Paper to be presented at CHI 2007 workshop on Tangible User Interface in Context and Theory, San Jose CA, 28 April 2007.

A Solid Diagram Metaphor for Tangible Interaction

Alan F. Blackwell, Cecily Morrison and Darren Edge

University of Cambridge Computer Laboratory
William Gates Building, J J Thomson Avenue, Cambridge UK.
{afb21, cpm38, dke22}@cam.ac.uk

ABSTRACT
Every approach to interaction design reflects an implicit metaphor held by designers of relationships between the system and user. In the case of interactive research prototypes, as found in ubiquitous computing applications, these implicit metaphors can be inappropriate to real usage scenarios. We propose a novel metaphor of tangible interfaces as being a solid diagram to which we can apply abstract notational analysis, resulting in a powerful analytic approach to interaction design.

Author Keywords
Tangible User Interfaces, Solid Diagrams, Cognitive Dimensions of Notations

ACM Classification Keywords
H.5.2 User Interfaces – Evaluation/methodology, Input devices and strategies, Interaction styles

INTRODUCTION
Interaction design practices are always founded on implicit metaphors. These metaphors establish certain relationships between technology and users as more appropriate than others. In the case of the familiar WIMP interface, the prevailing metaphors have converged and been reified [2], to an extent that both designers and users scarcely notice them any more. Ubiquitous computing technologies have the potential to change the way that users engage with digital abstractions. The options for innovation when generating or evaluating new kinds of interaction, however, can be obscured by reliance on old metaphors.

This paper reviews some of the most prevalent metaphors of interaction, and considers the ways that they influence ubiquitous computing research. We propose that a particularly productive interaction metaphor is to treat the arrangement of physical objects as if they were a notation, a kind of three-dimensional *solid diagram*. This is most useful in the case of tangible user interfaces (TUIs). We propose that reading the physical world as a notational system offers a valuable set of conceptual design tools that can be used to compare usability options and trade-offs, exploring the available design space from an analytic perspective.

IMPLICIT METAPHORS OF COMPUTING
The term "metaphor" is frequently used explicitly by UI designers as a presentational aid or training tool (for example the popular understanding of the "desktop metaphor"), but we are not concerned with explicit use of metaphor here. Instead, we are interested in the implicit metaphors of the relationship between user and system that might influence designers and researchers.

The Conduit metaphor
The *conduit* metaphor is a common conceptual metaphor of human language, which imagines conversations, reading and other human communication as being like a pipe along which a message is passed from one person to another [17]. Among technologists, the popular appeal of this metaphor is further reinforced by the apparent relevance of Shannon's communication theory. As a result, technologists naturally think of traditional UIs as providing a *conduit* from the mental intention of the user to the behaviour of the computer, via the keyboard as a channel. When the computer responds or informs the user, this duplex communication seems much like the exchange between two people having a conversation. If human experience of language is reduced to a conduit, we can describe machine communication as though it took place between two people – as "commands", or as a "dialog".

The Virtual World metaphor
The use of the mouse to point at and manipulate images on the screen strains the conduit metaphor. Although popularly described as *more* metaphorical than the conversational metaphors that it replaced, the characterisation of direct manipulation interfaces as a "desktop" has many well-known deficiencies. The idea of the desktop as a simulated *virtual world* is undermined by the fact that objects in the GUI cannot be grasped, but only pointed at. The virtual "world" represented on the screen is so impoverished by comparison to the physical world that it ought better to be described as a diagram, in which the rich physical properties of a real desk and office have been reduced to no more than structural relations, providing a notation that describes office behaviours rather than simulating them.

Metaphors of Autonomy and Sentience
In the case of ubiquitous computing technologies, it is no longer so clear that the device acts solely on the basis of explicit communication from the user, because location and other sensing interfaces can influence system state and

actions in ways not explicitly prompted by the user. In these circumstances, the implicit conduit metaphor of traditional human-computer interaction breaks down more seriously than in the "desktop" GUI. Instead, it has been necessary to construct new descriptions of principles for designing system behaviour. One option has been to characterise the designed system as being an autonomous actor with its own understanding and objectives, drawing on the metaphor of "sentient computing" to describe systems that are able to act on their own account, rather than depending on human agency via some conduit [1,15,16].

The Metaphor of Context

Metaphors of autonomy and sentience seem potentially to exclude human agency. An alternative is to focus on the way that a human might describe required parameters of behaviour by a system that is partially autonomous. This communication might still be regarded as conduit-based, but the additional inferences made by the system depend on an understanding of the "context" in which communication is grounded [9,10]. Context, too, can be applied as a design metaphor that draws attention to social agency, embodiment, and other pressing concerns in the philosophy of computing .

Solid Diagrams: a Notational Metaphor

These implicit metaphors involve varying perceptions of the degree of agency and collaboration shared between user and system. However, most focus on the immediate effect of communication to provoke system action or change of state. A further alternative is for the user to specify the structure of the required behaviour, rather than directly specifying the required actions. To a computer scientist, this might be considered as a programming or scripting language, but we prefer to focus on a metaphor in which tangible and manipulable objects might be considered as a solid diagram. This appeals to diagrammatic notation as a generic cognitive tool. It allows us to draw on traditions of engagement with social and physical contexts that extend well beyond the technological, for example in the performing arts [3,8] and professional design practice [11].

TECHNICAL PARADIGMS OF UBICOMP

Interactive demonstrators of ubiquitous computing technology tend to associate particular technical advances with different implicit metaphors of interaction, to an extent that might be considered as new technoscience paradigms.

The conversational paradigm in ubiquitous computing research provides an opportunity to demonstrate natural language processing technology, and offers an ideal of interaction in which the user holds a conversation with the system in order to come to some agreement on required system behaviour. This clearly owes much to the conduit metaphor, and expects to account for context, if at all by means of social agency on the part of the system [7], if it is to participate in a genuine conversation.

The inference paradigm in ubiquitous computing research provides an opportunity to demonstrate sensing and machine learning technology, by observing physical context, potentially including both inanimate objects and the posture of the user's body, and on the basis of this to infer and anticipate the user's requirements. It draws mainly on the metaphors of context and of sentience. It is characteristic of the great deal of pervasive and ubiquitous computing research that uses wireless sensors and machine vision to monitor and structure the lives of the users. Structure is emergent, not notated.

The solid diagram metaphor that we propose applies a notational paradigm that explicitly recognises how the ability of systems to sense fully the social and physical context is impoverished. It describes only basic structural relations rather than attempting to simulate rich human understanding, interpreting the world not fully, but diagrammatically. Examples of structural diagrammatic relations in the world include registering which objects are touching which other object, and which objects are contained within particular spatial regions.

From our perspective, the solid diagram metaphor allows users to treat the physical world as if it were a notation, arranging, touching, moving and reconfiguring objects (including their own bodies) to create abstract data structures that specify system behaviour It offers the possibility of sophisticated user control, without presuming any degree of intelligence or agency in the system. Although possibly disappointing to technology researchers seeking applications of AI techniques, we believe that analysis of ubiquitous computing interfaces in terms of solid diagrams offers a more human-centric approach to interaction design. In future, this approach might be further augmented with conversational or inference-based behaviours, but we suggest that the solid diagram metaphor provides the most appropriate starting point for interactive system design of ubicomp technologies.

Examples of Solid Diagrams in Ubiquitous Computing

If we analyse proposals for new ubiquitous computing interfaces, we can observe ways in which augmenting or enhancing objects in the physical world draws attention to possible diagrammatic interpretations of those objects, providing familiar physical objects with new abstract semantics. To take one recent example, the "Clutter Bowl" [19] takes an otherwise unremarkable household object – the bowl – and draws attention to its abstract semantics by providing a computational interpretation of the inside versus outside of the bowl, by drawing attention to the rim of the bowl as a continuum along which one can specify motion by placing a marker (one's finger), and by distinguishing between the bottom of the bowl and its inner sides, (as collection surface and display surface respectively). The physical bowl itself will have many other contextual meanings to potential users (fragility, colour, sentimental association), but its augmentation has led to its being diagrammatised.

An earlier example in our own work was the Media Cubes system for specifying network interaction between home appliances [6]. By interpreting proximity and contact between remote controls and appliances, we drew attention to the physical semantics of the remote control as a (portable, graspable) object, and interpreted these properties diagrammatically so that they constitute a tangible programming language for scripting future abstract behaviours and interactions.

It is clear that both of these systems might benefit from the use of conversational or inference technologies to further enhance the functionality or user experience. Nevertheless, the user experience of these particular objects belongs primarily to the class that we have defined as solid diagrams, insofar as they provide support for the user to inspect, specify and modify abstract relations through direct interaction with the physical world.

SOLID DIAGRAMS AS A DESIGN PERSPECTIVE FOR UBIQUITOUS COMPUTING

We have in the past described the use of this solid diagram metaphor as a generative approach, supporting ideational 3-D sketching of alternate physical forms for novel personal and ubiquitous information devices [4]. However, in the current paper, we adopt a more ambitious goal, considering the ways in which diagrammatic understanding of the physical world allows designers (and eventually users) to interpret any collection of objects, spaces and surfaces as a notational system, and thus apply the insights from prior design research analyses of abstract notation systems to interaction with any pervasive computing context.

Cognitive Dimensions Analysis

Our starting point for this work is the Cognitive Dimensions of Notations framework, the leading technique for analysis of usability in notational systems such as programming languages and other information artifacts [5,14,13]. We found that the framework also offered a valuable means of analysing programming systems that are based on physical elements, such as Media Cubes, the well-known early systems Slot Machine and AlgoBlocks, and other more recent tangible programming languages [12]. In the current paper, we now extend this perspective beyond the domain of programming, to consider all forms of abstract interaction that might be mediated by physical objects.

Cognitive Dimensions is constructed as an analytic vocabulary that helps designers understand the space of possible design alternatives, and the consequences of their decisions, for the eventual usability of a notational system. The conception of a notational system includes firstly some set of notational conventions (such as a diagram style, graphical syntax, or text format), and secondly an environment (such as an editor) for manipulating that notation. Notations can represent different kinds of information structure, including cognitive information structures held in the user's mind, and digital information structures held in computer memory. Each dimension describes a cognitively relevant property of the system that is likely to make it more or less useful for certain purposes. Overall, the dimensions describe a set of trade-offs, recognising that there is no perfect design that will be best for all users, and that designers should aspire to recognise and articulate the trade-offs they are making, rather than naively imagine that they are creating a perfect system. This design stance is obviously just as relevant to tangible interfaces and ubiquitous computing systems as it is to graphical notations (and even to conventional paper notations such as musical notation or design drawings).

A popular example of a generic cognitive dimension is the dimension of "viscosity" – defined as the extent to which a system resists small changes in the underlying information structure. Expert users who use a notation to explore many alternative abstractions (such as mathematicians or computer scientists) generally find viscous notations annoying and cumbersome. People who only read from a notation, or who create an information structure once and never change it (such as using pen on paper), are not likely to find viscosity a problem. Providing additional cues to the user to help them recognise the underlying structure (thereby reducing the dimension of "hidden dependencies") typically results in increased viscosity. Computer science researchers are often prepared to tolerate hidden dependencies in their own tools in order to reduce viscosity, for example in untyped programming languages such as Lisp. However this tradeoff is inappropriate for people who do not often manipulate abstract structures, with the consequence that computer scientists are surprised when other people do not like working with their tools. This example gives the flavour of cognitive dimensions analysis – it involves understanding the information structures inherent in users' activity, and recognising and choosing suitable trade-offs for those users and activities.

Cognitive Dimensions of the Physical World

We propose that this analytic approach be extended from on-screen and paper notations, to our view of the physical world as a solid diagram, a diagram which can be interpreted by ubiquitous computing systems as corresponding to some underlying information structures. Clearly there are some physical objects and combinations of objects that are more "viscous" than others, in the sense that they are hard to reconfigure. Furthermore, ubiquitous computing systems might also sense diagrammatic information from a particular configuration of objects that is not immediately visible to someone looking at those objects, thus leading to hidden dependencies. These two dimensions seem to trade off against each other in the physical world, much as they do in the virtual graphical domain. If a better indication of dependency between the information entities represented by two physical objects is provided (perhaps by gluing them together, or tying a string around them), the system then becomes more viscous because it is harder to modify that relationship. We might not always choose these particular terms, however, because while "viscosity" is an evocative metaphor when considering programming languages, it can be confusing when applied to the physical world (especially if there are liquids involved!). We have therefore proposed a complementary set of terms as tangible correlates of the cognitive dimensions [12], for use in this particular domain. The new vocabulary is intended to be used in conjunction with a traditional cognitive dimensions analysis as a means of addressing the salient dimensions of tangible interaction.

IMPLICATIONS FOR DESIGN

In this paper, we have proposed a generic approach to analysing the usability of tangible user interfaces. We apply the notational analysis paradigm of earlier HCI research, extending it to the physical world by application of a solid diagram metaphor. This is complementary to other analytic approaches such as the TAC paradigm [18], which provide the means of analysing possible correspondences between the physical and virtual parts of an interface. By focusing on the concept of notation as a way of interpreting the interactive possibilities of structure in the physical world, we have emphasised the way that interaction designers can anticipate the cognitive demands on the user when defining and manipulating physical objects as information structures.

ACKNOWLEDGMENTS

This research was supported by Boeing Corporation.

REFERENCES

1. Addlesee, M. Curwen, R. Hodges, S., Newman, J., Steggles, P., Ward, A. and Hopper, A. (2001). Implementing a sentient computing system. IEEE Computer, 34(8), 50-56.
2. Blackwell, A.F. (2006). The reification of metaphor as a design tool. ACM Transactions on Computer-Human Interaction (TOCHI), 13(4), 490-530.
3. Blackwell, A.F. and Collins, N. (2005). The programming language as a musical instrument. In Proceedings of PPIG 2005, pp. 120-130.
4. Blackwell, A.F., Edge, D., Dubuc, L.M., Rode, J.A., Stringer, M. and Toye, E.F. (2005). Using solid diagrams for tangible interface prototyping. IEEE Pervasive Computing, Oct-Dec, 18-21.
5. Blackwell, A.F. and Green, T.R.G. (2003). Notational systems - the Cognitive Dimensions of Notations framework. In J.M. Carroll (Ed.) HCI Models, Theories and Frameworks: Toward a multidisciplinary science. San Francisco: Morgan Kaufmann, 103-134.
6. Blackwell, A.F. and Hague, R. (2001). AutoHAN: An Architecture for Programming the Home. In Proceedings of the IEEE Symposia on Human-Centric Computing Languages and Environments, pp. 150-157.
7. Collins, H. and Kusch, M. (1998). The shape of actions: What humans and machines can do. Cambridge, MA: MIT Press
8. Delahunta, S., McGregor, W. and Blackwell, A.F. (2004). Transactables. Performance Research 9(2), 67-72.
9. Dourish, P. (2001). Where the Action Is: The Foundations of Embodied Interaction. Cambridge: MIT Press.
10. Dourish, P. (2004). What We Talk About When We Talk About Context. Personal and Ubiquitous Computing, 8(1), 19-30
11. Eckert, C.M., Blackwell, A.F., Stacey, M.K. and Earl, C.F. (2004). Sketching across design domains. In Proceedings of the Third International Conference on Visual and Spatial Reasoning in Design (VR'04).
12. Edge, D. and Blackwell, A.F. (2006). Correlates of the cognitive dimensions for tangible user interface. Journal of Visual Languages and Computing, 17(4), 366-394.
13. Green T.R.G. and Petre, M. (1996). Usability analysis of visual programming environments: a 'cognitive dimensions' framework, Journal of Visual Languages and Computing 7, 131–174.
14. Green, T.R.G. (1989). Cognitive dimensions of notations. In A. Sutcliffe and I. Macaulay, Editors, People and Computers V, Cambridge: Cambridge University Press.
15. Hopper, A. (2000). Sentient Computing. Philosophical Transactions of the Royal Society, 358(1), 2349-2358.
16. Pascoe, J., Ryan, N.S. & Brown, P.J. (1998). Context aware: the dawn of sentient computing? GPS World, 9(9), 22-30.
17. Reddy, M.J. (1993). The conduit metaphor: A case of frame conflict in our language about language. In A. Ortony (Ed.) Metaphor and Thought (2nd Ed). Cambridge University Press, 164-201.
18. Shaer, O., Leland, N., Calvillo-Gamez, E.H., and Jacob, R.J.K. (2004). The TAC paradigm: specifying tangible user interfaces. Personal and Ubiquitous Computing 8(5), 359-369.
19. Taylor A.S., Swan, L., Izadi, S. and Harper, R. (submitted). Containing Family Clutter, Hawaii International Conference on System Sciences, HICSS 07