xReality Interactions Within an Immersive Blended Reality Learning Space

Anasol PEÑA-RIOS [a,1], Vic CALLAGHAN [a,b], Michael GARDNER [a] and Mohammed J. ALHADDAD [b]

[a] *Department of Computer Science, University of Essex, UK.*

[b] *Faculty of Computing and Information Technology, King Abdulaziz University, KSA.*

Abstract. Multi-user virtual environments (MUVEs), used in training, entertainment and education, had been a reality for several years now, allowing users to perform collaborative activities using virtual settings and virtual objects. Nowadays, technological advances are opening the possibility of integrating multiuser virtual environments with so-called ubiquitous virtual reality (U-VR); extending and complementing real and virtual worlds in a blended reality space. In this work-in-progress paper we describe our efforts towards the implementation of a blended reality distributed system, to achieve integration between real and virtual objects., using smart objects (*xReality objects)* and immersive technology in a mixed reality learning environment, extending our previous work towards the creation of a holistic option to enable geographically dispersed learners to collaborate on laboratory activities.

Keywords. Ubiquitous virtual reality, mixed reality, dual reality, virtual laboratory, blended reality, xReality objects, interreality, human-machine interface (HMI), internet-of-things, tangible user interface (TUI), Multi-user virtual environment (MUVE).

Introduction

Traditionally virtual worlds have been regarded as a standalone entities separated from the real world, where users can create their fantasy worlds which could follow realistic physical laws and rules (e.g. gravity) or differ from reality (e.g. flying, non-human avatars, etc.). Multi-user virtual environments (MUVEs) provide a 3D virtual canvas for users to populate with 3D representations of the real world, extending human capabilities into a computer-generated space. However, new research in this field is moving towards the integration and correlation between virtual worlds and real worlds using ubiquitous computing to create different technologies and possibilities for users. Suh et al. [1] defined Ubiquitous Virtual Reality (U-VR) as the possibility to "*make VR pervasive into our daily lives and ubiquitous by allowing VR to meet a new infrastructure, i.e. ubiquitous computing*". Lee et al. [2] complemented this concept by stating that ubiquitous virtual reality produces intelligent spaces combining real and virtual worlds to create seamless connections, with the advantage of each world complementing the other. This encourages the concept of *interreality* as defined by

---

[1] Corresponding Author: Anasol Peña-Ríos, School of Computer Science & Electronic Engineering, University of Essex, Wivenhoe Park, Colchester, CO4 3SQ, UK; E-mail: acpena@essex.ac.uk

Van Kokswijk [3] "*a hybrid total experience between reality and virtuality*", generating on the user the illusion of the so-called Blended Reality [4], a "*switch context between real-local and virtual-distant environments and blend traces of one into the other in a socially unconscious manner (often seemingly simultaneously)*". To achieve this, Lifton et al. [5] proposed the creation of a cross reality (xReality) environment, a ubiquitously networked sensor/actuator infrastructure mirrored in real-time with a 3D virtual environment to complement both, virtual and real worlds, in real time bidirectional process.

But how do end users can interact with this sophisticated architecture, where real and virtual objects can host a complete bidirectional conference without talking a single word to the end user? Several gaming platforms had been using tangible user interfaces (TUI) to connect end-user to real objects and virtual worlds. Ishii et al. [6] defined a TUI as a user interface that "*augments the real physical world by coupling digital information to everyday physical objects and environments*". Video games industry has many examples: traditional game controllers, dance pads, sophisticated on-body gesture recognition controls (e.g. Nintendo Wii [2]), etc. A particular example we would like to cite is Ubisoft's Rocksmith[3] which connects a real electric guitar to a virtual interface in order to teach the end user to play the guitar in an individual learning session (or collaborative if the other user has an extra electric guitar/bass). Here the user can perform an action over the real object updating a state in the virtual world (e.g. a string pressed is reflected on the virtual fret board and playing the sound). However, this is a unidirectional communication, as actions performed over the virtual world cannot be reflected in the real world, and each real object (e.g. electric guitar) is tied to a single user at a time. In [7] Ibáñez et al. used a mobile device to interact in a learning activity, maintaining a bidirectional communication between real and virtual worlds. Although the activities described in both examples were collaborative learning activities, real world objects were designed to be use by a single user at a time.

In our previous works [8] [9], we proposed the use of *xReality objects* to achieve collaborative bi-directional communications between real and virtual objects, with the particularity to allow sharing both, real and virtual objects. *xReality objects* are smart networked objects coupled to their virtual representation, updated and maintained in real time to create a mirrored state (dual reality). The test bed scenario was a mixed reality collaborative laboratory activity to produce Internet-of-Things (IoT) applications emphasising computing fundamentals.

In the following sections of this work-in-progress paper we describe the conceptual model and architecture of our blended reality distributed system, present the implementation and discuss future work. The first phase of our research involved the creation of a single *dual reality* state and was described in [10]; in this paper we have moved on to include the creation of collaborative learning sessions, incorporating two users, and the management of multiple *dual reality* states, mirroring two or more *xReality objects* on a single virtual environment in synchronous time in different scenarios.

---

[2] Nintendo Wii - http://www.nintendo.com/wii

[3] Rocksmith - http://rocksmith.ubi.com/rocksmith/en-GB/home/index.aspx

1. A blended reality distributed system

In [10] we described the conceptual model and implementation of an *xReality object* using two different components: a main module, which communicates and identifies other components connecting them to the interreality system; and a group of interchangeable pluggable sensors and actuators, enabling the creation of diverse physical mashups.
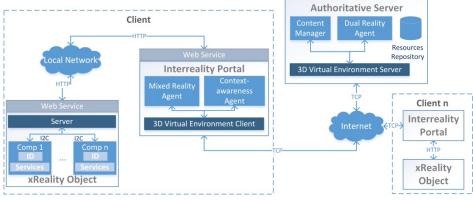

Figure 1. Blended Reality Distributed System architecture [10]

Figure 1 shows the architecture proposed. In this architecture we define three software agents to coordinate actions/responses and maintain mirroring between the virtual and real world. The Context Awareness agent (CAag) captures any changes in real or virtual world and sends the information to the Mixed Reality agent (MRag). MRag sends this data to the Dual Reality agent (DRag) and receives instructions from it to execute within the local virtual environment, representing/executing any change in either the virtual or real world. The DRag manages the multiple dual reality states, synchronising client environments using these predefined rules [10]:

a) *A change in any Virtual object of a given Interreality Portal results in identical changes to all subscribing Interreality portals.*
b) *A change in an xReality object of a given Interreality Portal results in changes in the representation of the real device on all subscribing Interreality portals.*

In figure 2 we describe concurrent state transitions between the three main elements of our Blended Reality Distributed System: the *xReality object*, the *Interreality Portal* and the server. The process starts when the *Interreality Portal* client connects to the server for validation. Once validated by the server the learner can choose to enrol in a learning session. At this moment the server sends an individual inquiry to all the *Interreality Portals* enrolled in the session asking for information on all the *xReality objects* available. The *Interreality Portal* connects to the local *xReality object* and sends a list of available services to the server. The server lists them, making them available for execution by any of the *Interreality Portals* active in the learning session. If a user chooses to execute a particular service (local or remote) the *Interreality Portal* sends the information to the server which locates the selected service and sends an execution call to the *Interreality Portal* that owns the *xReality*

3

object. The *Interreality Portal* sends the RPC to the *xReality object* and waits for the result. At this point the service is marked as locked for both, the server (to deny any subsequent call for a different client) and the *xReality object*, until the latter finishes execution and sends data to the *Interreality Portal*, which conveys the data to the server. Finally before unlocking the service, the server updates its data and synchronises all subscribing *InterrealityPortals*.
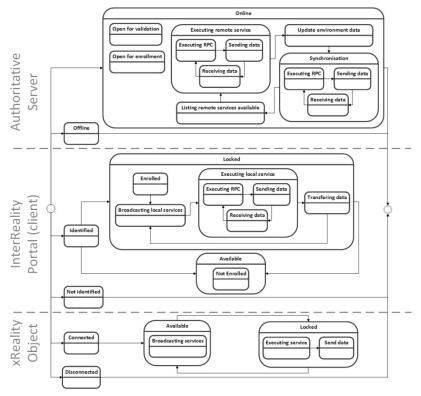

Figure 2. Blended Reality Distributed System states diagram

1.1. Implementation

Figure 3 shows a 3D virtual representation of an *xReality object* being explored by two learners within a learning session. These virtual representations were created using Unity3D[4], a cross-platform game engine for creating interactive 3D content which supports C# and JavaScript routines. To create collaborative sessions and maintain the dual reality states we implemented an authoritative server using SmartFoxServer X2[5] (SFS2X), a middleware platform optimized for real-time multiplayer games, MMOs, virtual communities, etc. SFS2X API connect the clients to the central server via a persistent connection (using the TCP protocol), while the central server is responsible for maintaining object states and sending back synchronisation messages to every client.

---

[4] Unity3D Game Engine - www.unity3d.com
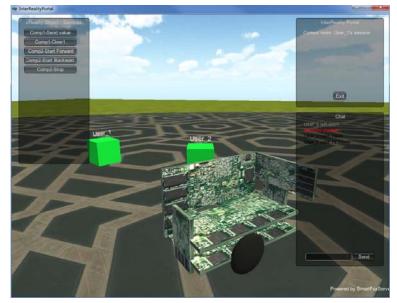[5] SmartFoxServer 2X - www.smartfoxserver.com

Figure 3. 3D Virtual Interface – Interreality Portal.

To start a learning lesson, the user sets up an *xReality object* (fig. 4) by turning on the main component and connecting it to the local network. The *xReality object*'s main component was implemented using a Raspberry Pi[6] (RPi), a small low-cost computer with a linux-based operating system. At this point the learner can connect one or more interchangeable pluggable physical components (BuzzBoards) on RPi's general purpose I/O pins, which communicate using a python library via the Inter-Integrated Circuit bus (I^2C). I^2C is a multi-master serial single-ended computer bus created by Philips in 1982 for attaching low-speed peripherals [11]. Once a BuzzBoard is detected by the RPi, its available services are broadcast to the network via a RESTful web service (WS). This WS was implemented using Bottle[7], a distributed python-based Web Server Gateway Interface (WSGI) micro web-framework. Python[8] is an open-source general-purpose programming language promoted by the Python Software Foundation, and selected by the Raspberry Pi Foundation as its official programming language. Pluggable components were implemented using Fortito's BuzzBoard Educational Toolkit[9], a collection of pluggable network-aware hardware boards which can be interconnected to create a variety of Internet-of-Things (IoT) prototypes by using combinations of modules plugged together. For example, if the learner connects the BuzzBot module, formed by a bot servo motor, light sensors and IR Rangers, the RPi will advertise the movement/direction options for the servo motor (forward, backward and stop) and the "read" service for all the sensors (fig. 4) [10].

---

[6] Raspberry Pi Foundation – http://www.raspberrypi.org

[7] Bottle: Python Web Framework - http://bottlepy.org/docs/dev/

[8] Python - http://www.python.org/

[9] Fortito Ltd – http://www.fortito.mx/en

[10] In [9] we presented a detailed description of the implementation of an xReality object.

5

To use the system a learner starts the 3D learning environment and, once he is authenticated the 3D virtual environment, it will display the "lobby" space, where he can chat with other learners and arrange a learning session. To create these learning sessions we used SFS2X's Room object, allowing it to create and destroy sessions at runtime from either the client or the server. As soon as the learners join one session from the list located on the left side of the "lobby" space, they enter into a shared virtual world where they can see each other as avatar representations, and they can see the virtual representation of the xReality object(s) linked to any of the users of the current learning session. These where detected using the broadcasted list of services available, which is located on the left side of the main screen (fig. 3). A chat window is located on the right side of the screen to allow the users to communicate during the learning session.

Figure 4. xReality object

In [12] Fowler points that research on 3D virtual learning environments (VLEs) cannot truly be considered as 3D from a sensory point of view, as visually these environments are accommodated in a 2D desktop computer. Therefore our testbed was deployed on an immersive environment using Immersive Display's ImmersaVu [11] platform (fig. 5), a composite moulded panoramic dome screen, which allows a free-range of head movement without the need of any special instrumentation such as glasses or other devices that can interfere with the learning session.

---

[11] Immersive Display - www.immersivedisplay.co.uk/pdf/immersavu.pdf

6

Figure 5. ImmersaVu

1.2. Learning scenarios based on xReality interactions

In [13], Alrashidi et al. described a 4-Dimensional Learning Activity Framework (4DLAT) (fig. 6), where they classified the learning activities by the elements that compose the activity: number of learners and complexity of the task. In a previous work [14] we proposed a classification of activities based on its nature and characteristics (fig. 7). Similarly Lee at al. [2] identified three key dimensions in the creation of Ubiquitous Virtual Reality (U-VR) implementations:

- Reality; which refers to the point where the implementation is located in relation with Milgram's virtuality continuum [15].
- Context; which refers to the flexibility to change and adapt according to time and space. Context can be presented as a static-dynamic continuum.
- Activity; which refers to the number of people that will execute an activity within the implementation, going from a single user to large community.

All these classifications explore learning activities from a technological point of view, based on the nature and complexity of the task. It is necessary to consider pedagogical challenges in the implementation of these activities, evaluating the learning benefits. Elliot et al. [16] argue that "technical components are part of the learning environment, and as such should not be treated as separate, but interconnected constructs". From this perspective, we integrate these classifications to define the learning scenarios described in table 1.
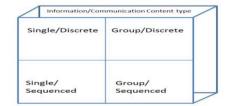

Figure 6. 4-Dimensional Learning Activity Framework (4DLAT) [13]

Figure 7. Classification of Activities in Mixed Reality Learning Environments [14]

Figure 8 illustrates the possible scenarios for learning activities using a combination of virtual objects and *xReality objects* in an individual or collaborative session. In this first implementation we only consider the possibility of using single services through the 3D virtual world. Future research will include the creation of sequences of services designed by the learners, similar to Chin's virtual appliances approach [17], therefore 4DLAT's sequenced activities (Single-Sequenced / Group-Sequenced) were not considered on this preliminary learning scenarios.
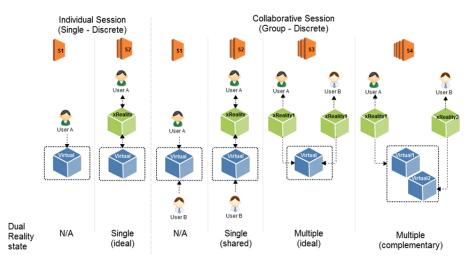

Figure 8. Possible scenarios for virtual and xReality objects

Scenario 1 (S1) examines the use of only virtual objects, either in an individual or collaborative session, similar to virtual laboratories where simulation is the key to performing an action; in this case although there is synchronisation between virtual representations within a collaborative session, there is no dual reality state. A dual reality state involves the coupling of a real object to their virtual representation, updated and maintained in real time [10].

Scenario 2 (S2) describes the use of virtual objects and an xReality objects in a learning activity; the individual session represents the ideal case of dual reality state. In

8

the case of a collaborative activity it shows the possibility of having just one xReality object shared by the users; in this case both, the remote and the local user, have access to the services available for the physical mashup, and both can control it using its virtual representation. Although only one the local user will see the result of the activity in the physical object, the remote user can follow the execution via the 3D virtual environment, as the xReality object keeps updating its virtual representation in real time.

Scenario 3 (S3) exemplifies the synchronisation of two (or more) xReality objects with the same hardware configuration (e.g. two desktop robots) within a collaborative session, creating real-time multiple dual reality states. This could be considered as the ideal scenario for multiple dual realities.

Finally, Scenario 4 (S4) describes a collaborative session where users do not share the same xReality object hardware configuration. In this scenario User A could have a part of the final mashup (e.g. a led display) and User B in a remote location could have a different device (e.g. a bot servo motor) that complements the one in possession of User A. Both objects have a mirrored virtual representation within the virtual world, therefore both are xReality objects but the mashup is only configured in the virtual world, creating multiple complementary dual reality states. In our example this will happen when User A and User B create the desktop robot using a combination of their virtual pieces. Once the mashup is created in the virtual world, both users could execute the available services. Both can follow the result in the virtual world, although only the user with the respective physical device will see the result of the activity reflected in the physical object.

Summary and future work

In this work-in-progress paper we briefly explained the fundamental concepts and rationale of our research. We reviewed our previous work towards the creation of a Blended Reality Distributed System, explaining how this paper extends our research from the creation of a single dual reality state, available to a single user, to the creation and management of a multiple dual reality states in a collaborative session, by 1) proposing an architectural model and 2) exploring different learning scenarios, including the possibility of having collaborative sessions with a real world object shared by multiple users at a time (single shared dual reality) and completing mashups within the virtual environment (multiple complementary dual realities).

Our current implementation manages single dual reality states (ideal and shared), the first one in an individual session and second one within a collaborative activity. However the architecture proposed allows the implementation of multiple dual reality states. This opens a possibility for learning sessions in places where laboratory resources might not be available due to place or money constraints. Future work advances towards implementation of multiple dual realities (ideal and complementary) using two or more *xReality objects*; and the integration of sequenced groups of services to be executed within our *Interreality Portal*, encouraging teamwork, creativity and innovation. Our main contribution from this paper is the proposed Blended Reality Distributed system architectural model and the implementation that enables

9

geographically dispersed learners to control a single *xReality object*. In particular we introduced the possible interactions within real and virtual objects, and classifications for each type of associated dual reality. We look forward to presenting further outcomes of this research, as our work progresses, in subsequent workshops and conferences.

Acknowledgments

References

[1] Y. Suh, K. Kim, J. Han and W. Woo, "Virtual Reality in Ubiquitous Computing Environment," in *International Symposium on Ubiquitous VR*, 2007.

[2] Y. Lee, S. Oh, C. Shin and W. Woo, "Ubiquitous virtual reality and its key dimension," in *International Workshop on Ubiquitous Virtual Reality*, 2009.

[3] J. van Kokswijk, Hum@n: Telecoms and Internet as Interface to Interreality : a Search for Adaptive Technology and Defining Users, Bergboek, 2003.

[4] S. A. Applin and M. Fischer, "A Cultural Perspective on Mixed , Dual and Blended Reality," in *IUI - Workshop on Location Awareness for Mixed and Dual Reality (LAMDa'11)*, Palo Alto California, USA, 2011.

[5] J. Lifton and J. Paradiso, "Dual Reality: Merging the Real and Virtual," *Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering,* vol. 33, no. 1, pp. 12-28, 2010.

[6] H. Ishii and B. Ulmer, "Tangible bits: towards seamless interfaces between people, bits, and atoms," in *Conference on Human Factors in Computing Systems (CHI'97)*, Atlanta, 1997.

[7] M. B. Ibáñez, C. Delgado Kloos, D. Leony, J. J. García Rueda and D. Maroto, "Learning a Foreign Language in a Mixed- Reality Environment," *Ieee Internet Computing,* vol. 16, no. 6, pp. 44-47, 2011.

[8] A. Peña-Rios, V. Callaghan, M. Gardner and M. Alhaddad, "Remote mixed reality collaborative laboratory activities: Learning activities within the InterReality Portal," in *The Intelligent Campus International Symposium (IC'12)*, Macau, 2012.

[9] A. Peña-Rios, V. Callaghan, M. Gardner and M. J. Alhaddad, "End-user programming & deconstrutionalism for collaborative mixed reality laboratory co-creative activities," in *2nd European Immersive Education Summit (EiED'12)*, Paris, 2012.

[10] A. Peña-Rios, V. Callaghan, M. Gardner and M. J. Alhaddad, "Developing xReality objects for mixed - reality environments," in *Workshop Proceedings of the 9th International Conference on Intelligent Environments*, Athens, Greece, 2013.

[11] NXP Semiconductors, "NXP," [Online]. Available: http://www.nxp.com/documents/user_manual/UM10204.pdf. [Accessed 25 02 2013].

[12] C. Fowler, "Virtual Reality and Learning: Where's the Pedagogy?," *British Journal of Educational Technology,* in press.

[13] M. Alrashidi, V. Callaghan, M. Gardner and J. B. Elliot, "ViewPoint : An Augmented Reality Tool For Viewing and Understanding Deep Technology," in *Workshop Proceedings of the 9th International Conference on Intelligent Environments*, Athens, 2013.

[14] A. Peña-Ríos, V. Callaghan, M. Gardner and M. J. Alhaddad, "Towards the Next Generation of Learning Environments: An InterReality Learning Portal and Model," in *8th International Conference*

on Intelligent Environments 2012 (IE'12), Guanajuato, Mexico, 2012.

[15] P. Milgram and F. Kishino, "A taxonomy of virtual reality displays," *IEICE TRANSACTIONS on Information and Systems,* Vols. E77-D, no. 12, pp. 1321-1329, 1994.

[16] J. B. Elliot, M. Gardner and M. Alrashidi, "Towards a framework for the design of mixed reality immersive education spaces," in *2nd European Immersive Education Summit (EiED'12)*, Paris, 2012.

[17] J. Chin, V. Callaghan and G. Clarke, "Soft-appliances : A vision for user created networked appliances in digital homes," *Journal of Ambient Intelligence and Smart Environments,* vol. 1, no. 1, pp. 69-75, 2009.