Ontology for Multi-surface Interaction

Joëlle Coutaz, Christophe Lachenal, Sophie Dupuy-Chessa
CLIPS-IMAG, BP 53, 38041 Grenoble Cedex 9, France
{Joelle.Coutaz, Christophe.Lachenal, Sophie.Dupuy-Chessa}@imag.fr

Abstract: Digital computation is a powerful source of functional support. However, it has been confined to the augmentation of single objects only. In this article, we are interested in the combination of physicality with computation in the context of multiple objects. We propose the notion of multi-surface interaction as a unifying paradigm for reasoning about both emerging distributed UI's and known interaction techniques such as GUIs, tangible UIs, and manipulable UIs. Multi-surface interaction is expressed within an ontology that shows how our concepts feed into the design of sound foundational software for the development of ubiquitous user interfaces.

Keywords: multi-surface interaction, interaction technique, ubiquitous computing.

1 Introduction

Surfaces play a predominant role in our daily life. In civil architecture, they structure the space into places (Harrison et al, 1996) to support the accomplishment of specific activities or to favor the emergence of new ones. From the earlier ages, surfaces such as frescoes and art paintings, have served as efficient communication means. Similarly, public walls, blackboards, desks and tables, the annotated page of a book, or the back of an envelope, are crossroads for human activities. In HCI, the display screen is still the most familiar surface for interacting with computation.

All of the surfaces in the world serve a purpose based on their interactional properties: some surfaces are wearable or fit in the hand while others are too heavy to be moved around; some surfaces can be folded, torn off, and thrown away, while others are perennial. Some surfaces, such as a rain curtain, can be traversed while others form a rigid boundary. In short, physical surfaces are pervasive.

With the emergence of ubiquitous computing, physical surfaces are augmented with computational capabilities. The Mixed Reality trend exemplified by the Digital Desk pioneering work is one such approach (Wellner et al, 1993). Wireless connectivity and miniaturization go one step further, offering new opportunities for innovation: from centralized display on a single surface, user interfaces can now migrate freely and be distributed across multiple surfaces. For example, proximal PDA's could provide a reconfigurable mosaic on which the user interface can expand and shrink as required.

Because digital computation is a powerful means for interpreting information, we are interested in bringing together physical surfaces with digital information. This leads us to the concept of information surfaces. However, information surfaces gain added utility when they can be manipulated. Manipulation coupled with observation form the essence of interaction. We call this "multi-surface interaction".

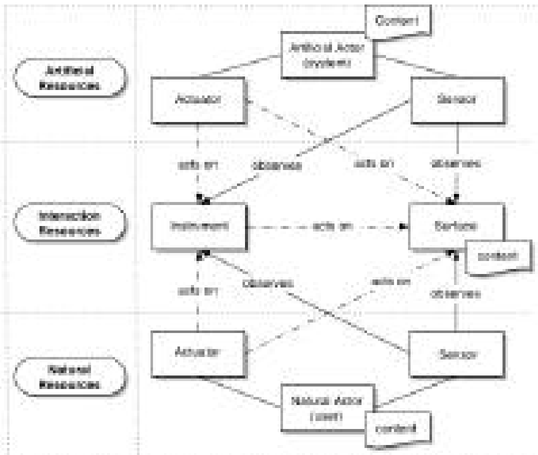

Figure 1: Informal overview of multi-surface interaction.

In this article, we propose an ontology that shows how the concept of multi-surface interaction can serve as a unifying framework for reasoning about both emerging distributed UI's and current interaction techniques such as GUIs, tangible UIs, and manipulable UIs. The article is structured as follows. The ontology is outlined in the next Section. Then the central concepts of multi-surface interaction are presented in details: surfaces, instruments, as well as the nature of their coupling with computational content. In the last section, we briefly analyze the implications of multi-surface interaction for the development of new foundational software tools.

2 Ontology: Overview

Figure 1 shows an informal overview of multi-surface interaction. It is completed with the UML formal description of Figure 2. Interaction resources serve as mediators between an artificial actor (e.g., a ubicomp system) and a natural actor (e.g., a user). An interaction resource may serve as an instrument and/or as a surface. As an instrument, an interaction resource mediates the actions of an actor. As a surface, the outmost boundary of a physical entity serves as a recipient for making information observable to an actor.

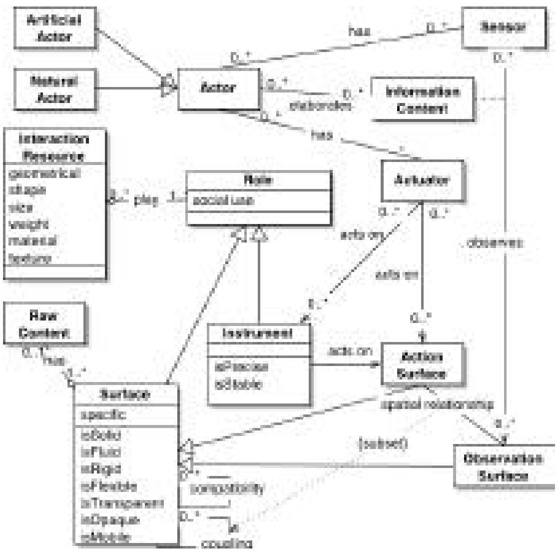

Figure 2: UML description of the Multi-surface Interaction ontology.

For example, in the GLOSS multi-surface system shown in Figures 3 and 4, a laser pointer and the "GLOSS clicker" are instruments to select raw content displayed on two surfaces: a wall and a table. The pucks of the Senseboard, on the other hand, serve both as instruments and surfaces (Jacob et al, 2002). The Senseboard provides a tangible user interface for manipulating and organizing pieces of information as in spreadsheets. Each puck is a small magnetized plastic tag that denotes a piece of information. It can be placed into grid cells to move information. As such, it is an instrument. In addition, a video-projector displays specific information on the top of the puck: it serves as a surface as well. As expressed in Figure 2 (using Coad's role pattern (Coad, 1992)), an interaction resource may play multiple roles: that of a surface and/or that of an instrument.

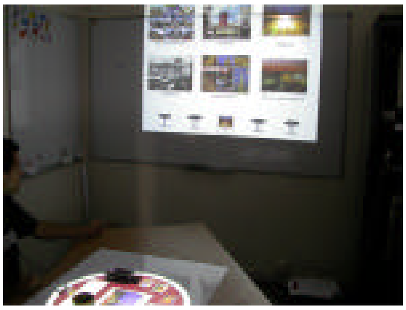

Figure 3: The GLOSS system multi-surface setting.

As shown in Figure 1, actors have actuators to modify the state of an interaction resource, and have sensors to observe the state of interaction resources. For example, actuators of the GLOSS system are two beamers that project information on the wall and the table surfaces. Its sensors are two video cameras that observe and track light beams.

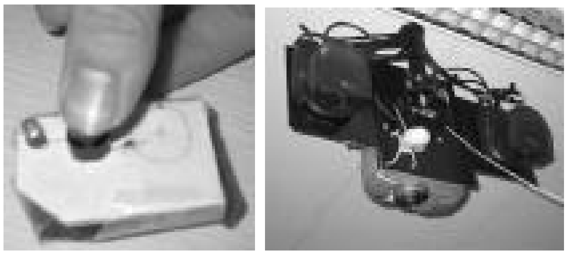

Figure 4: On the left, the "GLOSS clicker", a wireless "home made" pointing instrument that includes a press button and a LED mounted on top of a 2X4 cm foam box wrapped with paper. A battery embedded in the box provides power to the LED. On the right, a GLOSS system actuator (video-beamer) and two camera sensors.

Actors draw upon information content to perform computation. As shown in Figure 2, information content, is built from many sources including raw contents that can be observed from a surface. Markings on a sheet of paper, whether digital or physical, constitute the raw content of the sheet surface.

Inspired from the Model Human Processor (Card, 1983), our representation of a human actor in terms of "sensors-actuators-information content" is simplistic but is sufficient to elicit two relations:

"acts on" and "observes" which, as discussed next, set the foundations for analyzing multi-surface interaction.

3 Surfaces

The relations "acts on" and "observes" call for the definition of two classes of surfaces: action surfaces and observation surfaces. The adequacy of a surface for action and/or for observation depends on its attributes and properties. In turn, relationships between multiple surfaces provide a way to analyze the interaction space they form. These issues are discussed next.

3.1 Action and Observation Surfaces

An action surface is a subset of a physical surface on which an actor can act directly with actuators and/or indirectly with instruments. Similarly, an observation surface is a subset of a physical surface that an actor can observe with sensors (see Figure 2).

For example, the GLOSS table is a plastic woodcut surface in which an off-the shelf white board has been incrusted. As shown in Figure 4, a camera and a projector look over the table. In this particular setting, the camera is calibrated to observe the exact surface of the white board. Therefore, the white board is, for the GLOSS actor, an observation surface. The video-projector displays information on the white board within a circular area. The circular area is, for the GLOSS actor, an action surface. For human actors, on the other hand, the entire surface of the table can be acted on and observed. As another example, the top surface of a Senseboard puck is an action surface for the system and an observation surface for the user.

The analysis of the spatial relationships between action and observation surfaces may provide useful insights for design. Typically, an empty intersection between the action and the observation surfaces of a human actor, predicts discontinuity (e.g., the user can't see what he is doing). In the GLOSS example, the user's action and observation surfaces include the observation and action surfaces of the system. In turn, the observation surface of the system includes its action surface. This situation triggers a number of design issues. For instance, will users be aware that among the actions they perform outside the circular area, some of them are observed while others are not, given that, in both cases, GLOSS is unable to provide feedback outside the circular area? Should GLOSS and user's observation and action surfaces be redesigned so that they overlap as in Rekimoto's Pick-and-Drop setting (See Figure 5)? Alternatively, do the physical attributes and properties of the surfaces provide users with the "right" affordance?

3.2 Attributes of a Surface

Because multi-surface interaction is grounded on physicality, we suggest the following (non-exhaustive) list of attributes to characterize surfaces: the geometrical shape (e.g., sphere, polygon, a human face as in Hypermask where computer-generated expressions are projected on a blank mask), size and weight, material (e.g., wood, plastic, vapor, water), color, texture (e.g., homogeneous and smooth), and social use (public, private).

Attributes of a surface determine the modalities that are necessary for observing and for acting: Modalities for observation denote the sensors involved in observing the content of a surface. For human actors, these include sight, hearing, etc. For the GLOSS actor, the modality for observation is based on computer vision. Modalities for action denote the classes of actions that are applicable to the surface such as writing, folding, and moving.

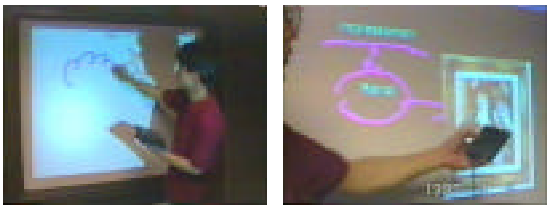

Figure 5: Pick-and-Drop examples used with the permission of Rekimoto (Rekimoto, 1997). On the left, the painter metaphor where the user's and system's action and observation surfaces overlap. On the right, coupling between two surfaces based on proximity.

Attributes are characteristics of an entity. They determine the properties of this entity.

3.3 Properties of a Surface

A property is the capability of an entity to fulfill a particular function. In HCI, properties provide a useful structure for design. As for attributes, our list of properties is incomplete but can be easily extended. For the purpose of this article, the list is intended to show how the ontology feeds into the design process, and from there, into the design of sound foundational software for the development of ubiquitous user interfaces.

The *solidity/fluidity/nebulosity* of a surface is inferred from the material it is made of. Liquid or vapor, it is ephemeral but it can be traversed. For example, a rain curtain (Koleva et al, 2000) on which images are projected, serves as a passage between virtual and real worlds.

The *rigidity/flexibility* of a surface expresses its capacity to change its shape and size. Most interactive surfaces are rigid: iRoom (Johanson et al, 2002), i-LAND (Streitz et al, 2001), Rekimoto's and Bérard's augmented surfaces (Rekimoto et al, 1999; Bérard, 1999) to name a few. On the other hand, the electronic paper, from Xerox and MIT, as well as the electronic fibers developed in (http://www.fibercomputing.net; Deflin et al, 2002) open the way to foldable surfaces. Illuminating Clay, which allows the user to shape a surface made of clay, is another promising approach to non-rigid interactive surfaces (Piper et al, 2002).

The *opacity/transparency* of a surface is widely exploited in civil architecture. A transparent surface enriches itself with environmental information. It favors openness while forming a boundary. Clear-Board is an early example of transparent surfaces applied to computer-mediated communication (Ishii et al, 1992). Windowpanes augmented with piezo-electric transducers allow pedestrians to interact with the shop by tapping.

Mobility coupled with *lightness* and *smallness* opens the way to new usage. A lightweight small size surface like a PDA can be carried in the hand. If so, it may also serve as a pointing instrument as in Pick-and-Drop (Cf. Figure 5) and manipulative UIs (Harrison et al, 1998).

Writa-bility/erasibility denotes the capacity of a surface to be modified with "write actions" and its capacity to be erased. A public wall is writable but tagging a wall is socially incorrect: it cannot be erased in a simple way. On the other hand, light and sound shows, public-animated walls based on tracking human movements (Maynes-Aminzade et al, 2002) are appropriate since they use digital ink, an erasable material.

Heterogeneity may enforce partitioning of a surface into areas for actions and areas for observation. For example, the whiteboard area of the GLOSS table affords scribbling, the wooden-look plastic area suggests piling up private paper documents, and the central bright circular area serves as the primary focus for interaction with the system. However, these hypotheses about the affordance of surface heterogeneity, need to be verified.

Refraction and *reflexion* of a surface have a direct impact on its *observability*. In particular, shiny surfaces are nightmare for computer vision based sensors. *Reachability* denotes whether the surface is physically accessible directly: too high, a surface may not be used for action or may require a dedicated instrument (e.g., a ladder or a laser pointer).

So far, we have analyzed a surface as a single entity. We need now to address the composition of multiple surfaces.

3.4 Relationships between Surfaces

By composing surfaces in space, we build geometric configurations that can be described with a topology. In turn, a topology sets the foundation for analyzing two properties of the spatial relationships between surfaces: coupling and compatibility.

3.4.1 Topology between surfaces

The purpose of a topology is to describe the location and orientation of entities in a reference coordinate system. Here, the entities of interest are those of our ontology, i.e., the actors, actuators, sensors, instruments, observation and action surfaces involved in a particular interactive situation.

The user's position in a multi-surface space matters. For example, the GLOSS circular area can be rotated on user's request by clicking on the white border of the circle. Alternatively, this articulatory task could migrate to the system if GLOSS were able to maintain the location of the user with regard to the table.

Similarly, the relative positions of surfaces matter. For example, in Built-IT (Rauterberg et al, 1998) and Rekimoto's augmented surfaces, objects are represented as 3D graphics interactors on laptops, whereas 2D rendering is used for objects placed on a horizontal surface. In these examples, the orientation of the rendering surfaces relative to the user (e.g., "horizontal" and "vertical") determines the nature of the output modalities.

In the Pick-and-Drop example of Figure 5, the user brings a PDA close to an electronic wall-board. In this configuration, the user can pick any information from the PDA and drop it on the board at the location denoted by the PDA position. This capacity to detect proximity provides one way to control surface coupling.

3.4.2 Coupling

Coupling between surfaces denote their mutual dependency. Two surfaces are coupled when a change of state of one surface has an impact on the state of the other. The Pick-and-Drop presented above is one such example. Another example is the need for the maintenance of spatial constraints between surfaces.

In modern classrooms, blackboards are comprised of several panes that can be moved up and down as needed. Teachers frequently project prepared slides on the boards. They augment the slides opportunistically by writing on the board with ink-pens. When they scroll the board, the ink inscriptions are scrolled but the projected slide is not. In this setting, the system, which has no sensing capacity, is unable to adjust its action surface to that of the user. The everywhere display projector is one option to solve our problem (Pinhanez et al, 2001).

3.4.3 Compatibility

Compatibility between surfaces expresses the possibility to use them conjointly, by complementarity, redundancy, equivalence, and assignation.

The painter metaphor illustrated in Figure 5, is an example of multi-surface interaction based on *complementarity*: tools palettes are displayed on the PDA, the PDA is held in the non dominant hand, the user holds a stylus in the dominant hand, and like the painter artist picks the appropriate tool on the palette with the stylus, then draws on the canvas supported by the wall-size electronic screen. In this example, the complementarity between the PDA and the electronic board is grounded on the differences between their size and weight attributes. Complementarity between surfaces may also rely on the similitude of their attributes to define new functions. The Data Tiles are a good example of this where the topology of the tiles defines the semantics of the composition (Rekimoto et al, 2001).

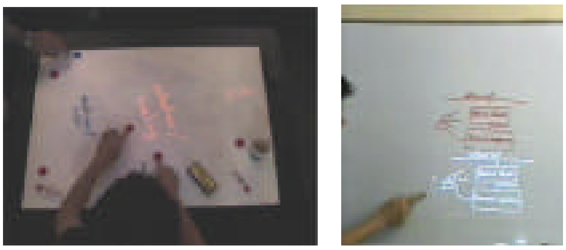

Figure 6: Coupling instruments within the Magic Table (left). On the right, direct coupling of human actuator with content in the Magic Board (Bérard, 1999; Crowley et al, 2000)

Surfaces are composed in a *redundant* way when they are used simultaneously to accomplish the same task. For example, connecting a Smart Board to the video output of a PC allows users to duplicate the user interface on both the electronic board and the display screen of the PC.

Surfaces are functionally *equivalent* when they can be used alternatively to accomplish a given set of tasks. For example, it is increasingly popular to access web services through a workstation, a PDA, or a cellular phone. In general, the user interface is adapted to the display surfaces to satisfy plasticity requirements (Thevenin et al, 1999; Calvary et al, 2001), but at the functional level, they support the same set of tasks.

Surface *assignment* means that each surface of the configuration plays a particular role. For example, in a meeting, personal information editing is assigned to PDA's whereas collaborative editing of a document is assigned to the public electronic board.

4 Instruments

Currently, keyboards and mice are the standard user's instruments. With graspable user interfaces, any object that holds in the hand can serve as an instrument. Fitzmaurice's bricks (Fitzmaurice et al, 1995), phicons (Ishii et al, 1997), and the Xerox pan-tilt-zoom PDA (Harrison et al, 1998), are examples of this approach.

As shown by our ontology, a system actor can act on instruments as well, for example orienting a video-projector mounted on an articulated arm to track a physical entity. The Actuated Workbench uses magnetic forces to move magnetic pucks across a table surface (Pangaro et al, 2002). In this example, pucks serve as instruments for both the user and the system.

4.1 Attributes and properties of instruments

As for surfaces, attributes of an instrument are grounded on physicality: shape, size, weight, material, social use (private, sharable, etc.). From these attributes, one infers its modalities for observation (e.g., touch, sight) as well as its modalities for action (e.g., physical/digital ink scribbling, pointing, moving, reshaping, illuminating). Properties of an instrument are measured against the functions expected from the instrument. For example, *precision* and *stability* are relevant properties for pointing instruments. Scribbling is concerned with *manipulability*.

4.2 Relations between instruments

As for surfaces, instruments belong to the topology mentioned earlier, and their relationships can be analysed according to their level of *coupling* and *compatibility*.

Figure 6 shows how instruments, plastic colored tokens, are coupled within the Magic Table. Two tokens are coupled by bringing them into contact. When coupled, the user can select physical/digital markings by forming a rectangle with the two tokens, one in each hand. Selected physical markings are digitised. Then, selected markings can be simultaneously resized and rotated using the tokens. Coupling ends when one of the tokens is hidden with the hand.

So far, we have discussed intra-relations, essentially relations between surfaces, and relations between instruments. We need now to consider how instruments and surfaces are coupled with content.

5 Coupling with Content

Spatio-temporal coupling and generality/specificity of coupling are concerned with associating interaction resources to information content.

5.1 Spatio-temporal Coupling with Content

Spatio-temporal coupling between entities defines how these entities are associated in time and space. To illustrate the discussion, let us consider the coupling of instruments and actuators to content (whether it be raw content or informational).

Fitzmaurice's observed that in conventional GUIs, the coupling of instruments to logical functions (i.e, information content) is "time-multiplexed": there is only one such instrument attached at a time. As a result, instruments are repeatedly coupled and decoupled to content. With Graspable UIs, the coupling can be "space-multiplexed": different instruments can be attached to different content, each independently (but possibly simultaneously) accessible to the user.

As indicated by our ontology, actuators, such as fingers, can be coupled directly to content without any intermediate instrument. For example, in the Magic Board shown in Figure 6, content is manipulated directly with fingers.

As a generalization of Fitzmaurice's work, we suggest to consider the numbers of actuators, sensors, instruments, and surfaces that can be simultaneously coupled to information content. These numbers can be used as a metric to characterize the system capabilities in relation to human performance and needs. For example, the Magic Table, which is able to track more than 4 tokens at a time, makes possible multi-user interaction. Similar remark holds for Smartskin (Rekimoto, 2002).

5.2 Genericity/Specifity of Coupling with Content

An interaction resource is generic if it can be coupled to any type of content. The mouse, Fitzmaurice's bricks, The Magic Table tokens, and the GLOSS clicker are generic instruments. The GLOSS table and wall are generic surfaces.

Alternatively, an interaction resource is specific when it is dedicated to a particular class of content. Phicons, and more specifically the MIT Dome developed for the Metadesk are specific instruments (Ullmer et al, 1997). In HyperMask, masks which have the shape of a human face, are specific surfaces (http://web.media.mit.edu/~pinhanez/).

Specificity works by analogy with real world entities. As demonstrated by early work on physical programming (and many others), specificity facilitates understanding and exploration (Montemayor et al, 2002). On the other hand, they are not reusable. They can't satisfy the scale factor when the number of information types increases. Therefore, the right mix of generic/specific interaction resources needs to be identified in relation to information space and finality of the system.

So far, we have used our ontology as a classification space for reasoning about current and future user interfaces. In the following section, we analyze the use of the ontology to devise new software requirements for multi-surface interaction.

6 Software Requirements

Windowing systems and toolkit, which set the foundations for the development of user interfaces, primarily address the implementation of user interfaces confined to a single PC, with limited models for display surfaces and input devices.

For example, windows are modelled as rectangular drawables whose borders are constrained to be parallel to that of the display. This model is based on the (wrong) assumption (for ubiquitous computing) that users keep facing a vertical screen and use a single pointing device at a time. Current windowing systems and toolkits are screen centric and have no knowledge of where they are in the world. Although connected to a network, computers sit side by side in front of users, each one forming an independent static set of interaction resources.

MID (Hourcade et al, 1999), which supports the connection of multiple mice to control a single screen surface, as well as Pebbles (Myers et al, 1998), which supports the use of PDA's as control instruments for a large screen display, are significant improvement over conventional toolkits. However, they do not address clusters of computational units (e.g., multiple computers of various kinds). BEACH (Tandler et al, 2001), a software infrastructure for smart rooms, supports an homogeneous cluster of PC's. It provides the programmer with a single logical output display mapped onto multiple physical screen displays. It does not support the dynamic connection of instruments nor does it support an explicit topology between users and interaction resources.

Topology is a difficult problem. The Aura project proposes an interesting hybrid approach to this problem, but it has not been applied to the fine grained location of interaction resources (Jiang et al, 2002). The architecture under development for iRoom (Johanson et al, 2002), EasyLiving (Brumitt et al, 2001), as well as our own work on an Interaction-Abstract Machine (Coutaz et al, 2002), are attempts to address both topology and the dynamic discovery of interaction and computational resources. However all of them are under development.

7 Conclusion

The emergence of ubiquitous computing calls for the definition of new conceptual frameworks for reasoning from both the user and the system perspectives. Our notion of multi-surface interaction is an attempt in this direction. In the article, we have illustrated the unifying power of our notion with examples drawn from graphics. However, multi-surface interaction extends to sonic rendering as well. Any surface such as a door can render sound when augmented with a soundbug (http://www.soundbug-us.com)

Our ontology for multi-surface interaction makes explicit the following important concepts: the dual view, surface-instrument, that any interaction resource may support; the distinction between action surfaces and observation surfaces; the symmetrical role between natural and artificial actors, both of them being characterised by actuators, sensors and information content; the spatio-temporal coupling with content; and the dynamic configuration of interaction resources and actors within a topology.

Topology along with a unifying software infrastructure that manages dynamic heterogeneous clusters of interaction resources, actuators and sensors, are the next software challenge to be addressed.

Acknowledgments

This work has been partly supported by the IST-FET GLOSS project (IST-2000-26070) and IST FAME project (IST-2000-28323).

References

Bérard F., “Vision par Ordinateur pour l’Interaction Homme-Machine Fortement Couplée”. Thèse de doctorat, Université Joseph Fourier, Novembre 1999http://iihm.imag.fr/publs/1999/THESE1999_Berard.pdf

Brumitt B., Shafer, S. “Better Living Through Geometry.” *Personal and Ubiquitous Computing* 2001. Vol 5.1 Springer.

Calvary G., Coutaz J., Thevenin. D. “A Unifying Reference Framework for the Development of Plastic User Interfaces”. *IFIP WG2.7 (13.2) Working Conference, EHCI01*,Toronto, May 2001, Springer Verlag Publ., LNCS 2254, M. Reed Little, L. Nigay Eds, p. 173-192.

Card S., Moran T., Newell A. *The Psychology of Human-Computer Interaction*, Chap. 2. The Human Information Processor, Lawrence Erlbaum Ass. Publ., 1983.

Coad P., “Object-Oriented Patterns”, *Communication of the ACM*, 35(9), Sept. 1992.

Coutaz J., Lachenal C., Rey, G. “Initial Reference Framework for Multi-surface Interaction”, *Deliverables D17*, Project GLOSS IST-2000-26070, Sept. 2002.

Crowley J., Coutaz J., Bérard F. “Things that see: Machine Perception for Human Computer Interaction”, *Communications of the ACM*, ACM, Vol 43 (3), March 2000, p. 54-64.

Deflin E., Weill A., Conkar V. “Communicating Clothes: Optical Fiber Fabric for a New Flexible Display”, *Proc. Avantex Symposium* ,13-15 Mai 2002, Frankfurt

Fitzmaurice G., Ishii H., Buxton W. “Bricks: Laying the Foundations for Graspable User Interfaces”. *In Proc. of the ACM conf. On Human Factors in Computer Human Interaction (CHI95)*, ACM, 1995, p. 442-449.

Harrison, B. L., Fishkin, K. P. ,Gujar, A., Mochon C., Want, R. “Squeeze me, Hold me, Tilt Me ! An exploration of Manipulative User Interface”. *In Proc. ACM conference on Computer Human Interaction (CHI98)*, ACM, p. 17-24.

Hourcade J., Bederson B. “Architecture and Implementation of a Java Package for Multiple Input Devices (MID)”, 1999, Available at: http://www.cs.umd.edu/hcil/mid/

Ishii H., Kobayashi M., Grudin J. “Integration of Interpersonal Space and Shared Workspace: Clearboard Design and Experiments”. *Proceedings Computer Supported Collaborative Work, CSCW 92*, ACM, Nov. 1992, p. 33-50.

Ishii H., Ullmer, B. “Tangible Bits: Towards Seamless Interfaces between People, Bits and Atoms”. *In Proc. of the ACM conf. On Human Factors in*

Computer Human Interaction (CHI97), ACM, 1997, p. 234-241.

Jacob R., Ishii H., Pangaro G. Patten J. "A Tangible Interface for Organizing Information using a Grid.". *In Proc. of the ACM conf. On Human Factors in Computer Human Interaction (CHI2002)*, ACM, 2002, p. 339-346.

Jiang C., Steenkiste P. "A Hybrid Location Model with a Computable Location Identifier for Ubiquitous Computing". *In Proc. of the 4th International Conference on Ubiquitous Computing (Ubicomp 2002), Springer, LNCS 2498, p. 246-263.*

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

Koleva B., Schnädelbach H., Benford S., Greenhalgh C. "Traversable Interfaces Between real and Virtual Worlds". *In Proc. of the ACM conf. On Human Factors in Computer Human Interaction (CHI2000)*, ACM, 2000, p. 233-240.

Konomi S., Müller-Tomfelde C., Streitz N. "Passage: Physical Transportation of Digital Information in Cooperative Buildings". *In Proceedings of the Second International Workshop (CoBuild'99), Cooperative Buildings - Integrating Information, Organizations, and Architecture*, Streitz N., Siegel J., Hartkopf V., Konomi S. (Eds.), LNCS 1670, Heidelberg, Germany, Springer, 1999. p. 45-54. ISBN 3-540-66596-X

Maynes-Aminzade D., Pausch R., Seitz S. "Techniques for Interactive Audience Participation". *In Proc. IEEE 4th International Conference on Multimodal Interfaces (ICMI02)*, Pittsburh, Oct. 2002.

Montemayor J., Druin A., Farber A., Simms S., Churaman W., D'Amour A. "Physical Programming: Designing Tools for Children to Create Physical Interactive Environments". *In Proc. of the ACM conf. On Human Factors in Computer Human Interaction (CHI2002)*, ACM, Minneapolis, Minnesota, 2002, p. 299-306.

Myers B., Stiel H., Gargiulo R. "Collaboration Using Multiple PDAs Connected to a PC". *Proceedings Computer Supported Collaborative Work, CSCW98*, ACM, 1998, Seattle, WA. p. 285-294.

Pangaro G., Maynes-Aminzade D., Ishii H. "The Actuated Workbench: Computer-Controlled Actuation in Tabletop Tangible Interfaces". *In Proceedings of User Interface Software and Technology (UIST 2002)*, ACM, 2002, p. 181-189.

Pinhanez C. "The everywhere Displays Projector: A Device to Create Ubiquitous Graphical Interfaces". *In Proc. of the 3rd International Conference on Ubiquitous Computing (Ubicomp 2001*), Atlanta Sept. 2001, Abowd G., Brumitt B. and Shafer S. Eds., Springer, LNCS 2201, p. 315-331.

Piper B., Ratti C., Ishii H. "Illuminating Clay: a 3D Tangible Interface for Landscape Analysis". *In Proc. of the ACM conf. On Human Factors in Computer Human Interaction (CHI2002)*, ACM, Minneapolis, Minnesota, 2002, p. 355-362.

Rauterberg M. et al. "BUILT-IT: A Planning Tool for Consruction and Design". *In Proc. Of the ACM Conf. In Human Factors in Computing Systems (CHI98) Conference Companion*, ACM, 1998, p. 177-178.

Rekimoto J. "Pick-and-Drop: A Direct Manipulation Technique for Multiple Computer Environments". *In Proceedings of User Interface Software and Technology (UIST 1997)*, ACM, 1997, p. 31-39.

Rekimoto J., Masanori S. "Augmented Surfaces : A Spatially Continous Workspace for Hybrid Computing Environments". *In Proc. of the ACM conf. On Human Factors in Computer Human Interaction (CHI99)*, ACM, 1999, p.378-385.

Rekimoto Y., Ullmer B., Oba H. "DataTiles: A Modular Platform for Mixed Physical and Graphical Interactions". *In Proc. of the ACM conf. On Human Factors in Computer Human Interaction (CHI2001)*, ACM, Seattle, 2001, p. 269-276.

Rekimoto Y. "Smartskin: An Infrastructure for Freehand Manipulation on Interactive Surfaces". *In Proc. of the ACM conf. On Human Factors in Computer Human Interaction (CHI2002)*, ACM, 2002, p. 113-120.

Streitz N., Tandler P., Müller-Tomfelde C. , 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, p. 553-578.

Tandler P. "Software Infrastructure for Ubiquitous Computing Environments : Supporting Synchronous Collaboration with Heterogenous Devices". *In Proc. of the 3rd International Conference on Ubiquitous Computing (Ubicomp 2001*), Atlanta Sept. 2001, Abowd G., Brumitt B. and Shafer S. Eds., Springer, LNCS 2201, p. 96-115.

Thevenin D., Coutaz J. "Plasticity of User Interfaces: Framework and Research Agenda". *In Proc. Interact99*, Edinburgh, Sasse A. and Johnson C. Eds, IFIP IOS Press Publ. , 1999, p.110-117.

Ullmer B., Ishii H. "The MetaDESK: Models and Prototype for Tangible User Interfaces". *In Proceedings of User Interface Software and Technology (UIST 97)*, ACM, 1997, p. 223-232.

Wellner P., Mackay W., Gold R. (Eds). "Computer-Augmented Environments: Back to the Real World". *Communication of the ACM*, ACM, 1993, 36(7).