

Using mixed-reality to develop smart environments

Anasol Peña-Ríos^{*}, Vic Callaghan[†], Michael Gardner^{*}, Mohammed J. Alhaddad^{*†}

^{*} Department of Computer Science, University of Essex, UK. E-mail: acpena@essex.ac.uk

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

Abstract— Smart homes, smart cars, smart classrooms are now a reality as the world becomes increasingly interconnected by ubiquitous computing technology. The next step is to interconnect such environments; however there are a number of significant barriers to advancing research in this area, most notably the lack of available environments, standards and tools etc. A possible solution is the use of simulated spaces; nevertheless as realistic as strive to make them, they are, at best, only approximations to the real spaces, with important differences such as utilising idealised rather than noisy sensor data. In this respect, an improvement to simulation is emulation, which uses specially adapted physical components to imitate real systems and environments. In this paper we present our work-in-progress towards the creation of a development tool for intelligent environments based on the interconnection of simulated, emulated and real intelligent spaces using a distributed model of mixed reality. To do so, we propose the use of physical/virtual components (*xReality objects*) able to be combined through a 3D graphical user interface, sharing real-time information. We present three scenarios of interconnected real and emulated spaces, used for education, achieving integration between real and virtual worlds.

Keywords—intelligent environments; interreality; hyperreality; ubiquitous virtual reality; blended reality; mixed reality; HCI.

I. INTRODUCTION

An Intelligent Environment could be defined as a physical space populated with numerous networked controllers in which actions are orchestrated by self-programming processes to create an interactive holistic space that enhance an occupants experiences [1]. These environments combine sensing/effecting/control capabilities (*automation*) with artificial intelligence (*intelligent environment*) based on the pervasive/ubiquitous availability of services. Main actors within an intelligent environment are: users, objects (with at least one controllable service) and the environment itself, understood as a space where all the actors are located and interact with each other. The mix of ubiquitous computing and distributed systems has opened new opportunities for the interconnection of environments located in geographically distributed places.

A key factor in the development of these environments is the use of multimodal intuitive human-computer interaction (HCI) interfaces. Human computer interfaces allow interaction between the actors and can take graphical or physical forms. Moreover, some researchers have explored decomposing the boundaries between real and virtual components, creating composite systems or so-called mixed reality (MR) environments [2]. One example is the use of simulated environments that reproduce

features and behaviour of equipment and users via artificial means, such as virtual environments.

Ubiquitous Virtual Reality (UVR) has been defined as the possibility to “*make VR pervasive in our daily lives and ubiquitous by allowing VR to create a new infrastructure, i.e. ubiquitous computing*” [3]. Lee et al. enriched this concept by stating that ubiquitous virtual reality can produce intelligent spaces by combining real and virtual worlds to create seamless connections, with the advantage of each world complementing the other [4]. These, and similar terms, have been used to create this idea of a seamless interaction between an intelligent environment and its users. Some examples include Interreality which is “*a hybrid total experience between reality and virtuality*” [5], and Blended Reality which has been defined as an interactive mixed-reality environment where the physical and the virtual are intimately combined, blending traces of one into the other, to achieve communication and interaction [6]. Figure 1 shows the position of blended reality within the Physical-Virtual continuum. Tangible user interfaces (TUI) need to be considered as part of this concept as they “*augment the real physical world by coupling digital information to everyday physical objects and environments*” [7]. In this work, emulation is defined as the imitation of behaviour of an electronic or computer system using another type of hardware, with the advantage of representing faithfully physical variables. It is within this mixture of tangible devices and virtual components that hybrid hardware/software emulation is located.



Figure 1. Physical-Virtual continuum[8, 9]

From a human perspective, to achieve this degree of interaction it is necessary to avoid the so-called “*vacancy problem*”, which involves a user’s capacity to be present and engaged to a single reality at a time [10]. Therefore it is necessary to consider how technology can create a real-time synchronised copy of a physical environment with sufficient fidelity to be effective. Lifton et al. [11] proposed the use of a ubiquitously networked sensor/actuator infrastructure mirrored in real-time with a 3D virtual environment, emphasizing the importance of a real-time bidirectional process (cross reality environment). This is clearly related to management and identification of services within an intelligent environment, linked to a 3D graphical user interface.

The concept of Hyperreality adds an extra layer of complexity by incorporating intelligence (human and artificial) to the creation of mixed reality environments. It can be defined as the mix of virtual reality (VR) with physical reality (PR) and artificial intelligence (AI) with human intelligence (HI) allowing seamless interaction between all the parts [12]. In our previous definition of an intelligent environment, we considered artificial intelligence (ambience intelligence) as a fundamental component; however this concept considers human intelligence as a fundamental part of the equation. Figure 2 illustrates the possible relationships between reality and virtuality, having a real environment (HI+PR), a virtual environment (VR+AI) and a hyperreal environment (VR+PR+AI+HI). The ultimate goal should be to achieve hyperreality in an intelligent environment.

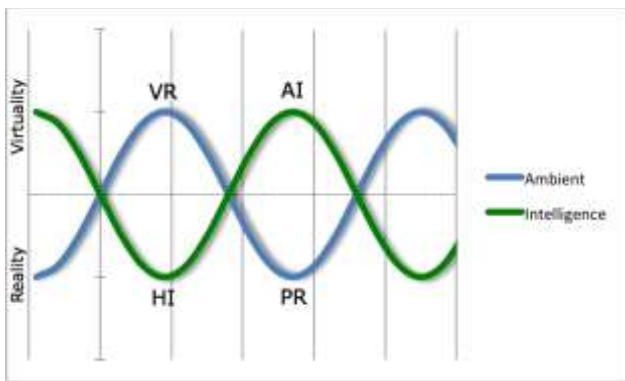


Figure 2. Representation of Ambient and Intelligence in Hyperreality.

Our perspective of an environment encompasses a setting with objects and users. These objects provide services based on a network of sensors and actuators. This definition (to some extent) might be applicable to virtual environments as virtual environments are virtual settings with virtual objects and virtual users. Figure 3 shows possible interactions between objects and users within an environment (real or virtual).

The Internet of Things (IoT) is intimately linked to intelligent environments, as usually the components are connected (and sometimes associated) via a local network; updating their state according to users settings and sensor/effector/controller values, thereby enabling the production of diverse services. The conventional way to interact with these services is via hardware interfaces (e.g. a light switch or volume control). Moreover, the Web-of-Things (WoT) adds a virtual layer to establish communication between physical objects [13] by converting real objects into RESTful resources available over HTTP, creating virtual-physical and physical-physical mash-ups. Chin et al. [14] proposed the concept of virtual appliances, based on a deconstructionist model. This model argues that a space contains objects, which provide services. Once these services have been identified, the user can 1) mix them, creating novel virtual mash-ups (meta-applications and meta-appliances), and 2) determine when these virtual appliances should be used (rules). For example, by combining a telephone, a room's curtains and a television it is possible to create a virtual appliance for

entertainment (e.g. if TV = ON then CURTAINS = CLOSED and TELEPHONE_VOL = LOW). This enriched interaction with an intelligent environment via the use of digital services. In [15] [2] the authors explored the use of 3D graphical user interfaces (GUI) to control services embedded in objects within a intelligent environment. Both examples enabled users to interact within a local environment, via physical objects (e.g. switches) and virtual objects (meta-applications, virtual appliances).

In the next section we introduce our work-in-progress research towards the implementation of a distributed system that is able to link simulated, emulated and real intelligent environments using smart objects that can be combined into different physical mash-ups linked to a virtual representation, enabling the possibility of creating mixed reality mash-ups.

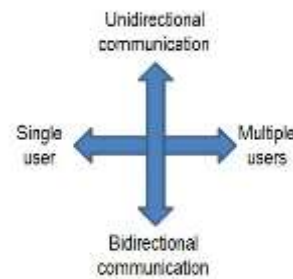


Figure 3. Interactions between actors within a mixed reality environment.

II. A BLENDED REALITY DISTRIBUTED SYSTEM

In previous work [16] [17] we proposed a model to create blended reality within a distributed computing architecture, allowing bidirectional communication between multiple environments, smart objects and users. Figure 4 illustrate the architecture proposed, based on a client-server design.

In this model each client represents an intelligent environment (real or emulated/simulated) and the objects that enable services within it. The interface to, and between these differing realities is what we label as an Interreality Portal. The primary function of an Interreality portal is capturing any changes in real or virtual environments/objects (via the Context Awareness agent) and synchronising changes in mirrored objects and environments (via the Mixed Reality agent). The combination of a Mixed Reality agent (MRag) and a Context Aware agent (CAag) forms what we call an *xReality object*, which is a type of smart networked object with a virtual representation that is updated and maintained in real-time creating a mirrored state (dual-reality). This achieves collaborative bi-directional communication between real and virtual objects. In more detail an xReality object is formed by:

- a) A manager component, which identifies and controls all the subscribed components that form each smart object.

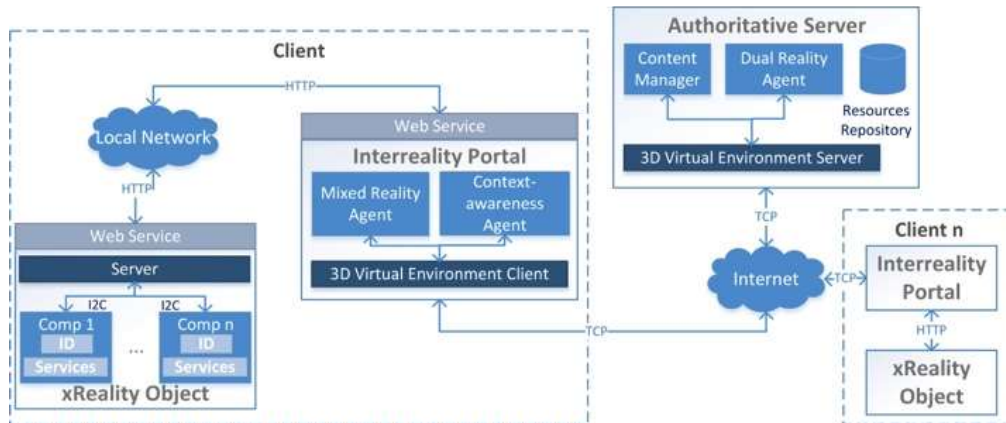


Figure 4. Blended Reality Distributed System architecture [17].

b) A group of interchangeable pluggable sensors and actuators, which provide a list of services.

c) A virtual representation that is linked to a physical mash-up.

The client side of the architecture shows the interaction between the 3D GUI and the local mixed-reality environment (xReality objects and users). Communication is achieved using RESTful web services, following the ideas of the Web-of-Things. Although multiple instances of a virtual environment can be deployed and linked to a real environment at the same time, our system employs just one virtual representation for each environment/object, creating a one-to-one relationship.

To interconnect multiple environments we use a Dual-Reality agent (DRag) situated in the server. This is the responsible for the orchestration of multiple intelligent environments using dual-reality states synchronised according to the following predefined rules [17]:

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.

III. CONNECTING SIMULATED, EMULATED AND REAL INTELLIGENT ENVIRONMENTS

To implement the architectural model described above we utilised the following:

a) Virtual environment components were created using Unity3D¹, a cross-platform game engine for creating interactive 3D content.

b) Physical components were based on a Raspberry Pi² (a credit card sized low-cost computer that runs Linux) and Fortito's BuzzBoard Educational Toolkit³ (a collection of pluggable network-aware hardware boards designed to create a variety of Internet-of-Things prototypes). Communication between the Raspberry Pi (RPi) and the

pluggable components was achieved using an Inter-Integrated Circuit bus (I²C). I²C is a multi-master serial single-ended computer bus created by Philips in 1982 for attaching low-speed peripherals.

c) The server was based on SmartFoxServer X2⁴ (SFS2X), a middleware platform optimized for real-time multiplayer games, MMOs, virtual communities, etc. SFS2X provides an API able to connect multiple clients to the server via a persistent connection (using the TCP protocol). Using this connection the server was able to maintain object states, sending back synchronisation messages to every client.

In the following paragraphs we present three different scenarios for connecting a mix of simulated, emulated and real intelligent environments based on the blended reality distributed system proposed. We used two learning activities, the first one to explain the creation of behavioural rules for inhabitants of an intelligent environment; and the second one to explain the operation of a small desktop robot. In both activities we used our system as a collaborative teaching tool whereby students gain experience of developing complex intelligent environments.

A. Connecting a simulated and an emulated environment

The scenario adopted for this experiment was a collaborative session in which the students created a set of behavioural rules, which in this case, automatically controlled the lights (as part of a smart alarm-clock design). This activity was inspired by earlier pioneering work on the embedded-internet (the forerunner of the Internet-of-Things) [20] and Internet-Appliances [21]. In this activity we used Fortito's BuzzBox³, a desktop sized emulation of an intelligent environment, designed for education, training and R&D. It allows students to design and program intelligent environments based on assembling modularised sets of software and hardware components, plus writing code that can be transferred later to a real sized smart space. The desktop-based intelligent environment emulation (BuzzBox) is controlled using a Raspberry Pi (RPi) via an I²C bus, connecting embedded sensors and actuators. It contains a variable speed

¹ Unity3D Game Engine – www.unity3d.com

² Raspberry Pi Foundation – <http://www.raspberrypi.org>

³ Fortito Ltd – <http://www.fortito.mx/en>

⁴ SmartFoxServer 2X – www.smartfoxserver.com

ventilator fan, a controllable heater, 4 dimmable ‘warm white’ LED lights, 8 push buttons, 8 tricolour LED’s, 4 attachable sensors (light, temperature, motion and magnetic) and an oLED display. We added an audio system using RPi embedded audio hardware and a pair of speakers.

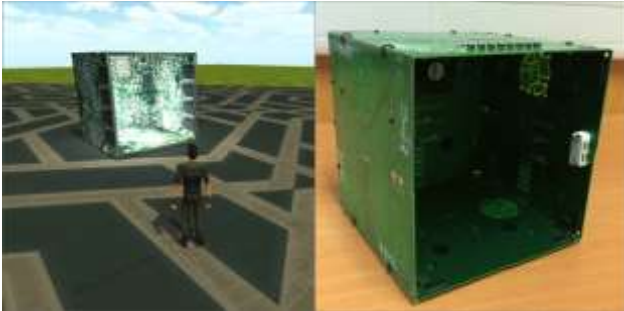


Figure 5. A desktop sized emulated intelligent environment (right) and its virtual representation (left).

The activity was designed to connect the emulated environment (e.g. BuzzBox) with a virtual representation within a simulated environment (e.g. virtual world) linking both to mirror each other in real-time. To do so, two students logged into the virtual environment. The student in possession of the *xReality object* (e.g. BuzzBox) activated it by connecting the RPi to the local network and powering it up. This student then connected a set of attachable sensors to the interior of the box. Once the RPi had detected services it broadcasted them using web services. The 3D GUI listed all available services (lights, ventilation/heating system, and audio system via the RPi) through the virtual world. In the scenario described, the students interacted with the static components (e.g. systems embedded into the box) and the dynamic components (e.g. attachable sensors which can be added or removed according to user’s needs), creating novel mash-ups. Such mash-ups can take a variety of forms from hard (the real components) to soft (simulated components) or a mixture of both. For the first part of the task, the students attached a light sensor to measure light intensity. When ambient light level detected was low it turned the lights on. Actions could be followed and triggered via the virtual representations as the real and virtual objects were tightly synchronised. Once the students agreed on a certain minimum light level they were able to create a rule to control the lights. Moreover, this scenario could be also replicated using a mix of virtual/real components. (e.g. having a box in each station and using both light sensors, having a box in each station and using just one light sensor, or using a virtual sensor within the virtual environment and changing from day to night within the virtual world) enhancing the flexibility and collaborative options.

For the final part the students created a behavioural rule to work as a smart clock alarm. The scenario proposed was: “Imagine that the box (virtual or real) represents a bedroom, with a bed and all the devices mentioned before. What type of rule would you create to wake up the inhabitants? Consider the use of a snooze button to differ the alarm timer and the use of different behaviours to encourage people not to stay in bed.”. Figure 6 exemplifies some of the rules created.

```
If TIMER = 8.00AM and SNOOZE = 0 then
DIM_LIGHTS = 25% and FAN = ON
```

```
If TIMER = 8.10AM and SNOOZE = 1 then
DIM_LIGHTS = 50% and PLAY = TUNE1 and
VOLUME = 40%
```

```
If TIMER = 8.15AM and SNOOZE = 2 then
BLINK_LIGHTS = ON and PLAY = TUNE2 and
VOLUME = 100%
```

Figure 6. Smart Alarm Clock Behavioural Rules

By using the virtual interface both students were able to see the results of rules execution (in real or virtual), and trigger the events, as the emulated and simulated environments were mirrored in real time.

B. Interconnecting a real emulated and simulated environment

For the execution of this activity we used a real intelligent environment: the University of Essex iClassroom. This is a purpose-built classroom used as a test bed for pervasive computing research applied to education [18]. This setting contains a diverse set of web service-controllable object/services: a dimmable lighting system, HVAC, a smart glass window able to turn from opaque to transparent, a door lock device and 7 light sensors. The scenario adopted for this experiment was similar to the previous one, where the students created a set of behavioural rules. The difference in this case was that the rules created were reflected in two different sized intelligent environments, and in the virtual representation.

This particular scenario was implemented in two different ways. Firstly the emulation (BuzzBox) was directly mirrored to the real environment (iClassroom) via our 3D virtual interface. In this activity one user was inside the iClassroom and the other, in a different location, was in possession of the BuzzBox. As in the previous example both users were able to control lights and HVAC system by triggering changes on the virtual world or via the light sensors implemented on each end. For example, if one user clicked on the light service within the 3D GUI, lights were turned on in the iClassroom, inside the BuzzBox and in the virtual object. This scenario could be extended to use and mediate data from light sensors within the iClassroom and the BuzzBox in a similar way to the previous example.

For the second activity, we considered both environments to be independent. For each environment was given its own virtual representation, which provided control via a 3D GUI. Given both physical spaces were independent of each other users could only follow the current interact with the state of the other environment via the virtual world.

C. A Perspective of Scale

Real intelligent environments are generally physically large systems, comprising, for example, buildings or towns. Development stations are emulations of intelligent environments are generally small, for example the desktop emulation of a room used in this work; the BuzzBox. Thus, when building intelligent environments, scale is an important issue. Figure 7 presents depicts these issues from

the perspective of physical and computational size. From this diagram it can be seen that, although the emulated space and the real space are not of the same physical scale, they can be of the same computing scale, thereby illustrating how such tools can be used to emulate large-scale interconnected intelligent environments. In our example we are using rooms as an example, but this approach could be applied to other larger spaces and systems thereby scaling up these spaces.

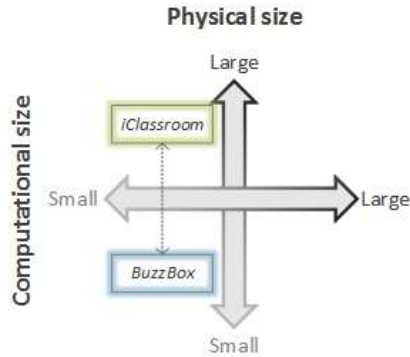


Figure 7. Scale-Scoping Diagram.

D. Adding a remote real object to a real environment using mixed reality

An alternative test activity was created using a real environment (iClassroom) with a different emulated object (a robot) via our 3D GUI. It concerned to a collaborative laboratory activity, which aimed to produce Internet-of-Things (IoT) applications, emphasising computing fundamentals based around FortiTo’s range of BuzzBoards. The objective of the activity was for geographically separated students to collaborate in the construction and control of a small robot. To do so, two students in different locations logged into the virtual environment. One of the students set up an *xReality* object by, assembling a series of BuzzBoards, connecting the RPi to the local network and powering the system up. Once the RPi detects a BuzzBoard, all the services linked to that particular board are broadcast to the network via a RESTful web service. These web services were implemented using Bottle⁵, a distributed python-based Web Server Gateway Interface (WSGI) micro web-framework. Thus, if the user connects a BuzzBot module, formed by plugging together servo motors, light sensors and IR Range finders, the RPi will broadcast all the services available. As a result the 3D GUI will present options to interact with these services. Figure 8 shows a 3D virtual representation of a BuzzBoard-based robot being constructed by two learners within a session.

In a similar way, the 3D GUI will detect the services being broadcast by the iClassroom (figure 8) and, as a consequence, enabling options in the virtual environment to use these services. In this learning scenario, we have two different types of *xReality* object: first static objects that are embedded into the environment (i.e. modification can only be achieved via a planned maintenance task, such

as adding an extra door lock or an extra set of ceiling-mounted dimmable lights); and second, nomadic objects that can be added as needed by the users (i.e. in this case the educational toolkit) creating a dynamic environment.



Figure 8. A real object and its virtual representation.

In our implementation, one student was in the iClassroom and the other was in at a different location; the modular educational kit was in possession of this last remote user. Using the 3D GUI, both of the students could see what was happening to all the objects as they were coupled to virtual representations. Figure 9 shows the main GUI screen. The list of detected objects/environment services is displayed on the left. On the right side the GUI offers a chat window to enable communication between users. In this way both students can access all the services regardless of where they or the objects are. Once the robot was assembled, any of the participants can use the broadcast services such as activating the robot’s motors. Sensor/effector services can be used in more complex ways complementing or enhancing environments/objects, such as using the light sensors embedded into robot chassis to detect ambient light intensity adjusting, for example, the iClassroom’s smart glass window or room lights as required which may depend on the path followed.



Figure 9. Interreality Portal 3D GUI.

IV. XREALITY INTERACTIONS

The scenarios explained in the previous section illustrate different combinations between real and virtual objects. For scenarios A and B, we considered both, the iClassroom and the BuzzBox as a complete *xReality* object each, regardless its physical or computational size. In the

⁵ Bottle: Python Web Framework - <http://bottlepy.org/docs/dev/>

same way, scenario C used the iClassroom and the BuzzBoards as individual *xReality objects*, able to complement each other. Figure 10 shows all the possible combinations of *xReality objects* [17]. In this figure, combinations S1 and S3 exemplify the use of simulated environments, either in an individual or collaborative session, where interaction and synchronisation occurs only between virtual representations.

Examples S2 and S4 describes the use of a real object connected to its virtual representation within an individual (S2) or collaborative (S4) session. An example of this is our scenario A, where we connected a simulated environment (3D GUI) with an emulated one (BuzzBox). Here both users can follow the result of the activity within the virtual world but only one will be able to see the execution in the real world, creating a single *dual-reality state*.

Combination S5 illustrates first implementation of scenario B, where we mirrored the iClassroom (real environment) with the BuzzBox (simulated environment) via the 3D GUI, having just one virtual representation. In this case every change reflected in any of the real objects (environments) was reflected in the virtual representation first and then in the distant counterpart, creating an ideal case of multiple *dual-reality states*. The second implementation of scenario B is represented by figure S7 where each object/environment has its own virtual representation and both can interact or complement only through the virtual environment. Here, the real environment and the emulated one have its own individual *dual-reality state*.

Finally, scenario C corresponds to figure S6, which describes a collaborative session where users do not share the same *xReality object hardware configuration*, but using its *dual-reality states* complement or enhance each other.

Here, user 1 could have a part of the final mashup (e.g. the sensors in the BuzzBot) and user 2 in a different location could have a different object/service (e.g. the lights in the iClassroom). Both could be combined to create interaction that can be followed completely in the virtual world, and partially in each of the respective physical devices/environments.

V. CHALLENGES

In the preceding sections we have introduced an innovative new tool for developing intelligent environments ranging from large real environments such as inhabited buildings through desktop emulation to simulations. The capability of this tool supporting work across these different sized environments in important to enable development of large scale connected environments; however during implementation and preliminary user evaluation, we identified a number of challenges and opportunities that are necessary to consider when using this approach. In this section we introduce some of them.

A. Connectivity between environments

For our implementation we utilise videogame technology used for real-time multiplayer games, however, network latency could be a major challenge for mixed-reality synchronization, as it is necessary not only to update the virtual representation but the real object linked to it. This creates small delays that can break the synchronisation and performance of the system. Our current implementation has been performed and tested within a controlled network. As part of our future work we will investigate this issues performing some trials to connect environments in a large-scale network configuration between the UK, Mexico and Saudi Arabia.

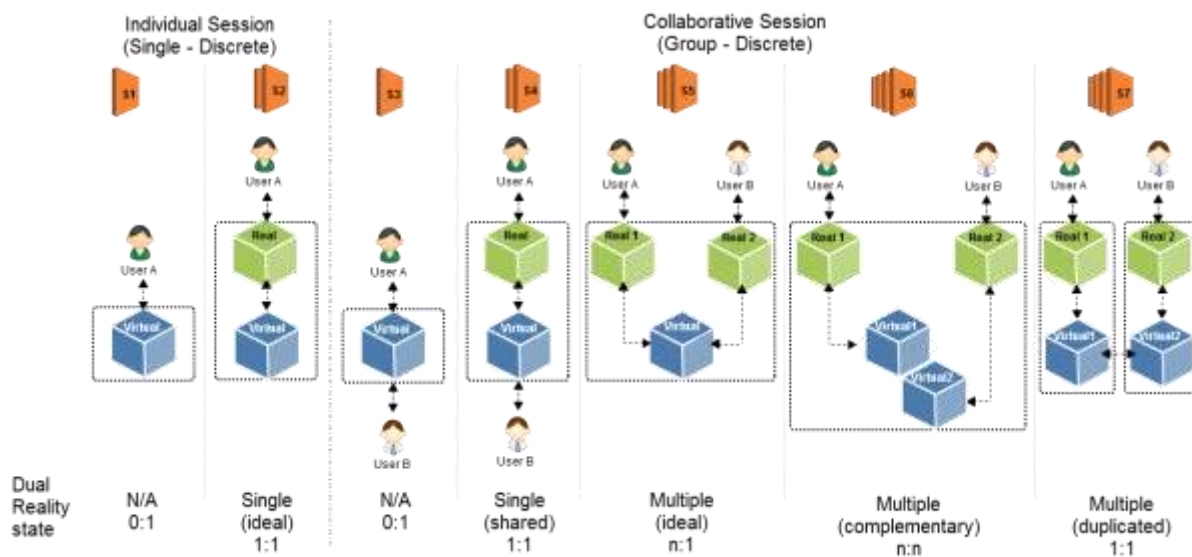


Figure 10. Possible scenarios for *xReality objects* [17].

B. Maintain the fidelity of the environment

Fidelity and integrity becomes more complex depending on the type of environment used. In addition to the system needing to synchronise status, it needs to check both environments, identify the differences and analyse the consequences of any difference, giving the users options to deal with this differences (e.g. to create a virtual object linked to just one environment or, perhaps, not to create the object at all). In doing this, it is necessary to consider the integration of environments and objects with different schemes for object detection and location.

C. Control, safety and privacy

Another challenge comes when establishing limits to control and ownership of the objects. In a scenario where there is only one real object, the ownership privileges can be assigned automatically to the one in possession of the physical object. This case becomes more complex when there are two or more shared objects, especially in the case of identical objects. Theoretically, privilege assignment should work in the same way as the previous case, but when shared within blended reality both objects “become” the same and should maintain the same state, involving safety and privacy issues (e.g. a robot being in the centre of a table in one environment but at the edge of another).

D. User acceptance

It is necessary to consider user acceptance and engagement as our conceptual model aims to alter their perception of space, time and architecture. In [19] Applin & Fisher describe different scenarios for interaction between single/multiple users within technology mediated connected/non-connected environments. Although they focused on asynchronous communication, issues they raised also apply to our synchronous communication. For example, they argue that “*when technologies become 'unmarked' (e.g. absorbed) into the 'unawareness' of daily life in society, there is a successful technology acceptance*”. Preliminary user evaluation of our learning scenarios has shown a positive acceptance to the use of this platform. However, it is necessary to create environments able to simultaneously merge traces of one into the other, enabling users to unconsciously think of it as one entity. The creation of a truly blended reality illusion is directly linked to real world fidelity and reliability of the interconnected blended space which are all goals our system would need to achieve.

VI. SUMMARY AND FUTURE DIRECTIONS

In this work-in-progress paper we introduced an innovative new tool for developing intelligent environments ranging from large real environments such as inhabited buildings through desktop emulation to simulations. The tool integrates real, emulated and virtual environments. We introduced the conceptual architecture underpinning this tool, illustrating its implementation using three collaborative learning scenarios. In the first scenario we interconnected a simulated and emulated environment; introducing a novel desktop-sized intelligent environment (the BuzzBox) used in combination with our conceptual architecture; enabling the use of static and nomadic xReality objects, in a similar way as the real-size scenario. Our second scenario connected an emulated

environment and a real environment (in this case an intelligent classroom) via a common *dual-reality state*, thus mirroring both realities. Our model considered both environments as offering equivalent *xReality objects*, regardless of their physical or computational size, making them available to be used and shared by distant users. Finally, our third scenario connected two different real entities: an intelligent environment (iClassroom) and a smart object (a small robot created using a physical mash-up of BuzzBoards) placed in different locations. In this example, we considered both shared but different xReality objects, allowing users to create a mash-up of complementary mixed reality objects, where one part of the mash-up can be in one location and the other in a different one; reflecting its status in real-time. Table 1 summarises the current affordances of our implementation.

TABLE I. BLENDED REALITY DISTRIBUTED SYSTEM AFFORDANCES

| BLENDED REALITY DISTRIBUTED SYSTEM | | |
|------------------------------------|---|--|
| | <i>Affordances</i> | <i>Description</i> |
| 1 | Simulation of real objects | Enable the use of virtual objects. |
| 2 | Emulation using a mixture of real and virtual mirrored objects (xReality objects) | Instantiation of diverse scenarios of dual reality states (single, multiple, shared, complementary, duplicated). |
| 3 | Creation of physical-virtual mashups using a deconstructionist model (virtual appliances) | Creation of mashups using services available in static and nomadic xReality objects. |
| 4 | Collaborative sessions between 2 or more users | Support the use and sharing of xReality objects within an environment, regardless its physical location. |

Our main contribution from this paper is the proposed model and implementation that attempts to mix intelligent environments, smart objects (static and nomadic) and services regardless of their physical and computational size, thereby providing a tool to enable development of scaled-up an intelligent environment. Our implementation enables the possibility of having collaborative sessions involving physical objects shared by multiple dispersed but concurrent users building mixed-reality mash-ups; and extends our previous work towards the implementation of the xReality interactions described in section IV. Additionally, we discussed some of the challenges identified in preliminary user evaluations, when implementing this technology.

We have built this system and published videos of it on YouTube (<http://youtu.be/akKPHnDY9bw>). Our plans for future work are to address some of the challenges described and to conduct more formal and exhaustive evaluations; from technical performance to user evaluation. We look forward to presenting further outcomes of this research, as our work progresses, in subsequent workshops and conferences.

ACKNOWLEDGMENT

We are pleased to acknowledge King Abdulaziz University, Saudi Arabia for their generous funding of this research project, including the provision of a PhD scholarship to the lead author. In addition, we wish to

thank Immersive Displays UK Ltd. and Fortito Ltd. for their support. Finally we are pleased to acknowledge Malcolm Lear (Essex University) for technical support relating to BuzzBoard technology and Dr. Marc Davies (Essex University) for his inspirational ideas and assistance on technical issues.

REFERENCES

- [1] Augusto, J., Callaghan, V., Cook, D., Kameas, A., and Satoh, I.: "Intelligent Environments: a manifesto", *Human-centric Computing and Information Sciences*, 2013, 3, (1), pp. 12
- [2] Davies, M., Callaghan, V., and Gardner, M.: 'Towards a mixed reality intelligent campus', in 4th International Conference in Intelligent Environments, Seattle, Washington, 2008
- [3] Suh, Y., Kim, K., Han, J., and Woo, W.: 'Virtual reality in ubiquitous computing environment', in 'Book Virtual reality in ubiquitous computing environment' (2007, edn.), pp. 1
- [4] Lee, Y., Oh, S., Shin, C., and Woo, W.: 'Ubiquitous virtual reality and its key dimension', in 'Book Ubiquitous virtual reality and its key dimension' (2009, edn.), pp. 5-8
- [5] van Kokswijk, J.: 'Hum@ n: Telecoms and Internet as Interface to Interreality: a Search for Adaptive Technology and Defining Users' (Bergboek, 2003. 2003)
- [6] Hoshi, K., Pesola, U.-M., Waterworth, E.L., and Waterworth, J.: 'Tools, Perspectives and Avatars in Blended Reality Space', *Studies in health technology and informatics*, 2009, 144, pp. 91-95
- [7] Ishii, H., and Ullmer, B.: 'Tangible bits: towards seamless interfaces between people, bits and atoms', in 'Book Tangible bits: towards seamless interfaces between people, bits and atoms' (ACM, 1997, edn.), pp. 234-241
- [8] Bower, M., Cram, A., and Groom, D.: 'Blended reality: Issues and potentials in combining virtual worlds and face-to-face classes'. *Proc. Curriculum, technology & transformation for an unknown future. Australasian Society for Computers in Learning in Tertiary Education (ascilite)*, Sydney, Australia, 2010.
- [9] Milgram, P., and Kishino, F.: 'A taxonomy of mixed reality visual displays', *IEICE TRANSACTIONS on Information and Systems*, 1994, 77, (12), pp. 1321-1329
- [10] Lifton, J., and Paradiso, J.A.: 'Dual reality: Merging the real and virtual': 'Facets of Virtual Environments' (Springer, 2010), pp. 12-28
- [11] Lifton, J., Laibowitz, M., Harry, D., Gong, N.-W., Mittal, M., and Paradiso, J.A.: 'Metaphor and manifestation-cross-reality with ubiquitous sensor/actuator networks', in 'Book Metaphor and manifestation-cross-reality with ubiquitous sensor/actuator networks' (Institute of Electrical and Electronics Engineers, 2009, edn.).
- [12] Terashima, N., and Tiffin, J.: 'Hyperreality: Paradigm for the third millennium' (Routledge, 2001. 2001)
- [13] Guinard, D., and Trifa, V.: 'Towards the web of things: Web mashups for embedded devices', in 'Book Towards the web of things: Web mashups for embedded devices' (2009, edn.), pp. 15
- [14] Chin, J., Callaghan, V., and Clarke, G.: 'Soft-appliances: A vision for user created networked appliances in digital homes', *Journal of Ambient Intelligence and Smart Environments*, 2009, 1, (1), pp. 69-75
- [15] Dooley, J., Davies, M., Ball, M., Callaghan, V., Hagra, H., Colley, M., and Gardner, M.: 'Decloaking big brother: Demonstrating intelligent environments', in *Proc 6th International Conference on Intelligent Environments, Kuala Lumpur, Malaysia (IEEE, 2010, edn.)*, pp. 324-327
- [16] Peña-Ríos, A., Callaghan, V., Gardner, M., and Alhaddad, M.J.: 'Developing xReality objects for mixed-reality environments'. *Proc. 1st Workshop on the Cloud of Things, Athens, Greece 2013* pp. Pages
- [17] Peña-Ríos, A., Callaghan, V., Gardner, M., and Alhaddad, M.J.: 'xReality interactions within an immersive blended reality learning space' in *Proc. 3rd European Immersive Education Summit (Immersive Education Initiative, 2013, edn.)*.
- [18] Dooley, J., Callaghan, V., Hagra, H., Gardner, M., Ghanbari, M., and Al-Ghazzawi, D.: 'The intelligent classroom: Beyond four walls', in *Proc 7th International Conference on Intelligent Environments, Nottingham, UK (2011, edn.)*, pp.
- [19] Applin, S.A., and Fischer, M.: 'PolySocial Reality: prospects for extending user capabilities beyond mixed, dual and blended reality', *Proc. LAMDa, 2012, 12*, pp. 1-4
- [20] Chin J, Callaghan V., 'Embedded-Internet Devices: A Means of Realising the Pervasive Computing Vision', *Proceedings of the IADIS International Conference WWW/Internet 2003, ICWI 2003, Algarve Portugal, 2003*.
- [21] Scott G and Chin J, "A Smart Internet Alarm Clock; A DIY approach to Pervasive Computing on Internet of Things", *IEEE the 5th Computer Science and Electronic Engineering Conference, Essex, UK, 17-18 September 2013*.