一九九六分散式系統技術及應用研討會
1996 Workshop on Distributed System Technologies and Applications

An Infrastructure for Mobile Computing with Intelligent Messaging: Implementation Issues

Wen-Shyen E. Chen*, Yao-Nan Lien**, and Wen-Yee Hsien*

*Institute of Computer Science
National Chung-Hsing University
Taichung, Taiwan, ROC
email: echen@cs.nchu.edu.tw

**Department of Computer Science
National Cheng-Chi University
Taipei, Taiwan, ROC
email: lien@cherry.cs.nccu.edu.tw

Abstract

The past few years have witnessed the new developments in computer networking: Applications such as the World-Wide Web (WWW) has become a popular means of information dissemination for users with or without computer skills. Since the client/server architecture for WWW have been designed for workstations connected by high-speed networks, their fixed policies and fixed interfaces prevent them from working effectively on other computing platforms. In particular, since wireless mobile computers do not have high-bandwidth connections to the Internet and are limited by small memory and display area, they are quickly overwhelmed by the data included in many multimedia WWW documents if conventional architecture of accessing WWW information are used.

In this paper, we propose an infrastrcuture that employs intelligent messaging/agent techniques to provide agent mobility across heterogeneous networks to address the issues in allowing mobile users to access the WWW. A prototype that implements this infrastructure is also described. The current WWW server structure will remain intact to provide maximum compatibility.

Keywords: Mobile Computing, Intelligent Messaging, World-Wide Web, Email System, Script, Tcl, Java

1. Introduction

Information dissemination is likely to be one of the driving applications of mobile computing. New, small mobile devices that integrate the functionality of phones, faxes, pagers, and other communication tools are becoming available at low price. Popular information retrieval tools need to be supported on this new technology to allow users to have information at their fingertips everywhere. Recently, the World-Wide Web (WWW) [1,2], together with its graphic browsers, such as NCSA Mosaic [3], have become popular for information dissemination. Since most of the WWW browsers have been designed to run on workstations connected by high-speed networks, their fixed policies and fixed interfaces prevent them from being able to easily incorporate new applications or running on other computing platforms. In particular, since wireless mobile computers[1] do not have a high-bandwidth connection to the Internet and are limited by small memory and display area, they are quickly overwhelmed by the data included in many multimedia WWW documents. As a result, to provide the mobile computers with the compatible power to that of the stations with fixed connections so that they can participate in the information distribution in the Internet is an important problem to be addressed.

In this paper, we propose an infrastructure that employs intelligent messaging techniques, coupled with an enhanced electronic mail system, to address the issues in allowing mobile users to access the vast amount of information provided by the WWW. The current WWW server structure will remain intact to provide maximum compatibility. A prototype system is implemented to illustrate the feasibility of the infrastructure and provide some insights for the future full implementation.

The rest of the paper is organized as follows. Section 2 gives an overview of wireless communication and mobile computing. Section 3 introduces the current approaches to providing WWW browsers for mobile users. Section 4 discusses the intelligent messaging approach we propose. Section 5 presents the implementation issues of our approach. Concluding remarks are given in Section 6.

[1] In this paper, we will use the term *mobile computer* to denote a portable computer that is connected to the network through a mobile link.

2. Overview

2.1 Characteristics of Wireless Communication

Wireless communication is much more difficult to achieve than wired communication because the surrounding environment interacts with the signal, blocking signal paths and introducing noise and echoes [13,14,15]. As a result, wireless connections have lower bandwidth, high bandwidth variability, higher error rate, and more frequent spurious disconnection. These factors can in turn increase communication latency due to retransmissions, retransmission timeout delays, error control protocol processing, and frequent disconnections. They also more expensive than wired counterparts.

2.3 Mobile Computing

Wireless networking/communication greatly enhances the utility of carrying a computing device. It provides the mobile user with versatile communication to other people and expedient notification of important event, yet with much more flexibility than cellular phones or pagers. Continuous access to the services and resources of the wired network is also made possible. The combination of networking and mobility will create new applications and services. However, the impairments of the underlying wireless communication infrastructure, along with the physical constrains of the mobile computers, handicap the establishing of this paradigm of computing [13,14].

Compared with traditional wired communication, wireless communication has low bandwidth, long connection time and low reliability. Radio signal is easily affected by noise and interference. In addition, wireless computing is more susceptible to disconnection. Consequently, traditional protocols, such as TCP/IP, may suffer much performance degradation or even cease to function when used directly in such an environment. For a mobile application to become viable, all these factors must be taken into consideration and the underlying protocol has to be efficient, robust, be able to cope with disconnections more gracefully and work around them whenever possible.

Intelligent Messaging

Simply put, an intelligent message [12] is an electronic message that carries a computer program, whether procedural or declarative, which can be executed by the computer system of the recipient on behalf of the message. The program can instruct the recipient computer to forward automatically the message itself to another server to be executed. Consequently, after submitting an intelligent message to the network, the mobile computer can then disconnected from the network until the final results are returned. As mobile computers requiring wireless network access often face more frequent disconnection, lower bandwidth, greater variation in available bandwidth, and greater network heterogeneity, it is argued that the intelligent message can better cope with the challenges presented by mobile computing, especially in the WWW environment, as illustrated in Fig. 1.

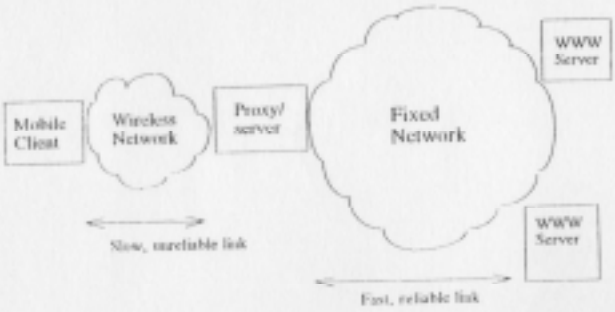

Figure 1. An Environment for Mobile Host to Access the WWW.

3 Current Approaches

There are several approaches in the literature [16-21] to help the mobile computer to connect to the WWW. The consensus is that full implementation of the browser on the mobile computer, along with the TCP/IP protocol stack and the HTTP [9] transport protocol for the WWW, might be inadequate for mobile computers because the hardware limitations. In addition, HTTP might perform poorly with TCP/IP in the mobile environment because the unreliable nature of the mobile link might tend to reset the congestion control window and restrict the flow of data [10].

One of the more promising approaches employs *indirect client-server interaction* model, where the WWW browser application is split between a mobile computer (mobile host) and its base station (mobile support host), called *proxy* [18]. Using this approach, a client-server interaction in the WWW environment involves at least two separate interactions - one on the wireless medium between a mobile host and its mobile support host, and another on the wired medium between the mobile support host and a fixed server (WWW server). A possible implementation is shown in Fig. 2. To adapt to the

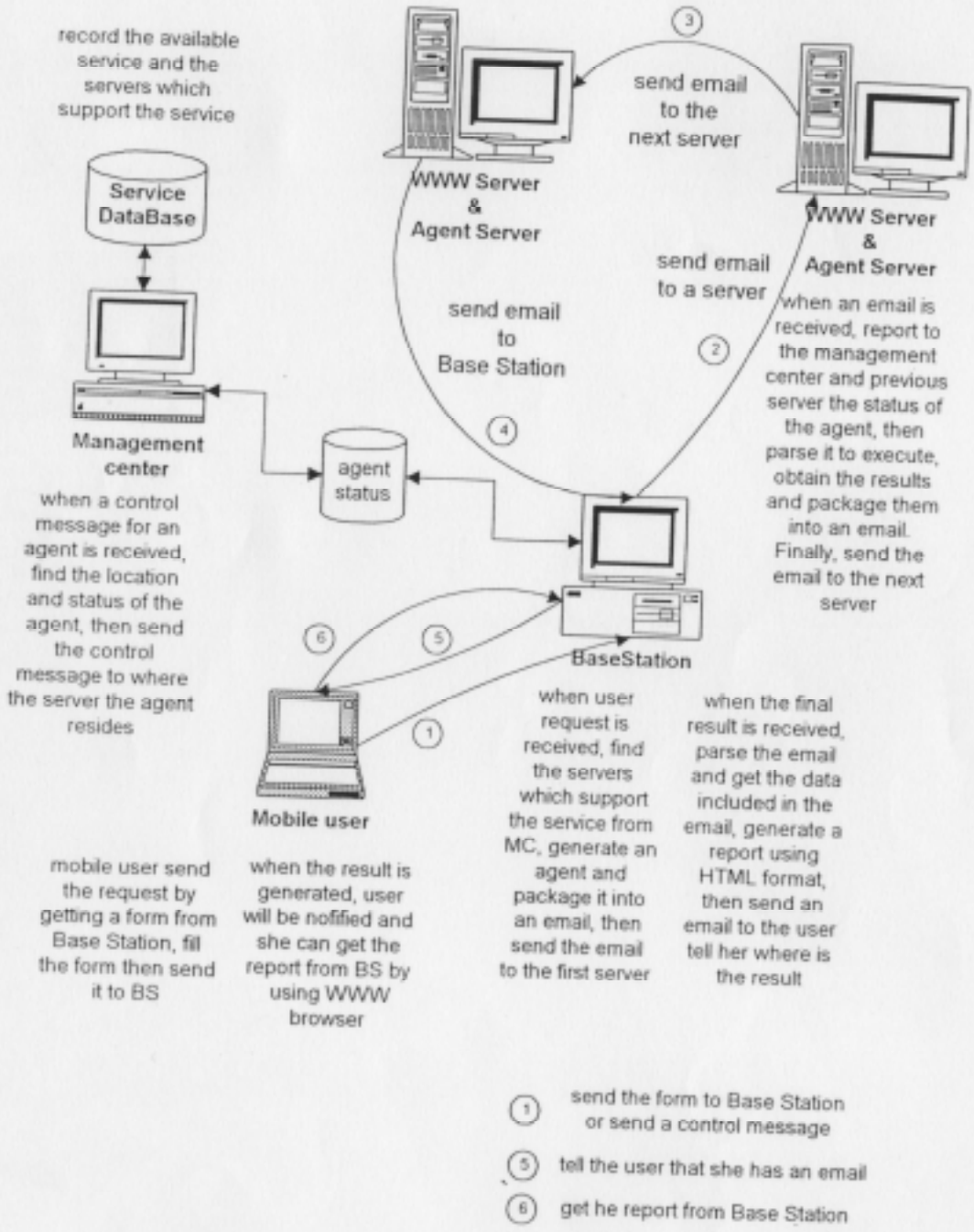

Figure 3. Intelligent Message in the WWW architecture

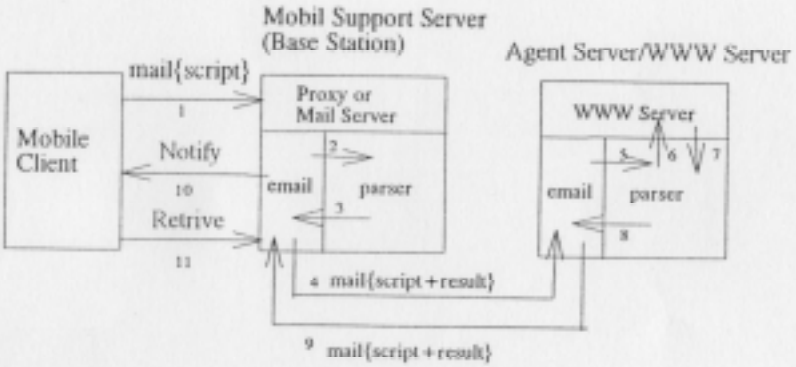

Figure 4. The flows of the intelligent messaging in WWW architecture.

along with the results, to the second destination, the WWW server 2. WWW servers 2 and 3 repeats this operation and then forwards the results back to the originating mobile support server. It then formats the result receives and signals the mobile user about the arrival of the final result, and the mobile user will retrieve the information from the mobile support server. We expect this approach, to be compact, more user-customizable and efficient enough to expand the capability of information retrieval in the mobile environment. Fig. 4 shows the flows of the intelligent messaging in the WWW architecture. Some implementation issues of this approach will be described in the next section.

5 Implementation Issues

We have presented in the previous section the conceptual design of the architecture for applying intelligent messaging technique in the WWW.

- **Intelligent Message Transport Mechanism**: Since the intelligent message will be traveling about the WWW servers, there must be a mechanism that enables the shipping of the script and the intermediate results from one server to another. The mechanism should requires **minimum system modification, be available in the current systems**. and **be compatible with existing WWW servers implementations**. Some candidates come to mind are remote shell [22] in UNIX system and "MIME Extension" [25,26] for electronic mail (email), since they are readily available in many systems.

  In our prototype implementation, we use e-mail as the intelligent messages transport mechanism. A special content type is defined for the e-mail system and therefore requires minimum changes to the existing systems. The normal e-mails can be delivered as usual. Only when a message with the special content type for the intelligent message is received will a processing engine be involved to process the script enclosed in the message.

  When an e-mail message arrives at a server, it is forwarded to a "parser" specified in the ".forward" file for processing. If the content-type [29] is not of "application/Intelligent_Messaging," the message will be processed normally. Otherwise, the script and the intermediate results included in the message will be retrieved and passed to the script interpreter for processing. When the interpreter encounters the "mailto" command in the script, it packages the script and the results, then sends it to the next server determined by the script or other conditions. Note that "mailto" is an extension to the script language of choice. Since the *sendmail* mail delivering system does not require that the mobile client to be connected to the mobile support server at all time, the client can disconnect the mobile link to the mobile support station after it submits the intelligent message. The base mobile support station will notify the mobile user with an email when the final result arrives.

  An example script for retrieving two pictures from two different servers is shown in the Appendix. The Appendix also shows the email message generated for user notification and the final results being viewed by an WWW browser

- **Selection of Script Languages:** Currently, there are two good candidates: **Safe-Tcl** [25] and **Java** [28]. Their characteristics are described as follows:

a. **Safe-Tcl:** A Tcl [23] based language to provide the basic syntax and many of the primitives of Tcl. Tcl is a well-designed language to be embedded as a computational extension to a larger application. It has the virtual of being simple, well defined, and available in a high-quality, extremely portable public domain implementation. Safe-Tcl provides improvements over vanilla Tcl to meet the constrains of security, interface portability, and cross-platform availability presented in intelligent messaging system. It's code is also available in public domain and offers a X/MIT style license.

b. **Java**: A language similar in syntax to C++. The Java [28] language environment is interpreted and dynamic. The Java compiler generates byte codes for the Java Virtual Machine. The notion of virtual interpreted machines is not new. But the Java language brings the concepts into the realm of secure, distributed, network-based systems. The system includes a byte code compiler and a virtual machine runtime. The runtime is typesafe and supports a form of secure loading, so that the code from untrusted source can be added dynamically. The Java language virtual machine is a strictly defined virtual machine for which an interpreter must be available for each hardware architecture and operating system on which you wish to run Java language applications. Once the Java language interpreter and run-time support become available on a given hardware and operating system platform, any application in Java language script can be imported from anywhere.

We use Java in our prototype implementation because of its descriptive power and rich features in security.

- **Security:** Since the intelligent messages have to be executed the remote servers for this mechanism to work, the messages have to be trusted, i.e., they perform only actions not harmful to the remote servers. As a result, when selecting the script language, special attention has to be paid to the security aspect. Three levels of security need to be considered:

  a. **Script Language:** The script language needs to be designed so that no harmful effects can be done.

  b. **Server Verification:** The server needs to be able to verify the integrity of the script it receives to ensure no harmful action is embedded in it.

  c. **Access Control/User Authentication:** The server will want to define actions that are available only for intelligent messages from certain trusted senders. In addition, different users might be given different access rights to various resources at the server. This implies that the server should be able to understand and validate authenticated intelligent messages.

- **Other service considerations:** Mechanisms has to be provided so that after the user submitted a intelligent message, some controls can still be applied to that message. The services for control of the message include atomicity maintenance, location of the agent, status inquiry, etc. [12].

6 Conclusion

Since mobile computers do not have a high-bandwidth, reliable connection to the fixed network and are limited by small memory and display area, to provide the mobile computers with the compatible power that of the stations with fixed connections so that they can participate in the information dissemination in the Internet will be an important problem to address.

In this paper, we have proposed an infrastructure that employs intelligent messaging techniques, together with an enhanced electronic mail system, to allow mobile users to access the vast amount of information provided by the WWW. The current WWW server structure will remain intact to provide maximum compatibility. The major difference between this approach and the conventional information retrieval in the WWW is that servers collaborate according to the scripts travels across them and perform functions specified in the scripts on behalf of the mobile client. Therefore, a mobile computer can better cope with its physical limits and frequent link disconnection.

A prototype system is implemented to illustrate the feasibility of the infrastructure and provide some insights for the future full implementation. However, for such an infrastructure to be viable, more management functions, such as maintenance of transaction atomicity, agents location and collaboration, need to be studied.

References

[1] T. Berners-Lee, *et al.* "The World-Wide Web," *Communications of the ACM*, vol. 37, no. 8, August 1994, pp. 76-82.

[2] T. Berners -Lee, R. Cailliuau, J.-F. Groff, and B. Pollermann, "World-Wide Web: The Information Universe," *Electronic Networking: Research, Applications and Policy, vol. 1, no. 2*, Meckler, Westport, CT., Spring 1992.

[3] R. J. Vetter , C. Spell, and C. Ward, "Mosaic and the World-Wide Web," *IEEE Computer Magazine*, pp. 49-57, October 1994.

[4] "NSFNET Traffic Distribution Highlights," Available by Anonymous FTP from nic.merit.edu.

[5] J. Postel and J. Reynolds, "File Transfer Protocol (FTP)," Internet RFC-959, October 1985.

[6] F. Halasz and M. Schwartz, " The Dexter Hypertext Reference Model," In *Proc. of the Hypertext Standardization Workshop*, January 1990.

[7] C. F. Goldarb, Information Technology-Hypermedia/Time-Based Structuring Language (HyTime), ISO/IEC CD 10744 (Draft).

[8] J. Postel and J. Reynolds, "File Transfer Protocol (FTP)," Internet RFC-959, October 1985.

[9] T. Berners-Lee, "Hypertext Transfer Protocol (HTTP)," IETF Internet Draft (work in progress), 1994.

[10] V. Jacobson, "Congestion Avoidance and Control," In *Proc. of ACM SIGCOMM'88*, pp. 314-329, August 1988.

[11] D. E. Comer and D. L. Stevens, *Internetworking with TCP/IP*, Vols. I and II, Prentice-Hall, NJ, 1991.

[12] Y.-N. Lien, "An Open Intelligent Messaging Network Infrastructure for Ubiquitous Information Service," In *Proc. of the First Workshop on Mobile Computing*, Hsing-Chu, Taiwan, April 1995.

[13] G. H. Forman and J. Zahorjan, "The Challenges of Mobile Computing," *IEEE Computer Magazine*, pp. 38-47, March 1994.

[14] T. Imielinski and B. R. Badrinath, "Mobile Wireless Computing: Challenges in Data Management," *Communication of ACM*, August 1994.

[15] T. Alanko, *et al.* "Measured Performance of Data Transmission Over Cellular Telephone Networks," *ACM SIGCOMM Computer Communication Review*, pp. 24-44, August 1994.

[16] A. DeSimone and S. Nanda. "Wireless Data: System, Standards, Applications," Technical Report, Performance Analysis Department AT&T Bell Laboratories.

[17] S. Gessler and A. Kotulla. "PDAs as Mobile WWW Browsers," *In Proc. of The Second World Wide Web Conference '94*, October, 1994.

[18] T. Watson, "Wit: An Infrastructure for Wireless Palmtop Computing," Technical Report UW-CSE-94-11-08. University of Washington, November 1994.

[19] J. F. Bartlett, "W4 - The Wireless World Wide Web," In *Proc. of the IEEE Workshop on Mobile Computing Systems and Applications*, December 1994.

[20] M. F. Kaashoek, T. Pinckney, and J. A. Tauber. "Dynamic Documents: Mobile Wireless Access to the WWW," In *Proc. of the IEEE Workshop on Mobile Computing Systems and Applications*, December 1994.

[21] G. M. Voelker and B. N. Bershad, "Mobisaic: An Information System for a Mobile Wireless Computing Environment," Technical Report, Department of Computer Science and Engineering, University of Washington, September 1994.

[22] K. D. Kotay and D. Kotz, "Transportable Agents," In *Proc. of the Third International Conference on Information and Knowledge Management*, December 1994.

[23] J. Ousterhout, *Tcl and the Tk Toolkit*, Addison-Wesley, Reading MA., 1994.

[24] N. Borenstein, N. Freed, "MIME (Multipurpose Internet Mail Extensions)" Part One and Two, Inrternet RFC 1521-1522, September 23, 1993.

[25] N. Borenstein, "Email with A Mind of Its Own: The Safe-Tcl Language for Enabled Mail," Available by Anonymous FTP from ftp.ics.uci.edu/mrose/safe-tcl.

[26] N. Borenstein and M. Rose, "MIME Extensions for Mail Enabled Applications: application/Safe-Tcl and multipart/enabled-mail," Internet Working Draft, November 1993.

[27] J. E. White, "Telescript Technology: The Foundation of the Electronic Marketplace," *General Magic White Paper*, 1994.

[28] *The Java Language Environment: A White Paper*, Sun Microsystems Computer Company, May 1995.