OPENURC: STANDARDIZATION TOWARD 'USER INTERFACES FOR EVERYONE, EVERYWHERE, ON ANYTHING'

Jürgen Bund[1*], Gregg Vanderheiden[2], Gottfried Zimmermann[3], Ingo Zinnikus[4], Jan Alexandersson[5], Martin Klima[6], Eduardo Carrasco[7], Gorka Epelde[7], Elena Urdaneta[8]

[1] Meticube Lda, Centro de Empresas de Taveiro, Estrada de Condeixa, 3045-504 Taveiro, Portugal, +351 962 406 400, juergen.bund@meticube.com
[2] TRACE R&D Center, University of Wisconsin-Madison, USA
[3] University of Tübingen, Germany
[4] DFKI GmbH, Germany, Ingo.Zinnikus@dfki.de
[5] DFKI GmbH, Germany
[6].Czech Technical University, Czech Republic
[7].VICOMTech, Spain
[8].ingema – Instituto Gerontológico Matia, Spain

ABSTRACT

In this paper we present an international, open, standard for the development, deployment and use of pluggable and personal user interfaces that can provide 'Accessible User Interfaces, for Everyone, Everywhere, on Anything', ISO/IEC 24752 for 'Universal Remote Consoles' (URCs).

We give an overview of available platforms and tools around this standard, their current status and technological road map.

We present reference projects that adopted the aforementioned URC standard and other URC case studies, and the range of applicability of the URC concept and technology.

Last but not least, we introduce the openURC Alliance, an international organization whose primary mission is the promotion of the URC standard.

KEYWORDS

Accessibility, Frameworks, Interoperability, Standardization, User-Centred Design

1 BACKGROUND AND MOTIVATION

Imagine you are on business travel, checking into a hotel in a foreign city. You enter the room and the air conditioning automatically sets to your preferred daytime room temperature. The TV displays a welcome screen. You pull out your smartphone and use it to switch to your favorite news channel. Even though all products and systems in the room are new to you, you

feel a familiarity because your smartphone is showing the same interface that you use for your home appliances. As this is your own personalized interface, the controls are shown in your native language, so you don't have to decipher the labels on the systems in the room which may be in a language foreign to you.

And think about elders or users with mild cognitive disabilities, some of whom would like a much simpler interface than that offered to the general public. They would no longer have to learn how to use a new interface each time a device has to be replaced or when they are travelling or visiting family and, even more important, in their own homes.

The conclusion is more-or-less evident: the evolution toward more and more (useful) devices and services in all aspects of people's lives (home, work, leisure, travel, ...) logically create demands for:

(1) 1 a wider availability of services (ubiquitous / pervasive services;
(2) 2 UIs that can integrate services (User Centric Design); and
(3) 3 UIs that can be personalized, namely, being adapted to the capacities / limitations of a specific user.

The reality, however, is that up to today, UIs tend to represent the 'stepdaughter' of many products and services. Typically dominated by concerns of brand protection, marketing, and focus on the 'sweet spot' of mainstream users, UIs frequently lack the ability to adapt to a specific user's needs.

2 THE URC STANDARD

Technology enabling wireless connectivity and networked computing is already available, providing methods for seamless discovery, controlling and eventing. At the moment, however, user interfaces still have to be authored separately for each controller platform. Furthermore, many existing interfaces are neither intuitive nor easy to understand for many users. What is needed is a standardized, versatile user interface description for products – a kind of 'user interface socket' to which any personal device or 'URC' (Universal Remote Console) can connect to discover, access and control a device, system or service. A solid user interface description alone could support diverse URC technologies – including direct manipulation techniques via desktop computers and personal digital assistants (PDAs), TV, voice recognition and natural language technologies used by PDAs and wearable computers, or even BCI (Brain Computer Interfaces). Such an approach could also enable older products to be controlled with new user interface technologies (e.g. natural language processing).

In early 2008, subcommittee SC 35, User Interfaces of ISO/IEC JTC1, Information Technology, published a new multi-part International Standard, promoting interoperable personal user interfaces, ISO/IEC 24752, Information technology – User Interfaces – Universal Remote Console (URC) [1].

The goal of URC standard and technology is to allow any device or service to be accessed and manipulated by any controlling unit. Users can then select a user interface that fits their needs and preferences, using input and output modalities, and interaction mechanisms with which they are familiar and which work well with them.

In the following, we refer to the devices and services that are to be controlled as ***Targets***, and to the controller devices and their user interfaces as ***Controllers***. To enable Controllers to access and control a Target without any prior knowledge of each other, some 'common understandings' need to be in place. The first part of ISO/IEC 24752, Part 1: Framework, defines the components of the URC framework and specifies the 'common understandings' between them as conformance requirements, stated in terms of high-level interaction. A key part of this interaction is the sharing of control and access information through XML documents.

ISO/IEC 24752 does not determine a specific networking protocol between a URC and a Target. It only defines requirements for such a networking platform. The idea is that the URC related interaction could be implemented on top of existing networking platforms that support device discovery, control and eventing – such as UPnP (Universal Plug and Play), OSGi, Web Services, PLC, ZigBee, Z-Wave, etc.

3 THE MIDDLEWARE APPROACH

Today, we are encountering a growing number of networked devices and services at home, at work, in schools and in public places. These devices and services use different networking platforms and technologies, however, including UPnP, OSGi, and web-based protocols such as UDDI, WSDL, SOAP, etc. A significant benefit would arise for consumers if manufacturers of these devices and services agreed first to adopt a common, UI-related standard like URC, and second to use a specific Target-URC Networking protocol for communication between URCs and Targets. In this scenario, a middleware solution is needed that can harvest the benefits of the URC standards, and still work with existing devices and services available on the market.

This is where the ***Universal Control Hub (UCH)*** comes into play. The UCH is a profiling architecture of the URC framework that provides connection points to existing Targets and Controllers that are not URC-compatible via a single, virtual 'box'. The UCH box is a middleware that establishes a control connection between non-compliant Controller and non-compliant Target devices/services that would otherwise not understand each other. The UCH is designed to be extremely extensible and scalable with regard to diverse Targets and Controllers. By using the mechanisms as defined in ISO/IEC 24752, it provides an open platform for personalized and adaptive user interfaces that are based upon user interface sockets. Please refer to [3] for more detailed technical information of the URC ecosystem.

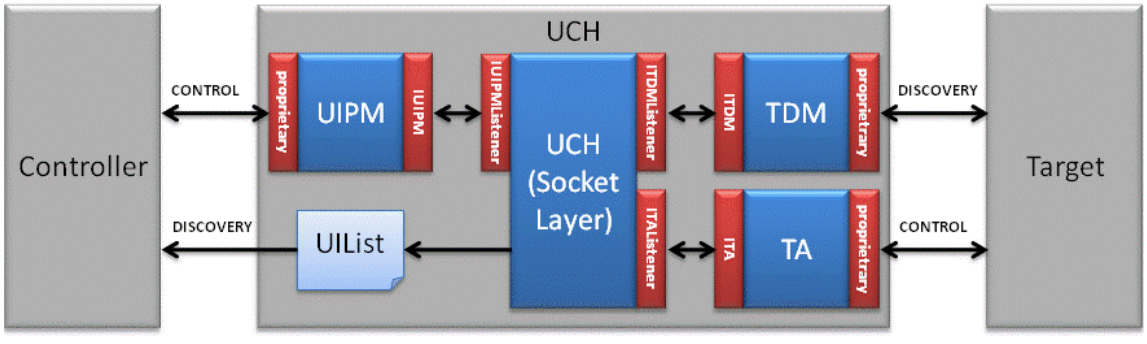

Figure 1 – UCH as middleware between Targets and Controllers

Thanks to the UCH, you can run a URC environment at home or connect to a URC environment hosted remotely (cluster, cloud, ...), and use pluggable user interfaces and similar resources in a constrained environment such as a local network. The real power of the URC framework unfolds, however, if applied to a global ecosystem. In such a scenario, different parties contribute the different parts necessary to build flexible user interfaces: providers of devices and services, providers of network services, providers of Controllers, providers of pluggable user interfaces, and the users. Here, a specific component in the URC ecosystem plays an important role, the so-called Resource Server, described in the next section.

4 THE URC RESOURCE SERVER

The URC Resource Server is a public platform for the storage, processing and retrieval of information related to URC-related user interfaces for service environments. The Resource Server allows the hosting and provision of pluggable, that is, downloadable user interfaces onto a specific UHC hardware, just-in-time. In a nutshell: the Resource Server is the user interface marketplace for the URC ecosystem, as much as the iTunes AppStore is the application marketplace for the iPhone.

The Resource Server is instrumental in providing the benefits of the URC framework regarding user interfaces, such as personalization, accessibility, context awareness, openness for third-party contributions, support for agent-based user interfaces, and support for management of user interfaces [4].

Currently, a pilot Resource Server is being operated by the US-based company dotUI (www.dotui.com).

5 CURRENT STATUS, AVAILABILITY AND SUPPORT

Today, besides the aforementioned Resource Server, a variety of platforms and tools are already available to the person or entity that is interested in evaluating the standard and/or has decided to apply it, whether in a research context or for the development of real products or services. More are already in the development pipeline. In this section we present the most important ones.

The central element for any URC-driven infrastructure is the UCH mentioned above. TRACE Center distributes an ***Open Source reference implementation*** (available in Java and C++). Meticube offers a range of UCH variants based on MS .NET, Java and C++. These are available in so-called ***Starter Kits (Basic, Professional, Enterprise)***. All versions are optimized concerning robustness, performance and scalability, equally suited for R&D and professional usage in real applications and production environments. Meticube's UCH also features a special ***Cascading Mode*** that allows interaction with devices and services that are connected to a remote UCH and supports a wide range of deployment scenarios (hierarchic, mesh, P2P, proxying, ...). DFKI disposes of an ***OSGi***-compatible implementation.

Furthermore, Meticube provides a range of tools and add-ons, like an ***Activity Management Extension Kit (EK)*** for its UCHs, based on CEA 2018, a ***Load Balancer*** for deployment of

the UCH in cluster / cloud environments, and a ***'UCH Workbench'***, a graphical tool that supports the development, installation, configuration, operation and control of UCH-driven systems and applications, from small local installations up to large-scale deployments with thousands of UCHs operating in parallel.

VICOMTech provides three EKs. First, a MS Windows Media Center (MCE) Extension Kit enables MS MCE to be used as both Target and Controller. Second, the ***Avatar EK*** enables the design and deployment of UIs that incorporate 3D avatars in combination with speech-driven interaction. Last but least, a ***Video Conference EK*** provides support for UIs that integrate video-based user interaction, on both the Controller and Target sides.

From the Czech Technical University comes the ***UITV EK***. Its purpose is the rapid development of UIs specifically designed for usage in conjunction with TV sets. UITV also features configurability and dynamic rendering of UIs.

Finally, C-LAB GmbH from Germany provides a simple but efficient ***GUI Tool EK*** based solely on JavaScript.

6 ADOPTION AND APPLICATION OF THE URC STANDARD

The URC standard found application already when it was in draft international standard status. Today, it is used in many different application domains, like Ambient Assisted Living (AAL), Independent Living, e-Health, Home Automation & Control, Energy, Automotive, Mobility, Public Transport, Industrial Maintenance and Marketing.

In the following we briefly present some of the most relevant projects and initiatives that adopted the URC standard and the UCH and related technologies as core building-blocks.

The first initiative to do so, and the very first in Europe, too, was the European research project ***'i2home'***, an ICT FP6 STREP project (www.i2home.org).The scope of i2home is the Intuitive Interaction for Everyone with Home Appliances based on Industry Standards. In this way i2home made devices and appliances at home more accessible to persons with mild cognitive disabilities and older persons. Besides various technical results (see Section 5 above), the project conceived, applied and successfully evaluated the so-called User Centric Design methodology.

i2home's sibling project ***'MonAMI'*** (www.monami.info), an ICT FP6 IP project, was the second initiative to incorporate the URC standard, along with the UCH developed by i2home. The objective of the MonAMI project is to demonstrate that accessible, useful services for elderly and disabled persons living at home can be delivered in mainstream systems and service platforms like OSGi. In MonAMI, URC and UCH extend OSGi's capabilities towards UIs by enabling the deployment of pluggable UIs.

The ICT FP6 STREP project ***'VITAL'*** (www.ist-vital.eu) was the third project that aimed to apply the URC standard and i2home's UCH as core building-blocks. The VITAL project aims to consolidate the technological framework that will enable the smooth transition of multimedia communications (including voice) from a circuit to a packet switched domain of the communications.

'BrainAble' (www.brainable.org), an ICT FP7 STREP project, is another recent research project, coordinated by the Spanish R&TD institution Barcelona Digital. BrainAble will conceive research, design, implement and validate an ICT-based human computer interface (HCI) composed of BNCI sensors combined with affective computing and virtual environments. This combination will dramatically improve the quality of life of people with disabilities by overcoming the two main shortcomings they suffer – exclusion from home and social activities – by providing inner functional independence for daily life activities and autonomy and outer social inclusion.

USG – Universal Service Gateway is a product development project carried out by T-Systems GmbH, a branch of Deutsche Telekom AG. Here, the URC standard and Meticube's UCH are applied in a completely different domain and market – ***Automotive***. The goal of the USG project is to provide a scalable, reliable and high-performance service platform that interconnects the Car Environment, its multimedia capabilities and sensor / actuator network with a variety of online services through a highly personalizable gateway – the USG. Another aspect is the seamless integration with other environments, namely the home, work and mobile environments.

In March 2010, openURC Founding Members Meticube and dotUI, along with partner TriDiVisions, established a Joint Venture agreement for the development of an innovative product for Facility and Energy Management called ***SEFA – Smart Energy for All***. Here the UCH, along with other technology and platforms provided by Meticube, will be adapted for managing and reducing the energy footprint of physical infrastructures. Facility Managers will benefit from a technical console that allows the central monitoring of all aspects of energy and water consumption, while end consumers / inhabitants will be provided with pluggable UIs based on the URC standard that enable them to interact with their environment ad-hoc and in an intuitive way.

Currently, the URC standard, along with the available tools and other software packages, is ***applied in R&D and product development projects that total more than €100 million and aggregate more than 100 entities on five continents*** (Europe, North and South America, Australia and Asia). This shows clearly both the applicability and usefulness of the URC standard, in a variety of domains and markets.

7 THE OPENURC.ORG ALLIANCE

The mission of the ***openURC Alliance*** (www.openurc.org) is to disseminate and promote the URC standard, to drive its continuous development, and to provide a communication and cooperation platform for the growing community of organizations and developers applying the URC framework. The impact of implementations will be maximized by sharing resources and following common guidelines. Everybody can join this community to help and build a URC ecosystem that will facilitate simple, flexible, and accessible user interfaces.

A key objective of our strategy is to avoid 'reinventing the wheel'. Hence, the openURC Alliance is seeking cooperation with complementary standards as much as possible. For that purpose, the Alliance is developing relations with a series of other standardization initiatives, e.g. OSGi, Zigbee, Z-Wave, enocean and digitalstrom, in order to establish cooperation agreements that seek the creation of synergies on a technological, marketing and political level.

At the current stage the Founding Members of the openURC Alliance are Trace Center, University of Wisconsin (USA), DFKI (Germany), ATG (Germany), dotUI (USA), Meticube (Portugal), VICOMTech (Spain), the Czech Technical University (Czech Republic) and Georgia Tech (USA). A membership model and conditions have already been ratified. Currently the focus is laid on the constitution of a legal entity that will act as the formal basis of the Alliance.

The legal entity will be constituted in the first quarter of 2011 as a not-for-profit organization. For that purpose we are supported by several international bodies, including the European Commission. If you are interested to learn more about the openURC Alliance and getting a member, please contact us at info@openurc.org.

8 FUTURE WORK

The OpenURC Technical Committee is working on technical specifications that provide precise descriptions of implementation aspects of the URC ecosystem. Examples of these technical specifications are the UCH specification, the URC-HTTP protocol specification [5], and the resource property vocabulary [6]. More recently, the Technical Committee has taken on work for a tight integration of web services and the URC technology, to allow for easy-to-develop, personalized user interfaces for Web services. Also, an overall security concept for user interface resources is being worked in, mandating any resource in the URC ecosystem to be digitally signed by its publisher.

For an ecosystem to be fully functional, various tools need to be available for authors, publishers and installers. Plans for the development of various tools are being pursued in existing projects and those for which applications are under way. Examples of such tools are URC runtime support for Android, plug-ins for mainstream application development frameworks and workbenches like Eclipse, MS Visual Studio and Flex and diagnostic and reporting tools, to mention just a few.

ACKNOWLEDGEMENTS

Work on the URC framework was funded by EU 6th Framework Programme under grant FP6-033502 (i2home); and by the US Department of Education, NIDRR, under Grant H133E030012 (RERC on IT Access). The opinions presented in this paper are those of the authors and not necessarily those of the funding agencies.

REFERENCES

[1] ISO/IEC 24752:2008. Information technology – User interfaces – Universal Remote Console. International standard in 5 parts. ISO/IEC, 2008.

[2] URC Consortium: Universal Control Hub. Latest specification: http://myurc.org/TR/uch/

[3] URC Consortium: URC Technical Primer 1.0, Draft Technical Report. Latest version available at http://myurc.org/TR/urc-tech-primer1.0

[4] Zimmermann, G. and B. Wassermann (2009). Why We Need a User Interface Resource Server for Intelligent Environments. 4th International Workshop on Artificial Intelligence Techniques for Ambient Intelligence (AITAmI'09). Paper to appear in: Augusto, J. C. (chief editor): *Ambient Intelligence and Smart Environments*. IOS Press, Amsterdam, The Netherlands. Paper available online: http://www.accesstechnologiesgroup.com/pubs/ZimmermannWassermann2009-AITAmI'09

[5] URC Consortium. URC-HTTP Protocol 2.0, Draft Technical Report. Latest version available at http://myurc.org/TR/urc-http-protocol2.0

[6] URC Consortium. Resource Property Vocabulary 1.0, Draft Technical Report. Latest version available at http://myurc.org/TR/res-serv-http1.0