Haystack: Per-User Information Evironments
MIT9904-08

Progress Report: July 1, 1999—December 31, 1999

David R. Karger and Lynn Andrea Stein

Project Overview

The Haystack project is investigating the ways in which electronic infrastructure can be used to triangulate among the different knowledges of an individual, of collegial communities to which we belong, and of the world at large. Its infrastructure consists of a community of independent, interacting information repositories, each customized to its individual user. It provides automated data gathering (through active observation of user activity), customized information collection, and adaptation to individual query needs. It also facilities inter-haystack collaboration, exposing (public subsets of) the tremendous wealth of individual knowledge that each of us currently has locked up in our personal information spaces. The Haystack-NTT project involves augmenting its customization, learning and adaptation, and inter-haystack communication.

Progress Through December 1999

Through the NTT MIT Collaboration, we were able to hire several graduate students to work on Haystack beginning in September, 1999. We have been making significant progress towards our short-term goals of a robust, scalable Haystack system and simultaneously beginning serious investigation of our longer term goals, including learning and clustering.

This summer and fall, we rewrote the core of our Haystack system to create a privileged kernel. This kernel assures the persistence and transaction-safety of the data in a Haystack. The creation of a kernel recognizes that Haystack is in essence recreating much of the functionality of an operating system. For example, the kernel includes a thread scheduler for Haystack tasks, a straw storage system that mimics a file system, and an interface for network connectivity. The new kernel separates future work on the user and collaboration side of the Haystack system from the underlying service model. The final step in this conversion is the integration of a transaction-based persistent object store.

We also began to investigate the ways in which a user's Haystack can organize itself to better meet the needs of the user. One piece of this work involves exploiting successive query attempts to help the user on subsequent queries. Much of the basic infrastructure to support this work has been implemented; we are ready to add specific learning algorithms to the system and to begin experimentation with them. We are simultaneously investigating the potential usefulness of clustering as a way to organize data for the user. We have begun some preliminary theoretical work and prototyping of these ideas, including modifications to the user interface to support better presentation of such organization.

Research Plan for the Next Six Months

Over the next six months, we intend to complete the kernel conversion and to integrate a relational database back end into Haystack. This latter project will make it easier for users to make structured as well as unstructured queries against the data in their Haystacks. There are several open research questions in dissecting a query into its structured and unstructured components and in joining the result sets obtained from such heterogeneous back ends as a relational database and an information retrieval system, and the investigation of such questions will be a part of our project.

We will also build on our initial phase work in adaptive Haystacks, i.e., Haystacks that learn from user query behavior. We expect to implement algorithms for ranking, relevance feedback, and query expansion to enhance a Haystack's ability to find the document for which its user is looking. We also expect to use clustering techniques to organize the data in a Haystack even in the absence of explicit user queries. This involves both basic research into clustering algorithms and more applied work on user interfaces for browsing and data presentation.

By the end of June 2000, we expect to be able to demonstrate a robust Haystack system running on a sizable corpus exploiting several of these new features.