write: "While most devices are networked, actually interconnecting these devices is painful […]. Even when devices are connected, performing tasks among them is usually tedious […]. In practice, […] the vast majority of devices are blind to the presence of other devices" [4]. Similarly, Oulasvirta considers today's digital ecologies as "a multilayered agglomeration of connections and data, distributed physically and digitally, and operating under no recognizable guiding principles" [11]. For him "achieving seamlessness" and "fluent multidevice work" are the key challenges of ubiquitous computing.

## The Vision

We envision ad-hoc communities of co-located devices that are not blind to the presence of other devices. All devices contribute their different input and output capabilities for the common goal to provide users with a seamless, usable, and accessible UI. Users should experience this community of cooperating devices as a single UI that always makes optimal use of the currently available interaction resources in the environment, e.g., available displays, pen/touch/speech input, tactile output. Ideally such a community 1.) automatically adapts to the many possible changes in the kind and number of available devices, their spatial configuration, and the number of users, 2.) is robust against breakdowns or loss of interaction resources, and 3.) readily adds or removes further interaction resources and devices. Furthermore, users should not be concerned with configuration and setup anymore. Instead this happens implicitly as a by-product of normal use (e.g., bringing multiple devices to the same room, placing them side-by-side and shifting them around, handing them over to others).

**Top left:** Size limitations of phone displays will be overcome by using tablets. Users can move tablets close to phones to make ad-hoc use of the tablet's screen. Both devices instantly cooperate and provide a navigation view on the phone synchronized to the content view on the tablet.

**Top right:** Thin bezel-less displays or tablets will be used to carry configurable display space around and provide it whenever and wherever needed. These display tiles will be multi-touch and pen-enabled and can be used to create larger interactive surfaces, e.g., by laying them out on a desk.

**Bottom Left:** Tiles can be moved around on a table to create different (split-)views of a document. When a tile is moved away from a cluster, it switches its role from a content display to a tool palette or a notepad. *(con't on next page)*

**Figure 1.** Different possible designs of future ad-hoc communities of devices (for details see text box on the left).

*(con't from prev. page)*
**Bottom Right:** By moving a tablet or tile above the table, it automatically switches roles and becomes a see-through magic lens that alters appearance of data seen through it, e.g., showing satellite imagery and photos in addition to maps.

**Self-Organization?**
Scientifically, self-organization is defined as "a process in which pattern at the global level of a system emerges solely from numerous interactions among the lower-level components of the system. Moreover, the rules specifying interactions among the system's components are executed using only local information, without reference to the global pattern. In short, the pattern is an emergent property of the system, rather than a property imposed on the system by an external ordering influence" (see Camazine et al. [1]).

## Previous Approaches

In the past, HCI research has tried different approaches for ad-hoc distribution of an UI across devices and an optimal use of the available interaction resources:

*Model-based UIs* use abstract high-level models of tasks, dialogs, or presentation to define UIs without specifying details of interaction and visualization. These models are used to (semi-)automatically adapt or generate detailed UI code for different contexts, devices, and modalities [8]. Such models can also be used to dynamically generate and adapt UIs that are distributed across many devices [9]. However, in practice, model-based UI development has not been successful [8]. In our opinion, one weakness of the approach is the overhead of detailed modeling and the assumption that UI design can be created from abstract models without iterative design cycles. This is in direct opposition to the dominant paradigm in HCI design that recommends a thoughtful, iterative user- and value-centered design that makes use of ethnographic methods, user studies, sketching, and the creativity and artistry of product and industrial design [5].

*Context Awareness* uses models to describe and identify different usage contexts and to adapt the appearance and functionality of a UI accordingly. It is based on an upfront prediction of all possible contexts and the necessary context-specific changes. The approach is criticized as being inherently flawed since it ignores that context is a "dynamic construct" [3]: "Designers may find it difficult or even impossible to enumerate the set of contextual states that may exist, know what information could accurately determine a contextual state within that set, and state what appropriate action should be taken from a particular state". A major breakthrough in context-awareness is considered questionable until problems of strong AI and "vagueness" of contextual states are not solved [2].

More recent approaches avoid modeling high-level concepts such as tasks and contexts and instead define simple rules of adaption based on physical states and dimensions. For example, *proxemic interactions* [4] use distance, orientation, movement, identity, and location of devices or persons, to create rule sets for small device communities, e.g., for a media player in a living room that reacts to the users' presence, their positions, their attention, or pointing gestures. Similar examples of multi-device interactions are the bumping of devices into another for display tiling [6], using multiple smart phones side-by-side to watch and share photos [7], or pushing "Siftables" into groups or piles to play games [10]. Such approaches work in simple scenarios in which the upfront definition of rules is comparably easy. However, "while it is easy to create believable scenarios where a rule set makes sense, there will always be many cases where applying the rule in a particular instance will be the wrong thing to do" [4]. It remains an open question how to design or combine robust rule sets for complex or unintended situations.

## Self-Organizing User Interfaces

In our vision of future HCI, *self-organization* is used to join many co-located interactive devices into an ad-hoc community of devices that serves users as a single usable and seamless UI. Self-organization can be observed in many biological or natural systems and often leads to an almost magical and nearly optimal use of resources that is not result of an external ordering influence or a *top-down* design. For example, self-organization can be observed in flocks of geese: By

**Centralized Control vs. Emergent Behavior**

In traditional *top-down* design, the desired behavior is defined in models or rules that are used to control the global state of the device community. A centralized control is in charge of executing and applying them.

In self-organizing *bottom-up* design, the global behavior of the community emerges from numerous rule-based local interactions between the devices and is based primarily on local information instead of global information.

following simple rules based on local information like the line of sight and air drag, each goose reacts individually to changes in the flock and the sum of these reactions results in forming a V-formation. This *bottom-up* emerging behavior greatly improves aerodynamics and the range of the flock.

Self-organization has already been used successfully in design and engineering, e.g., to solve the NP-hard Travelling Salesman Problem, to manage traffic lights and sensor networks, or to create self-assembling modular robotics. The reported virtues of self-organizing systems, as opposed to traditionally designed systems, are an increased *robustness*, *adaptability*, and *scalability* [8]. In our vision of self-organizing UIs, the way the UI reacts to each possible configuration and change in the environment is not determined by a centralized control as an "external ordering influence". Instead it emerges from the execution of many local rules defining proxemic interactions between devices and users. The more these rule sets focus only on local information and avoid reference to a global pattern, the more the device community will react sensible and stable to unpredictable changes in the environment. Thereby an illusion of "intelligent" behavior is achieved without the need for upfront abstract modeling of each possible state or solving problems of (semi-)automatic generation of UI design or strong AI. Our hope is that by defining simple rules of adaption based on proxemic interactions between devices, the deterministic preciseness of classic *top-down* design and modeling is traded in against less controllable, but more adaptable, robust, and scalable *bottom-up* designs that introduce a certain degree of vagueness but automatically adapt to the dynamics of ad-hoc real-world usage.

## Possible Implementation

Our vision is based on an environment that provides wireless communication between all devices and tracks the location and orientation of all devices and users. Present-day networks (e.g. TCP/IP or UDP via WiFi) and motion capturing systems (e.g., OptiTrack) enable us to simulate such a setting in the Media Room lab of our research group. A digital representation of this space can be managed and queried using tools such as the Proximity Toolkit [4]. Yet unavailable devices such as bezel-less display tiles can be simulated in our lab using tablets or by projecting them on desks covered with a large touch-sensitive foil and Anoto pattern. The true challenge, however, is the design and implementation of the base UIs for each device and in particular defining the rules for adapting these UIs in reaction to the proxemic interactions with neighboring devices or users. For example. to achieve a behavior similar to Figure 1, the base UI of each tile must be a distributed viewing application that shows content such as mails, documents, or maps on one or several adjacent interactive display tiles. As soon as one or multiple tiles are moved towards the smart phone and are laid out in a regular adjacent pattern, they must turn them into one virtual content display. Thereby zooming & panning or highlights or annotations on one tile must be applied synchronously to the others.

While this kind of distributed UI can be implemented using our ZOIL framework (http://zoil.codeplex.com), the formulation of good rule sets is more difficult. For example, there must be rules that define what happens when a virtual display is split vertically or horizontally (e.g., split view or copy of view?), when a single tile is moved out of a cluster of adjacent tiles (e.g., switch tile to toolbox when close to the cluster and to notepad

**Self-Organization in HCI?**

A valid point of criticism to make against self-organization in HCI is the sheer complexity of a UI. Although admirable for their structure and efficiency, emergent behaviors such as geese in V-formation or nearly-optimal throughput in sensor networks are simple compared to the complexity of a usable and attractive UI. Assuming that such an UI can emerge only from simple proxemic interactions between low-level components is comparable to the infamous thought experiment of millions of monkeys at typewriters reproducing Shakespeare by bashing random keys. Therefore we believe in taking a shortcut to avoid millions of iterations or years of evolution: By providing base UIs for each device as a starting point, self-organization is used rather to *adapt* the UI's content and behavior than to *create* it from scratch.

when close to user?) or when many tiles are added or removed. How the formulation of rules and their parameters will affect the emergent behavior of the UI on a global level might sometimes be impossible to predict for designers and will be subject to "educated guesses". To systematically approach this challenge, we propose to gradually increase the complexity and degree of self-organization: First, designs will start with homogeneous communities of devices (e.g., multiple identical tablets for display tiling) and weak self-organization (i.e., reference to global information and states is allowed). With growing experience, designs will increasingly focus on heterogeneous communities consisting of different interactive devices with strong self-organization (i.e., primarily or exclusively using local information and rules). For an efficient definition of UIs and rules that result in a "natural" and predictable user experience, it also seems promising to evaluate programming models that are unconventional in HCI, e.g., non-imperative declarative, logical, or functional programming languages.

## Conclusion

Self-organization could become a powerful tool for tackling the dynamics of future ubicomp environments. A successful use of self-organization could create more robust, adaptable, and scalable UIs that seamlessly span many devices. Such a seamless cross-device interaction and user experience in digital ecologies (e.g., seamless interactions across smart phones, tablets, TV, laptops, desktop PCs) would not only bring us closer to Weiser's vision of ubicomp, but would also be an important economic advantage for one of today's competing eco-systems (e.g., Android, Windows 8, iOS) and thus appears as a promising approach to explore.

## References


[1] Camazine, S. et al., *Self-Organization in Biological Systems*, Princeton (2001).

[2] Eliasson, J., Pargman, T. and Ramberg, R., Embodied Interaction or Context-Aware Computing?, *Proc. HCI '09*, Springer (2009), 606-15.

[3] Greenberg, S., Context as a dynamic construct, *Hum.-Comput. Interact.,* 16,2 (2001), 257-68.

[4] Greenberg, S., Marquardt, N., and Ballendat, T. Proxemic interactions: the new ubicomp?, *interactions 18*,1 (2011), 42-50.

[5] Harper, R., Rodden, T., Rogers, Y., and Sellen, A. *Being human: Human-computer interaction in the year 2020.* Microsoft Research, Cambridge, UK, 2008.

[6] Hinckley, K. Synchronous gestures for multiple persons and computers, *Proc. UIST '03*, ACM, 149-58.

[7] Lucero, A., Holopainen, J., and Jokela, T. Pass-them-around: collaborative use of mobile phones for photo sharing', *Proc. CHI '11,* ACM (2011), 1787-96.

[8] Meixner, G., Paterno, F., and Vanderdonckt, J. Past, Present, and Future of Model-Based User Interface Development, *i-com,* 10,3 (2011), 2-11.

[9] Melchior, J., Vanderdonckt, J., and Roy, P.V. A model-based approach for distributed user interfaces, *Proc. EICS '11*, ACM Press (2011), 11-20.

[10] Merrill, D., Kalanithi, J., and Maes, P., Siftables: towards sensor network user interfaces, *Proc. TEI '07,* ACM (2007), 75-78.

[11] Oulasvirta, A. When users "do" the Ubicomp, *interactions, 15*,6 (2008), 6-9.

[12] Prokopenko, M., Design vs. Self-organization, in *Advances in Applied Self-organizing Systems*, Springer (2008), 3-17.