

An Open Standardized Platform for Dual Reality Applications

Jochen Frey
DFKI GmbH
Saarbrücken, Germany
jochen.frey@dfki.de

Robert Neßelrath
DFKI GmbH
Saarbrücken, Germany
robert.nesselrath@dfki.de

Christoph Stahl
DFKI GmbH
Saarbrücken, Germany
christoph.stahl@dfki.de

ABSTRACT

In this paper we present a new approach towards assistance and awareness systems that exploits the duality of automated homes; the status of devices and appliances in the real world on the one hand, and their digital representation on the other hand. If the digital representation is embedded in a virtual world, i.e., a 3-D model that reflects the actual physical environment, we call this a Dual Reality (2R). The concept of Dual Reality emerges from the connection between real and virtual environments, so that they can mutually reflect and influence each other. This approach can be used to create new applications and to enhance already existing applications both in the real and also in the virtual world. In this paper, we show how the standard-based Universal Remote Console (URC) platform can be applied to Dual Reality applications and therefore can serve as an architectural foundation for synchronizing a virtual environment model with the real world.

Author Keywords

intelligent environments, ambient assisted living, dual reality, universal remote console, standard

ACM Classification Keywords

H.5.2 Information Interfaces and Presentation: Multimedia Information Systems—*Artificial, augmented, and virtual realities*

INTRODUCTION

In the scope of the DFKI Competence Center Ambient Assisted Living¹ (CCAAL) we aim to enhance the quality of life of older people through technology. According to the European Ambient Assisted Living (AAL)² research initiative, this includes increasing their autonomy, enhancing their security, and preventing isolation by staying socially connected. Our long-term vision is to promote an accessible in-

¹<http://ccaal.dfki.de>

²<http://www.aal-europe.eu/>

telligent environment based on open standards and architectures and innovative solutions where everyone can continue to play a role in society.

In this paper, we present an architectural foundation for realizing assistance and awareness systems that exploits the duality of automated homes; the status of devices and appliances in the real world on the one hand, and their digital representation on the other hand. If the digital representation is embedded in a virtual world, i.e., a 3-D model that reflects the actual physical environment, we call this a Dual Reality (2R) according to Lifton [3]. Lifton connects real and virtual environments, so that they can mutually reflect and influence each other. Similarly, Streitz and Wichert [6] put hybrid symmetric interaction on their research agenda, meaning that both worlds maintain consistency. In our living labs in Bremen BAALL and Saarbrücken INTELLIGENT KITCHEN we use home-automation technology to synchronize the status of the real environment with a virtual model thereof, so that the virtual world reflects the real lab situation. Vice versa, actions performed in virtual reality (e.g., to turn on a light by touching a virtual lamp) have a similar effect on the physical environment. We believe that for many users our approach provides a cognitively more adequate way of interacting with their environment. Finally, we investigate how geographically distant environments can be synchronized in order to create the sensation of virtual presence in conjoined environments, i.e., to let users feel like other family members were present and that they will keep an eye and know if something is wrong.

The paper is structured as follows. First, we show how the standard-based Universal Remote Console (URC) platform can be applied to Dual Reality applications and therefore can serve as an architectural foundation for synchronizing a virtual environment model with the real world. Developers will learn how the synchronization between realities can be implemented on the basis of existing development tools, methods and standards. The remainder of the paper describes two demonstrators that we have developed in the context of the INTELLIGENT KITCHEN and BAALL apartment as proof of concept. We close the paper with a summary and outlook.

UNIVERSAL REMOTE CONSOLE

One of the core concepts of our approach is the usage of standards-based open architectures. We propose the industrial Universal Remote Console (URC) standard [2], that focus on accessible and inclusive user interfaces by allowing

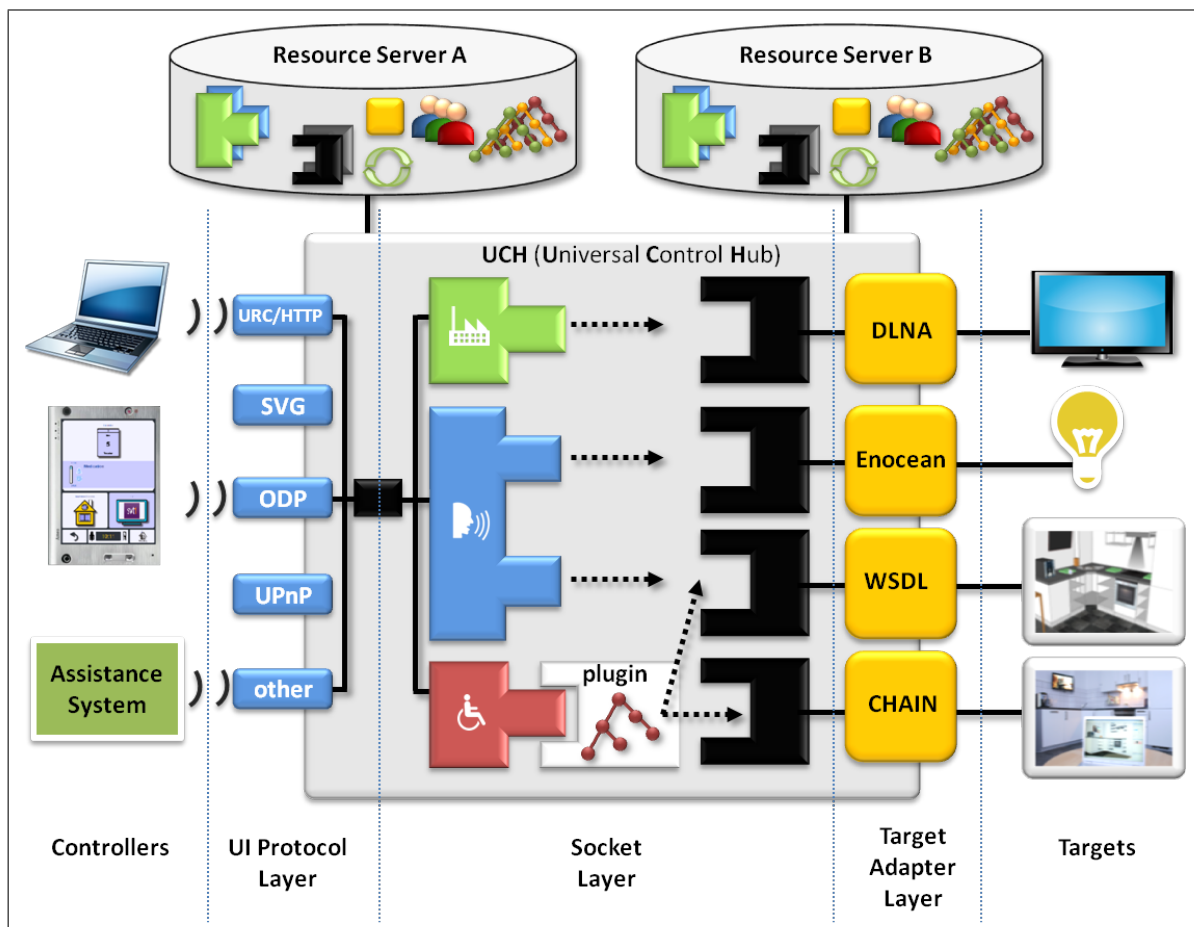


Figure 1. The URC framework provides an adaptive architecture that supports the flexible integration and reuse of heterogeneous software and hardware components, communication protocols and target appliances and therefore allows any target device to be accessed and manipulated by any controller device.

any device or service to be accessed and manipulated by any controller [1]. Like other middleware approaches, e.g., AM-ICO [4], the URC framework provides an adaptive architecture that supports the flexible integration and reuse of heterogeneous software and hardware components, communication protocols and target appliances. By contrast the focus of URC is the notion of *Pluggable User Interfaces* that allows for interfacing arbitrary networked appliances or services with secure, personalized and, perhaps most important, accessible user interfaces [1]. Users can select a user interface that fits their needs and preferences, using input and output modalities, and interaction mechanisms that they are familiar with and work well with them [10]. The first project in Europe using URC technology was i2home³, which had the ambitious effort to inject an ecosystem around the industrial URC standard [2] and to introduce URC technology in the field of AAL. Since i2home started, in Europe alone projects with in total 120 partners and an accumulated budget of about 100 million Euro are currently using

the URC technology, e.g., VITAL⁴, MonAMI⁵, Brainable⁶, SmartSenior⁷ or SensHome. Therefore, we are member of the OpenURC Alliance, which is a global initiative for exchanging experiences, ideas and, perhaps most importantly, continue the development of the URC standard.

The Universal Control Hub

Figure 1 shows the modular architecture of a Universal Control Hub (UCH) [9]. A UCH is a gateway-based architecture implementing the URC standard thus managing the communication between controller and target appliances: i) a *Controller* is any device for rendering the abstract user interface, e.g., TV, touch screen or smartphone; ii) a *Target* is any networked device or service intended to be controlled or monitored, e.g., kitchen appliance, home entertainment or eHealth devices; and iii) a *Resource Server* is a global service for sharing *Pluggable User Interfaces* and various resources necessary for interacting with the targets, e.g., language resources, icons or presentation templates defining the

³www.i2home.org

⁴www.ist-vital.org

⁵www.monami.info

⁶www.brainable.org

⁷www.smart-senior.de

arrangement of interface elements. Currently, a pilot resource server is being operated by dot UI⁸. The benefit of this approach is that it is possible to deploy consistent and, particularly, accessible interfaces which are tailored to particular users and their specific needs.

The UCH architecture mainly consists of three layers: i) the User Interface Protocol Layer is responsible for defining User Interface Protocol Modules (UIPM) which specify the communication protocols for the controllers, e.g., HTTP, SOAP, SVG, UPnP or proprietary solutions; ii) the User Interface Socket Layer defines the standardized *User Interface Socket* (UIS) that serves as a common contract between controllers and targets; and iii) the Target Adapter Layer manages the grounding and communication with actual targets, e.g., a DLNA television, Enocean lighting or CHAIN based kitchen appliances.

Standardized Contract for User Interfaces

The standardized definition of the *User Interface Socket* (UIS) describes the input and output behaviour of a specific target device on an abstract level and therefore is the common model of all user interfaces and communication protocols. A UIS contains a flat set of socket elements that provide a synchronized communication channel to the controlled device and its current state. Socket elements are either variables, commands or user notifications. The description also specifies how socket elements depend on each other, for example that the commands for confirming or cancelling the request are only relevant when the request notification is active. More advanced dependencies can also be described through the notion of pre- and postconditions.

Clearly, the UIS does not provide enough information for creating a nice-looking and accessible user interface. What's missing are concrete instructions how to build the user interface, what widgets to use, how to arrange and structure them and finally the language of labels and messages. These optional resources are provided by a *Pluggable User Interfaces* [8], a channel-specific user interface description that plugs into a particular UIS. The socket can then be rendered on some controller thus giving the abstract UIS a concrete implementation, or, in other words, plugging the socket. The concrete user interface connects to one or multiple sockets in two directions: first, getting and representing the values that reflect the current state of the target, and second, requesting changes in the target's state through variable changes and command invocations. On the backend side each target appliance is represented by a dedicated target adapter that is responsible for the grounding of abstract socket elements with any specific network protocol. The UCH architecture offers a flexible way of connecting different user interfaces with any UIS and therefore with any connected target device. Multiple controllers and targets can be attached, exchanged and detached at runtime.

An Open Plugin Infrastructure

There are several UCH implementations available, both open-source and commercial. The UCH described here is based

⁸www.dotui.com



Figure 2. The lights and sliding doors of DFKI's Ambient Assisted Living Lab (BAALL) in Bremen are synchronized with the 3-D model. They can also be interactively controlled from the model.

on an OSGI infrastructure for integrating loosely coupled modules in the form of OSGI bundles, e.g., UIPM's, Targets and *Pluggable User Interfaces*. In addition, our UCH implementation includes a dynamic discovery mechanism for optional plugin extensions (see Figure 1). With i2home's focus on activity management a task-model plugin based on ANSI/CEA-2018 [5] has been integrated. The plugin and its internal task-model engine allows for the implementation of home automation and task-based scenarios, e.g., leaving home or preparing a meal. The task-model plugin is itself represented by means of a UIS and therefore is accessible by all attached controller appliances. The synchronization of different realities is implemented by an additional synchronization plugin operating on the abstract UIS. For appliances in different realities, e.g., a kitchen in the real world and one in the virtual world, the UIS defines a common understanding of the properties and functions that are shared between realities. From a technical perspective the only difference between these two worlds is the actual grounding of the socket elements, i.e., *serve@home* communication in the case of the real extraction hood and API access in the case of the virtual YAMAMOTO model. The synchronization module is responsible for managing the values of the pertaining socket instance and for aligning attached targets. Due to the abstract socket description, URC defines a common bi-directional link between realities and thus perfectly supports the notion of Dual Reality.

DEMONSTRATORS

In the following, we will describe the synchronization between real and virtual worlds in case of the Bremen Ambient Assisted Living Lab BAALL and the INTELLIGENT KITCHEN . BAALL is an automated 60 m² apartment suitable for the elderly and people with physical or cognitive impairments. It comprises all necessary conditions for trial living, intended for two persons, as shown in the YAMAMOTO⁹ 3-D model in Figure 2. BAALL aims to compensate physical impairments of the user through adaptable furniture, such as a height-adaptable kitchen, and mobility assistance; an intelligent wheelchair and walker navigate their user within the lab. Accordingly, the lab has been equipped with five auto-

⁹<http://w5.cs.uni-sb.de/stahl/yamamoto/>



Figure 3. The programme on the TV set is reflected in the virtual model (laptop in the foreground). A context menu in the model acts as a remote control to switch channels; also the hood light is in sync between both realities.

mated KNX-based sliding doors that open automatically to let the wheelchair pass through. The doors can also be operated manually by pressing buttons or gently pushing the door itself. Furthermore, all lights in BAALL are automated. YAMAMOTO polls the status of the sliding doors and lights in the real environment and visualize it accordingly in 3-D, as shown in Figure 2. Light sources are visualized as cones that appear either black (off) or bright yellow (on). As the user switches a light, the cone in the model also immediately changes its color. Likewise, the sliding doors appear either open or closed, depending on their true state. In order to control the real world from YAMAMOTO we have extended the geometric scene description by a set of functions for each element. The functions are available through a context menu that is attached to the virtual objects. Hence the real lights can be switched on and off by manipulating the cones in the virtual environment, and the doors can be opened and closed. The INTELLIGENT KITCHEN is a fully instrumented room that allows for realizing kitchen and living room scenarios. We have fully modeled the kitchen with all its shelves, appliances, and its instrumentation with RFID sensors and a wall-mounted TFT display. Figure 3 shows a photo of the real kitchen that is synced with the YAMAMOTO 3-D model, shown on the laptop from the same perspective. We have focused on networked appliances; the fridge, cooker, oven, hood and dishwasher are connected to the Web via the proprietary *serve@home* protocol. The URC framework synchronizes for example the state of the hood light with the virtual environment model. Furthermore, the TV set (Windows Media Center) has been integrated into the URC framework. As a specialty, we stream the current content from the real TV screen to the virtual display in the 3-D world. Hence it is possible to watch the current programme with the avatar in the virtual model. The virtual TV's context menu allows to switch the channels of the TV, which affects both the real and virtual TV set. Using URC, it makes no difference whether virtual or real TVs are synced as their implementation is transparent to the system. In a user study described by [7] subjects also suggested to synchronize other devices in the house as well, for example the opening and closing

of curtains. Using URC, any connected devices can be synchronized with little effort.

SUMMARY AND OUTLOOK

We presented the open and standardized URC framework as an architectural foundation for realizing Dual Reality applications when real and virtual worlds mutually influence each other. We explained how the implementation of the synchronization process is supported by the URC framework. Finally, we presented two actual demonstrators, the BAALL and INTELLIGENT KITCHEN. Both allow the user to visualize and manipulate elements of the real world, such as lights, doors, and a TV set, from the virtual model. For the future, we plan to extend the capabilities of the demonstrators, i.e., include the real-time visualization of movable objects based on RFID sensor data and people that are localized e.g., by an indoor-positioning system.

REFERENCES

1. J. Frey, C. Schulz, R. Nesselrath, V. Stein, and J. Alexandersson, *Towards Pluggable User Interfaces for People with Cognitive Disabilities*, in: Proc. of the 3rd Int. Conf. on Health Informatics (HEALTHINF), Springer, 2010, pp. 428–431.
2. ISO, *ISO/IEC 24752: Information Technology — User Interfaces — Universal remote console — 5 parts*, "International Organization for Standardization", 2008.
3. J. Lifton, *Dual Reality: An Emerging Medium*, Ph.D. Dissertation, Massachusetts Institute of Technology, 2007.
4. Z. Obrenovic and D. Gasevic, *Open Source Software: All you do is put it together*, *IEEE Software*, Vol. 24, No. 5, Sept/Oct, 2007, pp. 86–95.
5. C. Rich, 'Building task-based user interfaces with ANSI/CEA-2018', *Computer*, **42**(8), 20–27, 2009.
6. N. A. Streitz and R. Wichert, InterLink Deliverable D4.2—Road-Mapping Research in Ambient Computing and Communication Environments, 2009.
7. H. Tsujita, K. Tsukada, and I. Siio, *SyncDecor: Communication Appliances for Couples Separated by Distance*, in: Proc. of UbiComp, ACM, pp. 279–286, 2008.
8. G. Vanderheiden and G. Zimmermann, *Use of user interface sockets to create naturally evolving intelligent environments*, in: Proc. of the 11th Int. Conf. on Human-Computer Interaction (HCII 2005), Lawrence Erlbaum Associates, 2005.
9. G. Zimmermann and G. Vanderheiden, *The universal control hub: An open platform for remote user interfaces in the digital home*, in: Human-Comp. Int., LNCS 4551, Springer, 2007, pp. 1040–1049.
10. G. Zimmermann and G. Vanderheiden, *A dream... The universal remote console*, *ISO Focus+*, Feb, 11–13, 2010. Article available online: trace.wisc.edu/docs/2010-URC-ISOFocus/