A Middleware Architecture for Designing TV-Based Adapted Applications for the Elderly

Carlos Gacimartín, José Alberto Hernández, and David Larrabeiti

Universidad Carlos III de Madrid
Avda. Universidad 30, 28911 Leganés, Madrid, Spain
{cgacimar,jahgutie,dlarra}@it.uc3m.es

Abstract. The elderly are begining to use more and more new technologies, although several times they find these are difficult to interact with, especially when they are accessed by a keyboard or the screen is too small.

This work proposes a middleware architecture for TV-based applications for the elderly, with a focus on social interaction services like webmail. The elderly feel very confident with their TVs since it is something they use on a daily basis, and the screen is sufficiently large. The architecture contains a Context Manager with an Ontology that models the elderly user impairments and actual context information collected from sensors located at the user's home, and an Interface Adaptation Engine which generates the adapted interface for a particular user, according to his profile and context of use. This architecture is part of an ongoing EU funded project: MyUI[1] and a first prototype is expected by mid 2011.

1 Introduction

In the last century, global life expectancy at birth has risen from 58 years in 1970-1975 to 67 years in 2005-2010, and is currently beyond 80 years for some developed countries, namely Japan (82.6 years, first ranked), Australia (81.2 years, fifth ranked) and Spain (80.9 years, sixth ranked). In developed countries, such an increase of longevity, together with the observed decrease of fertility, has a clear social impact: the elderly population is expected to nearly double by 2050 [1].

In light of this, Information and Communication Technologies (ICT) for the elderly are expected to increase their quality of life in the forthcoming years. The elderly are slowly embracing new technologies either because they are beginning to acknowledge their usefulness in daily tasks, but also because new technologies keep them close to the younger generations, especially their sons and grandsons. Also current generations which now demand technology will be technology consummers in the future. However, it is still the Television the preferred home appliance for the elderly. Indeed, it is the people beyond 65 who spend more

[1] http://www.myui.eu

J.A. Jacko (Ed.): Human-Computer Interaction, Part I, HCII 2011, LNCS 6761, pp. 443–449, 2011.
© Springer-Verlag Berlin Heidelberg 2011

time watching TV, with more than 294 minutes per day, 13 minutes more than the average TV consumption of people from all ages in Spain [2].

Hence, the combination of communication technologies and the Internet with TV will definitely open new services applications to people of all ages, and particularly to the elderly since the TV is the device they feel more confident with. A large number of TV vendors have forseen this opportunity and are producing TVs with Internet connection [3]. Additionally, several partnerships between TV vendors and Internet Service Providers have arisen to make true TV-based Internet services, see for instance Google TV with Sony and Logitech [4].

However, when developing ICT services for the elderly, a number of design requirements must be taken into account for its success, especially concerning accessibility issues in the Human Computer Interaction (HCI). Basically, the service presentation and navigation must be adapted to the aged user cognitive/visual/hearing profile of the user.

Indeed, elderly accessible software research is a hot research topic [6], see for instance Cybrarian [10], a webmail with a very easy-to-use interface intended for the elderly, but does not have interface adaptation. However, the use of a keyboard always hampers the adoption of ICT technologies by the elderly users. Ideally, a social interaction system for the elderly should be designed without the need to use a keyboard.

Take the webmail case for instance: A webemail without a keyboard is possible if the compose mail option is performed by recording the voice of the user. The user could navigate through the webmail on his TV and just press some buttons to record messages and send them to his family and friends. Additionally, font-size adaptation and even text-to-speech can be used for the hearing impaird users, where the emails font-size need to be increased or even read for the users.

In the following sections we propose a middleware architecture for the development of elderly-adapted applications, with a first focus on webmail applications, as part of the work conducted within the FP7 EU Project MyUI under code FP7-ICT-248606.

2 An Architecture for Services to the User

Fig. 1 shows a general system architecture for the design and implementation of web-based adapted services for elderly users. This architecture comprises three different parts: The Client side, the Context Manager, and the Middleware, which with a different role that is briefly described next.

2.1 The Client Side

The client side comprises the set of devices and sensors located near the user. These include:

- The Internet TV, which contains a web-browser that connects to the services allocated on the cloud. We assume that the elderly user accesses these services only with his TV remote controller.

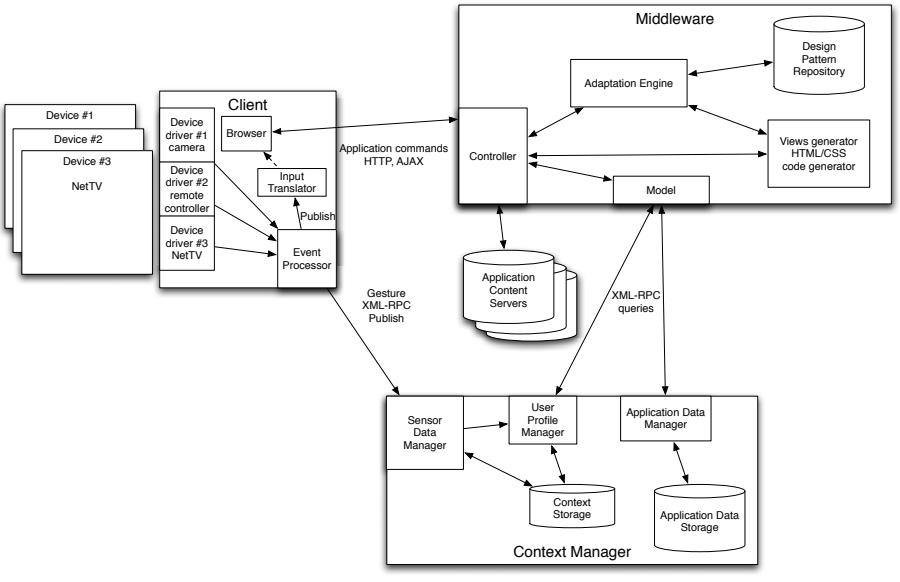

Fig. 1. System's architecture

- Hardware sensors, which are expected to capture real-time context information of the elderly user. Such information is first captured by the sensors' drivers and then published on the Context Manager using XML-RPC. Examples of important context information are: the user id that identifies who is accessing a given service (obtained from an RFID for instance), the user's distance to the TV, the current ambient light, etc. Such information is important in the generation of accessible interfaces.
- Software sensors, that are in charge of providing other inferred information about the user and his access to the service. For example, software sensors may measure the user's response time in accessing a given service or whether the user experiences difficulties in accessing the services. In summary, software sensors are expected to complement hardware sensors by inferring other contextual information about the user.

Hence, the client side not only must provide a front-end to access the adapted services, but also is expected to capture real-time context information about the user which helps in the generation of the personalised adapted interfaces. Such context information is first captured by the sensors, and then made available via XML-RPC on the Context Manager for further use by the Middleware.

2.2 The Context Manager

The Context Manager (CM) is the software component that stores both the user's profile and the measured context information. Both types of information

shall be used by the Middleware and its Interface Adaptation Engine for the generation of adapted user interfaces.

The information concerning the user's profile and his current situation is modelled and stored in the CM by an ontology. This ontology defines, in a conceptual way, the type of information necessary to make adapted user interfaces. In the case of adapted services for the elderly, two types of information are modelled and stored in the ontology, which follows the Open Ambien Assited Living (AAL) framework [7] format:

- User profile, which contains personal information about the user, for example visual acuity, color blindness, hearing impairment, etc. This information is static and does not change over time.
- Context information, which contains information about the actual context of the user, for example, current distance to the screen, etc. This information is more dynamic, and it is updated with real-time measurements.

Typically, two operations are offered by the CM to the other software components in the architecture: setValue and getValue, the former allows for an update on the ontology's information whereas the latter permits information retrieval from the CM. Technically, both operations are served as Remote Procedure Calls (RPC) from external applications, and the information is embedded within XML (this is the XML-RPC communication paradigm).

The communication flow with the client side follows the asynchronous publish/subscribe paradigm, at which the sensors publish updated information on the CM. A triple is necessary to identify every sensor event, namely:

(sensor id, measured property, measured value for that property)

After this information is updated in the CM, then it is made available to other software components, for instance the Middleware. The Middleware retrieves both user profile and context information from the CM before generating the any adapted interface.

2.3 The Middleware

The Middleware provides both the application logic and the interface adaptation engine in the whole system. The Middleware is structured following the Model View Controllelr (MVC) [8] approach which separates a typical web-based application into specific areas:

- The Model or data layer. This layer contains the information required by the application, including access to databases and file-systems. In our application, the model interacts with the CM by quering for the static user profile and the more dynamic real-time context information related to that user.
- The Views or presentation layer. This layer is in charge of generating the final HTML and CSS code to be sent to the browser of the Internet TV. The presentation layer comprises the Interface Adaptation Engine, which translates design patterns and users profile and context information into HTML and CSS code customised for a given user.

– The Controller or logic layer. This layer represents the operation workflow, which typically comprises several steps:
  1. The Controller asks for the information required to the Model;
  2. this further queries the Context Manager which contains the updated information of the user and his context, which
  3. is sent back to the Controller. Then,
  4. this information is passed to the Interface Adaptation Engine which generates the adapted interface for that particular user and his context. This information arrives at the Controller which just
  5. forwards it to the web-browser in the client side. A summary of these steps is given in Fig. 2.

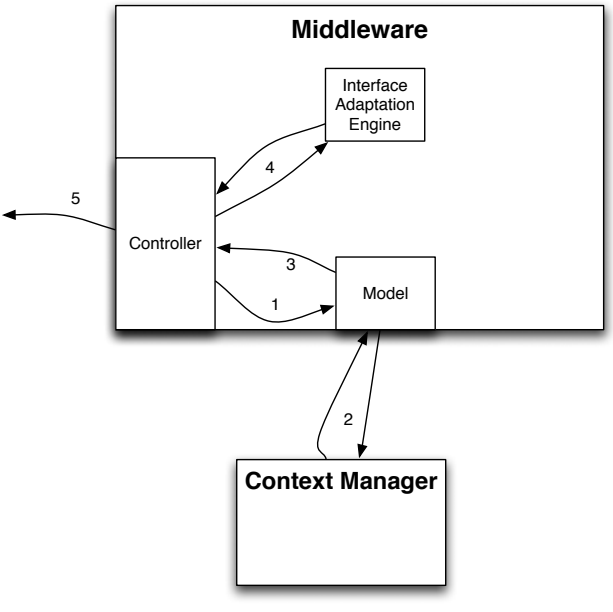

Fig. 2. MVC operation

Finally, the Controller also retrieves application-specific information from the Application Content Servers (Fig. 1). This refers to the contents of the application oriented for the elderly, for instance:

– Email application: The application contents in this case refers to the emails themselves.
– Physiotherapy exercise: In this case, the Controller needs to retrieve the flash videos that contain the specific exercises themselves.
– Cognitive games: The application information regards to the particular games along with their difficulty: Sudokus, Crosswords, Solitaire, etc.

Next, we examine how the CakePHP framework can be refined to provide such web-based TV services for the elderly.

3 Middleware Implementation with CakePHP

CakePHP is a native MVC framework that enables PHP developers to create robust web applications rapidly and reliably. CakePHP allows CRUD (Create, Read, Update and Delete) operations, necessary in most of web-based applications. By convention, urls in CakePHP always follows the same structure:

http://www.example.com/Application/Controller/Action/P1/P2/P3

which essentially calls some "Action" method from some "Controller" with some parameters "P1", "P2" and "P3". This url triggers all the steps in the Middleware: Contacts the CM and retrieves the user profile and context information, retrieves the application specific data and calls the Interface Adaptation engine with all this information to generate the HTML and CSS code adapted to the user.

Consider an email application, where an elderly user wants to read his emails after loging into the system with his username and password. In this case, the email controller would follow the next pseudo-code:

1. Identify the user, i.e. retrieve from the Context Manager using a XMLRPC call the RFID key of the user in front of the Internet TV. From that RFID code, the application has the necessary data for his session (login, email server, etc).
2. Retrieve the user's emails, namely, connect to the email server and retrieve the emails.
3. Text2Speech, that is, run some TTS code (for instance, the free Java-based TTS [9]) to convert the emails text to an audio format if necessary.
4. Dessign Patterns, in other words, apply a set of rules to translate the user's profile and his context into an appropriate interface with buttons, font-size, audio instructions, etc.

4 Summary and Conclusions

This work has proposed an architecture for a Middleware that provides adapted services to the elderly community. Essentially, the whole system contains:

1. An Internet TV and a set of sensors that measure real-time context information from the client.
2. A context manager, modelled by an ontology, which stores the user profile and the context information measured by the sensors.
3. A middleware platform that adapts the contents of some application with the context information obtained by the sensors.

The middleware follows the Model View Controller (MVC) approach which separates the data, presentation and logic layers. In our architecture, the Model refers to a set of functions that queries the Context Manager via XML-RPC, the

View comprises the Interface Adaptation Engine which generates HTML and CSS code from the model information and a set of pre-defined design patterns, and finally, the Controller handles all the operation workflow and logic, linking the application content, the model and the views. This architecture is currently under development within the EU-funded MyUI project, and a first version is expected to be released by mid-2011.

Acknowledgements

The authors would like to acknowledge the help of other partners from MyUI [5] in the development of this work, especially Matthias Peissner, Andreas Schuller and Dagmar Hbe from Fraunhofer IAO, Hester Bruikman and Roy van de Korput from Phillips Consumer Lifestyle, Peter Wolf and Oliver Strnad from Forschungszentrum Informatik, Ian Allen and Mark Ferguson from CleverCherry .com and Alberto Olmo, Víctor Sánchez and Alejandro García from ISOIN.

The authors would like to thank the support of the T2C2 Spanish project under code TIN2008-06739-C04-01 to the development of this work.

References

1. United Nations, Volume II: Sex and Age Distribution of the World Population "World Population Prospects, The 2008 Revision" (2008)
2. http://www.eldia.es/2011-01-03/sociedad/30-espanoles-vieron-television-media-minutos-persona-dia.htm
3. http://www.nettv.philips.com
4. http://www.google.com/tv
5. http://www.MyUI.eu
6. Sayaho, S., Blat, J.: Human-Computer Interaction with Older People: From Factors to Social Actors, Departament de Tecnologies de la Informaci i les Comunicacions, Universidad Pompeu Fabra (2009)
7. FZI Living Lab Ambient Assisted Living FZI Forschungszentrum Informatik, Germany
8. Rehman, K., Stajano, F., Coulouris, G.: An Architecture for Interactive Context-Aware Applications. Cambridge Syst. Associates (2007)
9. http://freetts.sourceforge.net
10. Dickinson, A., Newell, A.F., Smith, M.J., Hill, R.L.: Introducing the Internet to the over-60s: developing an e-mail system for novice Older computer users. Interacting with Computers, 1–22 (2005)
11. Age Concern, IT, Intemet and Older People, report on poll conducted for Age Concern by ICM Research, London, UK (July 2002), http://www.icmreseah.co.uk/reviews/2002/it-temet-old-peop1e.htm
12. Carmichael, A.R.: Style Guide for the Design of Interactive Television Services for Elderly Viewers, UK Independent Television Commission, London, UK (1999)
13. Hawthom, D.: Designing usable applications for older users an example, Conference, Melbourne, Australia (November 2002)
14. Rama, M.D., de Ridder, H., Bouma, H.: Technology generation and age in using layered user interfaces. Gerontechnology 1(1), 25–40 (2001)