Intelligent Messaging for Mobile Computing over the World-Wide Web

Wen-Shyen E. Chen and Yao-Nan Lien[**]

[*]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

Recently, the World-Wide Web (WWW) has become a popular architecture for information dissemination. Since WWW browsers, such as Mosaic, 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 mutil-media WWW documents if conventional architecture of accessing WWW information are used. As a result, to provide the mobile users with 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 architecture that employs intelligent messaging techniques, coupled with an enhanced electronic mail system, to address the issues in allowing mobile users to access the WWW. 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], has become popular for information dissemination. Since 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 mutil-media 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 architecture 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 used to illustrate the feasibility of the architecture and provide some insights for the future full implementation. We envision that the result of this study will make it possible to connect the mobile user to the "information superhighway" in an effective manner.

The rest of the paper is organized as follows. Section 2 gives an overview of the WWW architecture, mobile communication and mobile computing. Section 3 introduces the current approaches to providing WWW browsers for mobile users. Section 4 discusses the

[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.

intelligent messaging approach we propose. Section 5 presents the implementation issues of our approach. Concluding remarks are given in Section 6.

2. Overview

2. World-Wide Web

The past few years have witnessed the accelerated growth of the Internet and the proliferation of the information available in the Net. Along with the growth of the information comes the issues of manageability of the information.

WWW provides a good way to distribute and navigate the information/net [1,2]. The WWW project, started by CERN (the European Laboratory for Particle Physics), seeks to build a distributed hypermedia system. What the WWW project has done is provide users on computer networks with a consistent mean to access a variety of media in a simplified fashion. With popular software interfaces to the Web, such as Mosaic [3], the Web project has changed the way people view and create information -- it has created the first true global *hypermedia* (will be explained later) network. Although in its infancy, the WWW has attracted much attention and its success is evidenced by the fact that the WWW traffic has become the second largest in terms of byte counts in the NSFNET as of November 1994 [4].

The Hypertext Transfer Protocol (HTTP) [9] is the transport protocol for WWW. It is designed to be fast, stateless, extensible, and easy to implement for both clients and servers. However, from our initial observation and some reports in the literature, HTTP do have some deficiencies: it does not work well with TCP, the underlying Internet protocol. As a result, HTTP incurs much unnecessary overhead and leaves some room for improvements.

The operation of the Web relies mainly on hypertext [5] as its means of interacting with users. Hypertext is basically the same as regular text - it can be stored, read, searched, or edited - with an important exception: hypertext contains connections within the text to other documents. The browsers let you deal with the links in a transparent way -- select the high-lighted pointers (either text or icons), and you are presented with the hypermedia document that is pointed to. The advantage of hypertext is that in a hypertext document, if you want more information about a particular subject mentioned, you can usually "just click on it" to read further detail. In fact, documents can be and often are linked to other documents by completely different authors. For instance, take Fig. 1 for example, if you are reading a personal data that contains "links," in this case NTU, OSU, etc., to other documents. You can then double click on the high-lighted text to show more detailed information.

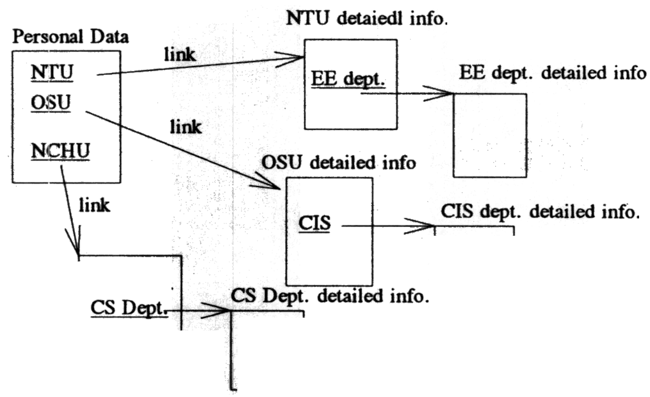

Figure Hypertex Data Model. All Information is local.

Hypermedia [5,6,7] is an extension to hypertext: hypermedia documents contain links not only to other pieces of text, but also to other forms of media -- sounds, images, and movies. Simply put, hypermedia combines hypertext and multimedia. For example, by double clicking on the high-lighted text in a story book, you will be presented not only a text document, but possibly also the pronunciation of the text or even a video clip.

The WWW project is now being carried out cooperatively by many groups and individuals at sites throughout the world. The use of WWW browsers and servers for a wide variety of applications has increased dramatically over the last year [4]. Fig. 2 illustrates that the user of the WWW can access graphic pictures, images, audio clips, or even full motion video through hypermedia retrieved locally as well as sites from around the world. Note that the WWW also supports search mechanisms, making it feasible in interacting with the other databases.

The user can run a browser program to access the WWW. The browser reads documents, and can retrieve documents from other sources. Information providers set up hypermedia servers, which browsers can get documents from. The browsers can, in addition, access files by FTP [5], NNTP (the Internet news protocol), gopher and an ever-increasing range of other methods. On top of these, if the server has search capabilities, the browsers will permit searches of documents and databases.

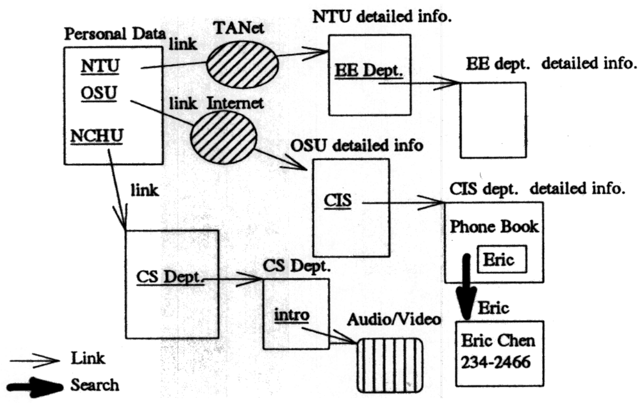

Figure 2. Distributed Hypermedia Data Model for the WWW.

2.2 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. They also more expensive than wired counterparts. These factors can in turn increase communication latency due to retransmissions, retransmission timeout delays, error control protocol processing, and short disconnection.

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 successful, all these factors must be taken into consideration and the underlying protocol has to be efficient, robust, be able to cope with disconnection 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 in a pipeline fashion. 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.

Possible Applications of Mobile Computing with Respect to the WWW

automatic stock quotes/stock purchase

A user can connect to an WWW server which offers the information of stocks through wireless communication network. The updated stock quotes can be delivered automatically to the user's mobile computer, depending on the conditions the user sets, such as when price changed or every 5 minutes. In addition, the user can place an order to a stock broker that offer services via the WWW and purchase stocks at a preset price. When the deal goes through, he or she will be notified.

remote monitoring

A good example for remote monitoring is when a doctor needs to watch for the conditions of his hospitalized patients. Dynamic information, such as the status of patients (pulse, blood pressure, etc.), can be sent to the doctor's PDA anytime, wherever the doctor is.

data entry/browsing

A salesman may need current data (such as inventory information) before a deal can be finalized. He or she can use a PDA to retrieve the data he needs from a base station in his company. A finalized deal can then be entered and the record can be updated instantly.

Though versatile and flexible, the mobile computer suffers from the impairments of mobile communication links and therefore cannot be connected to the WWW in the same way as that of the wired computer. Fig. 3 depicts such an environment. Note that the wireless link is slower, less reliable, and more expensive, whereas the fixed/wired network is fast, reliable, and less expensive.

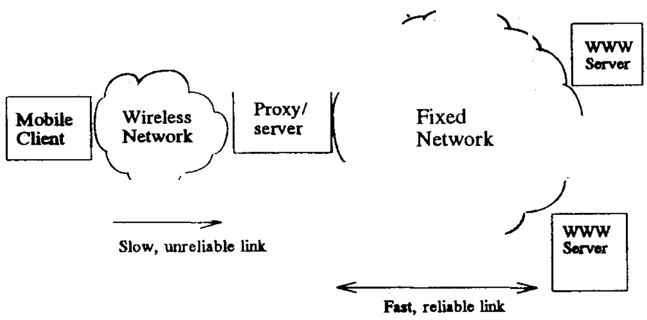

Figure 3. An Environment for Mobile Host to Accee the WWW.

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 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. 4. To adapt to the special display size of a mobile computer, the retrieved data might need to be reformatted to show on a small LCD screen.

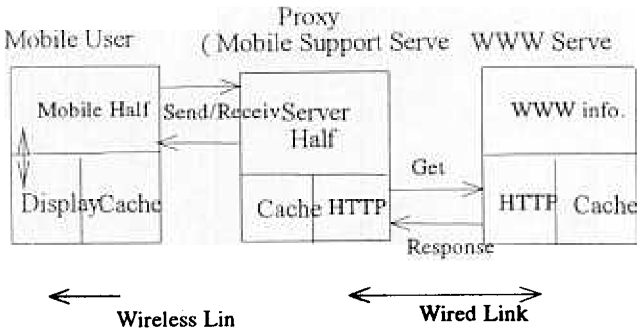

Figure 4. The Indriect Clinet-Server Interaction Model for Mobile Access.

This approach does address the issues of small memory/processing power in mobile computers. However, it might require continuous connection between the mobile user and the mobile support host, which might not be cost effective. In addition, as network information resources continue to grow in size, query-processing efficiency becomes more of a concern. Furthermore, this traditional client-server model requires human intervention in determining what actions need to be taken base on the results of the previously retrieved information, and therefore might require the server to send all the information to the client and being processed at the client side. This is inefficient in bandwidth utilization and ineffective in terms of performance, especially if slow mobile link is used.

4 The Intelligent Messaging Approach

Our objective is to come up with an environment to enable efficient, cost-effective accesses of the WWW from the mobile client. We propose to use "intelligent message," which is a script composed of instructions to be performed, as a means for mobile computers to access the vast amount of information presented in the WWW. Traditionally, the HTTP protocol is used for the communications between clients and servers in the WWW. The client has to interact with one server at a time to retrieve information, deciding what further actions needs to be taken based on the information retrieved. In addition, since human intervention is needed in this case, the mobile link needs to be up all the time for the HTTP traffic to flow in an interactive manner. This translates to higher cost.

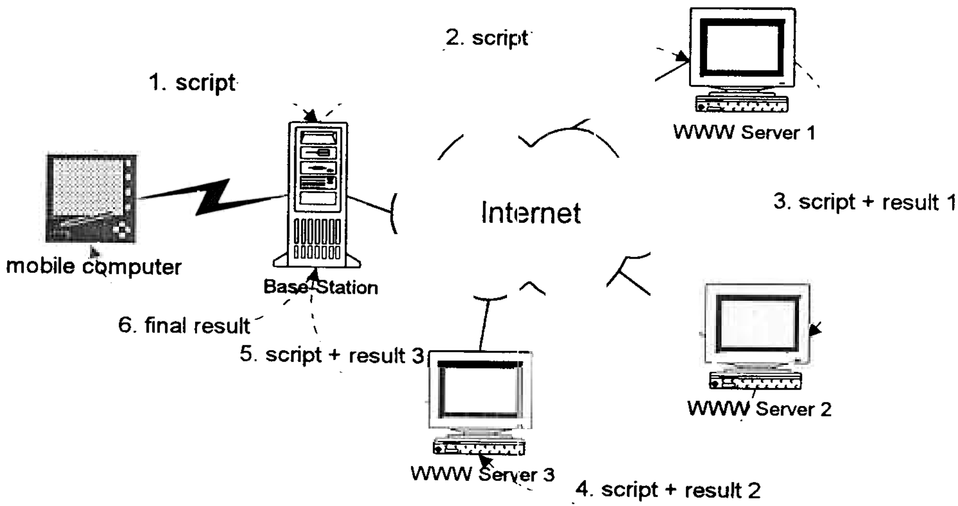

Figure 5. Intelligent Message in the WWW architecture

In our approach, the *indirect client-server interaction* model as mentioned in the previous section is adopted. However, in contrast to the conventional approach, the "request" in the intelligent messaging approach contains not only a simple "GET information," command, but a "program" to be executed at the remote sites (WWW servers in the case of WWW architecture) when they receive the message. This approach is well-suited in the mobile environment: it is faster, consumes fewer network resources when the interaction between a relocatable intelligent message and the information resource is local (in the server) rather than over a network. Furthermore, the ability to travel allows intelligent messages to support systems that do not have permanent network connections. The script in the intelligent message will enable the servers to perform cooperative works for the mobile client and therefore, providing a more flexible ways to access information in the Internet. The idea of this approach is illustrated in Fig. 5. **Note that the mobile computer does not need to have a mobile support server if the user choose to provide the mail server function on the mobile computer and he or she can provide continuous connection to the Internet. This makes the architecture more flexible. In addition, since the mail service locates above the transport layer, the "mobility" of the client is not visible to the server.**

As shown in Fig. 5, in the case of the WWW architecture, the mobile user selects his or her action via the WWW browser interface. A script is then generated and the mobile computer prepares to submit the script to its mobile support server. The mobile computer can then be disconnected from the network. The script lets the user specify not only static message information, but also "intelligence," in the form of a script language that gives additional instructions, such as telling the intelligent message how to move through the network or what to do when it arrives at a server. In the example of the WWW architecture, it specifies the stations an intelligent message needs to visit, the URLs it wants to retrieve, the task it wants to perform, and the format of the results it expects, etc. The mobile support server can be the mail server of the mobile computer. When the mobile support server receives the e-mail message, it examines the message, parses the information according to a predetermined format and deliver it to the script's first destination, assuming to be WWW server 1. The WWW server 1 accepts the intelligent message, interprets the script, carries out the task, and forwards the message, 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. 6 shows the flows of the intelligent messaging in the WWW architecture. Some implementation issues of this approach will be described in the next section.

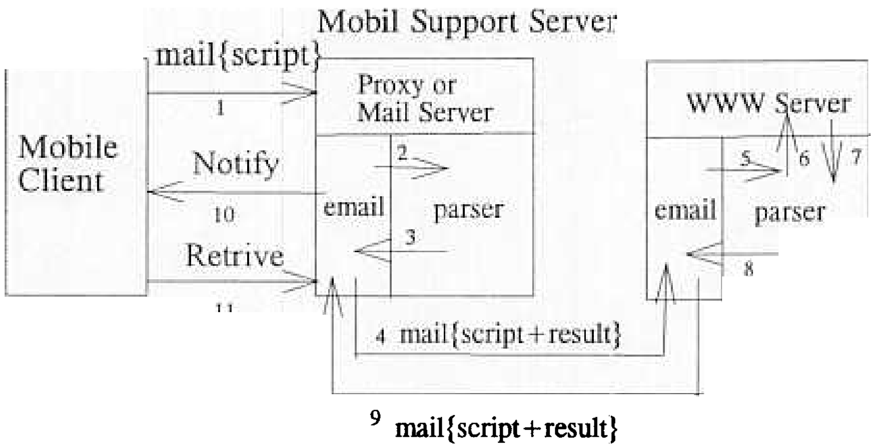

Figure 6. The flows of the intelligent messaging in the WWW architecture.

5 Implementation Issues

We have presented in the previous section the conceptual design of the architecture for applying intelligent messaging technique in the WWW. To realize the architecture, there are several issues to be addressed:

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 (e-mail), since they are readily available in many systems.

In the prototype, we use e-mail as the 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.

In our prototype implementation (with UNIX servers running *sendmail*), 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 possibly intermediate results, included in the message will be retrieved and passed to the script interpreter for processing. When the interpreter encounters the "SENDTO" 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 "SENDTO" 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 he mobile client can notify the mobile user when the final result arrives.

An example script for stock trading is shown in the Appendix.

- **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.

- **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 Internet and are limited by small memory and display area, they are quickly overwhelmed by the data included in many mutil-media 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 dissemination in the Internet will be an important problem to address.

In this paper, we have proposed an architecture that employs intelligent messaging techniques, coupled 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 used to illustrate the feasibility of the architecture and provide some insights for the future full implementation. We envision that the result of this study will make it possible to connect the mobile user to the "information superhighway" in a more effective manner.

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. -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.