Demonstration: Enabling Scalable Online Personalization on the Web

Kaushik Dutta
Georgia Tech
gte314q@prism.gatech.edu

Anindya Datta
Chutney Technologies
and Georgia Tech
adatta@cc.gatech.edu

Debra VanderMeer
Chutney Technologies
and Georgia Tech
deb@cc.gatech.edu

Krithi Ramamritham
University of Massachusetts
and IIT, Bombay
krithi@cs.umass.edu

Helen Thomas
Chutney Technologies
and Georgia Tech
helen@cc.gatech.edu

1 Introduction

Given the current hyper-competitive nature of the e-commerce marketplace, coupled with razor-thin margins, online personalization is of great interest to e-companies. The attractiveness of online personalization technologies derives from the claim by consumer behaviorists that a personalized experience leads to increased *buy probabilities* on the part of e-shoppers [11].

Virtually all personalization technologies are based on the idea of storing as much historical customer session data as possible, and then querying the data store as customers navigate through a web site. The holy grail of on-line personalization is an environment where fine-grained, detailed historical session data can be queried based on current online navigation patterns to formulate real-time responses. The problem, of course, is one of *scale* – it is extremely difficult to track tens of thousands of e-shoppers in real time, and even more difficult to access a database online to provide real-time responses. As a result, virtually all web-based customer interaction schemes use *static profiling* techniques, and provide either *delayed* or *canned* responses to users.

Real-time interaction management, the focus of our work, uses *dynamic profiles* to incorporate users' changing interests and needs over time. For example, a user who purchases Physics textbooks for his daughter from an online seller later returns to the site, looking for books for himself. After a few clicks the system should recognize that the user is not interested in Physics books in his current visit, and suppress its knowledge regarding his past behavior and instead adjust its current responses to be in tune with his more recent behavior.

We have developed the a system that (1) tracks a large number of users (potentially tens of thousands) in real time as they navigate through a site, (2) performs retrievals from a large data warehouse in real time, and (3) delivers an appropriate user response, in real time, based on the system's knowledge of the user's current behavior and the information retrieved from the data warehouse. To achieve our goals, we store three basic types of data: (1) navigational (i.e., where users go in the site), (2) transactional (e.g., what customers purchase), and (3) third-party data, e.g., demographic data. This knowledge is encoded simply as a set of rules of the type generated by standard data mining techniques. The foundation of this system is the *eGlue Server*, the subject of this demonstration proposal.

2 Architecture

eGlue Server is a component-based system which works with readily available web server and e-commerce application server systems. Figure 2 shows a graphical depiction of an "end-to-end" e-commerce system architecture, including the eGlue Server component. A full description of the system can be found in [16]. Perhaps the best way to de-

Proceedings of the 26th VLDB Conference, Cairo, Egypt, 2000.

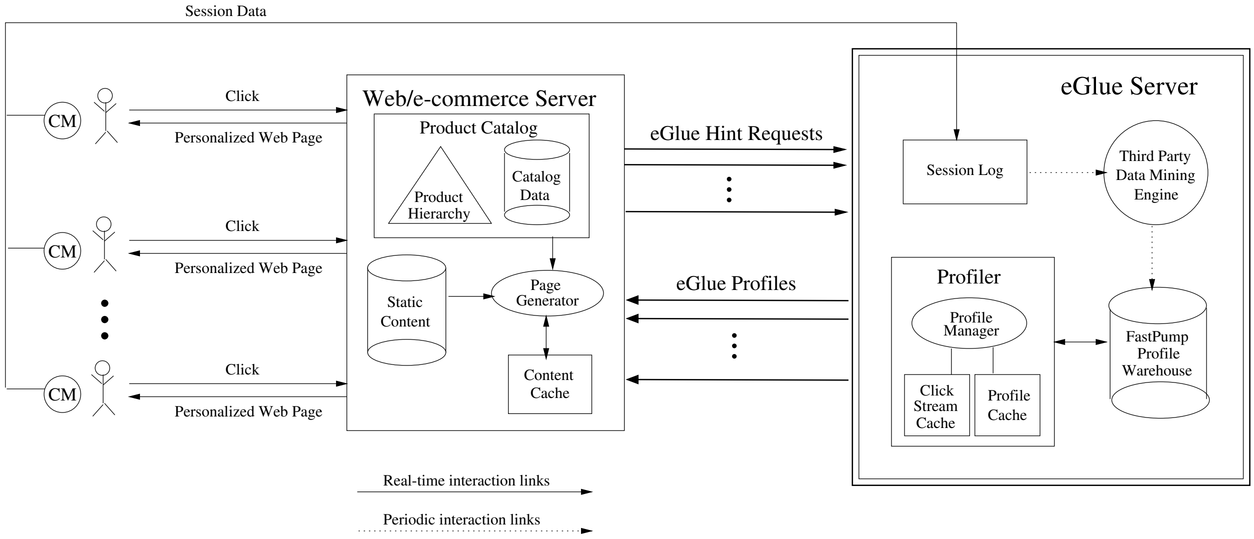

Figure 1: Real-Time Personalization System Architecture

scribe the overall workings of the system is to provide an example. Consider a user U, who clicks in an eGlue enabled e-commerce site. This causes an HTTP request to be sent to the *Web/E-commerce Server* (WES). (Note that, although the e-commerce and web servers are actually separate components, we describe them as a single component here for convenience). When the WES receives an HTTP request from U, it forwards U's click information to the *eGlue Server* (eGS) in the form of a *Profile Request* (HR). Upon receiving U's click information, the eGS performs two tasks.

1. *The eGS updates U's clickstream.*

2. *The eGS generates a profile.* A *profile* is simply a set of *action-probability pairs* (APPs), where the APPs represent actions U is likely to take, along with the corresponding probability that U will choose the action, given his current clickstream.

When the WES receives a *profile* from the eGS, it uses the profile to generate a customized web page for U. Precisely how the WES uses the profile to generate a personalized web page for a user is dependent on the needs of the web site.

3 The eGlue Server Components

Here, we provide an overview of the interaction of the components of the eGS, which is the primary focus of this demo. The *Profiler* consists of a set of data structures and algorithms designed to provide profile as to users' next actions on a site *in real time*, enabling a variety of interaction management applications, such as the delivery of customized pages or the prefetching of high-probability content from disk.

The Profiler consists of three components: (1) a *Clickstream Cache* (CC), which stores current clickstream information (of maximum length *CSL*) for each user in the system, (2) a *Profile Cache* (PC), which stores recently-used profiles, and (3) a *Profile Manager* (PM), which generates profiles for the WES.

To illustrate the mechanics of the profile-generation process, we return to our user U. Consider a situation where the WES has submitted a *hint request* to the eGS for U's i^{th} click. The PM first checks the CC to find U's previous clickstream (if any), and updates that to include U's latest reported action.

After determining U's current clickstream, the PM checks the PC for a profile matching U's clickstream (i.e., a profile whose RA matches the observed clickstream). If such a profile is found, the PM sends it to the WES. If, at this point, another user, say U', were to follow on the same path as U, the reader can easily see that the needed profile would be in the PC (assuming U' follows U closely enough that the profile has not been replaced by a newer profile).

If a matching profile is not found in the PC, the PM requests the information from the *FastPump Profile Warehouse* (FP) (described below). After FP returns the profile information, the PM sends it to the WES. This profile will also now reside in the PC until a decision is made to discard it.

The *FastPump* data warehouse engine, described in detail in [8] is an extremely fast storage and retrieval engine. Within the context of the eGlue server, FP serves as a *profile warehouse*, storing historical and navigational data in the form of rules, and retrieving this data in response to queries.

The rules stored in FP are generated by a *Third Party Data Mining Engine*. The data mining engine takes a *Session Log*, i.e., clickstream and transaction information generated by various clients clicking in the web site, as input, and generates a set of rules as output. Click and transaction data is added to the session log in real time (as noted by the solid lines in Figure 2), while the mining of the session log and update of the Profile Warehouse takes place offline (shown by the dotted lines in Figure 2).

4 eGlue Demonstration

We demonstrate our system by showing an eGlue-enabled e-commerce site. This demo has two objectives: (A) to demonstrate the workings of eGlue software, with special emphasis on scalability, and (B) to demonstrate the importance of the underlying database, i.e., FastPump, which was specifically designed for eGlue (as opposed to using a commercial data warehousing system, e.g., Oracle).

In the demo, we introduce two types of visitors concurrently into an e-commerce site: real (i.e., foreground) visitors and simulated (i.e., background) visitors. These visitors will navigate the demo catalog of the e-commerce site.

For foreground visitors, the demo will show various personalization features of the eGlue Server, e.g., ad targeting. The demo will allow the demonstrator to control the behavior of a foreground user, i.e., to "become" the visitor. Here, the demonstrator will be able to choose the visitor's navigation through the site, and see the eGlue profiles generated for each click, as well as the page generated in response to that click. The purpose of adding simulated background users is to show the scalability of the system. To this end, the demo shows each simulated user's current position, updated as users move through the site.

We show the performance of the system in two ways. First, we replace FastPump with a commercial data warehousing system, e.g., Oracle, to show the effect of the underlying database on the system. Second, we remove the Profiler, thus showing the performance of the system using the only web server's own cache.

5 Related Work

Within the academic literature, web personalization is emerging as a major field of research. We review some of the important literature in this field here, and mention a few commerical products. At present, there are two main approaches to personalization: collaborative filtering and data mining for user behavior patterns.

Collaborative filtering [12] is a means of personalization in which users are assigned to groups based on similarities in rankings of content of web pages, i.e., users with similar interests will rank the same pages in the same way. [5] presents an analysis of predictive algorithms for collaborative filtering. [7] describes a non-invasive method of collecting user interest information for collaborative filtering.

Much work has been published in the area of user navigation pattern discovery based on web logs, e.g., [6, 13]. A great deal of this work in web navigation and usage mining is based on fundamental work in data mining for association rules [2] and sequential patterns [3, 14]. This type of work is quite complementary to our work; in fact, we base our navigation pattern discovery methods on some of the ideas found in these papers.

On the commercial side, several companies offer products in the Internet Personalization space. The two functions supported are *recommendations*, (i.e., static, predetermined responses to actions) based on collaborative filtering techniques [12] and *reporting* (i.e., analysis of a web site's traffic). Recommendation products include Net Perceptions Recommendation Engine [9], Engage Technologies [15], and LikeMinds [4]. Personify [10] and Accrue [1] offer analysis and reporting based on log data.

While the general philosophy of our work is similar to that of the above personalization products in that we use a database and cache to store and retrieve our information, we depart from current personalization efforts in two key ways: (1) We do not use static profiles; rather, we make *dynamic decisions* based on the information available at the time a user clicks; and (2) We merge historical information with knowledge of a user's current behavior in order to make these dynamic decisions, as opposed to delivering canned responses to pre-determined actions.

6 Contribution & Conclusion

We demonstrate a component-based system to address the problem of real-time user interaction by generating *profiles*, which profile predictive information as to a user's future actions on the site, thus enabling a vast array of customization or prefetching applications on a site.

The foundation of this system consists of two components: FastPump, an extremely fast storage and retrieval engine; and the Profiler, an efficient profile caching mechanism. Perfomance studies indicate that our system gives an order of magnitude performance gain over an implementation using off-the-shelf database back end without the Profiler.

References

[1] Accrue. Accrue technologies. www.accrue.com, 1999.

[2] R. Agrawal, T. Imielinski, and A. Swami. Mining association rules between sets of items in large databases. In *Proceedings of the 1993 ACM SIGMOD International Conference on Management of Data*, pages 207–216, May 1993.

[3] R. Agrawal and R. Srikant. Mining sequential patterns. In *Proceedings of the Eleventh International Conference on Data Engineering*, pages 3–14, March 1995.

[4] Andromedia. Likeminds. www.andromedia.com/products/likeminds, 1999.

[5] J.S. Breese, D. Heckerman, and C. Kadie. Empirical analysis of predictive algorithms for collaborative filtering. In *Proceedings of the Fourteenth Conference on Uncertainty in Artificial Intelligence*, pages 43–52, July 1998.

[6] A.G. Buchner, M. Baumgarten, S.S. Anand, M.D. Mulvenna, and J.G. Hughes. Navigation pattern discovery from internet data. In *Proceedings of WEBKDD'99: Workshop on Web Usage Analysis and User Profiling*, 1999. http://www.acm.org/sigs/sigkdd/proceedings/webkdd99/toconline.htm.

[7] P.K. Chan. A non-invasive approach to building web user profiles. In *Proceedings of WEBKDD'99: Workshop on Web Usage Analysis and User Profiling*, 1999. Available at http://www.acm.org/sigs/sigkdd/proceedings/webkdd99/toconline.htm.

[8] A. Datta, H. Thomas, and K. Ramamritham. Curio: A novel solution for efficient storage and indexing in data warehouses. In *Proceedings of 25th International Conference on Very Large Data Bases*, pages 730–733, 1999. NB: Curio is the former name of the FastPump data warehouse.

[9] Net Perceptions. Net perceptions recommendation engine. www.netperceptions.com, 1999.

[10] Personify. Personify technologies. www.personify.com, 1999.

[11] F.F. Reichheld and W.E. Sasser. Zero defections: quality comes to services. *Harvard Business Review*, 68:105–7, September-October 1990.

[12] U. Shardanand and P. Maes. Social information filtering: algorithms for automating "word of mouth". In *Conference proceedings on Human factors in computing systems*, pages 210–217, May 1995.

[13] M. Spiliopoulou, L.S. Faulstich, and K. Winkler. A data miner analyzing the navigaitional behavior of web users. In *International Conference of ACAI'99: Workshop on Machine Learning in User Modelling*, 1999.

[14] R. Srikant and R. Agrawal. Mining sequential patterns: Generalizations and performance improvements. In *Advances in Database Technology - EDBT'96, 5th International Conference on Extending Database Technology*, pages 3–17, March 1996.

[15] Engage Technologies. Engage e-commerce product suite. www.engage.com, 1999.

[16] D. VanderMeer, K. Dutta, A. Datta, K. Ramamritham, and S. Navathe. Enabling scalable online personalization on the web. Technical Report CER-0004, Chutney Technologies, Atlanta, GA, 2000.