A Ubiquitous Zoo Guide

ABSTRACT

In this paper a novel visitor information system for zoos is described. Vistors going through the zoo hold PDAs close to RFID tags, which are attached to animal cages, and which thus identify the animals. Over WLAN, the PDAs send this ID to a Web server, which sends back a multimedia Web presentation for the respective animal. The novel idea is to additionally attach the Web server to a global zoo peer-to-peer system, thus enabling zoos world wide to share animal presentations. Each zoo contributes multimedia presentations for some of their animals in one or several languages, but can also download presentations for other animals in many other languages. This way, the zoos can keep the effort for creating presentations small, but are able to present multimedia information for all of their animals in many languages.

1 Introduction

Personal digital assistents (PDAs) based on standard operating systems have become popular as basis for mobile information systems. Mobile information systems are used for instance by visitors of museums or other exhibitions, visitors of large companies or universities, or in the context of mobile learning. They may provide general information about the institution the visitor is in, about displayed items, or may provide navigational information about specific tours the visitor may follow.

When presenting information about displayed items, the PDA usually is used for showing multimedia information, like text, pictures, or audio and video presentations. Such a visitor information system for exhibitions thus must master two tasks: (i) identify the exhibited item (or equivalently, determine the exact position of the PDA, and based on this information, the nearest shown item), and (ii) present the available multimedia information on the PDA. Task (i) can be achieved in various ways, which will be presented in the following related work section. For (ii), the multimedia content may either be stored directly on the PDA, since modern consumer PDAs usually have 65 MB of memory or even more, or may be fetched on-demand from a server via wireless LAN (WLAN), usually following the IEEE 802.11b standard, since PDA network interfaces following IEEE 802.11a/g are still unavailable or too expensive.

Alternatively to PDAs, dedicated audio players may be used for playing for instance pre-recorded MP3 files, here gradually replacing traditional audio guides using for instance music cassettes.

2 Related Work

Dedicated visitor information systems based on PDAs have been implemented mostly for museums, but also other institutions. The most important difference is given by the technology for identifying the displayed items. One way is given by manually entering an identification number into the PDA using the PDA soft keyboard or the PDA's character recognition facility, which is used for instance with the Personal Art Assistant (PAA).[1] Of course, such a scheme is tedious and error prone, with possible annoyance to the visitors due to too much work, small displays and typos.

Another approach for determining the position of a visitor is to use a WLAN based positioning engine, for example the Ekahau system.[2] In [5] this product is used for positioning museum visitors in the Zoological Museum of the University of Munich.

An interesting approach is carried out by the CoBIT system [6]. Here, each visitor wears a CoBIT device attached to the visitor's ear, and having a solar cell. A light beam mounted on top of each exhibit provides the electrical energy for the CoBIT system (via the solar cell), and also provides audio and positioning information.

Similarly, the Musical Instrument Museum in Brussels uses infrared-controlled headphones, which automatically play prerecorded tunes.[3]

Using GPS is a straight forward idea. However, GPS generally provides an accuracy of only about 3 m (Differential GPS providing much higher accuracy is extremely expensive and used only by professionals), and using GPS inside buildings usually is not possible (although some indoor GPS products may rise the sensitivity significantly). Thus, GPS is mainly used outdoor for applications which do not need high positioning accuracy, like car navigation, for instance.

Concerning visitor guides, a system for covering a larger areas is given by LoL@, a city tourist guide [7]. Here, the position is computed using partially GPS, but also the Open Service Access (OSA) as standardized by the 3rd Generation Partnership Project (3GPP). The positioning itself is carried out by the Gateway Mobility Location Center (GMLC) using cell-IDs of the involved base stations and terminal signal strength.

Some systems follow a more generic approach and can use several positioning technologies like irDA, GPS, UMTS cell-ID or barcode readers, for instance the AIDA system[4], products from the company Eyeled[5] or the HIPS project [1].

In our system we use the RFID technology for identifying the exhibits. Here, an RFID tag is placed near the exhibit, and a PDA holding an RFID reader is placed near the tag. For passive tags, the distance between reader and tag usually must be in the order of 10 cm or below. Here, first the reader sends a wakeup signal to the tag, the signal also carrying enough energy for the tag to operate. The RFID tag then sends its unique identification number back to the reader, which then may retrieve the information being associated with this ID from a server. This system requires only little action or knowledge from the visitors, but shows a stable and reliable performance. Information systems using RFIDs for identification are given for example, by the Museum of

[1]http://www.pocket.at/business/kunstforum.htm
[2]http://www.ekahau.com/
[3]http://www.mim.fgov.be/home_uk.htm
[4]http://www.aida-guide.de/
[5]http://www.eyeled.de/

Natural History in Aarhus, Denmark[6], the eXspot system used in the Exploratorium in San Francisco [2, 4], or the Plantscanner [8].

For comprehensive surveys of the work done in this field see, for instance, the CIMI project homepage[7] or Marc Jelittos Web site (in German).[8]

The previously described visitor guides share one common disadvantage. On the one hand, although creating a technological solution is far but easy, it mainly means plugging together mature off-the-shelf products, or it can be as easy as simply buying a product from a company. On the other hand, whatever the technology used, all multimedia presentations must be created by the respective institutions for presenting their usually unique exhibits. But creating the multimedia content is cumbersome, and requires many person month, a fact that may prevent institutions from doing so. We present a system and a case study, where the effort for the production of multimedia content can be decreased dramatically. This is achieved by *sharing* content with other institutions, a scenario that makes sense, for instance, in the context of zoos.

3 The Ubiquitous Zoo Information System Prototype

A zoo is an interesting area for an electronic user guide system, since zoos have to provide information about every single animal. In general, most zoos offer information boards that are either affixed to the animal's cages or next to them. Unfortunately this information access is often hindered through a crowd of people waiting in front of the cage, the information does not provide the details the user is interested in, or the information is only available in a specific language.

We therefore propose an interactive zoo guide which lets visitors obtain animal specific information right on their mobile devices. The requirements of such an interactive zoo guide include multi-language support, rich multimedia like audio commentary or selfexplaining video sequences, and on-demand information composition to different levels of details. The production of such detailed animal information however can result in an overwhelming effort for a single zoo. Therefore we propose the zoo to be connected to a global peer-to-peer system through which zoos share their animal descriptions and download missing information from other zoos to reduce time and costs for additional authoring in many languages.

In the beginning the system administrator would setup the animal categories and assign RFID identifications for the animals supported by the interactive zoo guide. Content creators then create rich multimedia presentations for the available animals (or a subset of them) in one or several languages. The content together with metadata describing the level of detail and a language abbreviation are then stored in a local database. To complement missing presentations, system administrators may periodically start a process within the zoo system, which searches for missing animal presentations within the zoo community. Received results are stored locally in the zoo information base and made available to the visitor's PDA (Figure 1).

[6] http://rfidjournal.com/article/articleview/1110/1/1
[7] http://www.cimi.org/whitesite/index.html
[8] http://ausstellungsmediumcomputer.de/technik/tragbar.htm

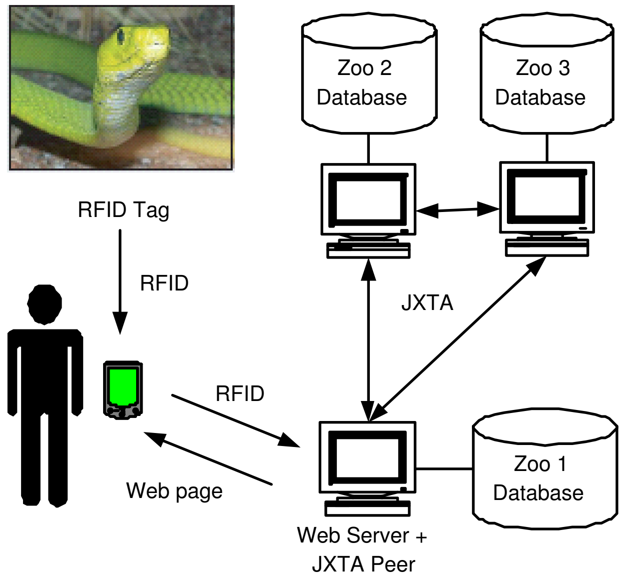

Figure 1: The information system architecture.

Zoo visitors are then equipped with PDAs having WLAN access to the zoo webserver, together with an RFID reader. When starting the zoo tour, visitors may first select their preferred language (Figure 2). Then, by holding the PDA close to an RFID tag which is attached to a specific cage, the PDA downloads the multimedia presentation for the animal living in the cage, and presents it on the PDA screen (Figure 3). In principle, no further interaction with the PDA is necessary, with the exception if a different level of detail is preferred. The main parts of this architecture (Figure 1) are given by the PDA part (Section 4), the webserver part (Section 5), and the peer-to-peer backend system (Section 6). These parts are described in the following.

4 RFID for PocketPC

On the client side we use a Toshiba e800 Pocket PC being equipped with a SanDisk WLAN (IEEE 802.11b) card plugged into the SDIO slot and an TAGflash RFID reader from TAGnology[9] for the CF II slot. The TAGflash reader and the used tags use a frequency of 13.56 MHz for communication. The PDA's operating system is Windows Mobile 2003 (Pocket PC 2003).[10] The PDA software consists of several parts (Figure 4). First a Java class called PerformRFID monitors the RFID reader, the used Java VM is IBM's J9 included into the WebSphere Device Developer Studio 5.5.0. The PerformRFID class acts as a permanently running thread, which calls two native libaries

[9] http://www.tagnology.com/
[10] http://www.microsoft.com/windowsmobile/

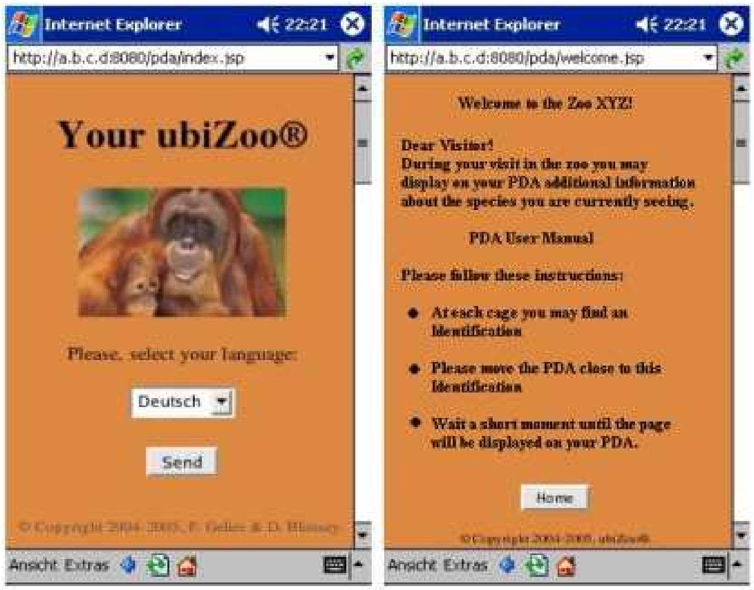

Figure 2: The PDA user interface: start screen for selecting the language.

via the Java Native Interface (JNI). The first is a DLL for reading data from the serial port based on the serial_function.DLL[11] for iPAQs. Using this library, the PerformRFID class continuously tries to read an array of size 19 bytes from the COM5 serial port, which is used for communicating with the RFID reader. A read attempt is carried out every two seconds. Valid RFID data is returned as soon as the RFID reader is placed near (around 5 cm) an RFID tag.

Once a valid ID has been read, the PerfomRFID class calls a method from the second native library IELaunch.DLL[12], which starts the Pocket PC's Internet Explorer, together with a URL pointing to the zoo webserver servlet RequestInfo and the read ID. If a presentation for the ID for the selected type and language is available on the webserver, the servlet returns it to be displayed on the PDA's Internet Explorer.

5 The Zoo Webserver

The core modules of the webserver include the *information usage module* to give visitors access to zoo relevant information, the *information creation module* enabling content creators to publish animal related information in a standardized way, and the *information sharing and requesting module* for enabling system administrators to download missing animal information from other zoos.

[11]http://www.ulrich-roehr.de/software/ipaq/serial/serial.html
[12]http://www.pocketpccity.com/software/pocketpc/IELaunch-2001-12-13-ce-pocketpc.html

Figure 3: The PDA user interface: animal presentations.

As webserver we use an ordinary Apache Tomcat running on a Linux machine. A web application called *pda* has been created and put into Tomcat's *webapps* directory. The web application is started by calling the index.jsp page. Here, the web application automatically creates a Java bean for uniquely identifying the user. The Tomcat server then answers with a conventional HTML page displaying a dropdown box for choosing the desired language (Figure 2 left). By pressing the *send* button, the chosen language is sent to the welcome.jsp page, which stores the language (specified in the parameter *lang*) in the user's session bean (Figure 2 right). From here on, a valid session is established, and the PerfomRFID class on the PDA may launch the Internet Explorer for any read RFID tag. Finding the right presentation for a given ID is implemented into the RequestInfo servlet. The read tag ID is passed to the servlet using a parameter called *rfid*. The RequestInfo servlet then opens the MySQL database table *animal* and looks for this ID in order to find the Latin name of the animal. For identifying each animal race, we chose the respective Latin name, which should be unique across the world. The Latin name, together with the language abbreviation and the content type are then used to construct a file name for an XML file, which should contain the requested animal presentation. Finally, if found, this XML file is turned into an HTML file which is returened to the PDA.

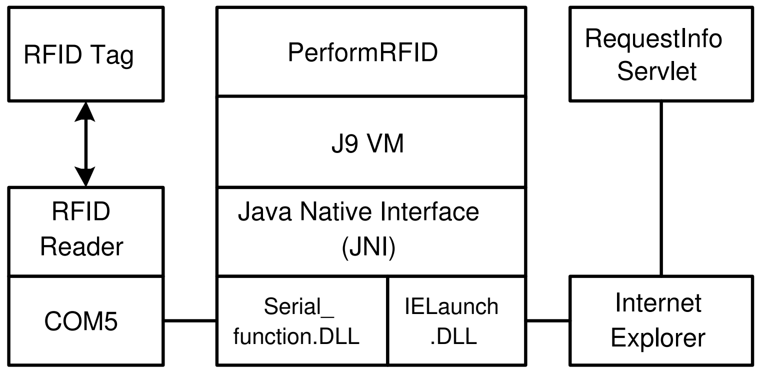

Figure 4: The PDA software architecture.

6 The Peer-to-Peer Backend System

Basic information about animals are generally always the same and can be standardized. Thus, it makes sense to exchange this information in order to obtain different language versions. As a consequence, zoos can save time as they do not need to create already existing content again. Peer-to-peer networking has a great potential to make a vast amount of resources accessible.

The prototype's information sharing and requesting module is based on JXTA[13], a widely used and mature peer-to-peer platform. From a high-level point of view, JXTA can be divided into three layers: core, service, and application layer. The core layer provides fundamental transport and security functionalities. The service layer relies on the core layer and provides services that are typically used in peer-to-peer applications, like searching and indexing. JXTA offers different application services, for the zoo community we will focus on the content management service (CMS), offering basic file sharing functionality. Simple methods exist for making files available to other peers, for searching for files, and for downloading files [3].

To establish a zoo content management system, first a peer group like "jxta-ZooGroup" for the zoo community should be created. Peers that belong to one application may assemble within peer groups, which form subnets within the global JXTA network and therefore contribute to the overall network scalability. By defining a custom peer group in addition to the global "Net Peer Group", discovery queries are later automatically restricted to members of the zoo community. During the CMS startup content advertisements are created for all shared files. These content advertisements contain information about the files, e.g., filename, unique content ID (cid), or length in bytes, and are needed to enable other peers to find these files in the JXTA network (Table 1). In our case, we additionally use meta information (XML <metadata> section) about the presentation language (here, "de" for German) and an expiration date, which defines a usage limit, after which the content should be refreshed.

[13]http://www.sun.com/software/jxta/

```
<!DOCTYPE jxta:ContentAdvertisement>
<jxta:ContentAdvertisement>
    <name>
        Pelecanus-crispus_a_de.xml
    </name>
    <cid>
        md5:4c4bd7030f082baf56209ae5cbcd516d
    </cid>
    <length>
        1128
    </length>
    <metadata>
        de,13.01.2005
        <scheme>
            <name>
                keywords
            </name>
            <content-type>
                text/plain
            </content-type>
        </scheme>
    </metadata>
    <address>
        jxta://uuid-59616...636303/CMS/jxta-NetGroup
    </address>
</jxta:ContentAdvertisement>
```

Table 1: A sample JXTA advertisement for the Pelicanus Crispus.

Each peer stores shared content in the share folder. The share folder should be located within the webserver, such that all shared files are immediately available for download. File downloads rely on JXTA's pipe communication facilities, which is set up automatically by the CMS. Since every zoo probably has hundreds of animals, the number of requests in the P2P network would be immense by generating search strings for all possible combinations of the parameters animal name, description level of detail (level ”a” for a short description, level ”b” for a detailed presentation, and level ”c” for information concerning s specific zoo or zoo animal, which should not be shared with other zoos) and language. All peers in the jxta-ZooGroup in this demonstration receive the discovery request of the query. Then, the remote peers look into their cached advertisements, and if they find the appropriate content advertisements, reply with a discovery response message which contains a list of the found content.

After receiving the search results, the information sharing and requesting module downloads the corresponding XML file containing the animal presentation. Like in a web page, this XML file may also contain links to additional graphics files like JPG or GIF, which also must be downloaded. After the download is completed, the module then adds entries for the respective animals, languages and levels of detail to the corresponding tables in the MySQL database, thus enabling the webserver to send the new content to the zoo visitors.

7 Discussion of the Work and Future Outlook

Our system relies on the availability of modern PDAs with WLAN interfaces and RFID readers. This demands high investments when implementing the system. The current prototype also relies on the existence of a WLAN infrastructure covering the whole zoo area. Due to shadowing or fading, connection problems between PDAs and WLAN access points may arise for zoos with large premises. Using WLAN can be avoided if the whole content is stored directly on each PDA, which limits the amount of content and the number of languages used. Alternatively, the content could be loaded to a particular PDA on demand, i.e., at the time when the visitor is handed over the PDA.

Additional efforts must be spent on the maintenance of the PDAs, the Web server and the content sharing. On the other hand a lot of work is saved by sharing content in different languages. However, some effort must be put into the evaluation of downloaded content, to ensure a minium quality of the texts.

A straight forward improvement of our prototype would be to include audio and video presentations. Similar to text and pictures, videos may be streamed on demand to the PDAs from a server, if the visitor wishes so. When sharing such content, care must be taken in order to avoid infringing copyrights for audiovisual presentations. Also, audio visual presentations can be offered in various formats (MS WMV, MPEG4, Real, etc.), demanding special player software to be installed on the PDAs.

Additional services are of course thinkable. The PDA may offer special tours through the zoo, for instance by showing the tour path on a map. After the visitor leaves, a summary of the seen animals can be accessed over the Web or be mailed to the user by e-mail. Also, the web application may offer a quality rating to the visitor to give feedback to the zoo management.

8 Conclusion

In this paper we present a ubiquitous information system for zoos. Visitors are handed over a PDA with WLAN and RFID reader, by holding the PDA near a RFID tag, the visitor is presented multimedia information about the zoo animals.

The main innovation of our prototype is the use of a peer-to-peer system, which enables the sharing of animal presentations between zoos on a global scale. Especially the exchange of presentations in various languages is of great interest for zoo visitors. This sharing decreases the effort for creating content significantly. Of course, sharing only makes sense if zoos worldwide show animals of similar race, an assumption which is valid for sure for many zoos and animals.

References

[1] G. Benelli, A. Bianchi, P. Marti, D. Sennati, and E. Not. Hips: Hyper-interaction within physical space. In *IEEE International Conference on Multimedia Computing and Systems Volume II-Number 2*, 1999.

[2] M. Fleck, M. Frid, T. Kindberg, and R. Rajani. From informing to remembering: Deploying a ubiquitous system in an interactive science museum. HPL 2002-54, HP Laboratories Palo Alto, March 2002.

[3] J.D. Gradecki. *Mastering JXTA: Building Java Peer-to-Peer Applications*. John Wiley & Sons, 2002.

[4] S. Hsi, R. Semper, W. Brunette, A. Rea, and G. Borriello. exspot: A wireless rfid transceiver for recording and extending museum visits. In *The Sixth International Conference on Ubiquitous Computing (Ubicomp 2004)*, September 2004.

[5] B. Inderbitzin. Digitale annotation realer objekte aufgrund ortsabhngiger daten am zoologischen museum zrich. Master's thesis, Universitt Zrich, Institut fr Informatik, January 2004.

[6] T. Nishimura, H. Itoh, Y. Nakamura, and H. Nakashima. A compact battery-less information terminal for interactive information support. In *The Fifth Annual Conference on Ubiquitous Computing (Ubicomp 2003), Workshop: Multi-Device Interfaces for Ubiquitous Peripheral Interaction*, 2003.

[7] G. Pospischil, H. Kunczier, and A. Kuchar. Lol@ - a umts location based service. In *International Symposion on 3rd Generation Infrastructure and Services*, July 2001.

[8] A. R. Ramchandran. Plantscanner: A handheld pda using rfid tags for child visitors to the michigan 4-h children's garden. Master's thesis, Michigan State University, Department of Telecommunications, Information Studies and Media, 2004.