

# Chapter 17

## Multi-Device Applications Using the Multimodal Architecture

Nuno Almeida, Samuel Silva, António Teixeira, and Diogo Vieira

**Abstract** Nowadays, users have access to a multitude of devices at their homes, workplaces or that they can carry around. Each of these devices, given its features (e.g., interaction modalities, screen size), might be more suitable for particular users, tasks, and contexts. While having one application installed in several devices might be common, they mostly work isolated, not exploring the possibilities of several devices working together to provide a more versatile and richer interaction scenario. Adopting a multimodal interaction (MMI) architecture based on the W3C recommendations, beyond the advantages to the design and development of MMI, provides, we argue, an elegant approach to tackle multi-device interaction scenarios. In this regard, this chapter conveys our views and research outcomes addressing this subject, presenting concrete application examples.

### 17.1 Introduction

Mobile devices, smart TVs, media centers, and game boxes are widespread and used on a daily basis by millions around the world. These devices are important means for providing users with applications and services that support many aspects of their personal and professional life. Multimodal interaction (MMI) is a common feature for many of them. Each of these devices presents a set of characteristics (e.g., mobile nature and available input and output modalities) that make them more suitable for particular contexts, and being able to seamlessly change from one device to another while performing a task, commuting among environments (e.g., from a smartphone on the street to a tablet at home), following the concept of migratory interfaces [1], would provide a desirable degree of flexibility and user adaptation.

In addition, multiple devices, if used together, can provide new means for interacting with an application, whether providing complementary features to a

---

N. Almeida • S. Silva (✉) • A. Teixeira • D. Vieira  
DETI—Department of Electronics, Telecommunications and Informatics, IEETA—Institute of Electronics and Informatics of Aveiro, University of Aveiro, Campus Universitário de Santiago, Aveiro, Portugal  
e-mail: [sss@ua.pt](mailto:sss@ua.pt)

single user or supporting collaboration among users. The output modalities should also be able to provide feedback in multiple ways, adapting to the current device ecosystem, with the information made available in one device potentially adding to the contents provided by other accessible devices.

Therefore, the coexistence of multiple devices and applications provides (and demands) new interaction possibilities, posing challenges regarding how different devices can be used simultaneously to access a specific application, taking the most out of each device features (e.g., screen size) and sharing input and output modalities.

Designing and developing a single application to run on multiple devices, taking advantage of each device characteristics, and harnessing the power of multiple devices simultaneously, presents a number of challenges. These concern decisions on where the application logic will be instantiated and how the different modalities are managed and used to attain a seamless and fluid interaction experience. The main goal is that the experience with the different devices and modalities blends to the point that what is in focus is the interaction with the ecosystem for which each of the devices can be seen as a nested or complex modality component [2]. For multimodality, we aim at providing a versatile experience through a set of modalities, used simultaneously or not, to widen the data bandwidth between users and systems. With multi-device we aim to serve the same purposes at a higher level.

While working on the design and development of multimodal applications, in a variety of scenarios provided, for example, by projects AAL4All,<sup>1</sup> Paelife,<sup>2</sup> Smartphones for Seniors,<sup>3</sup> and Marie Curie IAPP project IRIS<sup>4</sup> [3], we envisaged several contexts where multiple devices are available and can be explored to enhance interaction, and in which several users may interact simultaneously over the same contents making use of their personal devices.

The support for multi-device interaction is the focus of several works presented in the literature. From the early multi-device proposals, such as those by Rekimoto [4], to recent works, e.g., Diehl [5], the design approaches vary considerably and despite interesting results having been attained, serving multimodal multi-device purposes, most approaches propose their own architectures with a potential negative impact on their dissemination and compatibility. Proposing approaches that build on existing standards should be, in our opinion, and to the extent that those standards serve the targeted use cases, our first option.

In our work context, we approach multi-device support starting from the multimodality perspective and we argue that the MMI architecture, adopted for the different projects, and proposed based on the W3C recommendations, provides enough flexibility to integrate all the features that would enable multi-device support. This is also emphasized by our view, conveyed above, that a multi-device

---

<sup>1</sup> <http://www.aal4all.org/>.

<sup>2</sup> <http://www.paelife.eu/>.

<sup>3</sup> <http://www.smartphones4seniors.org/>.

<sup>4</sup> <http://iris-interaction.eu>.

setting can be viewed as a set of complex (or nested) modality components (the devices) used to interact with the macro application made available to users on more than one device. This approach would exhibit two advantages: (1) an approach to multi-device mostly performed at the architecture level, rather than at application level, potentially enabling easier multi-device support to the set of applications running over the MMI architecture; and (2) a standards-based approach, serving the goals of a loosely coupled architecture.

This chapter presents our views and proposals to support multi-device applications based on the W3C MMI architecture. After a brief contextualization of the architectural and technical background at stake for our work, in Sect. 17.2, we describe two different multi-device approaches in Sect. 17.3. Then, Sect. 17.4 is devoted to two application examples extracted from our ongoing work on ambient assisted living and interactive visualization. While the first example shows how multi-device support can be easily added to an existing multimodal application, the second is a first step in exploring multi-device visualization. Finally, Sect. 17.5 presents some conclusions and discusses routes for further development.

## 17.2 Background

Our work on multimodal applications spans over a wide range of applications covering services in ambient assisted living, such as personal life assistants [6, 7] and telerehabilitation [8], medication adherence [9], and a recent evolution of other lines of research into the MMI domain concerning autism spectrum disorders [10], and interactive visualization [11]. Our work also includes research on interaction modalities, such as speech [12, 13] and gaze [14], in an effort to provide what we call generic modalities [11–13]. In many of these works there is a rising opportunity and need of addressing multimodal multi-device interaction and this section provides a very brief account of one of the most recent scenarios we are considering for requirement elicitation, in line with the use cases proposed, for example, in [15], and summarizes relevant aspects of our previous developments regarding the adopted MMI architecture and its role on multi-device support.

### 17.2.1 *Research Context: Multimodal Multi-Device Scenarios*

One of the scenarios motivating our interest in further developing and exploring multi-device MMI, and a representative example of our research context, is provided by the Marie Curie IAPP project IRIS [3]. One of the main goals of IRIS is to provide a natural interaction communication platform accessible and adapted for all users, particularly for people with speech impairments and elderly in indoor

scenarios. The particular scenario under consideration, a household, where a family lives (parents, two children, one diagnosed with autism spectrum disorder, and a grandmother), and where different devices exist around the house, and are owned by the different family members, is the perfect setting for evolving multi-device MMI. In our view, communication can go beyond the exchange of messages through these media and profit from the dynamic multi-device ecosystem, where similar contents (e.g., a photo blog from the latest weekend activities or the family agenda) can be viewed in different manners, adapted to the device and user preferences, and supporting a collaborative interaction effort.

### ***17.2.2 Multimodal Interaction Architecture***

Our multimodal architecture [6, 12, 16–18], initially developed for AAL [6, 19], adopts the W3C recommendations and, as depicted in Fig. 17.1, can briefly be described as being composed of four main components [2, 17]: interaction manager (IM), data component, modality components, and the runtime framework.

The runtime framework provides the infrastructure to run the entire system, start components, and manage communication. Modality components enable the different inputs and outputs. The data component stores data related to the application and only the IM has access to it. The IM is a central component as it manages all received events from the modality components. It is a state chart configured through the State Chart extensible Markup Language (SCXML) [20].

The communication between components (or modalities) and the IM is performed by exchanging life cycle events, which contain Extensible MultiModal Annotation (EMMA) markup language [21], used to describe the events generated by modalities.

The implementation of the IM uses the Apache Commons SCXML<sup>5</sup> to manage the state machine defining the application logic. We extended the use of the SCXML to parse multimodal architecture life cycle events and trigger them into the state machine. The extension also includes the generation of the life cycle events to be transmitted to the modalities.

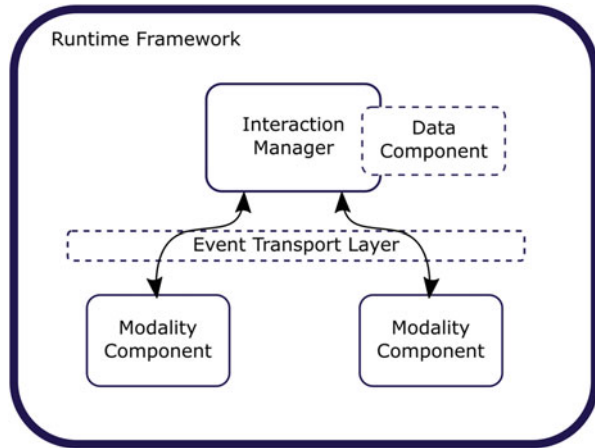
## **17.3 Multi-Device Support Using the W3C MMI Architecture**

Our work on making multi-device interaction possible started from a set of simple ideas we considered as grounds for our first experiments: (1) have a unique application, running in the various devices, to reduce development cost, integrating

---

<sup>5</sup> <http://commons.apache.org/proper/commons-scxml/>.

**Fig. 17.1** Main components of the multimodal architecture as recommended by the W3C (adapted from <https://www.w3.org/TR/mmi-arch>)



means to adapt to the device, user, context, and to the existence and status of other devices running the same application; (2) make information from each of the existing device modalities available to the applications running in other devices; and (3) rely on the W3C MMI architecture standards and on our implementation of the MMI architecture to accomplish all the required features.

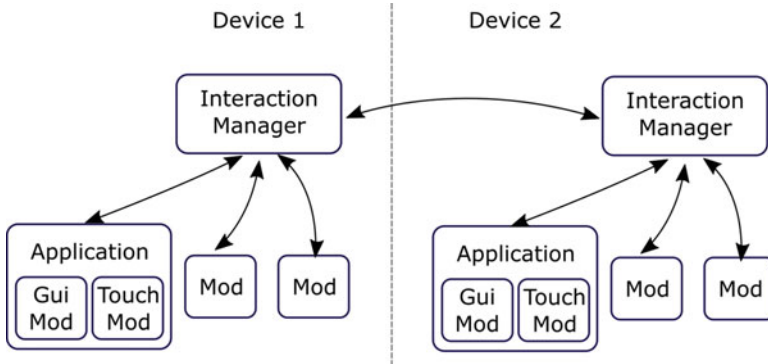
In our proposal, the IM module assumes the most importance. By using the IM as a pseudo-modality for applications running in other devices we were able to create a loosely coupled and extensible architecture that supports multiple modalities and the distribution of modalities across multiple devices, such as PCs, tablets, and smartphones. The architecture provides a flexible approach, allowing changing or adding modalities to the system without the other components being aware.

Two variants of the solution were proposed and prototyped, one considering an IM residing in each device, and other deploying the IM as a service, in the cloud, enabling the existence of a central IM per application, as detailed in the following sections.

### ***17.3.1 Per Device Interaction Manager***

Our first approach to multi-device applications considers that each device must run one IM. Figure 17.2 shows an illustration of a scenario with two devices, each running the application with a GUI modality, one IM, and any additional modalities, which can be different for each device. In this multi-device scenario, one IM behaves as a modality to the other. Following this approach, it is possible to disconnect the two devices and work with each device separately.

To enable each IM to discover the other, we use an UPnP server allowing all IMs to register their address. Each IM periodically sends broadcast requests to find UPnP servers with the service “MMI Discovery,” registers its address on it, and



**Fig. 17.2** Overview of the architecture supporting multi-device MMI considering an Interaction Manager for each device

obtains a list of existing IMs. From this point on, both IMs know of the other's existence and where to send the messages. Besides this process of discovery, all the communication between the IMs and the modalities is accomplished using the HTTP GET/POST protocol encapsulating the MMI life cycle events.

Only the IMs can exchange messages between devices. If a new event occurs in one modality of Device 1, this modality only sends the message to the local IM, which in turn, if it has information regarding the other device, sends the message to the other IM (Device 2). The IM for Device 2 processes that message as if it was sent by one of its modalities. Figure 17.3 presents an example of the messages between IMs and modalities after discovery.

Besides the simple registration of available devices, described above, a modality providing information regarding proximity among devices can be used in the multi-device application context as a trigger between single and multi-device use. As a proof of concept, and considering the scenario of a living room with a main unit (Home computer+TV) and a portable unit (tablet), we developed a proximity modality using wireless RSSI that computes the approximate distance between a tablet and the access point (placed near the main unit). Although this measurement is not accurate, it serves the purpose of identifying if the user is near or far away. Additionally, when the user is in front of the main unit, a Kinect is used to compute the distance between the devices. Proximity data is sent to the local IM that informs the other(s). With this information, the IM decides whether to use both devices for interaction. Figure 17.4 illustrates a scenario where the user is in different locations, enabling or not the use of the multi-device mode.



Fig. 17.3 Illustrative example of the communication between components for an MMI multi-device scenario. The touch and speech events, issued on one of the devices, are propagated to the other device through the Interaction Managers

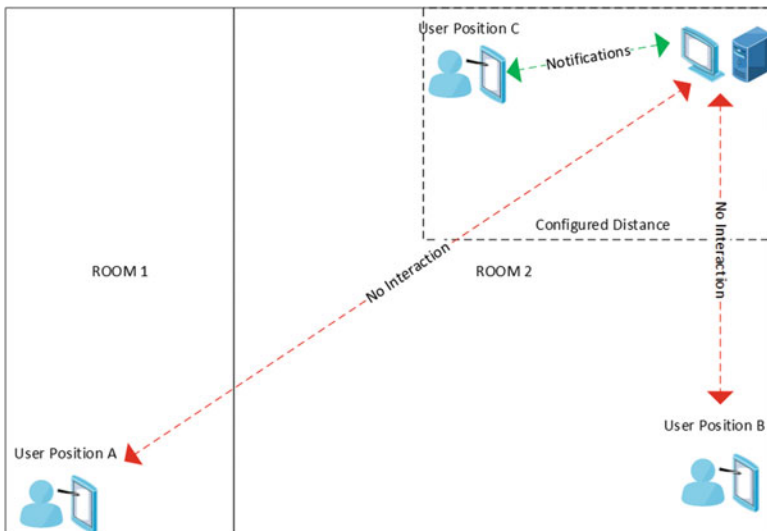
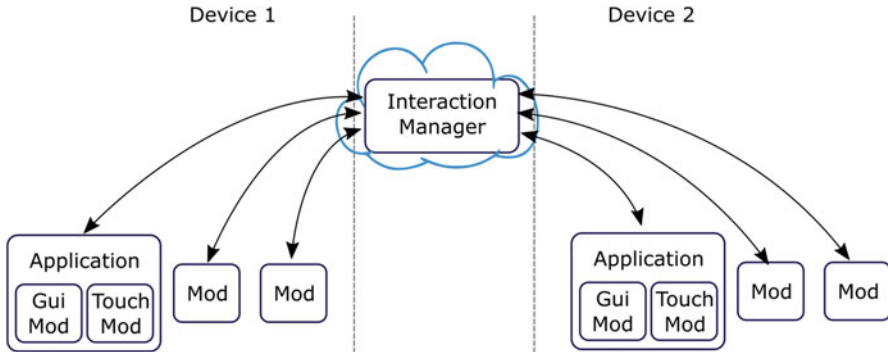


Fig. 17.4 Proximity modality in use illustrated for a fixed and a mobile device. The devices only enter multi-device mode when they are positioned near each other

### 17.3.2 Cloud-Based Interaction Manager

The approach described above allowed, to some extent, the use of multiple devices to improve the access to a particular application, profiting from all the available



**Fig. 17.5** Overall architecture for multi-device MMI supports using a single Interaction Manager located in the cloud

modalities and enabling a richer and more versatile output modality. Nonetheless, regarding the technical aspects of deploying these applications, a few issues arise, particularly for Microsoft Windows platforms. When making applications available through the Microsoft App Store, one of the limitations is that those applications, when installed, cannot communicate to internal services running on the device. Therefore, communication with the service providing the IM was compromised, which led us to move into a solution with the IM located externally, in the cloud. This change in how our MMI architecture was implemented yielded a more generic approach to multimodal multi-device interaction.

In this novel proposal to creating multi-device applications, only one central IM was considered, located in the cloud, and capable of managing multiple devices and multiple clients. To enable multiple clients in the central IM, each modality registers in the IM with a unique identifier. Figure 17.5 presents the overview of the target architecture.

This approach is more generic and can encompass a larger number of devices with less complexity than the first approach without a central IM. Furthermore, despite the overall differences between the two approaches, the way the modalities and IM communicate is the same, i.e., the same life cycle events, containing the same EMMA markup. So, applications adopting the first approach to multi-device support can be easily migrated to this more versatile solution.

## 17.4 Application Examples

The following sections illustrate how the described features have been used in the context of two different applications, each using one of the alternative approaches described earlier. The first example builds on our work for ambient assisted living. It enables the use of a personal life assistant taking advantage of multiple



devices to provide additional display space for accessing information by considering devices in the user's proximity. The second example concerns data and information visualization and is aimed at being an experimental platform to explore multimodal multi-device interactive visualization. Instead of the usual setting of custom applications to support collaborative multimodal interactive visualization and analysis, we argue that such features can be supported at the architecture level, enabling a more generalized use of such features among everyday applications.

### ***17.4.1 AAL Device-Rich Scenarios: A Multi-Device Personal Life Assistant***

This first example illustrates how an existing multimodal application, a personal life assistant [6] providing a set of modules such as news, weather, and a messaging hub, was evolved to support multi-device features.

The requirements for this application include not only the ones initially considered for the AALFred assistant [6], regarding the support for MMI including speech, touch, and gestures, but also new requirements were determined to create a multi-device experience. The application should be capable of running independently, connect among devices running it, change between autonomous or joint use, based on proximity, and allow showing the same or alternative content in each of the devices. Furthermore, the interface should be as similar as possible in both units, to minimize the need for additional learning.

In the first stage, only the news module was addressed, as a proof of concept for the multi-device features, updating the single device news module developed for the AALFred assistant. We considered, as the typical multi-device scenario, a static main unit (fixed computer) connected to a television and a mobile unit (tablet or smartphone), each working independently but simultaneously interoperable. In this multi-device scenario interaction can potentially be performed in three different ways: (a) through the main unit; (b) through the mobile unit; and (c) through both the main and mobile units.

Interaction through the main unit means that interaction is performed using only the modalities made available by the fixed-position device. In our prototype there are two main interaction modalities: voice (body), gestures and graphical output. In the interaction through the mobile unit, the user will interact by only using the modalities available on the tablet, particularly touch and graphical output. These two ways of interacting with the application are the typical single device scenario, although one should note that it is actually the same application running on each of the devices and not separate custom versions.

Interaction considering the two units takes advantage of the interaction capabilities of both devices to improve the usability of the system and implement new features. For example, when detecting that the user is within the range of the main unit, the application can allow using the main screen to visualize content while using the tablet as a controller.

In the news module there are three main information components that users can access: a list of the available news, an image illustration for the news content, and the news text. Considering these components, there are several content combinations possible when in the presence of two devices. For example, and without loss of generality, for a large TV set and a tablet, three combinations were considered, as depicted in Fig. 17.6: (1) TV and tablet showing the news content; (2) TV showing the image illustration in full screen and the tablet showing the complete news content; and (3) TV showing the whole news content and the tablet showing the list of news, serving as a news navigation device. If one of the applications is set to display only the news list and it starts working alone (the other device is not near) the application automatically reverts to working in single device mode.

This example depicts our first experiment with multi-device support. One of the first aspects that should be noted is that it consisted in adding multi-device capabilities to an existing multimodal application and, despite the changes to the IM, only minimal adjustments were required to the application, mostly concerning the extension of the output modality to support the different modes. In this example, even though, from the technical perspective, it would be possible to do so, we did not address the use of different input modalities connected to both devices, and the main unit was mostly used for its output capabilities. Although this example is for a particular scenario of two devices, this approach can be considered for any number of devices. Nonetheless, as previously mentioned, it would not serve the purpose of making the application available in the Windows App Store.

### ***17.4.2 Collaborative Data and Information Visualization***

The use of natural and MMI in Visualization, still a rather unexplored field of research [22], e.g., based on speech and touch, might bring advantages at different levels. The use of multiple interaction modalities can help bridge the gap between visualizations and certain audiences, by providing, for example, alternatives for the less technologically savvy, improving the visibility of certain aspects of the data, or by ensuring a richer communication channel between the user and the application. By supporting a multitude of interaction options, a system can also favor a more versatile scenario when it comes to the analysis of the data, enabling an active search for insight that otherwise might not have been foreseen by the interface designer [22]. In this regard, it is important to explore and understand the strengths and weaknesses of multimodality when used in the context of Interactive

**Fig. 17.6** Multiple ways to present the information available for a news content in a multi-device scenario including a TV and a tablet. From *top to bottom*: (a) both devices show the same content; (b) the TV shows a large image and the tablet the complete news contents; and (c) the TV shows the complete news contents and the tablet shows a news navigation menu



Visualization [23], exploring the potential advantages deriving from a richer interaction scenario, allowing adaptability to different contexts [24], and a wider communication bandwidth between the user and the application [25]. Furthermore, deriving from the wide range of devices available (smart TVs, tablet, smartphones, etc.), it is also relevant to explore how these multiple devices might be used to support visualization [26], whether individually, providing views adapted to the device characteristics [24], or simultaneously, providing multiple (complementary) views of the same dataset [27], fostering a richer interaction experience, or as the grounds for collaborative work [28].

To this purpose, we started the development of a prototype application that should allow for exploring the different challenges of collaborative multimodal interactive visualization [11] and how the MMI architecture could serve its requirements.

The application context that served as grounds for the prototype was inherited from our work on the proposal of evaluation frameworks for dynamic multimodal systems. Dynamic Evaluation as a Service (DynEaaS) [29] is a platform that supports the evaluation of complex multimodal distributed systems. The platform collects all data concerning the users' interaction with a system. The collected data is organized in a hierarchical form, according to the application components. Some insights of the users' performance with the application can be extracted by analyzing this data. Considering the amount and complexity of the resulting data, particularly in evaluation scenarios with complex tasks and several participants, it is important to create visualizations of the data allowing experts to interact, explore, and discuss the data.

To guide the design of the prototype application, we settled on a basic context scenario. In a meeting room, equipped with a TV connected to a computer, three experts meet to discuss some results of a previous system evaluation session. Each expert has a device capable of running the visualization application (each supporting multiple input and output modalities). One of the users is interacting with the TV, another user has a smartphone, and the other a tablet. The visualization modality adapts the default view to the screen size of each device. Also, users can choose a different visualization to be used in their device only. The outcomes of any user interaction over the visualization, through any of the available input modalities, are reflected in what the other users are seeing in their devices.

In the described context, the initial requirements for our prototype included: (1) Visualizations using different data representations, i.e., showing the same data but in a different way; (2) Multiple devices, adapting visualization to the screen size; and (3) Collaborative interactive visualization in a multi-device scenario.

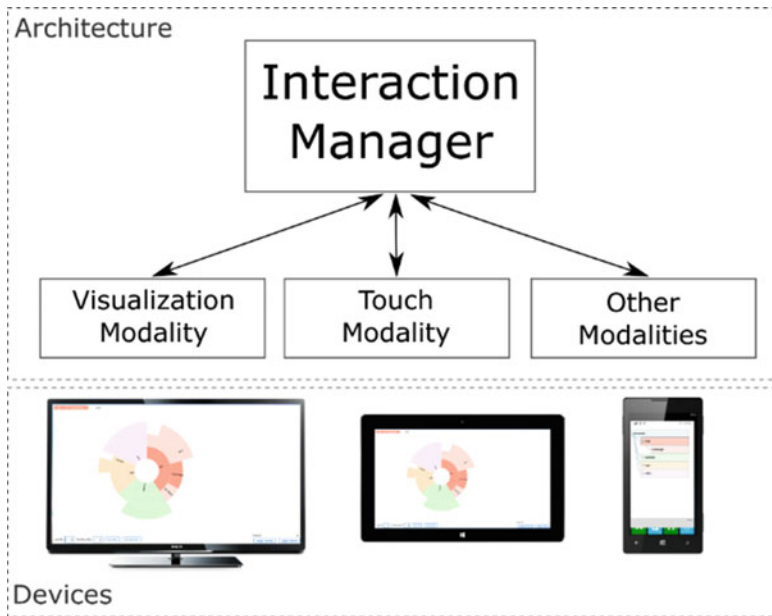
The visualization system adopts the multimodal framework and, in this approach, only one IM is used (see Fig. 17.7) and is responsible for managing all the life cycle events coming from all devices. At this stage, we managed all the visualization modes as part of a single modality, the touch modality is part of the application, and the remaining modalities are allowed to connect to the framework.

One additional modality was created that took into consideration a smartphone-specific capability, using the accelerometer to detect the smartphone motion. The user can rotate the smartphone 90° to the right or left to navigate through data.

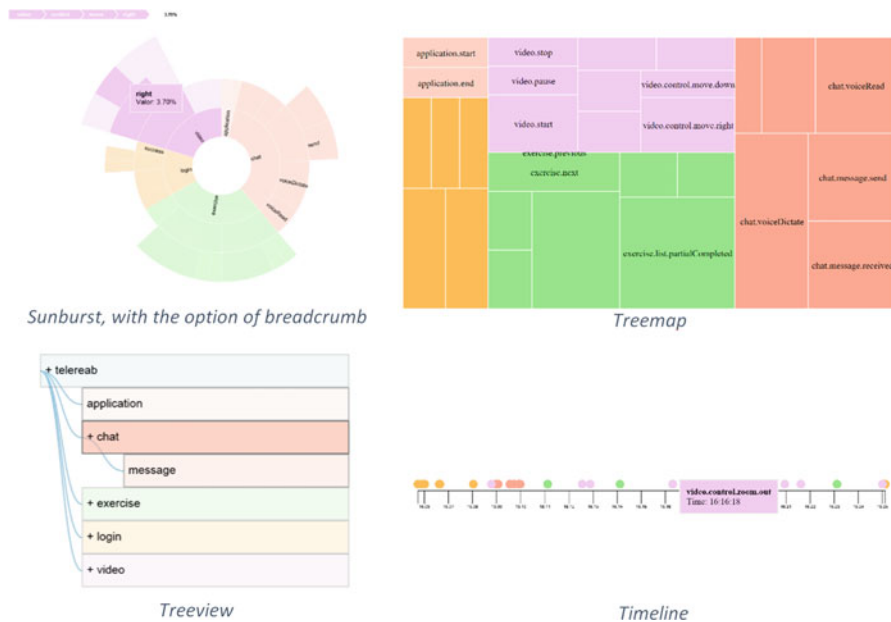
Following on the environment used in previous works, the application supports devices with Microsoft Windows, either desktop, tablets, and smartphones. To implement the visualization the D3js framework [30] was used as it naturally provides a large set of data and information representations.

For this first instantiation of the prototype, we opted for four different data representation alternatives, as depicted in Fig. 17.8: the sunburst visualization with breadcrumb and tooltips, the treemap, the treeview, and the timeline with tooltips. The consideration of the treeview, in particular, was meant to offer a compact representation that could be used in a device with a small display such as a small smartphone.

At its current stage, the prototype already illustrates some of the basic features we deem important as a proof of concept for a multimodal multi-device interactive



**Fig. 17.7** Overview of the multi-device Visualization application architecture. Multiple devices, allowing multimodal interaction with different representations of the considered information, are connected to a central Interaction Manager located in the cloud



**Fig. 17.8** Different data representations supported by the multi-device visualization application

visualization tool. By using the architecture to support the main features regarding the coordination between applications and the propagation of interaction, we place a complex aspect of such systems outside the application, yielding easier application development. In fact, from our point of view, this can be a first step towards a more general approach to multi-device support, where any application running over the multimodal framework can support, by default, multi-device features. One of the innovative features of using this second approach to multi-device is the deployment of the IM in the cloud, instead of an IM instance in each device.

## 17.5 Conclusions

Supporting interaction based on multiple devices is, we argue, fundamental to tackling the dynamic, device-rich interaction scenarios that have become so common nowadays. Supporting this feature at architecture level, as proposed, provides a simple and elegant approach that moves most of the need to support such features from the application into the architecture. In our view, such approach is critical to enabling widespread consideration of multi-device interaction not only in very specific applications, but also as a general feature available through the MMI architecture to all applications.

In our proposal we consider an approach to multi-device support where the IM is responsible for registering the available modalities. As to the discovery of modalities, in our first approach, we assume each device has an IM and finds other IMs in the network through an uPnP server; in a second approach, we consider a central IM located in the cloud and every modality connects to it. Regarding discovery and registration, it is worth noting that the W3C has recently published a first working draft for the discovery and registration of multimodal modality components [31].

As what is described in this chapter consists mainly of the adopted principles and some initial proofs-of-concept, there are several aspects that need to be addressed to attain the full extent of the desired multimodal multi-device interaction capabilities. A first line of work must address the scalability issues of the IM in the cloud. Another important aspect that should be further explored is the expansion of the output modality. For example, the graphical output modality should become more complex and autonomous to adapt to different devices and layouts in line with what we proposed for a complex speech modality [12]. New research is also needed to provide information regarding proximity between devices for the version with the IM in the cloud.

Finally, to get the most out of the multi-device capabilities, one could envisage complex interaction patterns considering, for example, that modalities made available by different devices can be used together. For instance, the user points to a large screen, equipped with a Kinect, and says “Show me this,” with the speech recognized by her smartphone. Or several users may be in a room, analyzing the animation of a dynamic dataset, but given various computational and display characteristics of their personal devices, different representations are used for

each (e.g., high resolution, graphical, text, and mixed). For both these examples, the timing for the different events assumes high relevance, and the importance of synchronization among the different devices and modalities is made clear. Addressing this challenge would largely benefit the evolution of multimodal multi-device interaction.

**Acknowledgements** The work presented in this chapter has been partially funded by IEETA Research Unit funding (Incentivo/EEI/UI0127/2014), Marie Curie IAPP project IRIS (ref. 610986, FP7-PEOPLE-2013-IAPP), project PaeLife (AAL-08-1-2001-0001), and QREN projects Smart Phones for Seniors (S4S), AAL4ALL and EMIF—European Medical Information Framework (EU FP7), co-funded by COMPETE and FEDER.

The authors thank all W3C MMI recommendations contributors for their insightful and inspiring approaches to MMI.

## References

1. Ghiani, G., Polet, J., Antila, V., & Mäntyjärvi, J. (2015). Evaluating context-aware user interface migration in multi-device environments. *Journal of Ambient Intelligence and Humanized Computing*, 6(2), 259–277.
2. Dahl, D. A. (2013). The W3C multimodal architecture and interfaces standard. *Journal on Multimodal User Interfaces*, 7(3), 171–182.
3. Freitas, J., Candeias, S., Dias, M. S., Lleida, E., Ortega, A., Teixeira, A., et al. (2014). The IRIS project: A liaison between industry and academia towards natural multimodal communication. In *Proceedings of Iberspeech, Las Palmas de Gran Canaria, Spain*, pp. 338–347.
4. Rekimoto, J. (1998). A multiple device approach for supporting whiteboard-based interactions. In *Proceedings of the Conference on Human Factors in Computing Systems (CHI'98), Los Angeles, CA*, pp. 344–351.
5. Diehl, J., & Borchers, J. O. (2014). *Supporting multi-device interaction in the wild by exposing application state*. (PhD thesis, No. RWTH-CONV-144030). Aachen:Fachgruppe Informatik.
6. Teixeira, A., Hämäläinen, A., Avelar, J., Almeida, N., Németh, G., Fegyó, T., et al. (2013). Speech-centric multimodal interaction for easy-to-access online services: A personal life assistant for the elderly. In *Proceedings DSAI 2013, Procedia Computer Science, Vigo, Spain*, pp. 389–397.
7. Hämäläinen, A., Teixeira, A., Almeida, N., Meinedo, H., Fegyó, T., & Dias, M. S. (2015). Multilingual speech recognition for the elderly: the AALFred personal life assistant. *Procedia Computer Science*, 67, 283–292.
8. Teixeira, A. J. S., Pereira, C., Oliveira e Silva, M., Alvarelhão, J., Silva, A., Cerqueira, M., et al. (2013). New telerehabilitation services for the elderly. In I. M. Miranda & M. M. Cruz-Cunha (Eds.), *Handbook of research on ICTs for healthcare and social services: Developments and applications*. Hershey, PA: IGI Global.
9. Ferreira, F., Almeida, N., Rosa, A. F., Oliveira, A., Casimiro, J., Silva, S., et al. (2013). Elderly centered design for Interaction—the case of the S4S Medication Assistant. In *5th International Conference on Software Development and Technologies for Enhancing Accessibility and Fighting Info-exclusion, DSAI, Vigo, Spain*.
10. Leal, A., Teixeira, A., & Silva, S. (2016). On the creation of a persona to support the development of technologies for children with autism spectrum disorder. In *Proc. HCI International LNCS 9739, Toronto, Canada*, 213–223. doi: [10.1007/978-3-319-40238-3\\_21](https://doi.org/10.1007/978-3-319-40238-3_21)



11. Almeida, N., Silva, S., Santos, B. S., & Teixeira, A. (2016). Interactive, multi-device visualization supported by a multimodal interaction framework: Proof of concept. In *Proc. HCI International. LNCS 9754, Toronto, Canada*, 279–289. doi: [10.1007/978-3-319-39943-0\\_27](https://doi.org/10.1007/978-3-319-39943-0_27)
12. Almeida, N., Silva, S., & Teixeira, A. (2014). Design and development of speech interaction: a methodology. In *Proc. HCI International, LNCS 8511, Crete, Greece*, 370–381.
13. Teixeira, A., Francisco, P., Almeida, N., Pereira, C., & Silva, S. (2014). Services to support use and development of speech input for multilingual multimodal applications for mobile scenarios. In *The Ninth International Conference on Internet and Web Applications and Services (ICIW 2014), Track WSSA—Web Services-based Systems and Applications, Paris, France*.
14. Vieira, D., Freitas, J. D., Acartürk, C., Teixeira, A., Sousa, L., Silva, S., Candeias, S., and Sales Dias, M. (2015). "Read that article": Exploring synergies between gaze and speech interaction. In *Proc. 17th International ACM SIGACCESS Conference on Computers & Accessibility (ASSETS '15). ACM, New York, NY, USA*, 341–342. doi: [10.1145/2700648.2811369](https://doi.org/10.1145/2700648.2811369)
15. Wiechno, P., Dahl, D., Ashimura, K., & Tumuluri, R. (2012). Registration & discovery of multimodal modality components in multimodal systems: Use cases and requirements. [Online]. <https://www.w3.org/TR/mmi-discovery/>. Accessed 1 Jan 2016.
16. Almeida, N., Silva, S., & Teixeira, A. J. S. (2014). Multimodal multi-device application supported by an SCXML state chart machine. In *Workshop on Engineering Interactive Systems with SCXML, The sixth ACM SIGCHI Symposium on Computing Systems, Toronto, Canada*.
17. Almeida N., & Teixeira A. (2013). Enhanced interaction for the elderly supported by the W3C Multimodal Architecture. In *Proc. 5a Conf. Nacional sobre Interação, Vila Real, Portugal*.
18. Teixeira, A., Almeida, N., Pereira, C., e Silva, M. O., & Pereira, J. C. (2013). Serviços de Suporte à Interação Multimodal. In A. Teixeira, A. Queirós, & N. Rocha (Eds.), *Laboratório Vivo de Usabilidade* (pp. 151–165). ARC Publishing.
19. Teixeira, A., Almeida, N., Pereira, C., e Silva, M. O., Vieira, D., & Silva, S. (2016). Applications in ambient assisted living. In D. Dahl (Ed.), *Multimodal Interaction with W3C Standards*. Springer.
20. Barnett, J., Akolkar, R., Auburn, R. J., Bodell, M., Burnett, D. C., Carter, J., et al. (2015). State chart XML (SCXML): State machine notation for control abstraction. W3C Recommendation. <https://www.w3.org/TR/scxml/>. Accessed 29 Jul 2016.
21. Baggia, P., Burnett, D. C., Carter, J., Dahl, D. A., McCobb, G., & Raggett, D. (2009). EMMA: Extensible multimodal annotation markup language. <https://www.w3.org/TR/emma/>. Accessed 1 Jan 2016.
22. Lee, B., Isenberg, P., Riche, N. H., & Carpendale, S. (2012). Beyond mouse and keyboard: Expanding design considerations for information visualization interactions. *IEEE Transactions on Visualization and Computer Graphics*, 18(12), 2689–2698.
23. Ward, M. O., Grinstein, G., & Keim, D. (2010). *Interactive data visualization: Foundations, techniques, and applications*. Natick, MA: CRC Press.
24. Roberts, J. C., Ritsos, P. D., Badam, S. K., Brodbeck, D., Kennedy, J., & Elmqvist, N. (2014). Visualization beyond the desktop—the next big thing. *IEEE Computer Graphics and Applications*, 34(6), 26–34.
25. Jaimes, A., & Sebe, N. (2007). Multimodal human-computer interaction: A survey. *Computer Vision and Image Understanding*, 108(1–2), 116–134.
26. Schmidt, B. (2014). *Facilitating data exploration in casual mobile settings with multi-device interaction*. Universität Stuttgart, Holzgartenstr. 16, 70174 Stuttgart.
27. Chung, H., North, C., Self, J. Z., Chu, S., & Quek, F. (2014). VisPorter: Facilitating information sharing for collaborative sensemaking on multiple displays. *Personal and Ubiquitous Computing*, 18(5), 1169–1186.
28. Isenberg, P., Elmqvist, N., Scholtz, J., Cernea, D., Ma, K.-L., & Hagen, H. (2011). Collaborative visualization: Definition, challenges, and research agenda. *Information Visualization*, 10(4), 310–326.
29. Pereira, C., Almeida, N., Martins, A. I., Silva, S., Rosa, A. F., Oliveira e Silva, M., & Teixeira, A. (2015). Evaluation of complex distributed multimodal applications: evaluating a



- telerehabilitation system when it really matters. In *Proc. HCI International, LNCS 9194, Los Angeles, CA, USA*, 146–157, doi:[10.1007/978-3-319-20913-5\\_14](https://doi.org/10.1007/978-3-319-20913-5_14)
30. Bostock, M., Ogievetsky, V., & Heer, J. (2011). D<sup>3</sup>: Data-driven documents. *IEEE Transactions on Visualization and Computer Graphics*, 17(12), 2301–2309.
  31. Barnett, J., Dahl, D., Tumuluri, R., Kharidi, N., & Ashimura, K. (2016). Discovery and registration of multimodal modality components: State handling. [Online]. <https://www.w3.org/TR/mmi-mc-discovery/>. Accessed 15 Mar 2016.