openURC: Standardisation towards "User Interfaces for Everyone, Everywhere, on Anything"*

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

[1] DFKI GmbH, Saarbrücken, Germany
[2] Meticube Lda, Portugal
[3] VICOMTech, Spain
[4] Czech Technical University, Czech Republic
[5] INGEMA - Instituto Gerontológico Matia, Spain
[6] TRACE R&D Center, University of Wisconsin-Madison, USA
[7] University of Tübingen, Germany

Abstract. As a consequence of the large interest in standardized, open, scalable and interoperable platforms that can implement scenarios and requirements posed by the AAL and eInclusion communities, the ISO/IEC 24752 standard Universal Remote Console (URC) has emerged from a single project infrastructure within the FP6 eInclusion project i2home into a steadily growing ecosystem with parties ranging from more technology oriented over business but also stakeholders that represent users. In order to coordinate and make transparent activities as well as provide a meeting place for different stakeholders, we have started building up the OpenURC Alliance. This paper presents the URC technology, the timeline and construction of the Alliance and, finally, point at some current and future activities.

Keywords: OpenURC, Alliance, Platform for Accessible User Interfaces, ISO 24752.

1 Introduction

One of the holy grails of Artificial Intelligence is to implement intelligent environments where persons are intelligent supported by the environment in various ways. We witness instant improvements but at the same time a diversification in different areas where different stakeholders focus on and implement specific aspects of such environments, such as *intelligent assistance while driving* which has additionally become an affordable technology. The development within other areas of daily living, e.g., smart homes,

* This work has been partially funded by the European Commission under the grant FP6-033502 (i2home) and FP6-030600 (VITAL), the Saarland funded SensHome project, the BMBF funded project SmartSenior and by the National Institute on Disability and Rehabilitation Research (NIDRR) US Dept of Education grant # H133E090001/ H133E080022 to the Trace R&D Center, University of Wisconsin-Madison. The opinions herein are those of the author and not necessarily those of the funding agencies.

or smart working places have not advanced in the same way. Really smart homes are today either practically nonexistent in large parts of the world or very expensive. The latter hinges on the fact that we still lack agreed-upon technological platforms that are capable of implementing the diversity of technology needed for realizing smart homes. Moreover, the "smartness" of the smart home is limited to the smart home itself and typically vanishes as soon as we leave our home and close the door.

Imagine instead a world where your 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 smart phone and use it to switch to your favourite news channel. Even though all products and systems in the room are new to you, they or more precise the interaction with them is familiar since you use the same interfaces on the smart phone 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.

Imagine additionally that you are buying to your home, a new TV since your old one broke along with a falling detector service since you have lately started to suffer from dizziness. Rather than throwing away your old remote and learn a new one, you simply replace the old TV with your one. The user interface remains the same as before. The falling detector service contains four sensors that you can even mount yourself in your flat and that seamlessly integrates with your existing environment. The appropriate user interface and additional software is installed automatically in your smart home.

Imagine finally 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, more relevant even, at their very homes. They too would become more autonomous.

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

1. services that are truly ubiquitous and/or pervasive;
2. architectures that can integrate new services and corresponding user interfaces on the fly as they are discovered
3. user interfaces that can be personalized, namely, being adapted to the capabilities 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. Indeed, we find well-functioning island solutions—there are in fact several which are concerned with controlling light and heating, e.g., Zigbee[1], enocean[2], knx[3] just to mention a few. However, since these

[1] http://www.zigbee.org
[2] http://www.enocean.com
[3] http://www.knx.org

technologies do not work together, or are unable to work with other equipments or services, the world increasingly become a sea of island solutions.

On a political level, Europe has started to implement the "Convention on the Rights of Persons with Disabilities"[4]. This convention is based on eight principles of which two are particularly relevant for the work described here: "3. Full and effective participation and inclusion in society" and "6. Accessibility". These principles have or will have very large impacts on different parts of the society not only in the areas of architecture and city planning but also on Information and Commmunication Technologies (ICT). Consequently, information and services made available by particularly government but also ideally hotels, shopping malls etc will have to be made accessible for everyone. This work describes an open, scalable architecture based on industrial standards that meets all these requirements and desires.

2 The Universal Remote Console Standard

Technology enabling wireless connectivity and networked computing is already available, providing methods for seamless discovery, controlling and eventing. But at the moment, 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-based UIs [6], voice recognition and natural language technologies used by PDAs and wearable computers, or even Brain Computer Interfaces (BCI) [7]. Such an approach could also enable older products to be controlled with new user interface technologies, e.g., speech-based UIs [9].

In early 2008, subcommittee SC 35, User Interfaces of ISO/IEC JTC1, Information Technology, published a new multi-part International Standard, promoting interoperable and personalizable user interfaces, ISO/IEC 24752, Information technology—User Interfaces—Universal Remote Console (URC), see [2].

One intuitive view point of the URC technology is the middleware approach called Universal Controle Hub (UCH) as depicted in figure 1. In line with many other middleware solutions, the UCH includes the heterogeneous appliances and services, called *targets* and make them available as abstract user interfaces ("UI Socket Layer" in the figure). A socket can be thought of as an abstract description of the input/output behaviour of the target. Furthermore, we say that socket is rendered or made concrete by typically running some user interface on some *controller*. 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

[4] `http://en.wikipedia.org/wiki/Convention_on_the_Rights_of_Persons_with_Disabilities`

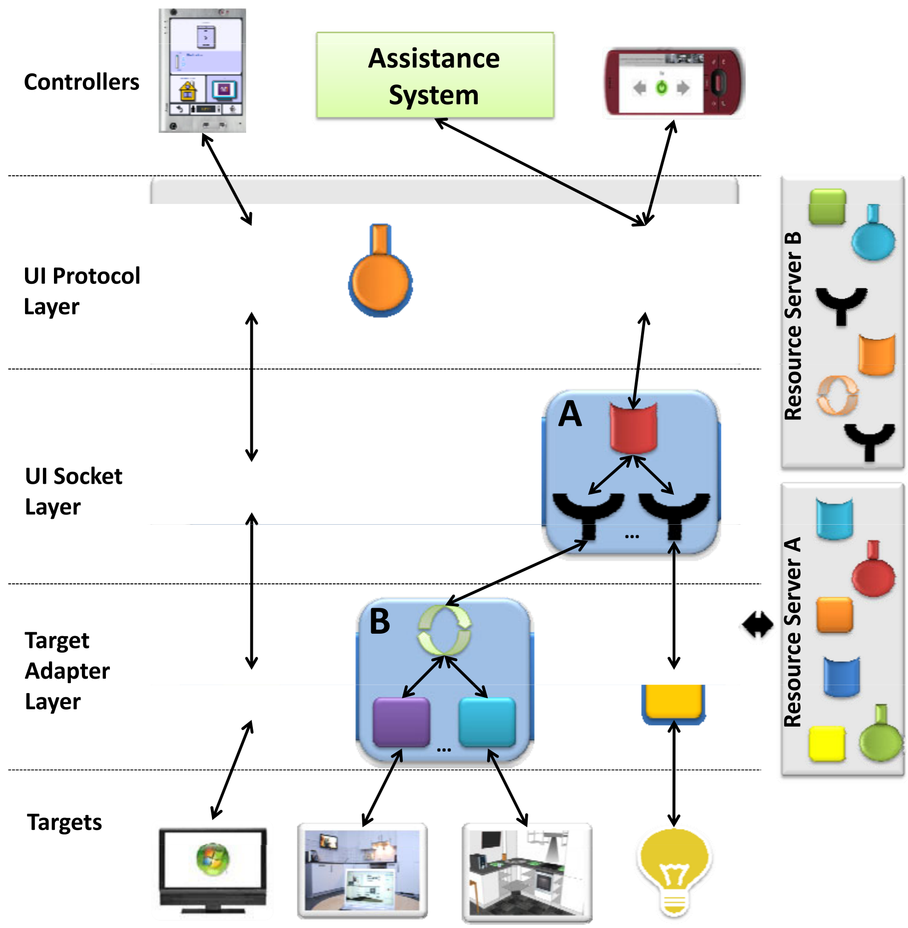

Fig. 1. The URC framework with its different layers (from the bottom): Target Adaptor Layer, Socket Layer, UI Protocol Layer. In the figure, two additional plugin extensions are depicted: A) A CEA-2018-based Task-Model Engine; B) A Synchronization Module used for implementing synchronized scenarios.

high-level interaction. A key part of this interaction is the sharing of control and access information through XML documents.

The URC technology does not pose any requirements on the targets other than being controllable over a network. Currently, UCHs can interact with UPnP devices, ENOCEAN, WSDL services, ZigBEE, bluetooth just name a few technologies. Integrated user interfaces range from simple automatically generated HTML pages, accessible HTML pages, flash-based UIs, multimodal speech- and gesture-based UIs. Appropriate controllers have been iPhones, Android phones, TV + Remote Control, touch-screen based computers and even Brain-Computer Interfaces (BNCs) etc. The URC technology has been applied in diverse scenarios, like comfort, security, information services, peer-to-peer gaming within the Smart Homes, i2home implementations of the UCH approach additionally allows for plugins, e.g., a CEA 2018 task model engine [8] which can be used to combine services forming compositions thereof.

The two USPs

Apart from the separation of concerns which allow user interface developers to focus on the development of user interfaces rather than technical issues, there are two unique selling propositions of the URC technology not present in most other frameworks:

– *Pluggable user interfaces* meaning that there is a clear separation between the appliances and services on the one hand and the user interfaces on the other. With this approach it is possible to dynamically install UIs with the same functionality depending on the user's profile and the context.
  We strongly believe that this approach should be incorporated into ICT in general since it provides the necessary ingredients for personal and thus potentially accessible user interfaces.
– *Resource Servers* A resource necessary for connecting a target with a UI is available via resource servers. In the last years, special forms of this concept have received a lot of attention, such as Apple's App Store. The URC approach, however, allows sharing arbitrary resources necessary for connecting a target with a user. Also, a UCH can interact with several resource servers thus allowing providers to maintain their own resources on own resource servers.
  The Resource Server is instrumental in providing the benefits of the URC framework regarding user interfaces, such as personalization, accessibility, context-awareness, openness for 3rd-party contributions, support for agent-based user interfaces, and support for management of user interfaces [10]. Currently, a pilot Resource Server is being operated by the US-based company dotUI, see `http://www.dotui.com`

Current status

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 decided to apply it, both in a research context or for the development of real products or services. More is already in the development pipeline:

The central element for any URC driven infrastructure is the already mentioned UCH. The TRACE Center distributes Open Source reference implementations 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 to interact with devices and services which are connected to a remote UCH and that supports a wide range of deployment scenarios (hierarchic, mesh, P2P, proxying, ...).

DFKI provides an implementation based on the OSGi platform. This implementation is scheduled for being re-implemented according to the common criteria methodology in order to receive a level four certificate. This correspond to the security level of today's home-banking systems. Furthermore, DFKI has developed a rapid-prototyping tool for drag-and-drop creation of UIs.

Furthermore, Meticube's .NET implementation provides a range of tools and add-ons, like an Activity Management Extension Kit (EK), 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 that allows to use MS MCE both as Target and Controller. The Avatar EK allows designing and deploying 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, both on the Controller and Target side.

From 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 solely based on JavaScript.

A large number of additional resources are available, such as several UI Protocal implementations, e.g., URC-HTTP modules, TAs for ENOCEAN, MonAMI residential gateway (including ZigBEE), google calendar, PICASA, Windows Media Center, etc.

3 Injecting the URC Ecosystem

During the course of the project, i2home [3] provided parallel running EU-funded projects, such as, the VITAL [5,11] and MonAMI [4] projects with access to the URC implementation. In fact, MonAMI's field trials (50 flats) include the i2home URC implementation. But also industry has started showing interest in the URC technology and so at this year's CEBIT, German Telekom demonstrated their Universal Service Gateway (USG) where the URC technology serves as a middleware for three different UIs (in car, PC-based and mobile-phone based) within a traveling scenario. In addition to India and the US, we also observe an adoption within academic research in Australia. Accumulated, we count projects using URC technology with about 120 partners and EURO 100 Mi, see figure 2.

This has encouraged us to initiate the international OpenURC Alliance, see `http://www.openurc.org` as an international platform for fostering the ecosystem that will be necessary for broadening the knowledge about and usage of the URC technology. The OpenURC Alliance provides a meeting place for different stakeholders interested in this technology.

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

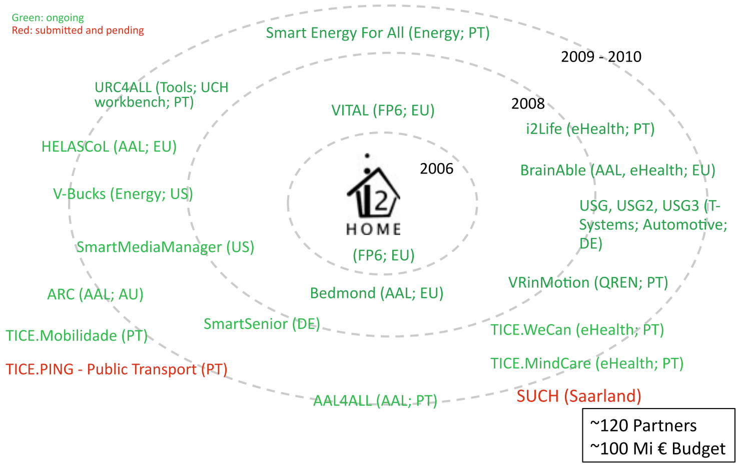

Fig. 2. International projects using URC technology.

possible. For that purpose, the Alliance is developing relations with a series of other standardization initiatives.

4 Structure and Timeline of the OpenURC Alliance

At the time of writing, the OpenURC Alliance consists of four Committees:

Governance: Here, the work consists of structuring the alliance as a whole. Also, work includes the writing of membership rules and By-Laws. We have developed four different membership categories: "Charter", "Core", "Associate" and "Basic" which all have different benefits. Additionally, experts might get invited in order to resolve or work on some issue(s).

Marketing: This committee develops material for presentation and marketing, organizes presentations at conferences and fares.

Technical: The technical committee further develops the URC technology by producing technical documents and perform standardization work. This committee also maintains open-source reference implementations and technical documentation of the URC standard, see section 5

User: We are currently developing a fourth committee called the User Committee whose purpose is to add to the alliance the perspective of the users, such as ethical considerations and how to execute user-centred design for persons with special needs, e.g., within gerontological scenarios. An important topic will be to host and develop personas [1].

4.1 Time Line

The OpenURC is already up and functioning as an ad hoc group performing meetings on weekly or bi-weekly basis. We are targeting a launch of the OpenURC Alliance as a non-profit legal body within 2011. The launch of a new web site is planned for late 2010. We are currently moving the content of the current Universal Remote Console Consortium web site `http://myurc.org` to the web site to the new new OpenURC Alliance web presence `http://www.openurc.org`.

5 Current and Future Technical 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].

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 projects that are being applied for. 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, just to mention a few.

We are currently working on an integration of the URC technology into the Web service world. Here, the goal is to allow for the design of personalized user interfaces and their substitution as web service front ends. Here, RESTful services and a WSDL interface play an important role.

Another vital challenge is the standardization of user interface sockets. Currently, an author can design their own user interface socket as needed. This results in different sockets for targets that may have many functions in common. Possible solutions is to publish "upper models" of prototypical sockets that can be used as a basis for new sockets. Extensions can be modelled by inheritance.

6 Conclusion and Outlook

We have argued and showed that the Universal Remote Console approach should play a role in any framework that takes accessibility seriously in the area of ICT. The URC approach provides an open standards-based fundamental functionality for any framework or platform where the diversity of users are to be supported and where accessibility is addressed. In order to provide a platform for different stakeholders within the AAL community to communicate and collaborate in this area, we are starting the OpenURC Alliance. The Alliance will provide a meeting place for providing information about the URC, exchanging ideas, meeting other stakeholders interested in the URC technology and, finally, manage open source resources.

[5] http://myurc.org/TR/urc-http-protocol2.0/

[6] http://myurc.org/TR/res-serv-http1.0/

References

1. Cooper, A.: The Inmates are running the Asylum: [why high-tech products drive us crazy and how to restore the sanity], 1st print edn. SAMS, Indianapolis (1999)
2. ISO. ISO/IEC 24752: Information Technology—User Interfaces—Universal remote console — 5 parts. International Organization for Standardization (2008)
3. i2home: Interaction for everyone with home appliances based industry standards (2006-2009), `http://www.i2home.org` (funded by the EU/FP6)
4. Monami: Mainstreaming on ambient intelligence (2006-2010), `http://www.monami.info` (funded by the EU/FP6)
5. Vital: Vital assistance for the elderly (2006-2010), `http://www.vital.org` (funded by the EU/FP6)
6. Smartsenior (2009-2012), `http://www.smart-senior.de` (funded by BMBF, Germany)
7. Brainable—autonomy and social inclusion through mixed reality brain-computer interfaces: Connecting the disabled to their physical and social world (2010-2012), `http://www.brianable.org` (funded by the EU/FP7)
8. Rich, C.: Building task-based user interfaces with ansi/cea-2018. Computer 42(8), 20–27 (2009)
9. Wahlster, W. (ed.): SmartKom: Foundations of Multimodal Dialogue Systems. Springer, Heidelberg (2006)
10. Zimmermann, G., Wassermann, B.: Why we need a user interface resource server for intelligent environments. In: 4th International Workshop on Artificial Intelligence Techniques for Ambient Intelligence (AITAmI 2009) (2009), `http://www.accesstechnologiesgroup.com/pubs/ZimmermannWassermann2009-AITAmI'09/`
11. Zinnikus, I., Fischer, K., Alexandersson, J., Diaz, U.: Bringing the elderly into the mainstream of e-society: the vital project. In: IADI e-Society Conference, Barcelona, Spain (2009) (outstanding paper award)