Location-Aware Web System

Amir Haghighat, Cristina Videira Lopes, Tony Givargis, Atri Mandal

School of Information and Computer Sciences
University of California, Irvine
{ ahaghigh, lopes, givargis, mandala}@ics.uci.edu

Abstract

We describe the Location-Aware Web System (LAWS), a location-aware system built on top of the Hypertext Transfer Protocol (HTTP). LAWS adds location-dependent virtual browsing experience to the users who are navigating in the physical space with any web browser-enabled roaming device (i.e.: PDA, Smartphone). The distinct characteristic about LAWS is its high degree of modularity and flexibility. By embedding the location information in HTTP requests, we can add location to the system while using existing browsers and existing web servers. This way, there is also, to some degree, independence between the overall system and the positioning system. LAWS can be used in a number of different environments using different positioning systems, and serving different purposes.

1. Introduction

Ubiquitous computing (Ubicomp) is a new paradigm in computing that envisions technology becoming virtually invisible in people's everyday life (Weiser 1991). Context-awareness is one of the major components of invisibility, as systems capture, interpret, and react to users' context. "Context" can be broadly defined; we focus only on location. Much research has been conducted on location-aware systems, and a variety of positioning systems, indoor and outdoor, have emerged ever since (Mandal et al. 2005, Fukuju et al. 2003, Priyantha et al. 2000, Bahl et al. 2000, Want et al. 1992, just to name a few). Location-aware applications interpret the location data obtained from positioning systems and provide the user with location-dependent information. Most location-aware systems lack generality, as they are mainly designed to serve a particular purpose in a particular environment using a particular positioning system. There is the need for a general architecture that can be used in building location-aware systems. In this general architecture, unplugging the positioning system and plugging-in another one into the location-aware system should not require heavy modification on the overall system. This architecture should also be general enough to work in different environments that we envision being powered by location-aware systems in near future.

In this paper we describe a location-aware system based on a web server, hence called the Location-Aware Web System (LAWS). In LAWS the users are provided with web pages on their roaming device's interface that are dynamically generated based on their location. In case of LAWS, we split the location-aware application into a client application running on the users' devices, and a server application that is a typical web server (i.e. Apache). In this approach the client application communicates with the server through the Hypertext Transfer Protocol (HTTP). The HTTP requests sent to the web server are appended with the client's location information, which is in turn used by the web server to dynamically generate location-dependent web pages. These pages are then transferred back to the user's roaming device through a wireless connection (802.11b) as the hardware infrastructure and HTTP as the protocol of communication.

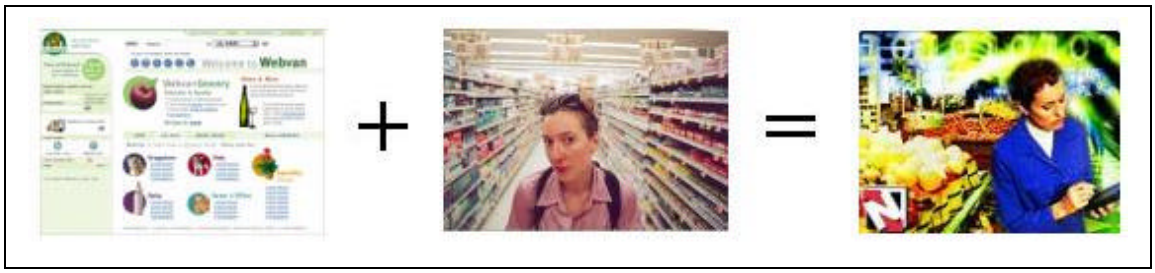

Figure 1: The goal of LAWS is to add location-aware virtual browsing to the users' experience in the physical space (i.e.: supermarket)

2. Location-Aware Web System

The main inspiration behind LAWS is to add location-dependent virtual browsing experience to the users who are navigating in the physical space with a roaming device. This way the users know where they are in the physical space (i.e.: grocery store) and are able to locate items or places of interest that they are looking for, either through a map that is shown on the roaming device or through a reference point to the item's location (i.e.: Aisle 2). The users are provided with location-dependent, dynamically generated web pages that are transferred to their roaming device through the wireless connection. An example use of LAWS in a supermarket is depicted in Figure 1. In this case the generated location-dependant web page can include information such as promotional items close to the user, items they have bought last time they were at that location, etc. The users can also search for an item and see the location of that item, as well as their own location on a map generated by the web server.
LAWS follows the "evolution, not revolution" strategy by simply using the commonly available and powerful Hypertext Transfer Protocol (HTTP) for passing the location information from the client to the server. Since HTTP is so ubiquitously available, we can generalize and claim that LAWS can work with any web server and any web-enabled roaming device regardless of its hardware, operating system, etc. In our approach the application installed on the user's roaming device fetches the user's location from the positioning system, appends the location information to a URL and sends the HTTP request to the web server. This use of HTTP as a location-aware "bridge" assists us on achieving our goal of modularity in the system. This way there is a high degree of independence between the positioning system and the location-aware application that provides the content to the user.

3. LAWS Architecture

Figure 2 depicts the high level structure of LAWS. Through the PDA's user interface, the user requests to be located within the physical space. This request is then forwarded to the URL Generator, whose responsibility is to fetch the user's location from the positioning system, append that location information to a URL, and send the HTTP request to the web server. The web server will then translate the X,Y coordinate to a logical location (i.e.: Aisle 2) and generate an HTML page for that specific user situated in that specific location. The web server and the positioning system components need no further explanation since they are not modified by LAWS. In other words any positioning system and any web server can be used in LAWS.

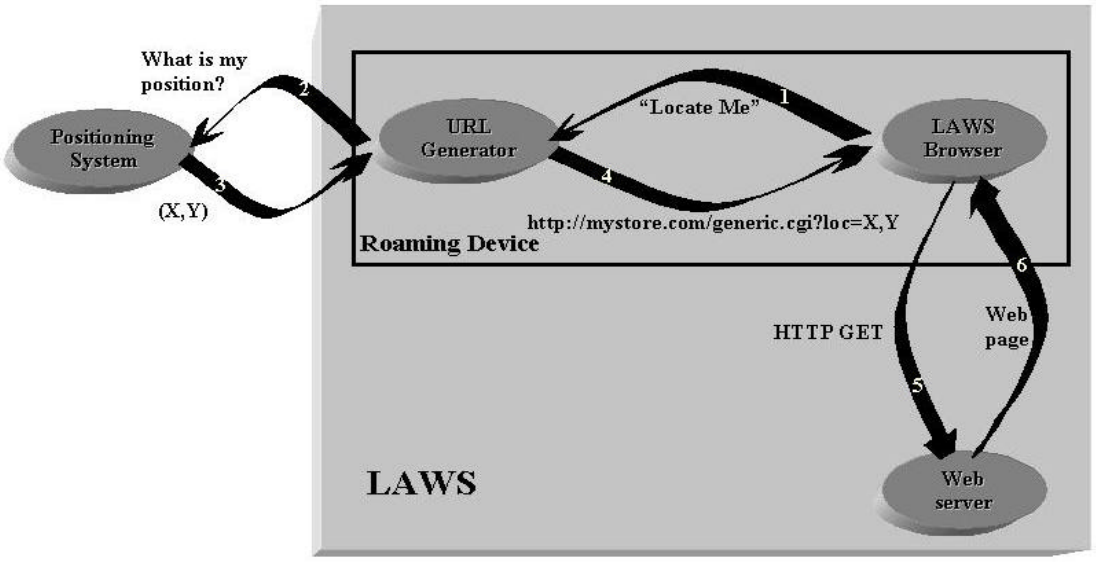

Figure 2: LAWS high-level architecture

3.1 LAWS Browser

As shown in Figure 3, the LAWS Browser is a modified version of the Pocket Internet Explorer that comes with most PDAs. The modification is so that there appears a button on the web browser with "Locate Me" as the caption. Clicking this button triggers a call to the URL Generator component, which interfaces with the positioning system. The URL Generator then passes a position-appended URL back to the LAWS Browser and the LAWS Browser in turn navigates to that URL.

3.2 URL Generator

The URL Generator is a software component that is installed on the user's roaming device and interfaces with the positioning system. It fetches the user's position from the positioning system, appends the position to a URL, and sends the generated URL back to the LAWS Browser which in turn will navigate to that URL. The URL Generator is the only component in LAWS that needs to know about the positioning system. For example in the case of our experiment, we used our own in-door position system. The resulting X-Ycoordinates are then passed by the positioning system to the URL Generator. The coordinates are appended to a URL and passed back to the LAWS Browser. The LAWS Browser will then navigate to that URL which results in displaying a location-dependent web page.

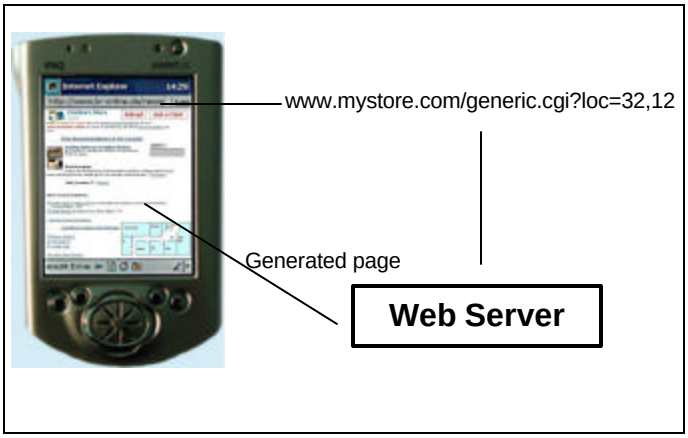

Figure 3: Snapshot of the LAWS Browser after the user attempts to be located

4. Related Work

Our work is based on the large body of work applied in traditional settings (i.e.: desktop/server computing platforms). However, it gets its inspiration from recent experimental ubiquitous computing systems developed, for example at PARC (Want 1995, Newman 2002), HP Labs (Cooltown), Georgia Tech (Aware Home, Abowd 2002) and Berkeley (Dey 2001, Mankoff 2003), among others.

The Cooltown project (Cooltown, Kindberg et al. 2000, Caswell and Debaty 2000) has been adapting the existing web and wireless infrastructures to support nomadic users, creating a bridge between physical entities and electronic services. In Cooltown, physical entities are divided into people, places and things, and they all have a web presence. Our project shares some ideas with Cooltown. Specifically, we advocate Cooltown's "evolution not revolution" strategy with respect to the computing infrastructures that are already in place. The use of HTTP and URLs as the basic messaging mechanism of the system is also a common theme. However, our project diverges from Cooltown in some important aspects. The notion of location does not exist explicitly in Cooltown. In Cooltown there are places, and places are "contexts for service provision, based on an underlying physical domain permeated by one or more networks." So Cooltown's places have the same intent as our physical places, but Cooltown stops short of locating users within those places. Our project is a lot more focused on addressing the issue of adding context-dependent virtual browsing to the user's physical navigation.

At Georgia Tech they have developed a system called iShop (Newcomb 2003), which can assist shoppers in a supermarket. In this case users upload their shopping list on the PDA, which will in turn generate a path that the user can take in the store in order to pick up all the items on their list. The main difference between our system and iShop is the fact that iShop does not locate the users in the physical space. It simply displays the map on which a path is outlined for the user to follow. Moreover, iShop is specifically designed for use in a supermarket, while we envision LAWS being applied in many different environments (i.e.: libraries, museums, etc).

A commercial software product named Vindigo provides PDA users with location-aware information such as restaurants and theaters around them. The difference is that in Vindigo the user has to know their location since the system does not have positioning capabilities. Another main characteristic about Vindigo is that the entire database is completely uploaded onto the user's PDA, hence eliminating a need for a wireless connection.

5. Conclusion and Future Work

We have developed a system that can be used in a variety of different physical places in order to provide the users with the added values that the virtual world brings to their experience. Shoppers in a supermarket can search by keywords, lookup the item's location on a map, and be reminded of items they bought during their last visits when they pass them by. In a museum people can view additional information on an art piece once they walk by it. In a library they can get suggestions based on their location and their profile, locate a book they are looking for, and read reviews on it. LAWS enables all that in a modular fashion in a way that the web server and users' roaming devices can change without imposing any modification on the system. The positioning system is also abstracted out of LAWS for the most part in a way that if it is changed, it will only result in the URL Generator component being updated in order to be able to interface with the new positioning system.

In the future we will focus our research in surveying the field of location-aware computing. In particular, we will survey the different architectures used in building location-aware systems.

6. References

Abowd, G.D., Mynatt, E.D. and Rodden, T. "The Human Experience," In IEEE Pervasive Computing, 2002.

Aware Home, http://www.cc.gatech.edu/fce/ahri/

Bahl P. and V. N. Padmanabhan, "RADAR: An In building RF-based User Location and Tracking System," In Proceedings of the Nineteenth Annual Joint Conference of the IEEE Computer and Communications Societies, INFOCOM 2000, Vol. 2, pp. 775-784, March 2000

Casewell D. and Debaty P. "Creating Web Representations for Places." In Proc. of the 2nd International Symposium on Handheld and Ubiquitous Computing (HUC2K) Bristol, UK, September 2000, pp 114-126

Dey, A.K., Salber, D. and Abowd, G.D. "A Conceptual Framework and a Toolkit for Supporting the Rapid Prototyping of Context-Aware Applications," Special issue on Context-Aware Computing, Human-Computer Interaction (HCI) Journal, Vol. 16(2-4), 2001, pp. 97-166.

Fukuju, Y. et al, "DOLPHIN: An Autonomous Indoor Positioning System in Ubiquitous Computing Environment," IEEE Workshop on Software Technologies for Future Embedded Systems,pp.53-56, May 2003

Kindberg, T. et al. "People, Places, Things: Web Presence for the Real World." In Proc. of 3rd Workshop on Mobile Computer Systems and Applications, Dec. 2000 pp 19-28.

Mandal, A., Lopes, C., Givargis, T., Haghighat, A., Jurdak, R., and Baldi, P., "Beep: 3D Indoor Positioning Using Audible Sound" In Proc. Of CCNC Las Vegas, NV, Jan. 2005 (to appear)

Mankoff, J., Dey, A.K., Hsieh, G., Kientz, J., Ames, M., Lederer, S. "Heuristic Evaluation of Ambient Displays," In Proc. of CHI 2003.

Newcomb, E., Pashley, T., and Stasko J. "Mobile Computing in the Retail Arena" In Proc. of CHI 2003.

Priyantha N.B., A. Chakraborty, H. Balakrishnan, "The Cricket Location-Support system," In Proc. ACM MOBICOM, August 2000.

Weiser M., "The computer for the 21st Century," Scientific American, 265(3):94--104, September 1991.

Newman, M.W., Sedivy, J.Z., Edwards, W.K., Smith, T.F., Marcelo, K., Neuwirth, C.M., Hong, J.I. and Izadi, S. "Designing for Serendipity: Supporting End-User Configuration of Ubiquitous Computing Environments." Proceedings of Designing Interactive Systems (DIS), (London, UK, 2002), 147--156.

Want, R. Hopper A., Falcao V. and Gibbons J. "The active badge location system." ACM Transactions on Information Systems 10(1):91-102. January 1992.

Want, R., Schilit, B, Norman, A., Gold, R., Petersen K., Ellis, J. and Weiser, M. "An Overview of the Parctab Ubiquitous Computing Experiment." IEEE Personal Communications, December 1995, Vol 2 No. 6, pp 28-43.