Public Displays and Mobile Devices in an Augmented Objects Framework for Ubiquitous Learning

Rui Neves Madeira
DSI/ESTSetúbal
Instituto Politécnico Setúbal
Setúbal, Portugal
rmadeira@est.ips.pt

Abstract— **We are facing promising developments in the area of mobile services where context-aware systems can take the location and position of a user, her preferences and 'smart' objects into account, adding (implicit) interaction between a mobile device (of the user) and the real world. The assimilation of ubiquitous computing by education marks an important step forward, as being pervasive and persistent it allows students to access education calmly, flexibly, and seamlessly. This paper focus on a framework proposal with a well contained and defined application goal in Ubiquitous Computing and a prototype directed for Ubiquitous Learning environments. This system can augment physical objects in an educational environment with additional multimedia information, which is displayed in public shared displays according to the user (student or teacher) that faces the system. The information is contextual, based not only in the object, but also in its specific location, in the environment and in the detected user profile. A generic architecture was designed to help on the implementation of this augmented objects framework in any physical space.**

Keywords- Ubiquitous framework, Mobile devices, Public displays, Augmented objects, Bluetooth, Ubiquitous learning context

I. INTRODUCTION

Ubiquitous Computing (UbiComp) has the main goal of building computing systems that support and facilitate the daily lives of users, but being the least intrusive possible.In 1991, Mark Weiser wrote the article "The Computer for the 21st Century" [1], which outlined his vision for UbiComp, based on the creation of environments saturated with computing and communication facilities, with a smooth and unobtrusive integration. Nowadays, mobile devices, particularly the handheld ones like mobile phones, PDA's and smart phones, are essential tools for our daily living. We are facing promising developments in the area of mobile services where context-aware systems can take the location of a user, her preferences and real objects into account, and add interaction between the user, with a mobile device, and the 'real world'.

The assimilation of UbiComp by education marks an important step forward, as being pervasive and persistent it allows students to access education calmly, flexibly, and seamlessly. The integration of adaptive learning with UbiComp and Ubiquitous Learning (u-learning) offers great innovation in the learning process, allowing its personalisation and customisation to student needs. The term mobile learning is frequently used to refer to the use of handheld mobile devices that enable the learner to be 'on the move', providing anytime anywhere access for learning [2]. With the move of computation into the physical environments in which we live and interact, mobile and ubiquitous computing also provide the opportunity to enhance and support learning in more ways than the 'anywhere, anytime' conception, moving into the u-learning concept. One major opportunity is the facility to digitally augment the 'real world' by linking digital information with physical objects or the environment. Information can be displayed in different ways, e.g., it can be serendipitously 'pinged' through a user passive interaction by using embedded sensor technology and according to various conditions in the learning environment, or it can even be requested with an active interaction by a learner [3].

Following these concepts, this paper introduces a framework proposal and a system based on recognized technologies and directed for u-learning environments. The developed prototype, u-learnEST, is designed for user interaction with situated public displays using mobile devices. This system augments the physical objects, in a learning environment, with additional digital information. It displays information according to the user (student or teacher) that interacts with the system. The information that augments an object is contextual, based not only in the object, but also in its specific location, in the physical space and in the profile of the detected user. A generic supporting architecture is also presented as it was designed to make possible the implementation of this computing framework in any physical space. The implementation of the system u-learnEST was made at the installations of ESTSetúbal (Polytechnic Institute of Setúbal), for an engineering course, although the same concepts could be applied to other places. The main idea was to convert the laboratories of a specific course into big objects 'visited' by multiple users, automatically augmenting them with additional information every time a user is detected in its area. An instance of the system is composed by a situated public display, by a sensor server to manage detections, presentations and communications, and by the mobile devices of the users.

This paper is organized as follows. Section II presents a set of systems with points in common with our project. The requirements and guiding principles for the supporting framework are described in Section III. Section IV is dedicated to the presentation of the supporting framework, highlighting

978-1-4244-6571-2/10/$26.00 ©2010 IEEE April 14-16, 2010, Madrid, SPAIN

the generic architecture. Section V is dedicated to the implementation of the prototype u-learnEST, with relevance to the architecture instantiation issues and used technologies. Finally, in Section VI, conclusions are made and future work is presented.

II. RELATED WORK

This section presents projects that have application issues, principles and/or applied technologies in common with the present work.

In [4], we presented for the first time a computational framework that would integrate interactions between mobile devices and large displays. The framework was in an initial phase, but led to the present work. Moreover, the divingForPearls prototype [5] is centered on activities distributed by objects/public displays in a museum kind of scenario. Based on ambient information systems [18], the museum's system should be the least intrusive as possible, not distracting the user from the activity considered normal.

We are working on another project, UbiSmartWheel [6], that is also based on the same supporting framework. It is a pervasive biomedical assistive environment for the elderly, with a wheelchair as the smart augmented object, having embedded sensors, an RFID reader and a touch-panel interface. Additionally, interactions with large displays are being integrated in the environment of the application.

ActivitySpot [7] is an interesting framework with relevant points to the present work. It is used to support located activities by occasional visitors of the application spaces. Users are directed to actively interact with the implemented system by personal handheld mobile devices, public displays, and other additional technologies, such as, RFID, Bluetooth or cameras. Therefore, each visitor must be registered as a user with a personal profile. Depending on the action, the system response can be personalized as, for example, presenting information in a display targeted to the user profile.

Rememberer [8], which is based on the Cooltown project [9], works as a tool to capture personal experiences while visiting a museum. The system allows the construction of a record of the visit and may be accessed during or after it. The record is composed by web pages about the visited exhibitions, including annotations and photographs taken during the visit. In the exhibitions, cameras have been installed to take sequences of photographs every time a user requests it making gestures. Initial tests were carried out with RFID technology, with passive tags, as cards or embedded in watches, to indicate that a visit would be to register. Alternatively, PDAs with Wi-Fi communication were also used.

A different project to consider is GroupCast [10] that uses infrared badges to detect audience and present content in a public display to start or enhance conversations between individuals or groups. The correct content is automatically selected according to previous profiles and preferences, defined for each user in the area where the system is implemented. On the other hand, BluScreen [11] uses Bluetooth-enabled mobile devices to influence a public display by detecting the audience around it. Users influence the content of a public display only by their detection, thus having a passive interaction with the system. BluScreen seeks to maximize exposure to the current audience as opposed to matching content to known interests, like GroupCast. The user's passive interaction with the screen (such as stopping in the zone to view the contents) is co-opted to incidentally provide feedback and thereby change the content a user receives in the future.

The DiABlu project is also based on Bluetooth technology and Bluetooth-enabled personal mobile devices [12]. It has the goal of developing a computational framework for the detection of people in an environment. In function of the detection type, there can be created several answers in the form of arts applications. The framework is divided in 3 modules: the server that manages the detections, the Bluetooth-enabled mobile device client, and the final application that receives messages from the server. There is an alternative scenario that allows active interactions between the server and Java-enabled mobile devices.

It is also interesting to focus on projects supporting learning application goals. One of them is the SHAPE project that has the aims: to enhance the conceptual understanding of how to undertake design of computing in public spaces; and to create exemplars for how new computing can be used to augment educational and social interaction in public environments, specifically galleries and museums [13]. A part of the project is used to simulate an archaeology dig, where the aim is to enhance children's collaborative learning in museums, through supporting sensorial experience and capturing embodied knowledge, and using a combination of WebCam shape detection and RFID sensing technology.

One emerging concept of digitally augmented classroom environments is that of 'embedded phenomena', where the augmentation is designed to enable learners to experience various scientific phenomena in the classroom, like in RoomQuake [14]. It implements a classroom embedded with a combination of sensor technologies and physical artifacts used to simulate scientific earthquakes. Pocket PCs served to provide dynamic readings of the simulated earthquakes, which students had to then re-represent as physical models using the physical artifacts.

In the Ambient Wood project [3], a playful learning experience was developed where children could explore and reflect upon a physical environment that had been augmented with a medley of digital abstractions. These were represented in a number of ambient ways, designed to provoke children to stop, wonder and learn when moving through and interacting with aspects of the physical environment. A variety of devices and multimodal displays were used to trigger and present the added digital information, sometimes caused by the children's automatic exploratory movements, and at other times determined by their intentional actions.

UNIWAP [15] is a mobile learning project designed to assist in teacher training. The project used relatively simple technologies, short message service (SMS) and digital pictures, to enable students to create digital portfolios in a central database built from materials created in the field. Messaging was used to enable the trainees, who were widely distributed when training in different schools, to collaborate with each other and share their experiences. Feedback from the participants showed that the immediacy provided by the mobile devices was an important factor. Messages and pictures could be shared immediately with other students. The supervisors benefited from being able to access the material generated by the students in the shared database while travelling between work placements.

Finally, The Augmented Knight's Castle project bases its infrastructure in RFID technology, pointing the pervasive augmentation to a Playmobil playset named “Knight's Empire Castle” [16]. The scenario of the game is educationally enriched and augmented by music, sound effects and verbal comments made by pieces that react to the moves executed by the players. Therefore, they don’t use displays with additional information. Some pieces of the set were chosen as the actors of the augmented scenario and were marked with RFID tags. Additionally, it integrates mobile devices equipped with RFID readers for the users’ active interaction with the game.

III. REQUIREMENTS AND GUIDING PRINCIPLES

In this section, we identify and describe main requirements and guiding principles for a proper specification of the desired supporting framework for systems like the proposed.

Although, initially, we didn’t have the concern to implement ambient information systems, we think that the framework should be directed to follow, as much as possible, the requirements and features of these systems. The other principles are natural options to what was pretended, being adopted as core guidelines in the orientation process of defining the pretended systems.

The framework definition is based on the detection of persons involved in their everyday life activities, which have passive interactions that might be considered incidental with the augmented objects. In a first scenario of application, the basic aim is to ensure that only the detection of the user’s co-presence near the object will activate the presentation of additional contextual information in a public display. There is a one to one relationship between an object, or part of it, and a public display. An object can be considered a physical, inanimate and easily viewed entity, or can even be considered as an institution or another user with activity in the environment of the application. In any of these cases, we are augmenting objects by providing embedded contextual information, not available by just looking at them.

The framework should take into account a second level of interaction. Though triggered by a passive interaction, even "incidental", coming from the co-presence of the user in the environment, there can be an active interaction with the object/display pair, using for this purpose ordinary personal handheld mobile devices. These devices also allow the provision of additional information in a private mode.

Fig. 1 shows a generic scenario of application where a user interacts with smart objects that are augmented, mainly, through a public situated display. The guiding principles are followed in a kind of scenario where the present work application is inserted.

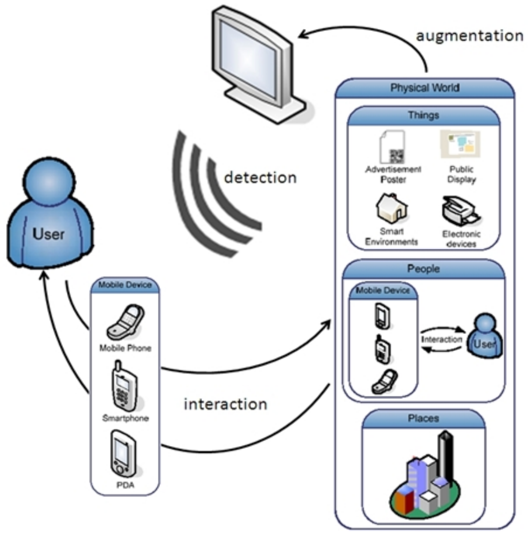


Figure 1. Generic scenario: user detection, object visual augmentation and active interaction between user and object (adapted from [17]).

A. Principles

After the definition of the main requirements, the guiding principles are clearly described as follows:

- Ambient Information System Concept

Considered a sub-domain of Ubicomp, the concept of Ambient Information System (AIS) has the main goal of presenting information through smooth and integrated changes in the environment, allowing that users are not distracted from their regular tasks, but being able to receive and be aware of not critical additional information. Being this a recent research area, there is still no commonly accepted definition for these systems, but Pousman et al. [18] attempted to unify the various existing terms, proposing the name used here and setting the following main features: 1) Show important, but not critical, information; 2) Getting attention from the periphery to the centre, and vice versa; 3) Aesthetically pleasing and environmentally appropriate integration; 4) Presentation of subtle changes to reflect updates in information (not to be focus of distraction); and 5) Preference for the use of existing resources in the environment for information presentation.

- Incidental Interactions with Co-Presence Notion

The notion of "incidental interactions" was proposed by Alan Dix to describe actions that are co-opted by a system to serve a purpose other than the one initially thought [19]. An incidental interaction can be seen as a situation where actions executed for another purpose (or unconscious signals) are interpreted in order to improve future interactions in everyday life. This notion is important for the present work, since the framework should start with the user interaction with an object to automatically augment it, without an obvious additional effort by the user.

It should be noted that incidentally does not imply that an action, or the subsequent goal, is not intentional. The user may have previously specified to the system its intention not needing to think consciously while performing the corresponding actions. Incidental interactions also differ in terms of feedback received by users. In a traditional application, the feedback received by the user about its actions will be direct, explicit. However, in an interaction called incidental, this feedback may be, in many cases, minimal, discrete and even delayed. Related with these interactions is the detection of persons that allow the computation of co-presence to be embodied within the real–world. The embodiment means that through normal real-world interactions, a user automatically participates in co-presence systems, without the need of participating in activities outside of his normal daily routine [20]. In a computational system, a technological solution, like RFID or mobile devices, should exist to detect the user co-presence.

- Use of Public Situated Displays

There are two trends, somewhat conflicting, with regard to providing information and services in collaborative environments [21]. The first directs itself to the use of public situated shared displays that can cover a wide audience while the second explores the mobility of small displays on mobile devices to offer more personalized content. Public displays, categorized as "boards" and preferably large, exist in various forms and scales, including, for example, street displays (used mainly for advertisement), projection displays used at conferences and in schools, kiosks, and even small screens next to objects. On the other hand, the second trend is related to mobile devices such as mobile phones, laptops and PDAs, which function not only as input for interactions with systems, but also as displays. These devices offer more personalization options, but the limited computational power and small displays, limit a more dynamic presentation.

Beyond other referred points, since public situated displays offer greater power and a better resolution for the presentation and combination of information, the framework's architecture presents a component that takes into account their integration in a system. In a first approach, the goal is to have the information that augments an object displayed next to it, with the display being a direct extension of the object. Thus, the action to augment the object becomes, initially, contained and dependent on the user approximation, being less intrusive and not forcing the user to take an active interaction. According to their availability and access modality, the framework's displays are categorized as: public, shared, proactive and even interactive [22].

- Ubiquity of Mobile Devices

Nowadays, handheld mobile devices are essential tools for our daily living, being present when interactions between users and objects occur in computational environments. So, they are becoming ubiquitous computational devices, since they find themselves in almost all locations, and are almost universally connectable. In parallel with the evolution of the available functionalities it also increased the diversity and capacity of wireless communication technologies of the devices.

Therefore, the proposed framework should take into account the ubiquity of the mobile devices. Depending on the context and type of physical space for application, the devices may be valuable tools when used for the detection of users. Thus, the space or the objects can feel the co-presence of a user in a natural and non-intrusive way. The user does not need to carry an extra identifier to be recognized in an application scenario. To accomplish this, using personal devices, it is assumed that there is a one to one relation between a user and her personal mobile device. An object/display pair can detect the co-presence of a device with the use of, preferably, one short-range wireless communication technology. Moreover, in a complementary option, the devices are used as tools to add to the framework active interactions and more personalized information. Knowing that situated displays are the choice in many Ubicomp systems, others prefer to use the small screens of mobile devices for mobility and personalization flexibility purposes [23].

The framework should enable the object augmentation through the combination of large public situated displays with small private mobile displays.

- Embedded Contextual Information

An environment may contain only one display or even a large number of display devices distributed in the environment and close to respective objects, casually available to environment users. The aim is to provide contextual information at decision points, which are where the objects are placed or exist. This process can be seen, merely, as a simple action of getting more and better information about an object, but also as an aid to make better decisions regarding the activities to be performed in an augmented space.

Therefore, it is considered that augmenting objects through situated displays is to provide additional information integrated, embedded, in the corresponding context. This option does not prevent the mobile devices displays to be used, as they can also be embedded in the context. Following this principle, it is essential to point that the information presentation is made where and when is needed, being as unobtrusive as possible, and with passive interaction [24]. Finally, the information that augments an object is contextual, based on not only in the object, but also in its specific location, in the physical space, or application scenario, and in the profile of the detected user.

IV. THE SUPPORTING FRAMEWORK'S ARCHITECTURE

This section is dedicated to the presentation of a generic and modular supporting architecture [25]. The architecture must be comprehensive, covering various types of scenarios. A generic modular architecture permits to maximize its application in different scenarios, decreasing the duration time of the systems' implementation phase. This approach allows us to understand how the various components can and should connect, adapting the architecture to the needs of each application project.

A. General Architecture

The implementation of a pervasive system to augment physical objects with additional contextual digital information involves multiple components and presents some particular issues. The proposed architecture for the system is based on three main basic components, which can be viewed as generic and independent modules, corresponding to three layers of implementation (Fig. 2). The components are the following:

- Tracking - Any application of this type must have a mechanism for identification and location of the user/device (tag). Additionally, this component can be extended to include an interface for short-range communication with the user.
- Application Core - This is the component responsible for implementing the application logic of the scenario. It must be able to analyze and process the information given by the module described above, crossing it with the contextual information related to the object. This component represents the independence of the application, which depends on the scenario and relative goals.
- Display Interface - In this work, displays were considered (the used technology varies with the application) as being versatile enough to achieve a reliable, rich and less intrusive representation of the extra information.

The three listed components are in the augmented object layer, considering the fundamental principle of pro-activity of the object/display pair in relation to the user. The latter can be seen as a simple tag (e.g., RFID or Bluetooth-enabled mobile phone) which has a passive interaction, resulting from her natural presence in the scenario of application. Additionally, for active interactions scenarios, the architecture provides a component for the mobile augmentation of the object. In this case, besides the public situated visual augmentation, the tracking component provides an extended augmentation communication module and the user is more than just a tag. Using a mobile device, which can be personal, enables communication with the object through short-range technologies or via GPRS/UMTS. It also allows viewing private information on the screen of the device, working as a supplement to the public display.

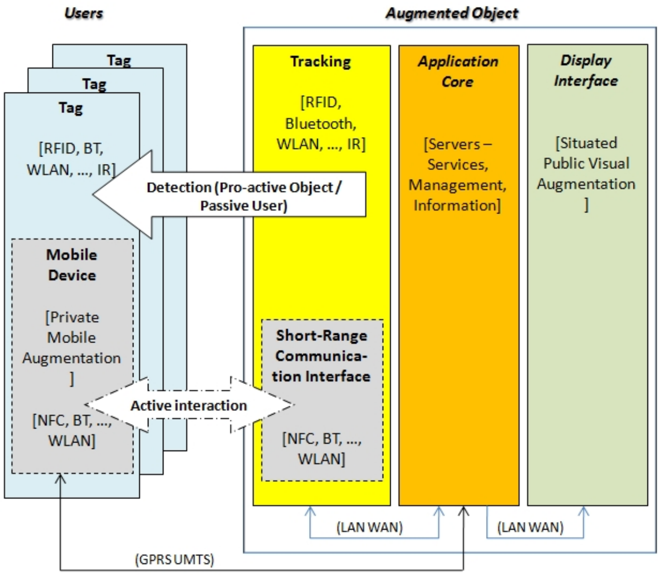


Figure 2. Main components of the general architecture.

The proposal can be seen as typical client/server architecture, given a set of servers dedicated to providing information to various clients. Therefore, the set formed by the user and the object can be seen as a client. Both the Tracking and the Interface components are considered close to the object, thus, being on the client side. The presence of the user in the area of the object is detected by the Tracking component that requests a service to the Application Core, the server. This one sends the results to the Display Interface. It can also communicate directly with the Tracking component, and even with the user's mobile device. Depending on the application scenario, the Application Core can be shared by several clients (objects) or be dedicated to only one object, with the latter having an extension by several users with mobile devices.

Since there are several modules/components communicating with each other, which may be remotes, it is necessary to have an interface for communications (which includes, e.g., NFC, Bluetooth, GPRS and/or WAN/LAN) capable of supporting, in a transparent and efficient way, the data communication between those modules.

B. Architecture's Main Modules

Fig. 3 presents a detailed architecture of the components corresponding to the augmented object layer, thus, only showing how the infrastructure faces the primary principle of user's passive interaction (or object's pro-activity). It follows a detailed description of all the architecture's main modules.

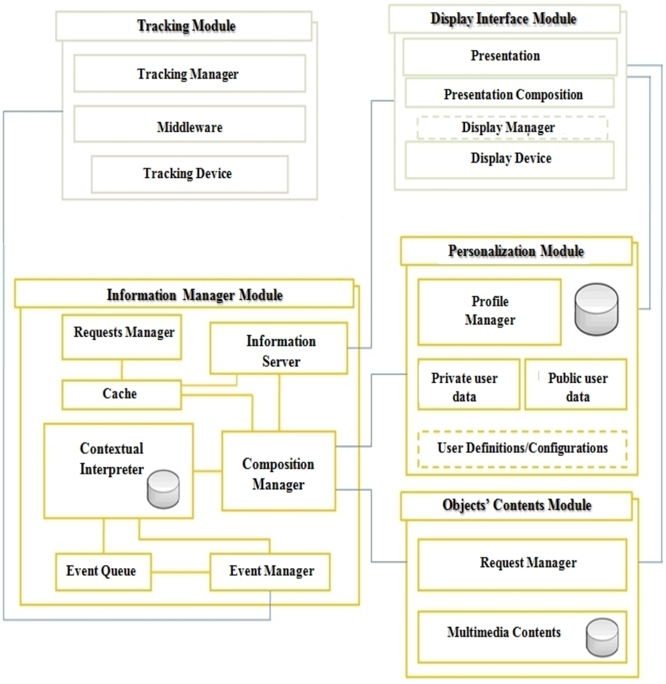


Figure 3. Architecture´s modules (augmented object components).

- **Tracking**:

This module is the basic complement to the object visual augmentation, and it is essential for any system implementation. It is responsible for the detection and identification of users, adding to the object the ability to perceive human co-presence (through technological means). Through this module, it is possible to know how much time a user spent in the pre-defined object area. There are several technologies that can be used with this module (e.g., RFID and Bluetooth). So, it is advisable to create an abstraction layer – Tracking Manager – to work as output interface, providing, transparently and uniformly, all data to the application core. Also noteworthy is the Middleware layer, which is dependent on chosen technology for a specific project development. This layer makes the abstraction of the Tracking Devices (hardware) used in the implementation, being essential for the top layer, which thus only has to integrate the interface(s) provided by the Middleware.

- **Display Interface**:

It is the real output module of an implemented system, being responsible for the final Presentation of the contextual information about an object. It can be programmed to integrate and provide other services, in case it is pretended to complement the main service of information presentation. The information should be presented to the user as soon as possible, verifying the lowest latency time between the user detection moment and information visualization. The application scenario, and in particular the users rate in the space, is crucial to the display technology choice. Since there is a great diversity for this choice, there may be a need to create an abstraction/management layer - Device Manager - for certain use cases, i.e., o create uniform protocols for the management of the Display Device(s), facilitating the programming of the Presentation Compositor sub-module. This module requests and receives messages from the server of the application module. These messages provide information about those involved in the interaction, i.e., object, user and her profile, and also the information type, which is the element that indicates what type of graphical user interface must be built/used. It also makes requests to the Personalization and Object's Contents modules to obtain the final content for presentation.

- **Personalization and Object's Contents (in Application Core component)**:

The first module acts as a data service that manages the application profiles and user data, and may include an optional sub-module for individual settings. The Object's Contents is another data service that is 'responsible' for the data repository about the scenario's objects. The repository is typically large and bulky, including a wide variety of multimedia content, such as text, sound, image and video.

- **Information Manager (in Application Core component)**:

The system architecture is event oriented because each time the user changes her status, i.e., being, or ceasing to be, detected next to an object/display pair, this is an event for which there is a handler, ensuring the application status update. The main entry of this module is the Event Manager, which controls the inputs generated by the Tracking module.

The module core is composed of a Contextual Interpreter (CI) and a Composition Manager (CM). After getting information from the Tracking module, the CI is responsible for correlating the detected user, which has a specific profile, with the detected object to obtain the corresponding additional information. As a next step, it sends a message (document) to the CM, giving knowledge about the executed interpretation. The CM sub-module combines this information with data gathered from data servers, producing another document to send to the Information Server. The document must also be produced according to the detection or request origin. If the CM has received a message from the CI, then the produced document must contain a minimum of information, because it is directed to the Display Interface module. In case the received request is from a mobile device (in an active interaction context), then the produced document must be the final, containing all the information asked or expected. Depending on the scenario of application, the CM can also take into account, for example, the local settings, user preferences, and object location. This composition module is the responsible for composing, structuring and selecting the information to be displayed.

The CM sub-module is dependent on the application, but the CI sub-module is completely independent, having a supporting model (architecture) prepared to suit any scenario. The model works for a scenario composed of multiple objects, such as a museum room, but also in cases where a conceptual large object is considered, such as an institution. It allows the association of multiple displays to one object, being useful in institutions scenarios.

The Information Server provides the final documents, created by CM, for both the Display Interface and the requests made by mobile devices, which are integrated in the infrastructure for active interactions. As for the Requests Manager, this is responsible for the information transmitted to the Tracking module, which can also perform a short-range communication with mobile devices.

C. Mobile Augmentation Modules

The generic architecture also presents an additional blue layer on the user side (Fig. 2). It will be implemented every time mobile devices are used in applications. Fig. 4 (left image) presents the Mobile Augmentation module that provides to the user an object additional augmentation, which can go outside the environment of application.

It allows the direct communication with the Tracking module, via short-range technology, and long distance communication with the Information Manager system. It has a Communication Manager that provides an abstraction in relation to the technology used for connectivity. This will make the programming of the Mobile Manager/Presentation (application) pair less complex and independent of used technology. The Mobile Manager includes an optional sub-module for management services, providing the facilities made available to the user in the process of enhancing the object, being strongly dependent on technology.

To accomplish the mobile augmentation purpose, on the Tracking module of the object layer, a new sub-module was integrated in Tracking Manager. Called SRCI (Short-Range Communication Interface), it permits the manager to be aware of devices that want to communicate with him. As an alternative to the near active interaction there is the possibility of the device to make direct requests to the application core (e.g., via GPRS), which provides augmentation documents in the Information Server. The purpose of this paper is not to present a detailed specification of the manager modules, but there are two important issues that should be described at this point, taking into account their importance to the present work. The first one is related to the possibility of SMS communication between the mobile client and the server and the second one is related to the CL competitive module.

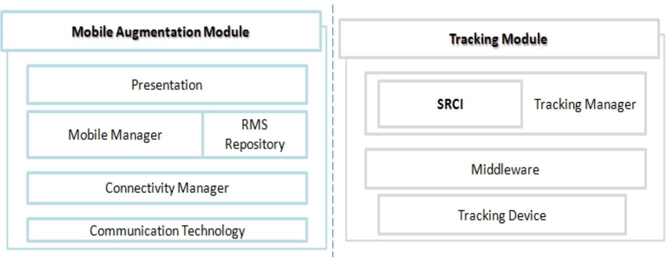


Figure 4. (left) Module of the mobile device and (right) SRCI addition to the Tracking module.

V. THE U-LEARNEST PROJECT IMPLEMENTATION

The implementation of the system u-learnEST was made at the installations of ESTSetúbal, for an engineering course, although the same concepts could be applied to other places. The main idea was to convert the laboratories of a specific course into big objects ‘visited’ by multiple users (students and teachers), automatically augmenting them with additional information every time a user is detected in its area. This information can be generic, for any kind of user, or personalized and directed to users with a know profile. For this system, the goal was to follow all the proposed principles, with particular focus on the ubiquity of mobile devices, thus, to implement an application with human active interactions.

The system is composed by a public shared display, a computer (the server) to manage the detections, and by the mobile devices of the users. The display was placed at the entrance of a main course laboratory, with the server close to it. Bluetooth (BT) was adopted as the communication and detection technology and Java, in the J2SE (Java 2 Standard Edition) and J2ME (Java 2 Micro Edition) platforms, was the basic language for software development. We can find almost a Bluetooth sensor attached to any person, through the mobile devices. This aligns well with the embodiment principle wanted for the system. A student going to a laboratory lesson would have an incidental interaction with the system, having the opportunity to interact actively with the u-learnEST system. For this, a client application was created for the mobile devices registered in the system. When an unregistered device appears, it is asked to register and to install the application to maximize the capabilities of the system. It is not necessary to have the client application installed to be registered.

Fig. 8 illustrates the implemented architecture for the present system. Regarding the instantiation issues, it is important to focus only on those considered determinant. In relation to the Tracking Device, a USB Bluetooth dongle connected to the server transform it in a Bluetooth device, being able to detect and to communicate with Bluetooth-enabled mobile devices. The Middleware module integrates three layers, with focus on the JSR-82 API [26]. The JSR-82 hides the complexity of the BT protocols stack, providing a simple set of API's for Java development. Since initially the JSR-82 API was only implemented for mobile devices, it was needed to use the BlueCove [27] implementation of it for J2SE. The Tracking Manager uses the BlueCove API and was programmed exactly as the correspondent in the other prototype. But, in this case, it interfaces directly with an Event Manager in the Application Core. Due to the fact that detecting “BT tags” does not work exactly as detecting RFID tags, this module has to manage the XML messages received from the previous module, filtering and organizing them for the Contextual Interpreter (CI). Its programming must be set and changed according to the conditions to be established for the system application.

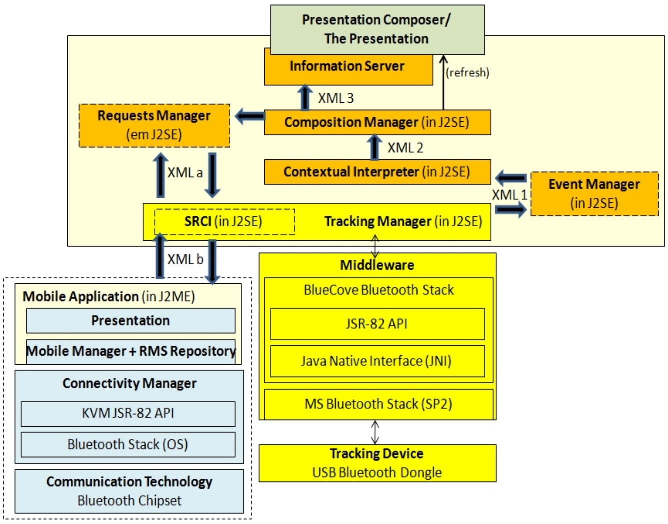


Figure 5. Architecture instantiation for the u-learnEST project.

The Application Core component includes three important implemented sub-modules: Contextual Interpreter, Composition Manager and Information Server. The first two are written in Java and the latter is a web server, which receives HTTP requests, and all the communication messages between them are XML documents (see Fig. 5). The Contextual Interpreter module is the system’s “decision maker”, processing data received from the Tracking Manager. Upon receiving a XML document with a detection indication, it makes an immediate processing, followed by the update of its contextual information database. Thus, knowing the object, next to which is the user, will determine what additional information to display and where to do it. With the contextual data determined, a second XML document (XML 2) is created and passed to the Composition Manager (CM) module. This one is responsible for placing a third document (XML 3) in the right folder of the Information Server. A folder for each object/display pair was created in the Information Server, with names according to the corresponding displays names. This is how the CM module knows where to place the generated XML

/

document. Before doing so, it processes the XML document passed by the previous module. It questions the Personalization module to retrieve some data about the detected user and the Objects' Content module to retrieve a portion of information that is essentially text. An element in XML 2 indicates if the additional information has text in its composition, beyond the generic summary that is already included. Content such as image or video will be retrieved from the database only by the Presentation Composer sub-module. Finally, the module automatically refreshes an XHTML file that is in the folder where the XML 3 document was previously placed.

The Composition Manager has one more task, the creation of XML documents with information to be accessed by mobile devices. The document is made available to the Requests Manager, thus, being available to be requested by a mobile device. In case a mobile device does not implement the API JSR-82, it may be placed in the Information Server. The server also has a folder for each mobile device, which using GPRS may request the most recent document.

In this prototype, the **SRCI** module was implemented to complement the Tracking Manager. It can start communicating with one of the discovered devices, but for this application it was programmed only to accept RFCOMM communication requests coming from users. When the mobile application requests a communication, the SRCI queries the Requests Manager database to obtain the XML document that refers to the device.

A. Mobile Application Development

The Mobile Application client is developed in Java (J2ME) for devices that support the MID profile [28]. It has two main modules:

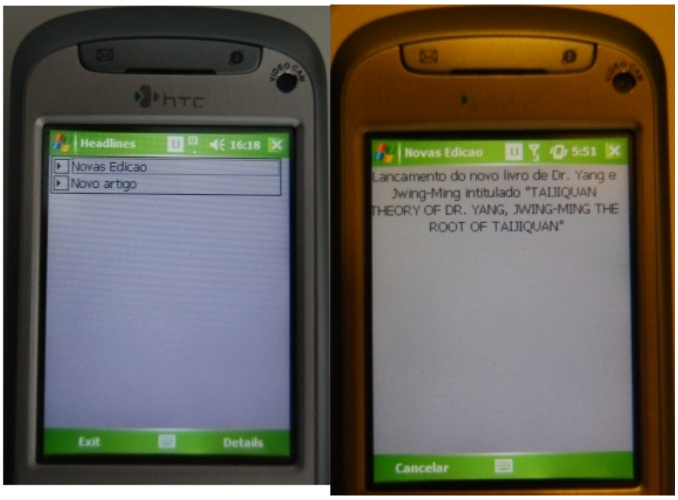
- One with services to interact with the server via Bluetooth (Connectivity Manager with two layers);
- Another one to manage the possible information received from the system (Mobile Manager).

Not all the mobile devices can run the first module to interact directly with the system. To accomplish that, the device must be embedded with the JSR-82 API. If a device does not embed the API, the solution is the direct communication with the Information Server, where the object has an URL address to be accessed through WAP. In this way, the user will be able to receive XML documents to consult later and to store them in the client repository, a Record Management System (RMS).

The module to manage the received information uses the kXML library in its implementation to make the parsing of the XML documents [29]. kXML provides an XML pull parser and writer suitable for all Java platforms, including J2ME. Because of its small footprint size, it is especially suited for Java applications running on mobile devices.

Summarizing, the client application allows the user (mainly, students) to:

- Discover nearby servers and connect to one (JSR-82 dependent). This makes it possible for the user to choose to interact with one from a number of nearby installations, although the existence of only one in the system implemented.
- Send a XML message or keystrokes to the u-learnEST server (via Bluetooth or web server).
- Receive information in a XML structure. This makes it possible for the user to receive personal and private information.
- Store received information.
- Visualize information after "disconnecting" from the system (Fig. 6).



Lancamento do novo livro de Dr. Yang e Jwing-Ming intitulado "TAIJIQUAN THEORY OF DR. YANG, JWING-MING THE ROOT OF TAIJIQUAN"

Figure 6. Earlier examples of mobile application screens.

VI. CONCLUSIONS AND FUTURE WORK

The current status of the prototype, using mobile devices and public displays, leads us to think that the system's framework is generic enough to deal with most interactions between the users and the 'laboratory object', even supported by different types of technologies. Although we have not yet carried out thorough usability and functional studies, we believe that the presented framework is also generic enough to accommodate different types of application scenarios, in terms of implementation issues.

The use of m-learning and u-learning tools, if correctly conceptualized and built, constitutes an efficient complementary tool to the traditional teaching methods. Students' information can be dynamically integrated over time and space, broadening and connecting students' understandings, both in the physical world and in classroom settings. Therefore, the u-learnEST project is an important asset to education, because it automatically complements the traditional learning environment without needing special devices from students, the main users of the system.

For future work, we are planning to update the mobile application interfaces with better design and functionality. In relation to interactions with users, we intend to carry out thorough usability, human-computer interfaces and personalization studies. Right now, we are conducting a questionnaire survey among 20 students that tested the system in order to determine their satisfaction and opinions.

ACKNOWLEDGMENT

I would like to thank to Prof. Nuno Correia for the great job coordinating our general work that led to the development of the framework and the system implementation.

REFERENCES

[1] M. Weiser, The Computer for the 21st Century, Scientific American, vol. 265, no. 3, Sept. 1991, pp. 66-75.

[2] S. Price, Ubiquitous computing: digital augmentation and learning, in Pachler, N. (Ed.), M-learning: towards a research agenda, WLE Centre occasional papers in work-based learning 1, London: Institute of Education, 2007.

[3] Y. Rogers, S. Price, G. Fitzpatrick, R. Fleck, E. Harris, H. Smith, C. Randall, H. Muller, C. O'Malley, and D. Stanton, Ambient Wood: designing new forms of digital augmentation for learning outdoors, Proc. of Interaction Design and Children, New York, ACM Press, 2004, pp. 3–10.

[4] R.N. Madeira, and N. Correia, Interaction between Shared Displays and Mobile Devices in an Augmented Objects Framework, Proc. Int. Conf. on Mobile Ubiquitous Computing, Systems, Services and Technologies (UBICOMM'07), IEEE Computer Society Press, Nov. 2007, pp. 189-194.

[5] R.N. Madeira, and N. Correia, "divingForPearls: A System with RFID based on a Ubiquitous Infrastructure", Proc. 4th International Conference on Ubiquitous Information Technologies & Applications (ICUT 2009), IEEE Computer Society Press, Dec. 2009.

[6] O. Postolache, R.N. Madeira, N. Correia, and P. Girão, UbiSmartWheel – A Ubiquitous System with Unobtrusive Services Embedded on a Wheelchair, Proc. ACM 2nd International Conference on PErvasive Technologies Related to Assistive Environments (PETRA'09), ACM press, Jun. 2009, pp. 1-4.

[7] H. Pinto, R. Jose, and J. C. Campos, An Interaction Model and Infrastructure for Localized Activities in Pervasive Computing Environments, Proc. ICPS'07: IEEE International Conference on Pervasive Services, perser, Jul. 2007, pp. 232-241.

[8] M. Fleck, M. Frid, T. Kindberg, E. O'Brien-Strain, R. Rajani, and M. Spasojevic, Rememberer: A Tool for Capturing Museum Visits, Proc. Ubicomp'02, Sep-Oct. 2002, pp. 48-55.

[9] J. Barton, and T. Kindberg, The CoolTown User Experience, Technical Report 2001-22, HP Laboratories, Palo Alto, Calif., 2001.

[10] J. F. McCarthy, T. J. Costa, and E. S. Liongosari, UniCast, OutCast & GroupCast: Three Steps Toward Ubiquitous, Peripheral Displays, in Proceedings of the 3rd Int. Conf. On Ubiquitous Computing (UbiComp '01), Springer-Verlag, 2001, pp. 332-345.

[11] M. Sharifi, T. Payne, and E. David, Public Display Advertising Based on Bluetooth Device Presence, in Proceedings of the Mobile Interaction with the Real World workshop (MIRW 2006), 2006, pp. 52-55.

[12] J. Cardoso, Enabling user interaction in installation art using mobile devices, British HCI Group Interfaces Magazine, vol. 58, 2006, pp. 6-9.

[13] T. Hall, and L. Bannon, Designing ubiquitous computing to enhance children's interactions in museums. Proceedings of IDC 2005, Boulder, Colorado, 2005.

[14] T. Moher, S. Hussain, T. Halter, and D. Kilb, RoomQuake: embedding dynamic phenomena within the physical space of an elementary school classroom. Conference on human factors in computing systems, New York: ACM Press, 2005, pp.1655–1668.

[15] P. Seppälä, and H. Alamäki, Mobile learning in teacher training, Journal of Computer Assisted Learning, No. 19, 2003, pp. 330-335.

[16] M. Lampe, and S. Hinske, Integrating Interactive Learning Experiences into Augmented Toy Environments, Pervasive Learning 2007: Design Challenges and Requirements Workshop At Pervasive2007, Springer, vol. 4480, May 2007.

[17] E. Rukzio, S. Wetzstein, and A. Schmidt, A Framework for Mobile Interactions with the Physical World, in Wireless Personal Multimedia Communications (WPMC'2005), 2005.

[18] Z. Pousman, and J. Stasko, A taxonomy of ambient information systems: four patterns of design, Proc. AVI '06: working conference on Advanced visual interfaces, 2006, pp. 67-74.

[19] A. Dix, Beyond intention - pushing boundaries with incidental interaction, Proc. Building Bridges: Interdisciplinary Context-Sensitive Computing, Glasgow University, Sept. 2002.

[20] P. Dourish, Where the Action Is, MIT Press, 2001.

[21] T. Paek, M. Agrawala, S. Basu, S. Drucker, T. Kristjansson, R. Logan, K. Toyama, and A. Wilson, Toward universal mobile interaction for shared displays, Proc. CSCW '04: 2004 ACM conference on Computer supported cooperative work, ACM Press, 2004, pp. 266-269.

[22] M. Koch, Supporting Community Awareness with Public Shared Displays, Proc. of 18th Bled eConference "eIntegration in Action", 2005.

[23] S. Agamanolis, Designing displays for Human Connectedness (book chapter), in Public and Situated Displays: Social and Interactional aspects of shared display technologies, Kenton O'Hara, Mark Perry, Elizabeth Churchill, and Daniel Russell (eds), Kluwer , 2003.

[24] A. Schmidt, M. Kranz, and P. Holleis, Embedded Information, Workshop Ubiquitous Display Environments on UbiComp 2004, Sept. 2004.

[25] R. N. Madeira, "A Computational Infrastructure to Augment Real Objects with Additional Information", MSc Thesis, FCT/UNL, 2007,

http://dspace.fct.unl.pt/bitstream/10362/1963/1/Madeira_2007.pdf (in portuguese).

[26] JSR 82 API, http://www.jcp.org/en/jsr/detail?id=82.

[27] BlueCove API, http://sourceforge.net/projects/bluecove/.

[28] Sun Microsystems, Inc., J2ME, Sun Developer Network (SDN), http://java.sun.com/javame/, 2009.

[29] kXML parser API, http://kxml.sourceforge.net/about.shtml.