

# Towards a Dynamic and Extensible Middleware for Enhancing Exhibits

Walter Rudametkin<sup>1,2</sup>, Kiev Gama<sup>1</sup>, Lionel Touseau<sup>1</sup>, Didier Donsez<sup>1</sup>

<sup>1</sup>Université Grenoble 1, BP 53, F38041 Grenoble Cedex 9, France

<sup>2</sup>Bull SAS, 1, rue de Provence, BP 208, F38432 Echirolles Cedex, France  
{firstname.lastname}@imag.fr

**Abstract**— Exhibit sites, such as museums and commercial conventions, do not usually allow or motivate visitor interaction with the exhibited items. In this work we present an architecture for providing an augmented experience for exhibits. In our concept, visitors can use wireless handheld devices to scan the identifiers (e.g. RFID tags, 2D barcodes) of exhibit items and receive information about them or trigger events in the surrounding environment. We present a generic middleware which is dynamic, extensible and configurable, and is capable of generating different responses to visitor interactions. Responses can consider the visitor’s profile, the exhibit’s or visitor’s history, administrative preferences, and other information. They include sending events to the visitor’s device or to devices surrounding the exhibit. The architecture hides development complexity and takes advantage of various technologies, integrating them into our middleware and our handset application. We also provide administration capabilities, such as reporting and exhibit configuration. Our concept has been implemented and validated in a museum scenario and uses multiple technologies for communication, unique exhibit identification, and software dynamicity and extensibility.

**Keywords:** NFC, RFID, Event Condition Action, OSGi, Dynamic Middleware.

## I. INTRODUCTION

Traditionally, exhibit environments (e.g., museums, commercial demonstrations, showrooms) provide information to visitors in a passive way, where visitors only observe the exhibited items (e.g., works of art, automobiles), but this scenario is gradually changing [1] [2] [3], exploiting the advantages of technology and moving towards an augmented user experience. Common utilizations of technology in such scenarios are audio gadgets that can be rented in museums or interactive computer applications available in strategically positioned kiosks at convention centers. Both cases of utilization involve hardware maintenance costs covered by the exhibit organizer.

Nowadays, ordinary people have become familiar with interactive scenarios (e.g., digital TV, multimedia applications, web pages) in their day-to-day lives. Wireless communication technologies such as Near Field Communication (NFC) [4], Bluetooth and Wi-Fi are available at affordable costs in consumer electronics, such as handheld devices, used on a daily basis.

Moreover, the exhibit total cost ownership (TCO) relies also on the software which manages the interactions with the visitors, plays the scenography and reports the exhibit activity. Most of those applications are vertical solutions that are developed specifically for exhibits management. General-purpose middleware implementing widely used industry standards can reduce the TCO by reusing modules already developed for other domains such as supply chain [5] or maintenance reporting.

In this paper we propose both an approach and architecture for interactive and evolutive exhibit applications. The approach consists in taking advantage of the visitors’ handsets for deploying the interactive exhibit application on them. Our proposed architecture introduces the customization of a general-purpose RFID middleware in order to respond to the interaction of the visitor with exhibited items. In this approach, the interactive system becomes the environment itself, and is no longer in a single precise location, but instead, is a series of surrounding objects and devices. The visitor can interact with the exhibits by using the client application for scanning the item’s corresponding RFID tags or 2D barcodes (e.g., DataMatrix, QRCode). The information is sent to the server application using fee-free wireless networks (e.g., Bluetooth, WiFi). The middleware server layer can then use this information to perform a diversity of tasks. The system can be configured to enact appropriate actions according to the user’s profile. We provide a useful museum scenario for evaluating our architecture and we have customized our implementation of an EPCGlobal middleware to develop a solution specific to exhibits. For this scenario, we have successfully integrated the usage of different types of wireless handsets and multimedia devices to provide user feedback.

The remainder of the paper is structured as follows. Section II presents an exhibit scenario. Section III describes the architecture of the middleware. Section IV details the proof of concept used for validating our proposition. Section V discusses related work. And finally, section VI exposes future work and concludes this paper.

## II. SCENARIO: THE MUSEUM EXHIBIT

In order to better understand the context and the expected capabilities of our middleware, we will present a popular scenario. We have used this scenario for designing our middleware and we have developed a use case application on top of it. Our application is used for enhancing museum exhibits. In this application, visitors can specify their preferences, including, but not limited to, whether or not they

suffer from any kind of handicap, their language of preference or their background knowledge of the subject at hand. The environment around a work of art will adapt itself and provide additional contextual information to a visitor according to his or her interaction with the solicited works of art. Such information can be visual, audio (or both), and aids the visitor by providing a more complete and satisfying environment. We will provide two perspectives when interacting with the middleware: that of the visitor and of that of the museum curator or scenographer.

#### A. Visitors

Visitors must initially install our client-side software on their NFC enabled device, such as an NFC-enabled telephone. Such a device must also provide a means of communication of longer range than NFC (e.g., Wi-Fi, Bluetooth). The client, when initiating a tour in the museum, is asked to configure the software with his or her specific preferences (e.g., language, specific interests, visual or hearing handicap). A visitor in the museum can then approach and use the NFC-enabled device to touch the corresponding RFID tags of the works of art, enacting a series of events in the visitor’s immediate surrounding environment. These events include a visitor receiving information pertaining to the solicited object directly on their NFC device (e.g., a webpage, a multimedia file as shown in the left of the figure 2), or information and events can be sent to the surrounding environmental devices, such as, audio-video devices (e.g., UPnP media renderers) or light dimmers. In case the device is not NFC enabled, the device must have a camera and be capable of reading the 2D barcode located next to the work of art, ultimately providing a similar result as in the RFID case. The figure 1 shows the elements used in this scenario, including the visitor equipped with a handset, the exhibit item that has been properly tagged, the communication paths between the server and the handset, and the rendering devices.

Visitors can then move on to the next exhibit item, halting the previous actions from the earlier exhibit if they remain unfinished, and creating new actions according to the current exhibit. Of course, information is recorded, so if a visitor returns to a previous exhibit the effects can be restarted, continuing where they had been halted or new events can be sent to the surrounding environment.

The system may propose a quick survey (right part of figure 2) to the visitor for evaluating the exhibit or for playing to a serious game [6][7] during the visit and at the exhibit exit as in the GUIDE system [8]. Surveys are sent to the NFCExhibit middleware. In the case of evaluations they are stored for later study by the exhibit organizer. In the case of a serious game, they are used to compute the player’s score and rank. In the case of NFC enabled handsets, the evaluations and the game responses can also be directly exchanged between visitors (this is not possible for handsets that only use 2D barcodes). A visitor, when encountering another visitor, can share his opinions regarding the previous exhibits using NFC peer-to-peer mode by simply touching the other visitor’s NFC handset. This gives visitors a possibility to compare opinions or game responses.

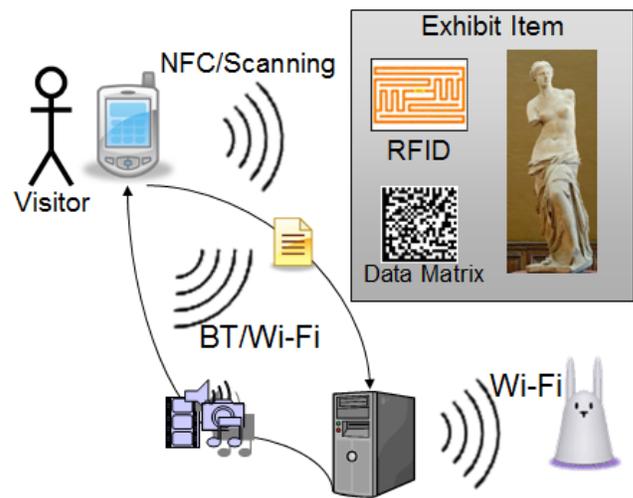


Figure 1: Museum exhibit scenario



Figure 2: Screenshots of the visitor’s NFC handset browsing information about a “touched” artwork (left) and rating it (right)

#### B. Museum Management

Museum management must be initially capable of expressing the design and behavior of the exhibit. While audio and visual devices, such as renderers, must be physically positioned in the environment, the drivers and controllers necessary for these elements must be activated and deployed by the scenographer. Also, the scenographer must be capable of enabling the actions of the surrounding devices when the system receives events pertaining to an exhibit being scanned or viewed (either by RFID or 2D codes).

Once the system is functional, the scenographer can use it to collect relevant information regarding visitors’ habits. This implies recording events such as visited exhibits and the events sent to external devices. Using this information, reports can be generated regarding visitors’ most common paths or the average time spent between works of art. With such information, correlated with the surveys filled out by the visitors themselves, people in charge of the museum can rearrange rooms, remove works of art that are not often visited or where visitors do not stop, or simply be aware of the museum’s most attractive pieces. Furthermore, the

scenographer is able to adapt and configure system preferences at any time, including adjusting peak period preferences, modifying the events and the media sent to external devices, change survey questions assigned to each exhibit, modify profile options, or simply generate reports.

### III. MIDDLEWARE ARCHITECTURE

The proposed architecture consists of handset-side and server-side software. The handset software is deployed on the visitor handsets, which are able to read information that can uniquely identify the exhibit item, either in the form of RFID tags or 2D barcodes. The server-side software, which is in charge of servicing the information read by the client software, is divided into two major parts: (1) a dynamic middleware that provides bridges and access to physical devices and evaluates events using an Event Condition Action (ECA) [9] rule engine, and software drivers that control the exhibit enhancing devices themselves; and (2) software that stores information and centralizes administration tasks and reporting,

The handset-side software has been developed using the J2ME framework. The installation on the visitor's handset can be triggered from the URL of the MIDlet description (e.g. .jad file) stored in the RFID tag or a 2D barcode located at the exhibit entrance. There are many differences between devices, and even though Java provides a level of abstraction over low level characteristics, it is still necessary to adapt applications in order to consider lacking or additional features. Installation is handled by the device and is often device specific, so varying procedures may be involved. The handset registers the visitor preferences and profile and enables communication with the server when RFID tags or 2D barcodes are read. The visitor profile is persisted on the handset and sent to the server when it is updated. The RFID tags contain a SmartPoster record completed by a geolocation record and a set of Bluetooth pairing records and Wi-Fi hotspot records<sup>1</sup>. The URL in the SmartPoster is used as a unique identifier. The geolocation record may be used to locate the visitor in the exhibit. The Bluetooth pairing record and Wi-Fi hotspot records enables to the application to skip long and unbound session initializations for the connection with the bridges.

The server-side of the architecture involves two main technologies: JavaEE and OSGi. By integrating JavaEE and OSGi we take advantage of the benefits provided by both solutions: robust tools and a proved application base found in enterprise applications versus dynamic middleware with device interaction facilities more commonly found in embedded systems.

The JavaEE part of the server-side application is used for the administrative web-interface, for recording the event history, and finally for creating the reports regarding visitors activity. The server software is mainly based on the EPCIS (EPC Information Services) which was initially specified by the EPC Global consortium for supply chain. The EPCIS, which is part of the OW2 (<http://wiki.aspire.ow2.org>) RFID

project, extends the EPC Global specification in order to store the geo-positions and the survey answers associated to visitor touch events.

The OSGi part of the server-side application is built around EPCGlobal's ALEServer. The ALEServer collects RFID events (with survey answers) sent by the visitors' handsets. The ALEServer publishes ECRports-compliant documents to the EPCIS and to the ECA rule engine. The ECA rule engine is implemented on top of the OSGi Event Admin service. The device drivers enable the communication with the physical devices that exist in the environment surrounding the exhibit.

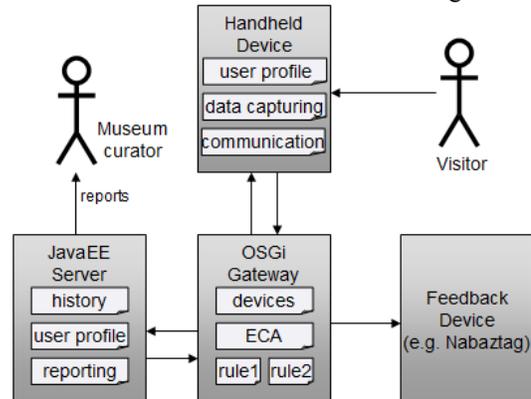


Figure 3: Platform Architecture.

The figure 3 shows the main elements of our architecture. We include the interaction devices and the main elements of the middleware. There are two basic actors that interact with the system, namely the visitor through his handset and the scenographer through the JavaEE administration interface.

When an NFC handset reads the unique exhibit identifier, it sends an event to the server. The event is recorded for later use in the reporting process and also to create an event history. Each event is a tuple consisting of an *ExhibitID*, a *UserID(PhoneID)*, and a *TimeStamp*. The server software receives the events and triggers the ECA rules that are currently active in the system. Conditions evaluate the events and can select actions to be performed. Each ECA rule has access to the event values, the user profile, the user's history (i.e., previous events related to the user), and the administration preferences that have been set (e.g., peak museum hours, number of visitors in the area). This provides the possibility of developing fairly complex rules. For example, at peak hours, actions sent to the handset devices are preferred because they are personal, and actions sent to media renderers should be avoided, since they can be used for group events.

In our system, ECA rules are provided dynamically and can be added, removed, and modified without restarting the application unlike well-known JSR-94 (Java Runtime API for rule engines) rule engines such as Drools or JRules. This provides the museum with the possibility of adapting its exhibits without causing a full application restart. Of course, exhibit sites are commonly closed at night and rules could then be updated, but this implies that the exhibit scenographer is capable of anticipating future activity, even if only one day in advance, something that is error prone at best and can reduce

<sup>1</sup> The SmartPoster record (RTD) is standardized by the NFC Forum whereas the Bluetooth pairing record is only in the Nokia's NFC development kits. Others are defined by the OW2 AspireRFID project.

visitor satisfaction. As shown in figure 4, ECA rules can also be associated to specific device handlers or drivers. When devices disappear or are invalidated, their matching rules are deactivated; when devices reappear the rules are thus reactivated. Finally, actions, associated with conditions that are evaluated to true, include sending events and media to physical devices (e.g., image, text, video, audio, choreography), including the handset itself, the media renderers and the light dimmers.

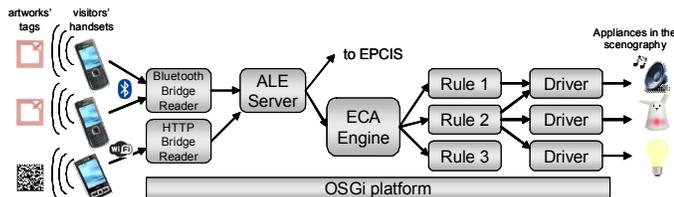


Figure 4: Event Condition Action rule engine.

#### IV. ARCHITECTURE VALIDATION: MUSEUM EXHIBIT

In our proof of concept, we have used two Nokia phones: An NFC 6131 handset (S40 firmware<sup>2</sup>) and a Nokia N95 (S60 firmware). The former is able to read RFID tags while the latter enables the reading of 2D barcodes. Each simulated work of art had an RFID tag (type ISO14443) and a printed 2D barcode, both uniquely identifying the same work of art. The external feedback was provided through Nabaztag electronic rabbits, and UPnP Media Renderers. For communication purposes, we setup a Bluetooth network between the NFC phone handset and the server, and a Wi-Fi network with the same purpose for the N95 phone (the N95 has Wi-Fi capability). For simplification matters, the same Wi-Fi network was used for communication between the server and the Nabaztag electronic rabbits. We attempted to demonstrate how the system would react in a real-world environment. Various user profiles were configured in the test, including multiple languages, different cultural levels (adults and children), and varying handicaps (i.e., visual or hearing). Responding to low attendance (one phone connected to the server) periods versus peak attendance (two phones connected to the server) periods was also tested.

Data capturing, either by RFID or datamatrix fired events to the server in order to get the interactive feedback. Depending on the cross-referencing of several factors (rules deployed in the system, user profile, work of art and peak period) the middleware received a response from the rule engine, which would either: send media to the UPnP media renderer or to the handset, play choreographies on the Nabaztag rabbits, send a sound file in the server computer (for testing purposes, since in a real scenario we would have other computers). Our implementation uses the JOnAS JavaEE (<http://jonas.ow2.org>), an OSGi-based application server, and the Apache Felix (<http://felix.apache.org>) implementation of the OSGi specification. The JOnAS server hosted the back office application which provided reports based on the

information logged when users scanned the tags of the works of art. The OSGi platform hosted the dynamic rule engine, which dynamically (i.e., without application restart) add or remove rules, as well as having its rules adapted depending on external device availability.

Finally, the middleware showed to be generic enough to be used in an arbitrary exhibit. For our scenario, in total we developed under twenty rules to be interpreted by the rule engine and three device bridges, one for each external device used. One profile was generated that included multiple questions to be filled out by the visitor. A multi-language description for each simulated exhibit was written, used to send responses back to the visitor's handset according to the profile language.

#### V. RELATED WORK

Museum and exhibit guides have been proposed as a trivial example for pervasive and context aware systems. Our work provides a dynamic and extensible middleware for enhancing exhibits. We take advantage of mass market devices, integrating different types of wireless communication in order to enhance the interaction with exhibited items. We take also advantage of a general-purpose middleware that enables visitors tracking for decision-supports and game accounting. In this section we present related work concerning attempts to enhance user experience and dynamicity in exhibits.

Grinter *et al.* [1] show the usage of a museum guidebook implemented as an interactive application in a handheld computer with a touchscreen display. The graphic interface of the application displays images (e.g., a wall with paintings) of the visited rooms, and the interaction consists of tapping the objects in the image to get audio descriptions of them. In this case there is no actual interaction with the exhibit item, since it is virtualized in the handheld application.

A web-based exhibit guide [11] provides a way to customize the visits to exhibit sites by allowing the audience to prepare the visit remotely (e.g., at home). During the visit users can interact with the system using a mobile device via a WLAN and the system proposes information (e.g., audio, animation, and texts) based on the users' context (e.g., the current exhibit room, position in the pavilion).

A more complete overview of context-based mobile applications for museums is provided by Raptis *et al.* [12]. It shows that in a context-based approach, information given to the user can be based on system decisions according to the user's location. Whereas in our approach we give the user the freedom to interact (e.g., pointing or scanning the item ID) with the exhibit items and the user only receives the information pertaining to the items that he or she expressively chooses.

We can find different utilizations of technology to augment exhibit experiences in the Exploratorium, an interactive science museum in San Francisco, United States. The Cooltown project [2] experiments an augmented experience by using handheld devices integrated with infrared and barcode readers which pick up URLs from barcodes located near the exhibited objects. The content of a scanned URL is displayed

<sup>2</sup> The Java platform of the S40 firmware is in reality too limited for downloading and streaming large amount of data. The GUI was limited to show textual information as it is shown in the figure 2.

in the handheld device as web pages with information about the Exploratorium exhibits and related topics. The PEACH project [10] uses IR beacons or RFID signals to locate the visitor and to deliver him a personalized presentation.

Other work [13][14] presents the usage of RFID technology in the Exploratorium for remembering visits. RFID cards are issued to visitors who are able to use such cards to bookmark their favorite exhibits. In the end of the tour, each visitor has a set of custom web pages with information and custom pictures taken during their interaction with the exhibits.

Finally, other usages of NFC handsets were proposed and experimented in the context of Museum exhibit. In the "Touch & Collect" project [15], the visitors can pick the URL by touching the RFID tags associated to exhibit items with their NFC mobile phones. The visitors can play afterwards the content either on their mobile phones, on home computers, or on displays scattered at the museum. The PLUG project [16] has experimented NFC-enabled handsets for a serious game similar to the Happy Families card game. Players can pick cards stored in RFID tags scattered in the CNAM Museum in Paris. Players can exchange cards using the NFC P2P mode. However, those projects are focusing currently more on the handset software than on the mediation infrastructure and the server backend or on reusable handset software libraries applicable to other contexts (e.g. maintenance, customer info).

## VI. CONCLUSION AND FUTURE WORK

The concept of enhancing exhibits is not recent, but our proposition provides an extensible dynamic middleware which has orthogonal pluggable aspects, such as, communication protocols and custom user feedback functionality.

Our initial use case prototype has been successfully tested and validates the architecture of our middleware by integrating different types of wireless communication. We have used RFID and 2D barcodes on the client side to obtain unique identifiers for each exhibit item, while using Bluetooth and Wi-Fi to communicate with the server side middleware. A pluggable mechanism to provide information to and obtain feedback from visitors has been implemented and can be used to create interactivity between the system and its users. These interactions can be modified at any time, and are also adaptable to users' profiles, hence customizing the overall interactive experience.

One of the main purposes of this work is to take advantage of different widespread information capturing technologies that are available in affordable consumer electronics, and of a general-purpose EPCglobal middleware with minor customizations. By deploying client software on the user's handheld devices we can minimize the required infrastructure necessary for such a system and approach a broad number of users. This is important when considering exhibits where potentially hundreds or thousands of people attend each day. Furthermore, our middleware allows exhibit organizers to provide a custom and interactive user experience with a lower investment, especially when regarding handheld device acquisition and maintenance.

## ACKNOWLEDGMENT

Part of this work has been carried out in the scope of the ASPIRE project (<http://www.fp7-aspire.eu>) which is co-funded by the European Commission in the scope of the FP7 programme under contract number 215417. Help and contributions from all partners of the project and also the OW2 AspireRfid community are acknowledged. Special thanks to our master students Maroula Perisanidi and Andrés Gómez for their invaluable effort in developing this middleware.

## REFERENCES

- [1] Grinter, R. E., Aoki, P. M., Szymanski, M. H., Thornton, J. D., Woodruff, A., and Hurst, A. Revisiting the visit: understanding how technology can shape the museum visit. In: Proceedings of the 2002 ACM Conference on Computer Supported Cooperative Work. New Orleans, Louisiana, USA, November 16 - 20, 2002
- [2] M. Spasojevic and T. Kindberg, "A Study of an Augmented Museum Experience". White paper, Hewlett-Packard. HPL-2001-178
- [3] Yglesias, A.E.; Moreno-Valle, L. The Amparo Museum Experience. Multimedia Computing and Museums: Selected Papers from the Third International Conference on Hypermedia and Interactivity in Museums (ICHIM'95 / MCN '95), Volume I. San Diego, USA, 1995.
- [4] Michahelles, F; Thiesse, F; Schmidt, A; Williams, J.R.; Pervasive RFID and Near Field Communication Technology, IEEE Pervasive Computing, vol. 6, no. 3, pp. 94-96, c3, Jul-Sept, 2007
- [5] Kefalakis, N., Leontiadis, N., Soldatos, J., Gama, K., and Donsez, D. Supply chain management and NFC picking demonstrations using the AspireRfid middleware platform. ACM Middleware Conference, Demo track, Leuven, Belgium, Dec. 2008.
- [6] Abt, C.C; Serious Games, Pub. University Press of America 1987, ISBN: 0819161489
- [7] Zyda, M.; From Visual Simulation to Virtual Reality to Games, Computer, Vol.38, No.9, Sept. 2005
- [8] Cheverst, K; Mitchell, K; Davies, N; Smith, G; Exploiting context to support social awareness and social navigation, ACM SIGGROUP Bulletin, Vol 21, No 3, Dec. 2000, DOI 10.1145/605647.605654
- [9] Chakravarthy, S; Le, R; Dasari, R; ECA Rule Processing in Distributed and Heterogeneous Environments, International Symposium on Distributed Objects and Applications, 1999.
- [10] Stock, O.; Zancanaro, M.; Busetta, P.; Callaway, C; Krüger, A; Kruppa, M; Kuflik, T; Not, E; Rocchi, C; Adaptive, intelligent presentation of information for the museum visitor in PEACH, User Modeling and User-Adapted Interaction, Vol 17, No 3, July 2007, DOI 10.1007/s11257-007-9029-6
- [11] Oppermann, R., Specht, M., and Jaceniak, I. Hippie: A Nomadic Information System. In Proceedings of the 1st international Symposium on Handheld and Ubiquitous Computing; Karlsruhe, Germany, September 27 - 29, 1999
- [12] Raptis, D., Tselios, N., Avouris, N. Context-based design of mobile applications for museums: a survey of existing practices. In Proceedings of the 7th international Conference on Human Computer interaction with Mobile Devices and Services. Salzburg, Austria, Sep. 19-22, 2005.
- [13] Hsi, S. and Fait, H. RFID enhances visitors' museum experience at the Exploratorium. Communications of the ACM 48, 9 Sep. 2005, pp 60-65
- [14] Fleck, M., Frid, M., Kindberg, T., O'Brien-Strain, E., Rajani, R., and Spasojevic, M. Rememberer: A Tool for Capturing Museum Visits. In Proceedings of the 4th international Conference on Ubiquitous Computing. Göteborg, Sweden, Sept, 2002.
- [15] Rieki, J.; RFID and smart spaces, International Journal of Internet Protocol Technology, Vol 2, No 3-4, 2007
- [16] Gentes A., Jutant C., Guyot A., Simatic M., RFID Technology: Fostering Human Interactions, IADIS International Conference Game and Entertainment Technologies 2009. Carvoeiro. Portugal.