Towards Model-Based development of 3D User Interfaces for Games

Juan Manuel González-Calleros, Jean Vanderdonckt
Université catholique de Louvain
Place des Doyens 1, 1348
Louvain-la-Neuve, BELGIUM
+32 (0)1 047 8349
{juan.m.gonzalez, jean.vanderdonckt}@uclouvain.be

ABSTRACT
Video games industry is the leader of the virtual reality (VR) market. Along as any VR reality application they used 3D User Interfaces (3DUIs) for the interaction with the user. Even that several games nowadays have very impressive realism and innovative representations, this is not the case of 3DUIs. In this position paper we state that 3DUIs for any 3D application, including video games, should be revised in order to reduce the gap between the innovative representation of VR application and their 3DUIs.

Categories and Subject Descriptors
D2.2 [**Software Engineering**]: Design Tools and Techniques – *Modules and interfaces; user interfaces*. D2.m [**Software Engineering**]: Miscellaneous – *Rapid Prototyping; reusable software*. H.1.2 [**Information Systems**]: Models and Principles – *User/Machine Systems*. H5.2 [**Information interfaces and presentation**]: User Interfaces – *Prototyping; user-centered design; user interface management systems (UIMS)*.

General Terms
Design, Experimentation, Human Factors, Verification.

Author Keywords
User Interface, 3D User Interface, 3DUI, Model-Based User Interface Development, Information Systems design.

INTRODUCTION
3DUIs are becoming the primary subject of interest of a growing community of researchers and developers adopting different approaches for specifying and creating 3DUIs. Providing development methods and software support for 3DUIs is a complex problem. Most research and development is focusing on technological issues, as reported in a survey of major publications on 3DUIs [11].

Environments, a workshop at EICS 2010, Berlin, Germany, June

Presented at Design and Engineering of Game-like Virtual and Multimodal DEnG-VE

20, 2010.The research is mainly focusing on how to overcome hardware and software issues [2]. Little or no attention is devoted to the design knowledge that should drive the development life cycle of 3DUIs.

This is also the case of VR for games or game-based VR applications. It is very frequent that 3DUIs are reduced to just imitate a 2DUI. For instance Figure 1, illustrates the representation of a 3DUI using a toolkit CEGUI (www.cegui.org.uk) for the game engine OGRE (www.ogre3d. org). Notice that the components regarding the User Interface remains in 2D. Consistency is one of the most important aspects to be considered for the usability of a system. It looks like this has been forgotten in the design of 3DUIs where there is always a break in the 3D metaphor.

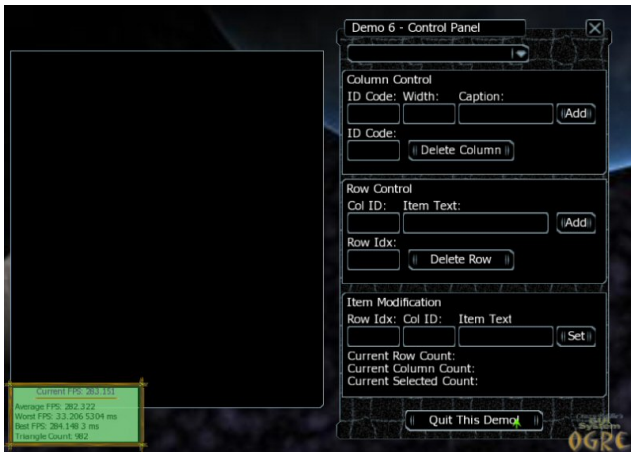


Figure 1 Game user interface created with Crazy Eddies' toolkit.

In this paper we argue that developing 3DUIs for VR applications is an activity that would benefit from the application of a development method composed of: (1) a set of models defined according to an ontology, (2) a language that expresses these models, and (3) a principle-based approach manipulating these models based on principles. Principles including guidelines and design knowledge helping in the task of creating 3DUIs more close to 3D than 2D.

STATE OF THE ART

Game engines, such as: Crystal Space 3D (www.crystalspace3d.org), Open Scene Graph (www.openscenegraph.com), Panda3D (panda3d.org), Microsoft XNA, are mostly promoting the development of 3DUIs by hand. Although there are some toolkits that help designers in this task, for instance, Crazy Eddie's GUI System assists the development of 3D GUI for the Object-Oriented Graphics Rendering Engine (OGRE). The set of elements that are supported are just limited to those existing in a GUI. There is no particular innovation in the representation as it just imitates a 2D GUI (Figure 1). There is a plethora of methods to develop 3DUIs; in [4] we explored some of them. Although, methods share some similarities on their steps, several conceptual dissimilarities differentiate them. Consequently, it is challenging to transfer one abstraction from one method to another in order to have one consistent framework supporting the development lifecycle of 3DUIs. Moreover, these methods rarely provide the design knowledge that should be typically used for achieving each development step. Most approaches cover aspects such as task modelling, dialog modelling and implementation aspects. While some knowledge in explicit of existing method most of the time it is hard to find the abstracted models, the transformational approach along with its rules.

3D USER INTERFACES GUIDELINES FOR SELECTING 3DUIS

In this section we define concepts related to 3DUIs, i.e., the representation of 3D widgets, their characteristics, and how we can derive the 3DUI. For doing that we need: models of the 3DUI, a method and a language. Our models were incorporated to UsiXML [8] to support:

- 3D Rendering of 2D User Interfaces [10]. The UsiXML model that considers 2D GUI was specialized to consider the attributes specific for their 3D rendering. Then, this is compliant to current approaches.
- 3DUIs involving a true 3D presentation of the 3DUI [5]. By true representation is meant going beyond an imitation of their 2D GUI counterpart. In that scenario it was not enough to rely on the existing model as concepts such as appearance, texture, shape, and behaviour were needed.
- Hapgets Haptically enhanced 3D widgets [7]. This to support another modality of interaction.

Regarding the method, the goal is not to come up with yet another Software Development Method but to reuse existing work and structure it accordingly. The result is a method that structures the development life cycle of a 3DUI of an IS in a principle-based way. The method follows an exploratory approach as its goal is to show a variety of possibilities to encourage design. It is said that is structured as it is based on a structured Framework, the Cameleon Framework [1]. Second, a set of instruments (evaluation guidelines [5], task patterns [5], a list of canonical task types [5], and rules for model to model transformations [4]) guide de use of the method then making it principle-based. The method (Figure 2) includes an evaluation of each development step, which can be performed manually or automatically. We rely on principles expressed as guidelines for modelling the task and abstract user interface (AUI) model. They are applied manually and refine the models. At the concrete user interface (CUI) model direct evaluation over the code is performed by evaluating rules and conventions or recommendations. The refined model is then used for the code generation. The next subsections details the development steps.

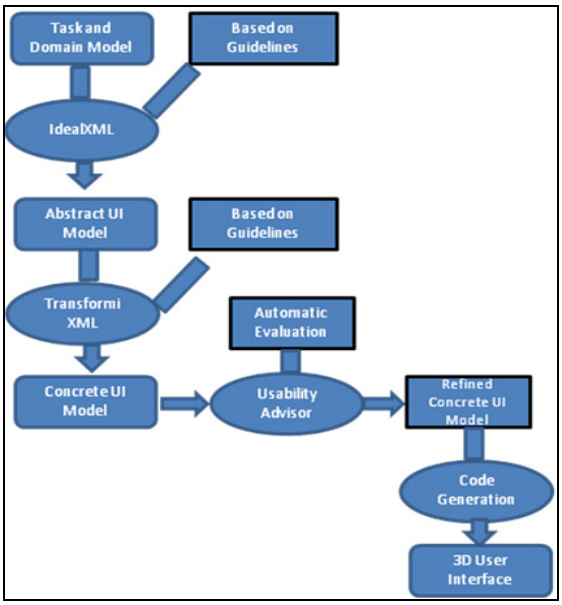


Figure 2 Outline of the method for developing 3DUIs

The method has been already reported in the literature but we would like to stress the use of the mechanism for selecting widgets based on a taxonomy of 3DUIs, this is done in the step from abstract to concrete user interface model.

After identifying the appropriate 3DUI interactive object that is needed for the VR application, for instance, a button. At this point the designer could ask himself what is the appropriate representation of a 3D button? Should the 2D desktop metaphor still be used or are there alternative visualizations or metaphors? Several attempts go towards defining a new toolkit of 3D objects which are natively appropriate to 3D applications. Again, this represents an advantage to have a predefined collection of such 3D-widgets, but then the interaction is reduced by what they offer natively.

To help designers in selecting more systematically the 3DUI elements we propose the following steps. We build a taxonomy of 3DUIs (Figure 3 illustrates the taxonomy of a radio player).

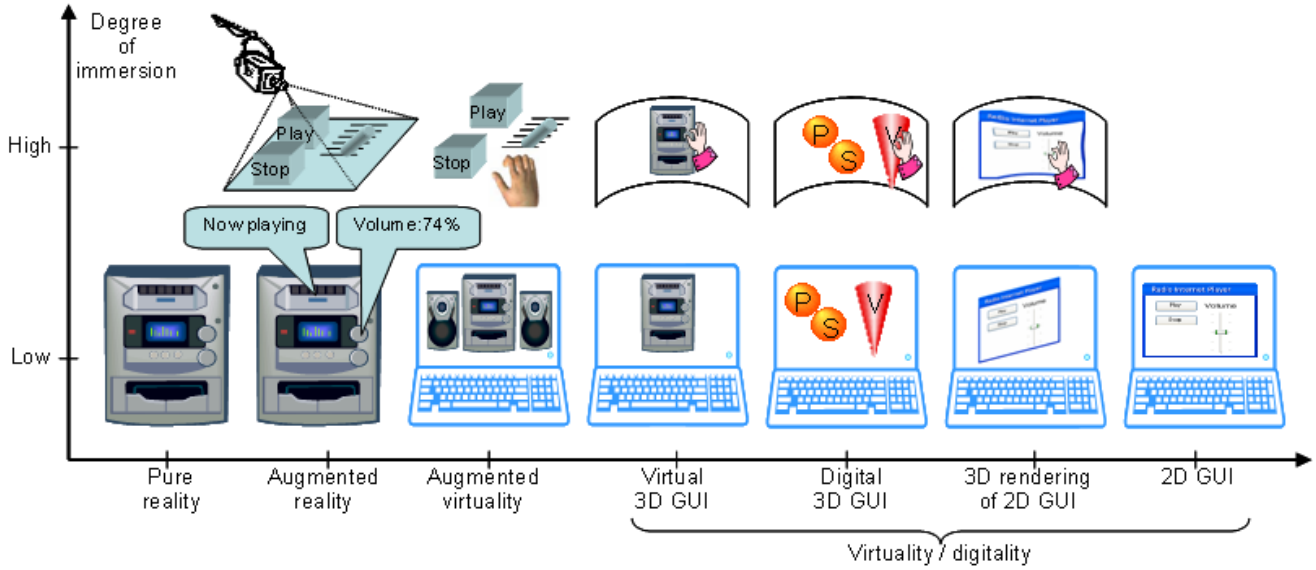


Figure 3 Three-Dimensional Sliders Taxonomy, source [4].

The taxonomy can be of some help in making that decision as a design space of potential representations of the 3DUI elements. The use of this taxonomy could changed by any other form that fits authors needs, for instance, Figure 4 shows a list of metaphors used to represent a toggle button.

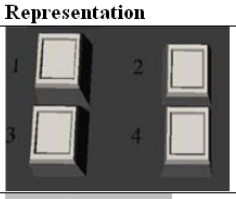
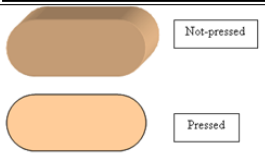
	Representation
2D Representation	
Switch	
Sphere	
Haptic	

Figure 4 Graphical representation for a toggle button

The use of the taxonomy provides hints to actual solutions for designers. More important is the fact that there can be more than the examples shown in each level, further investigation can fill the gaps or add more examples to the existing levels.

In a second step, the possible renderings are examined. The comparison between the different representations of the same component that are presented must be examined using criteria like the 2D and 3D consistency, the development complexity, the usability and how much intuitive the representation was. This evaluation of the representation could be as formal as the project demands. The result of evaluating the different renderings ends with a selection of the most appropriate representation for a 3DUI element, depending on the special needs of the users. For instance, when the target group is the blind users, there is no need to keep a 2D-3D consistency, as users cannot see the 2D components to make the comparison and more easily identify its 3D counterpart. In this case, the major criterion is to represent a 3D component in a way that is easy to comprehend and be used by the blind users, even if this representation has nothing to do with the 2D representation of the corresponding 2D components. The method for selecting the representation of the 3DUI element that we propose is the Question, option, criteria [9]. Accordingly to this method, a set of criteria can be evaluated when a question is raised and a set of possible answers exists (3D renderings in this case). In this case four criteria were identified to consider: 2D to 3D consistency, easy to develop, intuitional and the usability. The weight attributed to each criterion is based on the users who evaluate those representations.

In Figure 5 we illustrate this process for adding weight to the toggle button representation. The meaning of the links is: (++) means strongly supported, (+) means supported, (~) means neutral, (-) means denied and (..) means strongly denied. If we consider that the haptic channel is desirable and that the graphical representation is not of interest due to fact that our end user in visible impair. Then it is clear that the selection must be towards the haptic representation.

Notice that this analysis only provides a general view of the different properties related to the development of the 3DUI

elements and does not arrives at a conclusion about which representation is generally the best.

Question	Answer	Score	Options
What will be the representation of the 3D Toggle Button?	*Switch*	-	(1) 2D-3D Consistency
		--	(3) Easy to develop
		+	(4) Intuitional
		~	(5) Usability
	Sphere	~	(1) 2D-3D Consistency
		~	(3) Easy to develop
		~	(4) Intuitional
		~	(5) Usability
	2D representation	++	(1) 2D-3D Consistency
		+	(3) Easy to develop
		+	(4) Intuitional
		+	(5) Usability
	Haptic	+	(1) 2D-3D Consistency
		+	(3) Easy to develop
		+	(4) Intuitional
		+	(5) Usability

Figure 5 Questions and answer criteria to select a toggle button.

This step is part of a complete methodology supporting the development life cycle of 3DUIs [4,5,6,7] which is compliant with the Model Driven Architecture paradigm promoted by the OMG (www.omg.org). In order to be fully-MDA compliant, this works need a User Interface Description Language (UIDL). A review of the literature of existing UIDLs was conducted in [3] to justify the selection of UsiXML [8] as UIDL for our work.

CONCLUSIONS

The method presented is aimed at promoting a more systemic approach for the development of 3DUI for VR applications. This element we feel that has been considered of less importance due to the complexity of VR applications. The method shows the different steps to select 3DUI elements in a structured way. This development step reinforces existing knowledge on 3DUI development methods as we provide some means to identify the diversity of concretizations of the 3DUI (i.e. its representation).

ACKNOWLEDGMENTS

We gratefully acknowledge the support of the ITEA2 Call 3 UsiXML project under reference 20080026.

REFERENCES

1. Calvary, G., Coutaz, J., Thevenin, D., Limbourg, Q., Bouillon, L., Vanderdonckt, J.: A Unifying Reference Framework for Multi-Target User Interfaces. Interacting with Computers, Vol. 15, No. 3 (June 2003) 289–308.
2. Dünser, A., Grasset, R., Seichter, H., Billinghurst, M., Applying HCI Principles in AR Systems Design, *Proc. of the 2nd Int. Workshop on Mixed Reality User Interfaces: Specification, Authoring, Adaptation* (*MRUI'07),* Charlotte (USA), 11 March 2007.
3. Guerrero García, J., González Calleros, J.M., Vanderdonckt, J., Muñoz Arteaga, J. A Theoretical Survey of User Interface Description Languages: Preliminary Results, Proc. of Joint 4th Latin American Conference on Human-Computer Interaction-7th Latin American Web Congress LA-Web/CLIHC'2009 (Merida, November 9-11, 2009), E. Chavez, E. Furtado, A. Moran (Eds.), IEEE Computer Society Press, Los Alamitos, 2009, pp. 36-43.
4. Gonzalez-Calleros, J.M., Vanderdonckt, J., Muñoz-Arteaga, J.M., A Method for Developing 3D User Interfaces of Information Systems, *Proc. of 6th Int. Conf. on Computer-Aided Design of User Interfaces* (*CADUI'2006*), Bucharest (Romania), 6-8 June 2006, Chapter 7, Springer-Verlag, Berlin, 2006, pp. 85-100.
5. Gonzalez-Calleros, J.M., Vanderdonckt, J., Muñoz-Arteaga, J., A Structured Methodology for Developing 3D Web Applications, in T. Spiliotopoulos, P. Papadopoulou, D. Martakos, and G. Kouroupetroglou (eds.), Integrating Usability Engineering for Designing the Web Experience: Methodologies and Principles, Chapter 2, *IGI Global Inc.*, Hershey, 2010, pp. 15-43.
6. Gonzalez-Calleros, J.M., A Model-Driven Approach for Developing Three-Dimensional User Interfaces of Information Systems in a Principle-based Way, PhD Thesis, Université catholique de Louvain Press, February 2010.
7. Kaklanis, N., González-Calleros, J.M., Vanderdonckt, J., Tzovaras, D., Hapgets, Towards Haptically-enhanced widgets Based on a User Interface Description Language, *Proc. of Workshop on Multimodal Interaction Through Haptic Feedback* (*MITH'2008*), Naples (Italy), 31 May 2008.
8. Limbourg, Q., Vanderdonckt, J., Michotte, B., Bouillon, L., Lopez, V., UsiXML: a Language Supporting Multi-Path Development of User Interfaces, *Proc. of 9th IFIP Working Conference on Engineering for Human-Computer Interaction jointly with 11th Int. Workshop on Design, Specification, and Verification of Interactive Systems EHCI-DSVIS'2004,* Hamburg (Germany), July 11-13, 2004. Lecture Notes in Computer Science, Vol. 3425, Springer-Verlag, Berlin, 2005, pp. 200-220.
9. MacLean, A., Young, R.M., Bellotti, V., and Moran, T.P. Questions, Options, and Criteria: Elements of Design Space Analysis. Human-Computer Int. 6, 3-4 (1991) 201–250.
10. Molina, J.P., Vanderdonckt, J., Montero, F., Gonzalez, P., *Towards Virtualization of User Interfaces based on UsiXML*, Proc. of 10th ACM Int. Conf. on 3D Web Technology Web3D'2005 (Bangor, 29 March-1 April 2005), ACM Press, New York, 2005, pp. 169-178.
11. Swan, J.E., Gabbard, J.L. Survey of User-Based Experimentation in Augmented Reality? *Proc. of 1st Int. Conf. on Virtual Reality,* Las Vegas (USA), 2005.