FIGI: The Architecture of an Internet-based Financial Information Gathering Infrastructure

M. D. Dikaiakos

Department of Computer Science
University of Cyprus
Nicosia, CYPRUS

D. Gunopulos

Department of Computer Science
University of California, Riverside
Riverside, CA 92521

Abstract

In this paper we present the architecture of a Financial Information Gathering Infrastructure (FIGI). FIGI helps investors collect, filter, combine and integrate portfolio-related information provided through various Internet services like World-Wide Web sites and Web-databases. FIGI is being developed with Java-based Mobile Agent technology by Mitsubishi Electric Information Technology Center. The employment of Java and Mobile Agents provides us with a framework for unifying the various financial information services currently available on Internet and for sustaining continuous information provision even to mobile users.

1 Introduction

The convergence of computer and telecommunication technologies has brought an "information explosion" which is testified by the extensive amounts of information made available to people on wide-area networks like Internet. Access to this information is provided through a variety of networking tools designed for posting, searching, browsing, and integrating information; e.g. Usenet News, Wide-Area Information Servers (WAIS), the World-Wide Web [1], Web Search Engines [3], Information Dissemination Servers [10]. The surfeit of digital information has inundated the global networking infrastructure and is influencing heavily the interaction of people with information networks. This information explosion puts additional burden on information seekers and raises the need for advanced information acquisition, filtering and integration tools [9, 4].

This need is becoming obvious in the field of *portfolio monitoring*. Currently, Internet services like the World-Wide Web, electronic mail and Usenet News are used extensively to support investors with an ongoing, continuous provision of up-to-date information on financial markets. Such information includes market data, financial reports, technical models, analysts' reports, and breaking news. This information is being used by investors in their effort to establish and maintain an integrated financial picture that will help them manage their investment portfolios over time. However, the exploding volume of financial information provided on wide-area networks makes it difficult for investors to keep up with the fast pace of information generation.

For example, let us look at George, an investor who uses the Internet to keep track of his portfolio. George usually starts the day by seeing what has happened. To this end, he takes a tour of his *personal information space*. First, he checks his electronic mail; George is subscribed in a number of e-mailing lists for investors, where he exchanges opinions on investments. He reads some of the e-mails that have arrived at his e-mail box, deletes some messages and stores other. Then, George spawns a Web browser and navigates through various Web sites that provide stock quotes and indices, technical and fundamental analyses, breaking news, etc. As starting point for Web browsing he uses a number of Web pages registered in a personal meta-index directory ("bookmark" file). Depending on the information discovered, George may follow hyperlinks to sites outside his bookmarks or he may connect to a Web Search Engine to search for information on particular issues of interest. George collects and stores information from the Web pages visited either via cut & paste mechanisms or by saving content on his local disk. Furthermore, he updates his bookmarks with newly discovered Web sites of interest. Finally, George spawns a Usenet News reader and reviews a number of newsgroups circulating information on financial markets, investment opportunities, etc. Again, he reads and processes information, taking further actions as needed.

George's goal of getting informed about financial markets is by itself well defined and focused. However, in the process of seeking information, he is obliged to use disparate systems and to adapt his activity to their different functionalities and interfaces, which adds to his cognitive load. Furthermore, while reviewing information collected from e-mailing, searching and browsing, George easily becomes "disoriented" and goes astray in the information space.

The situation becomes even worse if George travels often. Being on the move, George cannot access his local network and desktop environment where he maintains the latest version of his Web bookmarks, e-mail boxes, e-mail aliases, Usenet news configuration files, etc. Thus, he has to rely on a portable computer or a Personal Digital Assistant device and a slow Internet connectivity to retrieve information on his portfolio. Essentially, he navigates the information space without his personal "roadmap."

To help an investor like George overcome the problems delineated above, we need to support him with an environment that addresses five main issues: First, the need to provide a user with local "views" of the global information space pertinent to his interests. Such views include "meta-information" about relevant information sources and a uniform interface for directly accessing and handling diverse information sources. World-Wide Web clients (browsers) have so far been a very successful platform in that respect but still lack essential features. Second, instead of making users seek and retrieve new information, it is desirable to have information collected on behalf of them on a continuous basis, according to their interests. Then, to feed to them information selectively, on a "need-to-know" basis and according to set priorities. Third, instead of letting users cope with fragments of data and information in diverse forms, it is important to present them with an integrated view tailored to their computing environments and accessible over Internet. Fourth, users should be offered the ability to record their ever-changing interests and priorities via "user-centered" interfaces, compatible with Internet technologies. Last but not least, seameless user interaction with information sources should be provided to mobile users, who do not have continuous access to their desktop environments.

Designing and building portfolio monitoring tools that address these needs is a challenging task since [8]:

- There is an enormous amount of continuously changing and generally unorganized available information.
- Information provision involves a variety of data types (market data, financial reports, breaking news, etc.) and formats (Database-query results, HTML-pages, XML, Usenet News feeds, Java-based "ticker tapes," etc.)
- The timely arrival of financial information to the user can be of critical importance. The timeliness of information depends on its classification according to user interests.
- Resource and cost constraints and popularity factors often dictate the provision of financial data. For instance, not all data are available for free.
- The seamless support of users, regardless of whether they access a portfolio monitoring system via mobile stations (e.g. Personal Digital Assistants) or desktop workstations, requires the use of software technology applicable to both environments.

In this paper we propose the development of a *Financial Information Gathering Infrastructure* (FIGI) that will address some of the issues and problems mentioned earlier. Using FIGI, an investor can register his interests and priorities on a profile server. This server acts like a user-proxy, retrieving and caching on a continuous basis information defined by user-profiles. Whenever a user connects to the system, he receives information of relevance according to priorities set and the urgency of information collected.

The remaining of this paper is organized as follows: Section 2 describes the architecture of FIGI. Section 3 presents Concordia, a Java-based Mobile Agent platform which is used for the implementation of FIGI. We end up in Section 4 with conclusions and future work.

2 The Architecture of FIGI

The Financial Information Gathering Infrastructure is comprised of servers with fixed Internet connectivity. A user connects to a FIGI server through his *Personal Information Roadmap* (PIR), an interface tailored to portfolio monitoring needs. The PIR can be implemented as a Java-applet, in order to guarantee its portability to fixed and mobile computing units. A user-investor can use PIR in three modes:

1. An interactive mode, where the user directly seeks and retrieves information from available information sources on Internet, accessing Web-resources, conducting text-search on Web-sites or querying Web databases.

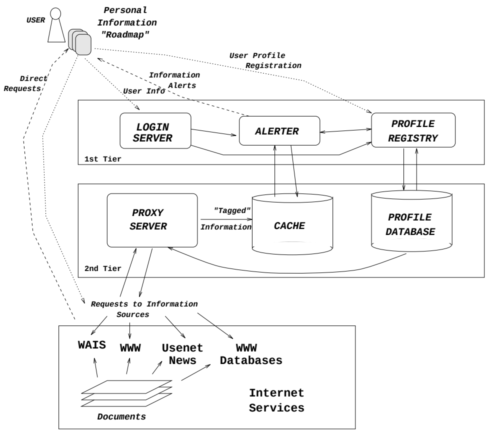

Figure 1: An overview of FIGI's architecture.

2. A FIGI-session mode, where the user downloads alerts and information collected by the FIGI server on his behalf.

3. A profile-registration mode, where the user registers or updates his portfolio monitoring interests on FIGI.

The FIGI server is organized as a two-tier architecture (see Figure 1). The first tier is comprised of three servers that receive user input and act upon it:

- The *Login Server* deals with user authentication. It notifies other FIGI components of user connections. Each time the user connects to the system he registers with FIGI the level of service he expects: whether he wishes to receive urgent alerts or a summary of new information or the full report of information gathered on his behalf by FIGI.

- Upon user registration, the *Alert Server* (Alerter) starts retrieving information pertinent to the particular user profile from the *FIGI Cache* and forwards it to the user through his PIR.

- The *Profile Registry* is a server where a user can register or update his interests.

Like in [10], a *user profile* is a set of long-term, continuously evaluated queries. In FIGI, these queries may include typical queries to Web databases, HTTP requests for World-Wide Web resources, access to general-purpose Search Engines or Subject Cataloging Sites, subscription to Usenet News, etc. Each profile is annotated by the user with a number of *data* and *control* parameters. Data parameters are query arguments (e.g., a stock symbol of interest), whereas control parameters determine the frequency of query execution, the expected amount of information gathered from queries (e.g., summary vs. full results), the priority of notification for a given query, etc.

The second tier of FIGI is comprised of a *Profile Database*, the *FIGI Cache* and a *Proxy Server*. The Profile Database stores investor profiles. The Proxy Server scans continuously the Profile Database, schedules and issues requests for information to Wide-area network services. The results of these requests are tagged with information denoting the corresponding user-profile and are stored in the FIGI Cache. When a user logs in to FIGI, these results are extracted from the Cache by the Alerter and forwarded to him.

3 Implementation Issues

3.1 Mitsubishi's Concordia platform

The implementation of FIGI is based on Concordia's Mobile Agent technology [2, 5], which is developed by the Mitsubishi Electric Information Technology Center. The term "agent" denotes an independent software program which runs on behalf of a user connected to a Wide-area network [5]. An agent may run when the user is disconnected from the network, even if the user is disconnected involuntarily. A Mobile Agent is an agent that can travel to multiple locations in a Wide-area network. While traveling, the agent performs work on behalf of the user, collecting information or delivering requests.

Concordia is a framework for the deployment of Mobile Agents written in the Java language. Concordia agents use Concordia services to travel to machines on a Wide-area network where they can access locally services required to accomplish their tasks. For a Concordia agent to visit and access services on an Internet host, the host must be running a *Concordia Server*. Concordia Servers run as NT Services on computers equipped with Windows NT or as daemons on computers running UNIX.

Besides the Concordia Server, the Concordia infrastructure provides a lightweight object called *Agent Transporter*, which enables any Java application (or applet) to send, receive, and execute agents [6]. The Agent Transporter object is designed to be embedded in a Java application that wishes to directly interact with Concordia agents. Once an application or applet embeds an Agent Transporter in itself, it can communicate with Concordia Servers, as well as other AgentTransporter objects, to transfer Concordia agents. The AgentTransporter also provides a way to directly interact with local agents.

Service Bridges are another interesting feature of Concordia, pertinent to FIGI. Service bridges are application specific service providers that operate within the Concordia server and provide functionality to agents [6]. Service bridges are Java objects which are written by agent application developers and can be installed into a Concordia server. Concordia agents can then travel to the server, retrieve references to the service objects and then interact with the service's public interface to gather the information they need or to perform the tasks they need to accomplish. According to [6], service bridges can be thought of as the non-mobile component of an agent application. Service bridges also can act as interfaces (or bridges) to existing services or servers.

3.2 Implementing FIGI on Concordia

The Concordia infrastructure is employed as a basis for developing FIGI. Concordia Mobile Agents are used in the implementation of the communication protocol between the users of FIGI and its servers. To this end, the Personal Information Roadmap is implemented as a Java applet with an embedded Agent Transporter, able to launch and receive Concordia Agents. A user can download and access his PIR via an Internet connection either to his desktop computer or to his Personal Digital Assistant in a mobile setting.

The Proxy Server of FIGI is implemented as a stationary Concordia Agent which launches Mobile Agents to access and combine information sources over the network, implementing the scheme described in [7]. Finally, the FIGI servers described in Figure 1 (Login, Alerter, Profile Registry) are implemented as Concordia Service Bridges.

4 Conclusions

The Mobile Agent technology is a flexible and efficient platform for implementing environments that will help mobile investors cope with the information overload experienced on Internet. In fact, Mobile Agents based on Java represent a framework for unifying the various financial information services currently provided or developed on Internet. The development of FIGI will provide an infrastructure upon which we can study several issues of interest, including the modeling of financial information provided to investors over Internet, the modeling of portfolio monitoring activities and the efficient caching of financial information.

4.1 Acknowledgments

The authors would like to thank G. Samaras for suggesting the employment of Mobile Agent technologies as a base for information gathering environments.

References

[1] K. Berghel. The World-Wide Web. *Communications of the ACM*, 39(1):31–40, January 1996.

[2] A. Castillo, M. Kawaguchi, N. Paciorek, and D. Wong. Concordia as Enabling Technology for Cooperative Information Gathering. In *Japanese Society for Artificial Intelligence Conference*, June 1998. http://www.meitca.com/HSL/Projects/Concordia/.

[3] H. Chu and M. Rosenthal. Search Engines for the World Wide Web: A Comparative Study and Evaluation Methodology. In *ASIS 1996 Annual Conference Proceedings*. American Society for Information Science, October 1996.

[4] K. Kamiya, M. Roscheisen, and T. Winograd. Grassroots: A System Providing a Uniform Framework for Communicating, Structuring, Sharing Information, and Organizing People. In *Fifth International World Wide Web Conference*. World Wide Web Consortium, May 1996. http://www.w3.org/Conferences/www5.

[5] Horizon Systems Laboratory. *Mobile Agent Computing. A White Paper*. Mitsubishi Electric ITA., Januray 1998.

[6] Mitsubishi Electric ITA. *Concordia Developer's Guide*, October 1998. http://www.meitca.com/HSL/Projects/Concordia.

[7] S. Papastavrou and G. Samaras. The Development of DBMS Client/Server Applications on the Web using Java Mobile Agents. Technical Report TR-98-5, Department of Computer Science, University of Cyprus, April 1998.

[8] K. P. Sycara, D. Zeng, and K. Decker. Intelligent Agents in Portfolio Management. In *Agent Technology. Foundations, Applications and Markets*, pages 267–282. Springer, 1998.

[9] K. Wittenburg. The WWW Information Glut: Implications for Next-Generation HCI Technologies. *ACM Computing Surveys*, 28(4es), Devember 1996.

[10] T. W. Yan and H. Garcia-Molina. SIFT - A Tool for Wide-Area Information Dissemination. In *Proceedings of the 1995 USENIX Technical Conference*, pages 177–186, 1995.