Meta-User Interfaces for Ambient Spaces

Joëlle Coutaz

Université Joseph-Fourier
CLIPS-IMAG, BP 53,
F-38041 Grenoble Cedex 9 (France)
Phone: +33 4 76 51 48 54 – Fax: +33 4 76 44 66 75 – Joelle.Coutaz@imag.fr

ABSTRACT

In this article, we propose the concept of meta-User Interface (meta-UI) as the set of functions (along with their user interfaces) that are necessary and sufficient to control and evaluate the state of interactive ambient spaces. This set is *meta*-, since it serves as an umbrella beyond the domain-dependent services that support human activities in an ambient interactive space. They are *User Interface*-oriented since their role is to help the user to control and evaluate the state of this space. We present a dimension space to classify, compare, and contrast disparate research efforts in the area of meta-UI's. We then exploit the generative power of our design space to suggest directions for future research.

Categories and Subject Descriptors

H.5.2 [User Interfaces] Theory and method; I.3.6 [Methodology and techniques] Interaction techniques.

General Terms

Design, Human Factors.

Keywords

Ubiquitous computing, ambient interactive spaces, design space, taxonomy, meta-UI, GUI desktop.

1. INTRODUCTION

The capacity for users to control and evaluate system state is fundamental to Computer-Human Interaction [33]. This principle, promoted in the early 1980's by cognitive psychologists and human factors specialists, has actually been applied twenty years earlier by computer scientists who introduced the concept of Job Control Language (JCL). JCL, used for batch processing, has been progressively replaced with the Unix Shell followed by graphical desktops. With the emergence of ambient computing, users are not limited to the system and applications of a single computer. Instead, ambient computing embraces a model in which users, services, and resources discover other users, services and resources, and integrate them into an ambient interactive space.

An *ambient interactive space* is a dynamic assembly of physical entities coupled with computational and communicational entities to support human activities. An ambient interactive space can be as simple as a workstation or a PDA connected to the services of the Internet, or as complex as a computational ecosystem that evolves to adapt to the context of use. Augmented rooms such as FAME [29], iRoom [25], i-LAND [40] and Dynamo [24], are early examples of interactive spaces where users meet in a dedicated place to collaborate. With Jigsaw, users can create domestic services by assembling augmented objects [38]. Coupling two tranSticks make it possible to extend a local interactive space to that of a distant machine [1]. An ambient interactive space can also be viewed as a computational aura that follows users as they move from place to place [16].

TAMODIA'06, October 23–24, 2006, Hasselt, Belgium.

These examples show that, with ambient computing, we are shifting from the control of systems and applications confined to a single workstation to that of a dynamic interactive space where the boundaries between the physical and the digital worlds are progressively disappearing. As a result, the pre-packaged well-understood solutions provided by shells and desktops are not sufficient [5], and many interaction techniques are being developed for ambient computing, although on a case-per-case basis. This ad-hoc approach is adequate for local exploration, but may not provide sufficient insights to the problem.

In this article, we propose the concept of meta-UI (meta-User Interface) to denote a kind of interactive system that allows users to control, mould and understand their interactive ambient spaces. In the following section, we define the notion of meta-UI and propose a taxonomic space to understand its nature more precisely. Then, using this space, we classify, compare, and contrast disparate research efforts in the area of meta-UI's to suggest directions for future research.

2. DEFINITION AND TAXONOMY

A *meta-UI* is an interactive system whose set of functions is necessary and sufficient to control and evaluate the state of an interactive ambient space. This set is *meta*- because it serves as an umbrella *beyond* the domain-dependent services that support human activities in this space. It is *UI*-oriented because its role is to allow users to control and evaluate the state of the ambient interactive space. In the context of this article, a meta-UI *is not* an abstract model, nor a language description, whose transformation/interpretation would produce a concrete effective UI. It is an over-arching interactive system whose role is to ambient computing what desktops and shells are to conventional workstations.

As shown in Figure 1, a meta-UI is characterized by its *functional coverage* in terms of *services* and *object types*. In turn, the services and objects are invoked and referenced by the way of an *interaction technique* (or UI) that provides users with some *level of control*. An interaction technique is a language (possibly *extensible*) characterized by the *representation* (vocabulary) used to denote objects and functions as well as by the way users construct sentences (including how they select/*designate* objects and functions). Given the role of a meta-UI, the elements of the interaction technique of the meta-UI cohabit with the UI's of the domain-dependent services that it governs. The last dimension of our taxonomy, the *integration level*, expresses this relationship.

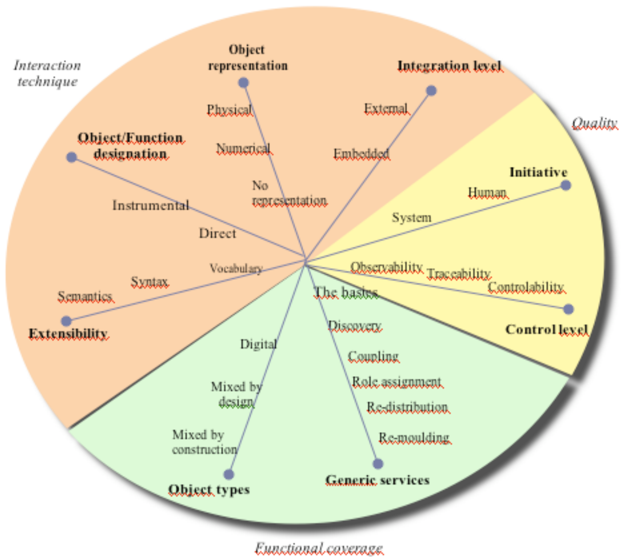


Figure 1. Dimension space for meta-UI's.

Functional coverage, interaction technique, and quality, are typical issues to be addressed when analyzing and designing an interactive system. We have refined these issues for the particular case of meta-UI's into a set of dimensions that are discussed next in more detail.

2.1 Object types

Objects involved in the services of a meta-UI may be *digital*, *mixed-by-design* and/or *mixed-by-construction*. Applications and files are typical examples of digital objects manipulated through the services of a meta-UI. Interactors such as windows, pointers, menus and forms, are other examples of digital objects. They are the conceptual units of windowing systems, which, according to our definition, are part of a conventional meta-UI.

A *mixed-by-design object* is an entity that results from the coupling, *by the designer*, of physical entities with digital services. A PDA and a mobile phone are mixed-by-design objects: the assembly of physical components with digital parts has been performed by designers beforehand.

A *mixed-by-construction object* is a mixed object that results from the coupling, *by the end-user*, of physical entities with digital service in order for that object to fulfill its *raison d'être*. For example, to function as a pointing device, the physical object that holds in the hand (called the mouse) must be coupled with the system mouse driver by the end-user. Similarly, the plastic rabbit shown in Figure 2 must be coupled by the end-user to an Internet service (such as time of the day or weather forecast) to serve as a communicating object.

The distinction between pure digital objects and mixed reality objects is now well understood. For mixed objects, the situation is still unclear. For example, Fitzmaurice's taxonomy applies to a particular type of phicons – the bricks [15]. Similarly, Holmquist addresses token-based access to information with the notions of containers (to move information between different devices or platforms), tokens (to access stored information), and tools (to manipulate digital information) [23]. Fishkin structures the problem space of Tangible UI's in terms of embodiment (to reflect spatial relationships between input and output), and in terms of metaphor (to reflect the analogy – or absence of analogy, between the mixed object and the real world) [14]. In particular, a noun metaphor means that an "<X> in the system is like an <X> in the real world", and a verb metaphor, "<X>-ing in the system is like <X>-ing in the real world."

Although these taxonomies can be used to refine the notion of object types, they are limited in scope or serve different purpose. Our notions of mixed-by-design and mixed-by-construction objects are more generic and make it explicit the capacity (or incapacity) for end-users to mould their own interactive space.

2.2 Generic services

Back in the 1960's, JCL provided end-users with generic services to control jobs execution and to perform files management. In the early 1980's, the Xerox Star introduced additional generic functions such as find, cut and paste, undo and redo. Starting/stopping the execution of a service, moving and renaming files, cutting and pasting data, as well as finding, are the *basics* of conventional meta-UI's. They are conceptually valid in ambient computing, but they need to be extended and refined.

In particular, the notion of finding can be extended with that of *objects discovery*. Objects discovery is key to building a sound mental model of the boundary and of the state of ambient spaces. For example, the Speakeasy browser allows users to explore lists of objects that satisfy some specified search criteria (such as location, object types, availability, etc.) [32].

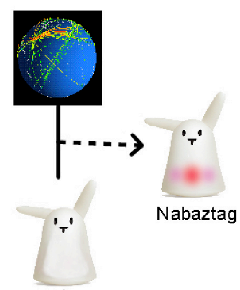


Figure 2. The Nabaztag is a mixed-by-construction object. It results from coupling a physical plastic rabbit with a set of Internet services. http://www.nabaztag.com

Because users are not simply consumers, but the designers and architects of their own interactive space, because the system must manage resource allocation dynamically, *coupling objects* becomes key. Coupling is the act of binding objects so that they can operate together to provide a new set of functions that these objects are unable to provide individually [11]. Two ConnecTables can be dynamically coupled by approaching them close to each other to enlarge the screen real estate [41]. With DataTiles, users obtain new services by configuring tagged transparent tiles on a flat panel display [37]. The analysis presented in [11] shows that coupling raises a large number of research issues.

Objects discovery allows users (and the system) to be aware of the objects that can be coupled. By coupling objects, users (and the system) build new constructs whose components play a set of roles (or functions). In conventional computing, roles are generally predefined. Typically, the screen of a laptop plays the role of an interaction resource, and this role is immutable by design. In ambient computing, where serendipity is paramount, *assigning roles to objects* becomes crucial. For example, in the Olympic

Café scenario illustrated in Figure 3, Bob and Jane uses spoons and lumps of sugar to denote the streets and buildings of the city they are planning together [10]. Bob couples a spoon with the table by laying it down on the table. The system can then discover the presence of the spoon and assigns it the role of interaction resource. (The spoon coupled with the system objects tracker and identifier is a mixed-by-construction object.) Then, by uttering the sentence "this spoon is Street Michel-Ange" while pointing at the spoon, Bob couples the interaction resource with a particular digital object known by the system as Street Michel-Ange. By doing so, Bob assigns the role of token[1] to the spoon.

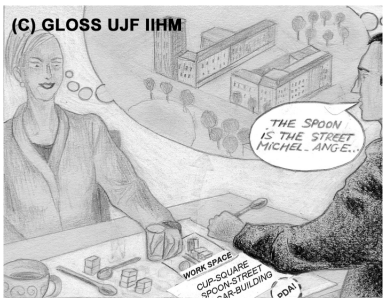


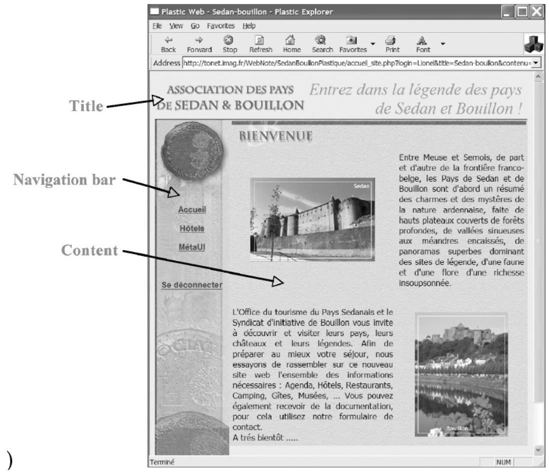
Figure 3. Bob and Jane use physical objects on the table to illustrate their ideas for the layout of the city they are planning together.

One particular role of interest in UI design, is that of input and output interaction resource. In conventional computing, these resources are connected to a single computer. In ambient computing, the platform is a dynamic cluster composed (by the way of coupling) of multiple interconnected computing devices whose interaction resources, all together, form an *habitat* for UI components. Instead of being *centralized*, user interfaces may now be *distributed* across the interaction resources of the cluster.

UI re-distribution, i.e. the application of *objects re-distribution* to UI components, denotes the re-allocation of the UI components of the interactive space to different interaction resources. For example, the Sedan-Bouillon Web site shown in Figure 4, whose UI is centralized on a single PC screen, is re-distributed in Figure 5 across the interaction resources of the PC and the PDA. Objects re-distribution and objects coupling may require object re-moulding.

Object re-moulding is to reshape objects without distorting their role. Applied to user interface components, *UI re-moulding* denotes the reconfiguration of the user interface that is perceivable to the user and that results from transformations applied to the source user interface. UI transformations include: *suppression* of the UI components that become irrelevant in the new context of use; *insertion* of new UI components to provide access to new services relevant in the new context of use, *reorganization* of UI components by revisiting their spatial layout and/or their temporal dependency. Reorganization may result from the suppression and/or insertion of UI components.

Re-moulding may result in using different modalities, or in exploiting multimodality differently. For example, because of the lack of computing power, the synergistic-complementarity [12] of the source multimodal UI (as in the example of Figure 3) may be transformed into an alternate-complementarity, or complementarity itself may disappear.



)

Figure 4. The Sedan-Bouillon Web site when centralized on a PC.

UI re-moulding is *intra-modal* when the source UI components that need to be changed are retargeted within the same modality. Note that if the source user interface is multimodal, then, the target UI is multimodal as well: intra-modal remoulding does not provoke any loss in the modalities set. UI re-moulding is *inter-modal* when the source UI components that need to be changed are retargeted into a different modality. Inter-modal retargeting may engender a modality loss or a modality gain. Thus, a source multimodal UI may be retargeted into a mono-modal UI and conversely, a mono-modal UI may be transformed into a multimodal UI. UI Re-moulding is *multi-modal* when it uses a combination of intra- and inter-modal transformations. For example, Teresa supports multi-modal re-moulding between graphics and vocal modalities [6]. As for inter-modal re-moulding, multi-modal re-moulding may result in a modality loss or in a modality gain.

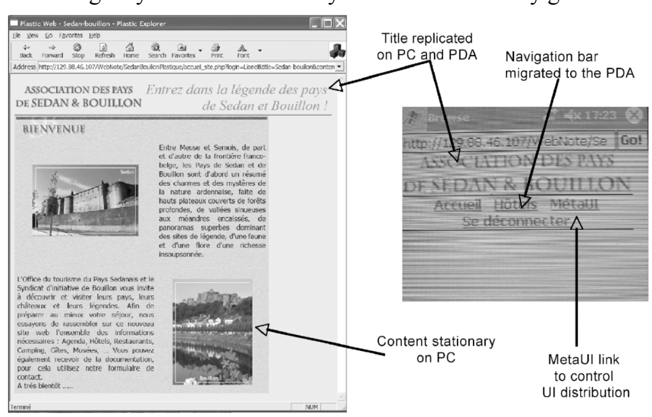


Figure 5. The UI of the Sedan-Bouillon Web site when distributed across the resources of the PC and the PDA.

All of the services provided by a meta-UI are executed under some level of human/system control.

2.3 Control

As for any interactive system, the services of a meta-UI may be executed on the system *initiative* and/or user's *initiative*. In the example of Figure 3, the system takes the initiative to discover

---

[1] Token, as in Holmquist's taxonomy [23].

and couple the spoon with the table. May be, Bob did not mean this. Initiative owning relates to implicit and explicit interaction which, in turn, relies on an appropriate model of context of use [10]. Context modeling as well as the balance between implicit and explicit interaction are still open issues.

Once a meta-UI service is launched, what kind of control does the user have? At minimum, *observability* should be supported, i.e. users should be able to evaluate the internal state of the service from its current perceivable representation. The next step is *traceability* by which users can observe the evolution of the service over time, but they cannot modify this evolution. With *controllability*, users can observe, trace, and intervene on the evolution of the meta-UI service. As shown in Figure 6, Sedan-Bouillon users can decide where to re-distribute the UI components of the web site.

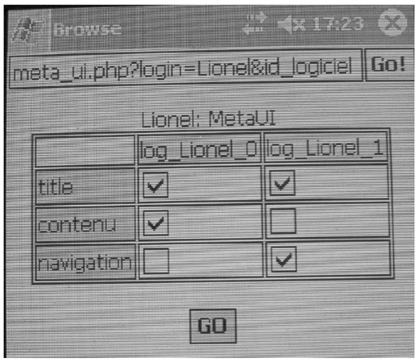


Figure 6. This form allows users to specify the re-distribution of the UI components of the Sedan-Bouillon web site.

Accessing a particular service of a meta-UI and controlling its execution is supported by the way of an interaction technique.

2.4 Interaction technique

In the context of this article, the term interaction technique denotes the user interface of the meta-UI. It is the set of sentences (i.e. the language) built from vocabulary elements assembled according to some predefined syntax, and whose semantics is expressed in terms of the generic services of the meta-UI.

Objects that can be involved in a meta-UI service are denoted by vocabulary elements. Some objects may be denoted by a representative which, in turn, may be *numerical* of *physical*. Others have *no representative*. They belong to the vocabulary. In the Olympic café scenario, a spoon is a physical representation of a street. It is not the street of the real world. Alternatively, in GUI geographical systems, real-world streets are represented numerically as lines in graphical maps. When coupling a mouse with its driver by plugging its connector into a USB port, the physical object mouse is part of the sentence. It is not represented. Coupling Hinckley's tablets is performed by bringing them in contact [20]. The tablets are not represented. The absence of representative may apply to digital objects as well: when moving a window, one acts on the window per se, not on a representative. Jigsaw, based on the noun metaphor, uses jigsaw pieces to represent objects and services for domestic environments [38]. Jigsaw pieces may be numerical or physical.

The elaboration of a sentence, which requires objects and functions designation, may be *direct* by acting on vocabulary elements, or indirect by the way of *instruments*. In turn, instruments may be physical or digital with various levels of indirection [4]. Moving a window with a finger is direct, whereas, moving it by the way of a pen is instrumental (the pen acts as a physical instrument). Alternatively, moving a window with a mouse is instrumental with one additional level of indirection: the window is not moved by the mouse, but by a pointer, a digital representative of the mouse. By assembling Jigsaw pieces, users can build sentences like "if someone rings the bell, take a picture and send it to my PDA" (voir figure 7). Selecting physical pieces is direct whereas selecting numerical pieces is instrumental.

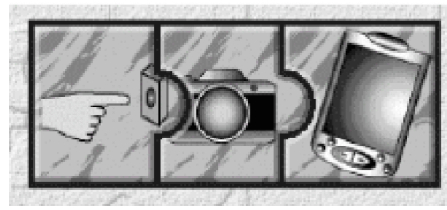


Figure 7. A Jigsaw sentence as an assembly of jigsaw pieces.

When reasoning about languages, *extensibility* is a typical issue to consider. Applied to our domain, is it possible for users to extend the *vocabulary* of the interaction technique, change its *syntax*, and from there, extend its *semantics*, thus create new meta-UI services? In an unbound domain like ambient computing, extensibility seems unavoidable, but the risk of introducing additional complexity is high. In conventional desktops, most users build simple sentences such as 'move this window here" or "cancel this process". Simple programs are elaborated by the way of macros. From end-user programming, we are now entering the area of end-user development. Although end-users are not programmers, they want to program their interactive space without meaning it. This is yet another key challenge that systems like Jigsaw, iCAP, and many others try to address (Cf. Table 1).

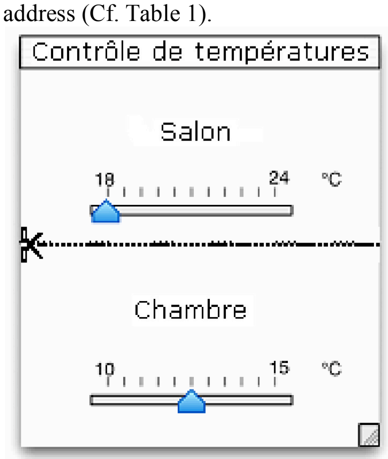


Figure 8. The scissors icon, which allows users to cut the window into two pieces, denotes a meta-UI service whose UI is weaved into the UI of domain-dependent services (here, controlling the heat level of two rooms in a domestic environment).

Given that a meta-UI provides users with the means to govern the domain-specific services of an interactive ambient space, how do its UI elements relate with that of the domain-specific services? What are they *level of integration*? We propose two approaches to this question: all or parts of the UI components of the meta-UI are *embedded* with (or weaved into) the UI components of the domain-dependent services. Collapse-to-zoom [3] and "Attach me, detach me, Assemble me" use the weaving approach [18]. Figure 8 shows another example. Alternatively, UI components of the meta-UI services are not mixed with the UI components of the domain-dependent services. They are *external*. As shown in Figure 6, Sedan-Bouillon users specify the re-distribution of the UI components of the web site using a dedicated form. Actually, a mix of weaving and external sounds reasonable. For example, in Sedan-bouillon, access to the re-distribution service is embedded in the navigation bar of the web site (see the "meta-UI" link in Figure 4), whereas the specification of the re-distribution is external.

3. ANALYSIS

Tables 1 and 2 synthesize research efforts we have selected for their relevance to the area of meta-UI. Although we have used a subset of our dimension space, the tables call for the following preliminary remarks.

Table 1. Meta-UI's according to our dimension space. (Legend: E denotes the existence of the service, O = Observability of the service, T=Traceability, C=Controllability)

	Discovery				Coupling				Re-distrib.				Re-mould.			
	E	O	T	C	E	O	T	C	E	O	T	C	E	O	T	C
Aris [7]	X	X			X				X	X	X	X				
a CAPella [13]	X	X														
DongleRelate [26	X	X		X	X	X		X								
MigreXML [30]	X	X			X				X	X	X	X	X			
AmbientDesk [11]	X	X			X	X		X	X	X		X				
SpeakEasy [32]	X	X		X	X				X			X	X			
Jigsaw [38]	X	X			X	X		X								
E-Gadget [27]	X	X			X	X		X								
iCAP [39]	X	X		X												
Lego Logo [28]					X	X		X								
Bope [34]					X			X	X			X				
MightyMouse [9]	X	X			X				X			X				
Dynamo [24]	X	X			X	X		X	X	X		X				
Peebles [31]	X				X				X	X		X				
PutThatThere [19]									X			X				
iStuff [2]	X				X											
Icrafter [35]	X	X		X												
Collapse [3]									X			X				
AttachMe [18]					X			X	X			X				
Stitching [21]					X			X	X	X	X	X				
DataTiles [37]					X	X		X								
Triangles [17]					X	X		X								
Hinckley [22]					X	X		X	X	X		X	X			
SyncTap [36]					X			X								
tranSticks [1]					X			X								

Table 2. Meta-UI's according to our dimension space (contd.).

	Object Types			Representation			Integration level	
	Mixed-by-Construct	Mixed-by-design	Digital	No-Repr.	Digital Repr.	Physical Repr.	Em-bedded	Exter-nal
Aris		X	X		X			X
a CAPella	X	X	X		X			X
DongleRelat		X			X			X
MigreXML		X	X		X			X
AmbientDes		X	X	X	X			X
SpeakEasy		X	X		X			X
Jigsaw	X	X	X		X	X		X
E-Gadget	X	X			X			X
iCAP	X	X	X		X			X
Lego Logo	X	X				X		X
Bope		X	X		X	X		X
MightyMous		X	X		X			X
Dynamo		X	X		X			X
Peebles		X	X		X			X
PutThatTh			X	X	X			X
iStuff	X	X		X				X
Icrafter		X	X		X			X
Collapse			X	X			X	
AttachMe			X	X			X	
Stitching			X	X	X			X
DataTiles			X		X	X		X
Triangles		X	X		X	X		X
Hinckley		X	X	X				X
SyncTap		X	X	X	X			X
tranSticks		X	X			X		X

In terms of functional coverage and human control:

- None of the systems proposes the full set of meta-UI services. In particular, role assignment is ignored (which means that it is hard-coded into the systems), and re-moulding is rarely supported. On the other hand, discovery and coupling are rather well represented. In addition, if discovery is supported, coupling is available as well: objects that can be discovered call naturally for their assembly. If coupling is available, re-distribution is generally supported, but, as in Put-That-There, the opposite is not necessarily true. In this case, the ambient space permits re-configuration but only for a predefined set of objects.
- Almost all meta-UI's of our list manipulate pure digital objects, and most of them include mixed-by-design objects

(that is objects assembled by designers). The construction of mixed objects by end-users is emerging, but still under-represented.

- So far, very little attention has been paid to the level of control left to the human actor. In particular, traceability has been systematically ignored. Discovery is systematically made observable, and, to a less extent, controllable. Coupling is clearly controllable whereas re-moulding is out of human control, not even made observable. In other words, system designers tend to adopt the principles of autonomic computing, rejecting the human from the loop.

In terms of interaction technique:

- Most systems use a mix of two styles of representation: on one hand, digital with physical representations, on the other hand, digital representation and absence of representation. Only two systems cover all three modes: a CAPella and ICAP. Interestingly, the absence of representation is progressively emerging, but then the objects to be manipulated must be closely co-located with the user.
- The UI of the meta-UI's of our list is almost always external, rarely embedded, and never uses a mix of embedded and external. Embedding the UI the meta-UI potentially increases a feeling of continuity at a risk, however, to overload the UI of the domain-dependent services. Not surprising then that observability is not well supported. For example, in Collapse-to-zoom, which allows users to collapse areas of web pages deemed irrelevant, the existence of the service for re-moulding is not observable. An external UI that is not observable has even less chance to be discovered. For example, Stitching [21], based on synchronous gestures [22], allows users to call upon two meta-UI services in one single gesture: using a stylus as an instrument, two tablets can be coupled, and at the same time, the UI components of the domain-dependent services can be re-distributed across the tablets. The interaction trajectory is very efficient, feedback is provided as the gesture is accomplished (re-moulding is both traceable and controllable), but the availability of coupling is not observable.

4. CONCLUSION: DIRECTIONS FOR FUTURE RESEARCH

The frenetic development of ambient interactive systems has entailed the creation of many sorts of end-user tools for controlling and shaping their interactive spaces. In this article, we propose the concept of meta-UI as a unifying umbrella along with a taxinomy to structure the problem space and to identify directions for future research.

Using our taxonomy, the analysis of the state of the art shows that we need to re-think the "basics" (find, cut&paste, etc.) in the light of multi-scale interactive spaces; The integration (embedded-ness) of the meta-UI with domain-dependent services has not been addressed explicitly, and services like role assignment and coupling have been overlooked [11]. Models and mechanisms are currently being developed for re-moulding and re-distribution under the umbrella of plastic UI's and context-aware adaptive UI's. But we should be very careful at considering the level of control left to end-users. We all agree that the Human should be kept in the loop, but the temptation is high for "systemers" to develop autonomic systems just for the sake of the scientific challenge.

Based on these observations, we have designed a prototype meta-UI that supports the discovery and coupling of mixed-by-design objects (PDA's and PC), as well as UI re-distribution. The integration of the UI of this meta-UI uses a combination of embedded-ness and externality using objects ownership as a driving principle: the user interface of the meta-UI services that act on the UI components of the domain-specific services are embedded (e.g., splitting and duplicating informational content) whereas the user interface of the meta-UI services that manipulate objects that are domain-independent (e.g., coupling PDA's and PC's) is external. For UI re-distribution, which allows end-users to re-allocate domain-dependent UI components to domain-independent objects such as PDA's and PC's, we use a mix of embedded and external approaches to bridge the gap between the system infrastructure and the application domain.

This early experience also demonstrates the difficulty to empower end-users with the appropriate programming mechanisms. End-user programming has been around for nearly twenty years. Graphical notations and programming by demonstration, all have shown limitations over textual programming. Combining expressive power and simplicity is yet another challenge to address for ambient interactive spaces.

5. ACKNOWLEDGMENTS

This work is supported partly by the EMODE ITEA project (if04046), and partly by the SIMILAR network of excellence (http://www.similar.cc), the European research task force creating human-machine interfaces similar to human-human communication (FP6-2002-IST1-507609). Our thanks to Anne Roudaut, Xavier Alvaro, Jean-Sebastien Sottet, and Lionel Balme for the early implementation of the CESAM meta-UI prototype.

6. REFERENCES

[1] Ayatsuka, Y., and Rekimoto, J. tranSticks: Physically Manipulatable Virtual Connections. In *Proceedings of the SIGCHI conference on Human factors in computing systems* (CHI 2005)*,* ACM Press, New York, NY, 2005, 251-260.

[2] Ballagas, R., Ringel, M., Stone, M., and Brochers, J. iStuff: A Physical User Interface Toolkit for Ubiquitous Computing Environments. In *Proceedings of the SIGCHI conference on Human factors in computing systems (CHI 2003),* ACM Press, New York, NY, 2003, 537-544.

[3] Baudisch, P., Xie, X., Wang, C., and Ma, W.Y. Collapse-to-zoom: Viewing Web Pages on Small Screen Devices by Interactively Removing Irrelevant Content. In *Proceedings of User Interface Software and Technology (UIST 2004)*, Technotes, ACM Press, New York, NY, 2004, 91-94.

[4] Beaudouin-Lafon, M. Instrumental Interaction: An Interaction Model for Designing Post-WIMP User Interfaces. In *Proceedings of the SIGCHI conference on Human factors in computing systems (CHI 2000),* ACM Press, New York, NY, 2000, 446-453.

[5] Bellotti V., Back M., Edwards K., Grinter R., Henderson A., and Lopes C. Making Sense of Sensing Systems: Five Questions for Designers and Researchers. In *Proceedings of the SIGCHI conference on Human factors in computing systems (CHI 2002),* ACM Press, New York, NY, 2002, 415–422.

[6] Berti, S., and Paternò, F. Migratory multimodal interfaces in multidevice environments. In *Proc. International Conference*

on Multimodal Interfaces (ICMI 2005), ACM Publ., 2005, 92-99.

[7] Biehl, J.T, and Bailey, B.P. ARIS: An interface for application Relocation in an interactive space. *In Proceedings GI 2004, Canadian Human-Computer Communications Society,* 2004, 107-116.

[8] Block, F., Schmidt, A., Villar, N., and Gellersen, H.W. Towards a Playful User Interface for Home Entertainment Systems. In *Proceedings of European Symposium on Ambient Intelligence (EUSAI 2004)* (Eindhoven, Netherlands), Springer, 2004, 207-217.

[9] Booth, K.S., Ficher, B.D., Lin, C.J.R., and Argue, R. The « mighty Mouse » Multi-Screen Collaboration Tool. *Proceedings of User Interface Software and Technology (UIST 2002)*, ACM Press, New York, NY, 2002, 209-212.

[10] Coutaz, J., Crowley, J. , Dobson, S., and Garlan, D. Context is Key, *Communications of the ACM*, ACM Publ., 48(3), March 2005, 49-53.

[11] Coutaz, J., Borkowski, S., and Barralon, N. Coupling Interaction Resources: an Analytical Model. In *Proceedings of European Symposium on Ambient Intelligence (EUSAI 2005) (Grenoble, France)*, 183-188.

[12] Coutaz, J., Nigay, L., Salber, D., Blandford, A., May, J. and Young, R. Four Easy Pieces for Assessing the Usability of Multimodal Interaction: The CARE properties. In *Proceedings of the INTERACT'95 conference (Interact'95) (Lillehammer, Norway)*, S. A. Arnesen & D. Gilmore Eds., Chapman&Hall Publ., June 1995, 115-120.

[13] Dey, A., Hamid, R., Beckmann, C., Li, Y., and Hsu, D. a CAPpella: Programming by Demonstration of Context-Aware Applications. In *Proceedings of the SIGCHI conference on Human factors in computing systems (CHI 2004) (Vienne, Austria),* ACM Press, New York, NY, 2004, 33-40.

[14] Fishkin, K.P., A taxonomy for and analysis of tangible interfaces. *Journal of Personal and Ubiquitous Computing*. Springer Verlag, 2004, 8: 347–358.

[15] Fitzmaurice, G., Ishii, H., and Buxton, W. Bricks: Laying the foundations for graspable user interfaces. In *Proceedings of the SIGCHI conference on Human factors in computing systems (CHI 95),* ACM Press, New York, NY, 1995, 432-449.

[16] Garlan, D., Siewiorek, D., Smailagic, A. and Steenkiste, P. Project Aura: Towards Distraction-Free Pervasive Computing. *IEEE Pervasive Computing*. Volume 21, Number 2, April-June, 2002, 22-31.

[17] Gorbet, G.M., Orth, M., and Ishii, H. Triangles: Tangible Interface for Manipulation and Exploration of Digital Information Topography. *In Proceedings of the SIGCHI conference on Human factors in computing systems (CHI 98) (Los Angeles, USA),* ACM Press, New York, NY, 1998, 49-56.

[18] Grolaux, D., Vanderdonckt, J., and Van Roy, P. Attach Me, Detach Me, Assemble Me Like You Work. In *Proceedings of 10th IFIP TC 13 Int. Conf. on Human-Computer Interaction Interact'2005 (Rome, Italy),* 2005, 198-212.

[19] Harada, S., Hwang, J., Lee, B., and Stone, M. "Put-That-There": What, Where, How? Integrating Speech and gesture in Interactive Workspace. In *Proc. International Conference on Computer Graphics and Interactive Techniques,* 262–270.

[20] Hinckley, K., Pierce, J., Sinclair, M., and Horvitz, E. Sensing Techniques for Mobile Interaction. In *Proceedings of User Interface Software and Technology (UIST 2000)*, ACM Press, New York, NY, 2000, 91-100.

[21] Hinckley, K., Ramos, G., Guimbretiere, F., Baudisch, and P., Smith, M. Stitching: Pen Gesture that Span multiple Displays. In *Proceedings of Advanced Visualization of Information (AVI 2004) (Gallipoli, Italy)*, ACM Press, New York, NY, 2004, 23-31.

[22] Hinckley, K. Synchronous gestures for multiple persons and computers. In *Proceedings of User Interface Software and Technology (UIST 2003) (Vancouver, Canada)*, ACM Press, New York, NY, 2003, 149-158.

[23] Holmquist, L.E, Redström, J., and Ljungstrand, P. Token-Based Access to Digital Information. *Lecture Notes in Computer Science 1707,* 1999, 234–245.

[24] Izadi, S., Brignull, H., Rodden, T., Rogers, Y., and Underwood, M. Dynamo: a public interactive surface supporting the cooperative sharing and exchange of media. *Proceedings of User Interface Software and Technology (UIST 2003) (Vancouver, Canada)*, ACM Press, New York, NY, 2003, 159-168.

[25] Johanson B., Fox A., and Winograd T. The Interactive Workspace Project: Experiences with Ubiquitous Computing Rooms. *IEEE Pervasive Computing Magazine 1(2)*, April-June 2002.

[26] Kortuem, G., Kray, C., and Gellersen, H. Sensing and Visualizing Spatial Relations of Mobile Devices. In *Proceedings of User Interface Software and Technology (UIST 2005)*, ACM Press, New York, NY, 2005, 93-102.

[27] Markopoulos, P., Mavrommati, I., and Kameas, A. End-User Configuration of Ambient Intelligent Environments: Feasibility from a User Perspective. In *Proceedings of European Symposium on Ambient Intelligence (EUSAI 2004)* (Eindhoven, Netherlands), Springer, 2004, 243-254.

[28] McNerney, T.S. From turtles to Tangible Programming Bricks: explorations in physical language design. *Journal of Personal and Ubiquitous Computing,* 8(5), 2004, 326-337.

[29] Metze, F., Gieselmann, P., Holzapfel, H., Kluge, T., Rogina, I., Waibel, A., Wölfel, M., Crowley, J., Reignier, P.,Vaufreydaz, D., Bérard, F., Cohen, B., Coutaz, J., Arranz, V., Bertran, M., and Rodriguez, H. The FAME Interactive Space. *2nd Joint Workshop on Multimodal Interaction and Related Machine Learning Algorithms (MLMI) (Edinburgh, UK)*, July 2005.

[30] Molina, J.P., Vanderdonckt, J., López, P.G, Caballero, F., and Pérez M. Rapig Prototying of distributed user interfaces, In *Proceedings of Computer-Aided Design of User Interfaces (CADUI'06)*, 2006.

[31] Myers, B. Mobile Devices for Control. In *Proceedings of the 4th International Symposium on Mobile Human-Computer Interaction (MobileHCI 02),* Springer-Verlag, 2002, 1-8.

[32] Newman, M.W., Sedivy, J.Z., Neuwirth, C.M., Edwards, W.K., Hong, J.I., Izadi, S., Marcelo, K., and Smith, T.F. Designing for Serendipity: Supporting End-User Configuration of Ubiquitous Computing Environments. In *Proceedings of*

Designing Interactive Systems (DIS 2002) (London UK), ACM Press, New York, NY, 2002, 147-156.

[33] Norman, D.A., and Draper, S.W. *User Centered System Design: New Perspectives on Human-Computer Interaction*. Erlbaum Associates, Hillsdale, NJ, 1986.

[34] Pering, T., Ballagas, R., and Want, R. Spontaneous marriages of Mobile Devices and Interactive Spaces. *Communications of the ACM,* Volume 48, Issue 9, (Sep. 2005), 53 – 59.

[35] Ponnekanti, S.R., Lee, B., Fox, A., Hanrahan, P., and Winograd, T. ICrafter: A Service Framework for Ubiquitous Computing Environments. In *Proceedings of Ubiquitous Computing Conference (Ubicomp 01)*. Springer-Verlag, Lecture Notes in Computer Science, vol. 2201, 2001, 56-75.

[36] Rekimoto, J., Ayatsuka, Y., and Kohno, M. SyncTap: An Interaction Technique for Mobile Networking. In *Proceedings of the 5th International Symposium on Mobile Human-Computer Interaction (MobileHCI 03),* Springer-Verlag, 2003, 104-115.

[37] Rekimoto, J., Ullmer, B., and Oba, H. DataTiles: A Modular Platform for Mixed Physical and Graphical Interactions. In *Proceedings of the SIGCHI conference on Human factors in computing systems (CHI 2001) (Seattle USA),* ACM Press, New York, NY, 2001, 269-276.

[38] Rodden, T., Crabtree, A., Hemmings, T., Koleva, B., Humble, J., Akesson, K.P., and Hansson, P. Configuring the Ubiquitous Home. In *Proceedings of the 2004 ACM Symposium on Designing Interactive Systems (DIS 04) (Cambridge, Massachusetts),* ACM Press, 2004.

[39] Sohn, T.Y., and Dey, A.K. iCAP: An Informal Tool for Interactive Prototyping of Context-Aware Applications. In *Proceedings of the International Conference on Pervasive Computing 2006 (Pervasive 2006) (Dublin, Ireland),* May 2006, 974–975.

[40] Streitz N., Tandler P., Müller-Tomfelde C., and Konomi S. Roomware: Towards the Next Generation of Human-Computer Interaction based on an Integrated Design of Real and Virtual Worlds. *In Human-Computer Interaction in the New Millenium*, Carroll J. (Ed.), Addison-Wesley, 2001, 553-578 http://www.ipsi.fraunhofer.de/ambiente/english/projekte/projekte/dynawall.html

[41] Tandler, P., Prante, T., Müller-Tomfelde, C., Streitz, N., and Steinmetz, R. ConnecTables: Dynamic Coupling of Displays for the Flexible Creation of Shared Workspaces. In *Proceedings of User Interface Software and Technology (UIST 2001) (Orlando USA)*, ACM Press, New York, NY, 2001, 11-20.