Metaphor

1 Initial Comments

1.1 The Role of Metaphor

Metaphor (and metonym and synechdoche) plays an important role in guiding expectation in user interfaces. Sometimes this is done through mapping between tangible real-world entities and data organisation, the most obvious example being the desktop metaphor used on PCs (files, folders etc). Metaphor is also applied to presentation of scalar values: traffic light colors (red, yellow, green) are used to indicate line quality in communications links, scrollbars are used to indicate relative position in a document. In fact, on investigation, most user interfaces will yield up a large number of metaphors in their operation. There is no guarantee that a user interface is consistent in its use of metaphor, especially if the interface supports multiple applications from different suppliers. A simple example here would be a web browser and a word processor. Both applications present “pages” to the user, but (generally) web “pages” have no fixed geometry, and word processor pages (generally) are the geometry of the printed page. Consequently, zooming and panning in on a web page may behave quite differently to doing the same thing in a word processor.

1.2 The Challenge to Assistive Technology

That so much information is communicated between user and application (potentially in both directions) using metaphor, presents significant challenges to assistive technology. Common Assistive Technology (AT), such as screen readers and zooming tools rely on a process of automated transliteration between the default form of user presentation provided by the product’s user interface, and that suited to the AT’s client users. How successful this process is, is largely dependent on how well the AT interprets the default presentation and catches content expressed by metaphor in addition to any explicitly presented content. To quote Barbosa [Barbosa-2000], "The appropriateness and sophistication of interpretations is directly proportional to the expressiveness of the underlying domain models". On a typical PC and PDA, this means that the AT requires access to a domain model of the platform itself (e.g. Microsoft PocketPC), and the domain model of the application (e.g. a web browser, a notepad application, a timetable application). In each case, AT needs access to the metaphors in the domain model used to express content. So, how does a typical screen-reader do this? In practical terms it can’t. It can, in theory at least, be pre-loaded with a domain model for a particular version of the platform and common applications, but it has no way to access other domain models. Consequently, screen-readers (e.g. Jaws) do not transliterate much of the content presented as metaphor beyond well-known metaphors in the Microsoft Windows platform. Metaphors associated with representation of scalar values, and those relying on relative geometrical positioning of visual elements are largely lost.

1.3 The Accessibility Layer

Since it is impractical to develop generic, self-contained, transliteration-based AT that can successfully extract content from metaphor, some transparency is required between application and AT, exposing the underlying domain model and its content (both explicit and implicit content) and the mapping between content and any presentation metaphors used. To some extent, this is what the accessibility layer in Microsoft Windows, Apple OSX, and the Java programming language, attempt. In each case however, what is exposed are instances of presentation elements, and potentially some alternative content for those elements, it is not a model of interface itself, or mapping between content and presentation. This is left to AT to hard-code into its own application, and is, by definition, version specific to the platform and each supported application. However, whilst limited in this way, even this level of exposure to AT allows existing transliterative AT to be at least partly successful.

1.4 Adapt/Extend/Replace

It's important to note that it does need to be content + metaphor that is exposed. If you consider the task given to AT: it must adapt/extend/replace default presentation elements and input methods. Unless the entire user interface is to be replaced, AT needs to seamlessly integrate its changes with the default interface, and to do that it must understand what elements of the interface represent what content. Sometimes this is additive, sometimes subtractive in nature (e.g. it may need to take control of sound hardware, overriding the application's normal use, and this may mean moving or changing the applications standard sound interface).

1.5 Expressing metaphor in UML

In relation to my work as it stands, I've developed an Executable UML domain chart which has Metaphor as an independently modelled domain mapping presentational elements to an arbitrary (defined by me) hierarchical classification of presentation metaphor. The domain chart counterparts objects within the Metaphor domain to information content objects in the Abstract UI domain, and to objects representing properties in the "senses" domains (Sight, Hearing, Haptic). Starting from the Capability domain, this means, we can map from user capability to sense properties, and from sense properties to usable metaphors for the given capability set. Starting from the Abstract UI domain, we can map content to appropriate metaphors (properly populated this means there should be typically multiple presentation metaphors available that are appropriate to any given content category). This leaves us with two sets of metaphors: those appropriate to the user's capability, and those appropriate to the content to present. The intersection of these two sets represents the appropriate metaphors to use in rendering the user interface. What the model does not yet do, is tell you what to do if the intersection contains more than one element; clearly there needs to be a mapping to the User Preference domain to help resolve the issue. Note:

I’ve simplified this description a little: clearly there is also a mapping to the capabilities of the device.

1.6 Metaphor as a Design Pattern

Noble [Noble-2002] suggest that design pattern are themselves examples of metaphors or metonyms, and goes on to classify some of the standard design patterns according to the category of metaphor they support.

In the same way, it’s possible to have a design pattern (though “analysis pattern” would be a rather better phrase) that represents the general case of metaphor. As always with Executable UML, design patterns tend to vanish into models of problem domains. This is because Executable UML is not an elaborative design approach, but rather a translative one, with design patterns limited in large measure to the translation technology inherent in the method. For this reason, it’s not so surprising that Metaphor is exposed in my existing work as a Metaphor domain rather than as a design pattern, however, I can see good reason for it to exist as a design pattern if a Booch/Rational Systems style elaborative development approach is taken.

1.7 Ways of Populating UML Models

From my own initial experiments at populating my Executable UML model with instance data, I can imagine how large and cumbersome manual maintenance of such a system would be. Clearly, to make such a large and complex domain chart manageable, it is necessary to provide developer tools. Barbosa’s [Barbosa-2000] work in this area is encouraging, using abductive reasoning to provide an interactive Q&A tool to allow extension of user interface metaphors and metonyms. Whilst Barbosa’s goal is End-User Programming, his work has use both to the UI developer and to the AT designer. To make his system work, he requires a model of the problem domain and its terminology, which maps neatly into my existing work with my Executable UML model which includes a model of Metaphor (though not currently metonym).

1.8 Quantitative and Qualitative Measurement

One of the problems when dealing with developing and managing UI metaphors is measurement. How do you know that a given metaphor is properly supported and implemented? How do you know if an implemented metaphor is effective for a given user group?

For mobile devices, Brewster and his associates [Pirhonen-2002] have developed usability testing strategies that may go some way towards answering the second of these questions. The usability question is considered both in terms of quantitative results attained by instrumentation of the UI under test, and in terms of qualitative measures through videoing use of the interface. Whilst their test strategies and Use-Cases are of direct interest, they don’t consider the effect of disability on their data collection, or on how you demonstrate validity when comparing results of users of different capability sets. The answer to that may lie in work on validity in terms of evaluation of on-line assessment tools in education (possibly work by Cetis or Techdis?)

Testing for the first question: validating implementation of a metaphor is perhaps something I may get partly "for free" in my Executable UML approach to modelling my user interface. Executable UML is a constrained use of UML notation design to produce precise descriptions of behaviour. To enable this, Executable UML has a very precise model of when data is valid, and uses a counterparting scheme between subject domains that make it easy to automatically validate a populated model. Such a validated model may not fully implement a metaphor, but at least the metaphor, if expressed in the Metaphor domain, must at least be correctly counterparted to equivalent objects in related domains. Because Executable UML models are stylized directed graphs, graph-theory lemmas should also allow for static analysis of metaphor implementations, though this is getting a bit outside of more core interest (and probably a PhD in its own right).

Even at this stage, work on selecting appropriate test criteria is important. Before any major builds of software are begun, it will be necessary to decide on what metrics will be collected and how, in order to develop an appropriate platform for implementing my Executable UML model.

2 Next Steps

a) Review remaining papers in part 4
b) Identify & review referenced papers from papers rated 4+
c) Order Kuhn's paper on formalising metaphors from library
d) Look at Cetis' and Techdis' work on validity
e) Look carefully at Brewster's test harnesses and metric collection techniques.

3 Considered Papers

Barbosa-2000	Extending software through metaphors and *metonymies, IUI 2000 New Orleans LA USA*, 2000
	Metaphor/p13-barbosa.pdf
	Keywords: Metaphor, Metonym, Abductive Reasoning, Extensible UI, End User Programming (EUP), Negotiated Meaning, Patterns of Use, Syntagmatic Expression, Computation of Analogies, Polysemy, Predicate Calculus Quotes: "The appropriateness and sophistication of interpretations is directly proportional to the expressiveness of the underlying domain models." Summary of Interest: Paper is weighted towards end user programming, replacing macros etc. with abductive reasoning allowing users to extend the application by analogy. Very interesting for me in that it shows two things: how difficult it is to infer meaning from grammar without a very strong domain model, and a way to express equivalence/metaphor to populate a model of UI e.g. a scroll bar is like a moving sound source, or color spectrum is like font-size range. The first point demonstrates the limitations of transliteration in AT, and the second point suggests a model to express metaphor/metonym in UML. Relevance Rating: 5/5

Fass-1988	Metonymy and metaphor: what's the difference?, *Proceedings of the 12th conference on Computational linguistics - Volume 1*, 1988
	metaphor/p177-fass.pdf
	Keywords: Metaphor/Analogy, Metonymy, Synechdoche, Collative Semantics, Anomalies, Semantic Relations, Transliteration Summary of interest: A paper that looks at how ideas are expressed, and presents an application (Meta5) that semantically analyses prose to ascertain meaning. Compare with transliteration techniques used in assistive technology. Also compare with Shlaer-Mellor and their semantic approach to modelling. Relevance rating: 5/5

Goldstein-2003	Providing proper affordances when transferring source metaphors from information appliances to a 3G mobile multipurpose handset, *Personal & Ubiquitous Computing 7: 372-380*, 2003
	metaphor/30070372.pdf
	Keywords: 3G, Phone Metaphor, Hypermedia, Affordance, Information Appliance, Source Metaphor, Usability Metric, The Invisible Computer by Donald Norman, Mental Model, Convergence Summary of Interest:

	Paper considering the difficulty of convergence of multiple single-aspect devices onto a multiple-aspect device, in this case a 3G phone. Looks at the difficulties in supporting the original source metaphors in a consistent way on a single device. Usability tests based around similar Use-Cases that I plan to use, and similar usability metrics I'm considering. Relevance rating: 5/5

Kuhn-1991	Specifying Metaphors Algebraically, *SIGCHI Bulletin 1991*
	metaphor/p58-kuhn.pdf
	Keywords: Formal theory of Metaphor, Cognitive Engineering, Morphism, Idealized Cognitive Model. Summary of Interest: Paper just an introduction to the idea o formalizing the theory of metaphor in HCI work. A good reference point for my mapping between metaphor, design space, and user capability. Relevance Rating: 5/5

Madsen-2000	Magic by Metaphors, *DARE 2000, April 2000, Elsinore, Denmark*
	metaphor/p167-madsen.pdf
	Keywords: magic, Metaphor, Epiphor, Diaphor, Computer as Theatre. Summary of Interest: Position paper the use and influence of metaphor on UI design. Compares and contrasts example of epiphor and diaphor metaphors, to suggest that too literal an epiphor can limit understanding of the UI. Also references the concept of computer as theatre and associated metaphors. Relevance Rating: 3/5

Noble-2002	Metaphor and metonymy in object-oriented design patterns, *Australian Computer Science Communications, Proceedings of the twenty-fifth Australasian conference on Computer science - Volume 4 CRPITS '02*, 2002
	metaphor/p187-noble.pdf
	Keywords: Design Pattern, Metaphor, Metonymy, Contiguity, Reification, HCI & Metaphor, Semiotics Summary of Interest: Paper suggesting that design patterns are examples of metaphor, or metonym. It takes a very semiotic view of UML class diagrams. Useful in supporting the idea of expressing metaphor as abstract re-usable patterns. Relevance Rating: 4/5

Pirhonen-2002	Gestural and Audio Metaphors as a Means of Control for Mobile Devices, *CHI 2002*, 2002
	P291-pirhonen.pdf
	Keywords: Gesture, Audio Metaphor, Mobile Device, Non-speech Audio, iPAQ 3630, Percentage Preferred Walking Speed (PWS), Mini-stepper as walking simulation. Summary of Interest: Paper is in 2 parts: creation of a gesture-based mp3 player with non-speech audio feedback, and evaluation of the device for usability. The gesture-based interface isn't new, it's in many of Brewster's other papers. The method of evaluation is a repeat of Brewster's work on usability of mobile devices, but with an extension to look at qualitative measures (usually he only does a quantitatve efficiency measure). The paper is useful in that it brings creation of novel audio metaphors together with useful evaluation techniques in one place. It also provides a useful reminder that input AND output metaphors can overlap: if you want to use gestures, you need to layout a touchscreen accordingly - so far I've tended to divorce the two aspects. The quantitative approach to evaluation used in the paper could also form the basis for my own evaluations though with some refinement: I would probably want more metrics collection, and I'd have to think carefully about the effect of user capability of the measurements. Finally it's interesting to note that part of the evaluation criteria come from assistive technology evaluation: PPWS. Relevance Rating: 5/5

Svanaes-2000	In Search of Metaphors for Tangible User Interfaces, *DARE 2000, April 2000, Elsinore, Denmark*, 2000
	metaphor/p121-svanaes.pdf
	Keywords: Tangible User Interface (TUI), Metaphor, Design Space, Magic, Marble Answerphone, Multi-dimensional design spaces Summary of Interest: A paper looking at the development of appropriate metaphors for use in TUI's. Experiments in extracting user-perceived metaphors from simple GUI and TUI elements. Largely based on the concept of multi-dimensional design spaces. Ideas on how to extract metaphors could be directly relevant to how I test the appropriateness of metaphors in my system. Relevance rating: 5/5

Walker-2005	Mappings and Metaphors in Auditory Displays: An Experimental Assessment, ACM Transactions on Applied Perception, Vol. 2, No. 4, October 2005, Pages 407–412, 2005
	metaphor/p407-walker.pdf
	Keywords: Auditory Display, Sonification, Metaphor, Usability Evaluation Summary of Interest: The paper attempt to identify good mappings between value ranges and audio properties. The actual detail is hidden in the appendix. It's interesting in the evaulation technique it applies to measure the effectiveness of the audio metaphors. It matches in with Brewster's usability strategy. This is something I'll need to build in to my own research. Relevance Rating: 4/5 (the actual mappings/results aren't that important)

4 Papers Awaiting Consideration

Howell-2005	Spatial metaphors for a speech-based mobile city guide service, *Personal & Ubiquitous Computing (2005) 9: 32-45*, 2005
	metaphor/779_2004_Article_271.pdf

Coschurba-2001	Metaphors and Context-Aware Information Access, *Personal & Ubiquitous Computing (2001) 5:16-19*, 2001
	metaphor/10050016.pdf

Weiner-1984	A Knowledge Representation Approach to Understanding Metaphors, *Computational Linguistics, Volume 10, Number 1, January-March 1984*, 1984
	metaphor/p1-weiner.pdf

Marcus-1994	Managing Metaphors for Advanced User Interfaces, *AVI 94-6/94 Bari Italy*, 1994
	metaphor/p12-marcus.pdf

Catarci-1995	Visual Metaphors for Interacting with Databases*, SIGCHI Bulletin Volume 27, Number 2, April 1995*, 1995
	metaphor/p-15-catarci.pdf

Carbonell-1980	Metaphor - A key to Extensible Semantic Analysis*, Proceedings of the 18th annual meeting on Association for Computational Linguistics*, 1980
	metaphor/p17-carbonell.pdf

Demasco-1994	The Application of Spatialization and Spatial metaphor to Augmentative and Alternative Communication, *Assets 94*, 1994
	metaphor/p31-demasco.pdf

Yousef- 2001	Assessment of Metaphor Efficacy in Unser Interfaces for the Elderly: A Tentative Model for Enhancing Accessibility, *WUAUC'01*, 2001
	metaphor/

Marcus-1998	Metaphor Design for User Interfaces, *CHI 98*, 1998
	metaphor/p129-marcus.pdf

Shafrir-1994	Visual Access to Hyper-Information: Using Multiple Metaphors with Graphic Affordances, *CHI 94* ,1994
	metaphor/p142-shafrir.pdf

Powell-2003	Composite Metaphor, Games and Interface, *Proceedings of the Second Australasian Conference on Interactive Entertainment, Sydney*, 2003 (probably....)
	metaphor/p159-powell.pdf

DeBoek-2004	Multisensory Interaction Metaphors With Haptics and Proprioception in Virtual Environments, *NordiCHI'04*, 2004
	metaphor/p189-de_boek.pdf

Duncker-2002	Cross-Cultural Usability of the Library Metaphor, *JCDL'02*, 2002
	metaphor/p223-duncker.pdf

Savidis-1995	Building non-visual interaction through the development of the Rooms metaphor, *CHI'95*, 1995
	metaphor/p224-savidis.pdf

Väänänen-1994	User Interfaces for Hypermedia: How to Find Good Metaphors?, *CHI'94*, 2004
	metaphor/p263-vaananen.pdf

Gaver-1995	Oh What a Tangled Web We Weave: Metaphor an mapping in Graphical Interfaces, *CHI'95*, 1995
	metaphor/p270-gaver.pdf

Kuhn-1996	Spatialization: Spatialization Metaphors for User Interfaces, *CHI'96*, 1996
	metaphor/p346-kuhn.pdf

Marx-1994	Using Metaphor Effectively in User Interface Design, *CHI'94*, 1994
	metaphor/p379-marx.pdf