Towards a Design Methodology for Virtual Environments

Clive Fencott

Virtual Reality Applications Research Centre
University of Teesside
p.c.fencott@tees.ac.uk

Abstract

Virtual Reality (VR) is currently the subject of much academic research and virtual environments, particularly in the form of 3D computer games, are the subject of much commercial activity. However, the development process for VEs is not well documented or fully researched. In this paper the available literature relating to VE design is briefly reviewed prior to the presentation of a possible design methodology for VEs. The paper then goes on to discuss the weak points in the methodology and problems with inter-relating the various stages. Finally, future areas of research are identified.

1 Introduction

Designing effective virtual environments (VEs) poses a number of problems, not the least of which being that the very term design itself has to be interpreted in two quite distinct ways in this context. First of all there is the aesthetic notion of designing something to create the desired perceptual responses and secondly there is the engineering notion of design as the creation of plans and models from which to test and build the desired artifact. Both forms of design are intrinsic to the process of designing effective VEs. It is the tension between the two notions of design that is at the heart of the arguments developed in this paper with a view to establishing an appropriate design methodology for VE construction. In the computer games field Church calls for a set of formal, abstract design tools to act as a common language across the whole range of games genres [5]. Such a common language for VEs in general would be of great benefit for both researchers and developers.

The particular notion of user centred design of VEs is also not straightforward. The nature of the interface and its role in human centred design is obviously of interest although much of the true interface between user and VE is embedded in the content and effectively separated from the interface technology itself, i.e. joystick, keyboard etc. The distinction is clear in computer games where the user might be physically pressing a key, or keys, whilst moving the mouse but perceptually aiming and firing a long bow. So the VE designer has to deal with both the traditional domain of HCI as well as the realm of the content creator. In computer games the technological interface does not receive a great deal of attention and in fact the game player is very often able to redefine the

mapping between input technology and the event model programmed into the game. Johnston [12] points out that a number of traditional HCI measures simply do not apply to many VR genres for reasons related to the purpose of particular VEs. See also [18].

One of the principle differences, although there are others, between WIMP interfaces and VEs is that the screen based objects of interaction of the former, buttons, scroll bars, etc., are perceptually static in that they constitute a small number of interaction objects with preset 2D viewpoints and well understood behaviors. Whereas, in VEs such objects may be perceived from a number of different viewpoints and may or may not have the behaviors associated with the real world - assuming such a correspondence exists. Informing the user, subtly and without adversely affecting their sense of presence, as to what can be interacted with, how, and to what local and global purpose is one of the major design skills for VEs.

In section 2 some of the relevant background research and theory is overviewed along with related design techniques. In section 3 a potential process model for VE design is introduced and suggestions made for a design methodology which might support it. Section 4 discusses the gaps and limitations within the methodology and identifies some areas for future research. Section 5 presents the concluding remarks.

Before proceeding, it should also be pointed out that this paper is very much a workshop paper and is intended to be a focus for discussion and not a finished piece of work.

2 Background Theory and Design Techniques

There are a number of active and relevant areas of research and development that can inform our development of VEs though, interestingly, there seems to be little cross-fertilisation at least in terms of citations. Due to the newness and dynamic nature of the field there it is also not always clear as to what constitutes a model from which to conduct research into the nature of VEs and what constitutes an actual technique to help VE designers in their task.

Kaur [14] rightly points out that VEs draw as much on hypertext systems as on traditional HCI and interaction theory. Thus at least some of the reference points we draw upon for VE design should be concerned with the design and theory of hypertext documents and other related media for which there is a small but established literature. Much of this work is located in what we might call media studies and is concerned with aesthetics. Murray [20] discusses interactive novels from the authoring point of view and uses the notion of the labyrinth as the design metaphor. She further notes that the comprehensible labyrinth lies somewhere between the single path maze and the rhizome or entangled web. In particular she documents such dramatic structures as choice points as major items of content. Aarseth [1] defines the concept of ergodic literature to survey the field of interactive media and locates VEs within it. The term ergodic is defined from the Greek words for work and path to emphasise the particular form of user interaction required by such media. Further sources of information in this respect are the various accounts given

by artists of the creation process [e.g.19]. Char Davies has given a number of particularly lucid accounts of the aesthetic theories underlying her works [6].

Presence, *the perceptual illusion of non mediation* [16], is concerned with the causes of the particular mental state of users of a VE being transported from the real world around them to the virtual world sensed through the input technologies. Presence thus lends support for the kinds of things we need to model and the way we need to model them in order to facilitate believable worlds. Presence is concerned with the direct relationship between mental state and both the embodying interface and the mediated content.

Fencott [10,11] draws on both the media studies and presence work to define a model of VE content, a mapping of perceptual opportunities, which focuses on the aesthetic design of the perceptual experiences over time which users are intended to accumulate.

Interaction theory is concerned with lower level structural models of the mental processes involved when users interact with VE content [14,13]. It can thus inform us as to why engrossing content was so involving and would seem to lend support to a deeper structural analysis of VE content concerned with what Ellis refers to as *the effective communication of causal interaction* [7]. We can thus perhaps see interaction theory as one of the bases for the engineering approach we have already identified [22,4,26].

In VE design we can observe the basic tension between aesthetic and engineering design in terms of perception and structure. Unlike 3D animation there is of course the added requirement for acceptable performance. Software engineering practice can indeed help us with structural modeling but cannot help us with perceptual modeling. Neither can the two proceed sequentially or independently but must proceed concurrently and interactively. We might therefore rewrite the second sentence of the opening paragraph of the call for papers for this workshop in the following way. As VE technology matures and becomes adopted in a range of applications there is, however, a need to better understand how software-engineering practice might be usefully accommodated in the design of applications using this new technology. Work in methods integration for structured and formal methods [8,9] suggest that however well intentioned and appropriate novel techniques may be they will not be adopted by a developer community unless they do indeed integrate easily with current practice. In attempting to introduce software engineering practice into VE design we must there proceed very carefully.

3 A Design Methodology for VEs

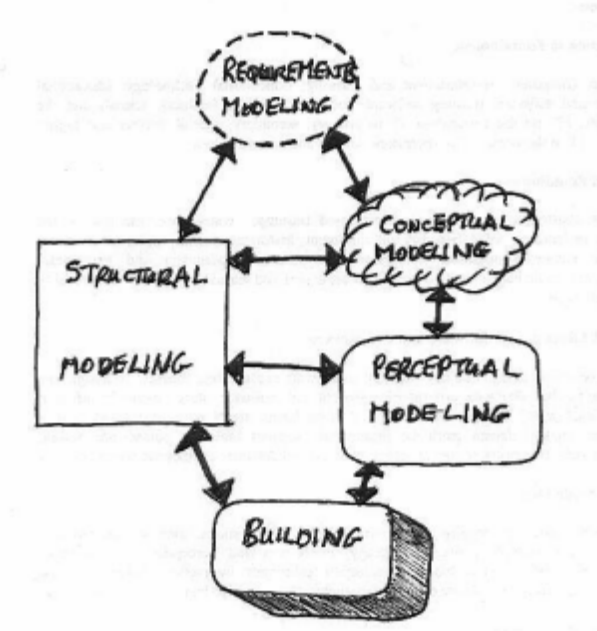
As part of a study of VE design practice Kaur [14] constructs the following outline VE design methodology:

1. Requirements specification:
2. Gathering of reference material from real world objects;
3. Structuring the graphical model and, sometimes, dividing it between designers;
4. Building objects and positioning them in the VE;

5. Enhancing the environment with texture, lighting, sound and interaction, and optimising the environment.

She also notes that there might also be a narrative design component missing here but this is probably because of the small scale of the VEs in the study. Certainly the narrative aspects of 3D games design are considered as soon as the principle subject and genre are established. Computer games are almost certainly the major examples of VEs large enough to benefit from software engineering practice.

With these arguments in mind a prototype design methodology for VEs is presented which juxtaposes the structural and perceptual modeling and attempts to empathise with current practice. The methodology is also based on practical experience gained in building a variety of desktop VEs, and in particular a virtual tourism project, as well as teaching VE design to several hundred undergraduate and masters students over a number of years. The diagram below represents an informal process model for the design methodology.



Requirements modeling equates to point 1 in Kaur's methodology above and parallels very closely the software engineering concept. One of the chief requirements is that purpose should be clearly established here.

Conceptual modeling equates to point 2 in Kaur's methodology and is effectively the background research activity common to many design project but in particular those with an aesthetic component. It is the gathering of materials, taking of photographs, sketches, sound and video recordings, etc. It might also include the construction of mood boards as well as potential storyboards etc. This is where the VE builder or builders get to know the world they have to build. Note that the world to be built might have no real world counterpart, which will of course impact on the kinds of activities that might be undertaken here. The artists' accounts and the techniques employed by animators etc. are

sources of applicable techniques [e.g.19]. An important outcome of this stage will be a choice of genre, to best achieve the purpose established at the requirements stage, with which to inform the nature of the meta-narrative structure to be developed in the perceptual modeling phase.

Perceptual modeling is the act of building up a model of the nature of the perceptual opportunities and their inter-relationships. It equates very roughly to 5 in Kaur's methodology. It is of course modeling the intended users' experience of the VE. In [10,11] perceptual maps are used to build up a meta-narrative structure, analogous to the comprehensible labyrinth of Murray [20], of perceptual opportunities which are categorised according to the role they play in the planned scheme of possible user activity. Perceptual opportunities deal not only with conscious experience - derived from *the specifically designed infidelities* of [25] - but also with unconscious experience, sureties, which deliver belief in the VE - perceptual realism in [16] - irrespective of any real world counterpart. The existence and importance of unconscious experience is identified and discussed in [23,3]. Perceptual opportunities are specifically defined on the notion of presence [16] but can be extended to deal with co-presence [21] as well.

Structural modeling. point 3 in Kaur's methodology, covers a variety of activities, which relate to the underlying realisation of the VE which the delivery platform, uses to construct the run-time sensory stimuli. Structural modeling would seem to commence alongside conceptual modeling and to run on alongside perceptual modeling. It starts with decisions on scale, the construction of plans and diagrams. It draws on the techniques from interaction theory to further decompose the perceptual map constructed in the perceptual modeling phase. The conclusion of the structural modeling phase will result in a scene graph diagram that lays out the code structure of the VE and its programmed behavioral components. In terms of software engineering practice the Unified Modeling Language (UML), for example could be incorporated here [24]. By way of an example, the relationship between UML and VRML is illustrated in [17]. In the early stages *use case diagrams* can be constructed to identify the relationships between user and VE etc. In later stages object models would lay out the actual structure of nodes in the scene graph as well as class diagrams for programmed components.

Building here relates more closely to the software engineering coding phase that should occur after all requirements, specification and design activities have been completed. Building refers to authoring using a WIMP based tool, direct coding of scene graph and program code itself, in VRML and Java/Javascript for example, and using an API such as World Tool Kit.

4 Discussion

The process model for a VE design methodology appears to be reasonably well populated with appropriate techniques but there are problems.

Of particular interest is the inter-relationship between structural and perceptual modeling

and the way in which both can help to deliver a VE, which runs acceptably on the target machine. Whilst the actual activities carried out can be identified in both models their inter-relationships are less clear. The structure of the perceptual map, the patterns of perceptual opportunities, may well be quite different from the optimum structure of the scene graph itself which will have to be organised to best accommodate the computation of interaction, animation and behavior, collision detection, selective lighting, and level of detail control. Further, in order to allow for acceptable performance, frames per second, on the chosen delivery platform the specific levels of fidelity of perceivable components will also have to be carefully controlled. There is thus a need to investigate the relationship between perceptual maps and the scene graphs, which implement them. For instance, in a computer game death match level, the scene graph descriptions of users' avatars within the scene graph and their actual appearance, quite often at a distance and obscured by columns etc. bear little relation except in the user's context based perception.

When are usability issues to be addressed? Due to the wide range of VE applications and genres the purpose and then genre of the VE have to be established before usability can be considered. A similar point is made in [18]. This is because such things as acceptable interface controls, accessibility, and help systems vary according to genre. We can consider different user requirements in this respect between a game such a Ritual Entertainment's 'Sin' and the Hubble Space Telescope VTE [15]. In the former, mastery of the complex keyboard interface with little or no help facility is at the centre of gameplay of the VE. While for the Hubble Space Telescope VTE familiarisation with the EVA activities requires a transparent interface and copious help, in this case, delivered by audio through headphones. However, it would seem quite possible that usability could be addressed as part of the perceptual modeling phase using the perceptual map.

It seems from Kaur's research that coding of VEs takes place throughout the development process both in the form of prototyping and evolutionary development [14]. The author's experience and the perceived wisdom from software engineering are that this is a dangerous practice. This is particularly so if, as is often the case, prototypes are directly subject to evolutionary development. However, prototyping of particular components and behaviors is often necessary in order to establish their practicality on the given delivery platform, for instance. Earlier, it was noted that a design methodology would only gain acceptance in the developer community if it aligned with developer preferences. The relationships between design and prototyping therefore need further consideration and research.

In software engineering verification and validation would directly affect the nature of the activities which populate the kind of process model proposed here. But what exactly does verification and validation mean for VEs. Certainly beta testing is regularly used by the games industry. There are evaluation techniques for VEs but the nature of evaluation is problematic because we have variously the correctness of code in not generating runtime errors, appropriate responses from particular objects to user's actions, and the appropriateness of such responses to meta-narrative structure established in the perceptual map. In the case of validation the perceptual map can play the role that a

specification, formal or semi-formal, plays in software engineering. More work needs to be done to establish qualitative and quantitative techniques to assess the relationship between users' responses to content and the predictions of the perceptual map.

Having considered a process mode for VE development it is clear that tool support is patchy. Tools currently available do not seem to support much, if any, of the design phases of the model. There is thus a need to consider what kinds of tools would best support design and how they could be integrated with the far more developed authoring tools.

5 Conclusions

In this paper a proposed design methodology for VEs has been outlined against a background of known work in the field. The methodology is intended as a focus for further discussion of the whole process of VE design and implementation. In this respect a number of weaknesses in the methodology, as well as some strengths, have been identified. No doubt, the list of weakness is a suitable target for expansion and the methodology itself may radically change or be replaced. However, a number of areas of possible future research have been identified some of which are already under consideration at Teesside.

References

[1] Espen J. Aarseth, "Cybertext: Perspectives on Ergodic Literature", John Hopkins University Press, 1997.

[2] Colin Anderson, "Using Audio to Create More Realistic Environmnets", in proceedings of Virtual Systems and Multimedia '99, University of Abertay Dundee, Dundee, Scotland, 1999.

[3] Susan Blackmore, "The Meme Machine", Oxford University Press, 1999.

[4] David Boyd and Lakshmi Sastry, "Development of the INQUISITIVE Interactive Toolkit: Concept and Realisation", Workshop on User Centred Design and Implementation of Virtual Environments, York, England, 1999.

[5] Doug Church, "Formal Abstract Design Tools", Games Developer Magazine, August 1999.

[6] Char Davies, list of publications at http://www.immersence.com/publications.htm.

[7] Stephen R. Ellis, "Presence of Mind: A Reaction to Thomas Sheridan's "Further Musings on Psychophysics of Presence"", in Presence, Volume 5, No. 2, Spring 1996.

[8] Fencott P C, Galloway A J, Lockyer M A, O'Brien S J and Pearson S, "Formalising the Semantics of Ward/Mellor SA/RT Essential Model using a Process Algebra", in Proceedings of FME '94. Lecture Notes in Computer Science (873), Springer Verlag, 1994.

[9] Fencott P C, Fleming C and Gerrard C "Practical Formal Methods for Process Control Engineers", Proceedings SAFECOMP `92. Zurich October l992, Pergamon Press.

[10] Clive Fencott, "Presence and the Content of Virtual Environments", 2nd International Workshop on Presence, University of Essex, 1999.
[11] Clive Fencott, "Content and Creativity in Virtual Environment Design", in proceedings of Virtual Systems and Multimedia '99, University of Abertay Dundee, Dundee, Scotland, 1999.
[12] Chris Johnston, "On the Problems of Validating Desktop VR", People and Computers XIII: Proceedings of HCT'98, eds. H. Johnson, L. Nigay and C. Roast, Springer Verlag, 1998.
[13] Chris Johnson, "Taking Fun Seriously: Using Cognitive Models to Reason about Interaction with Computer Games", http://www.dcs.gla.ac.uk/~johnson/papers/um99/games.htm.
[14] Kulwinder Kaur, "Designing Virtual Environments for Usability", PhD Thesis, City University, London, 1998.
[15] R. Bowen Loftin and Patrick J. Kenney, "The Use of Virtual Environments for Training the Hubble Space Telescope Flight Team", http://www.vetl.uk/edu/Hubble/virtel.html.
[16] Mathew Lombard and Teresa Ditton, "At the Heart of It All: The Concept of Telepresence", in Journal of Computer Mediated Communication, Volume 3, No 2, September 1997 http://jcmc.huji.ac.il/vol3/issue2/
[17] Patricia McIntosh, course notes on UML/VRML, www.public.asu.edu/~galatin/
[18] William L. Mitchell and Daphne Economou, "Understanding Context and Medium in the Development of Educational Virtual Environments", Workshop on User Centred Design and Implementation of Virtual Environments, York, England, 1999.
[19] Mary Anne Moser, "Immersed in Technology", The MIT Press, 1996.
[20] Janet H. Murray, "Hamlet on the Holodeck: The Future of Narrative in Cyberspace", Free Press, 1997.
[21] Mel Slater, "Co-Presence as an Amplifier of Emotion", 2nd International Workshop on Presence, University of Essex, April 1999, http://www.essex.ac.uk/psychology/tapestries/.
[22] Shamus Smith, David Duke and Peter Wright, "Using the Resources Model in Virtual Environment Design", Workshop on User Centred Design and Implementation of Virtual Environments, York, England, 1999.
[23] L. Spinney, "I Had a Hunch ...", in New Scientist, 5th September, 1998.
[24] UML Version 1.1 Summary, http://www.rational.com/uml/resources/documentation/summary/.
[25] Denise Whitelock, Paul Brna and Simon Holland, "What is the Value of Virtual Reality for Conceptual Learning? Towards a Theoretical Framework", http://www.cbl.leeds.ac.uk/~paul/papers/vrpaper96/VRpaper.html.
[26] James S. Willans and Michael D. Harrison, "Extended Abstract: Requirements for Prototyping the Behaviour of Virtual Environments", Workshop on User Centred Design and Implementation of Virtual Environments, York, England, 1999.