Haystack: A Platform for Personalized Information Management

David Karger, Karun Bakshi, David Huynh, Dennis Quan & Vineet Sinha

Artificial Intelligence Laboratory
Massachusetts Institute of Technology
Cambridge, Massachusetts 02139

http://www.ai.mit.edu

Problems and Motivation: The personal computer, though acclaimed for its ever increasing capabilities to process data, has yet to become a place convenient enough for the user to store and manage all of his or her various types of information. Currently, the user is restricted to storing only a limited set of information, including e-mail, text documents, pictures, audio files, contacts, and meetings. Without becoming a database administrator, the user struggles to find a place to record useful daily information (e.g. medications taken, shopping lists) and accumulated long-termed data (e.g. blood type, history of illnesses, a list of favorite restaurants, an inventory of insured and uninsured properties). Text editors and spreadsheets allow for content composition, but they lack capabilities for inputting semantics that would facilitate automatic information processing. Using existing software applications, the user cannot query for the favorite restaurant closest to a friend's house, ask for a list of uninsured items of more than $100 in value, or find out how often he or she buys milk.

Not only are the kinds of easily storable information limited, but so is the number of fields available for each type of information. In few address books can one store a history of physical addresses for a contact, should the contact move often. There is no support for recording a contact's lists of allergies and dietary restrictions, both of which prove useful in preparing a dinner party. One cannot query for "all baking recipes that contain chicken, egg, and salmon, and that do not conflict with any dietary restriction or allergy of any of my dinner guests."

Furthermore, there are still few tools available that aggregate these different types together in more meaningful ways: it is not readily apparent how to associate a meeting with the e-mail threads spawned from that meeting or the documents and to-do items resulted from the discussions in the meeting. Without creating one's own database, one cannot specify which contacts are present in a photograph and later search for "all photographs of those friends of mine who went to my high school."

These problems come in two flavors. First, the software applications support only a limited number of information types and very restricted schemata for those types. The user has few options for storing freeform information, which is more commonly encountered than structured data. Second, there exist few tools that draw information from different sources to answer complex yet practical search queries. Consequently, either the user is unable to enter the information he or she wishes to record, or he or she fails to see the benefits of entering such information as the information cannot be easily managed and retrieved later. Frustration often ensues as a result of the considerable effort to input and organize information that has then become unmanageable and unrecoverable.

Approach: Recognizing that currently, user information is segregated by limited and restrictive software applications' data formats, the Haystack project adopts one of the Semantic Web's core technologies, the Resource Description Framework (RDF), as its primary data model[1], in order to support a generic and unified data framework.

On top of this data modeling framework we have built a platform that facilitates sophisticated exploitation of such a framework. The platform provides a common data repository, a unified data access abstraction, a data-oriented programming language, a software agent infrastructure, a data navigation framework, and a rich semantics-based user interface paradigm for visualizing semi-structured information. On this platform we are exploring a variety of research issues, including user interface personalization, computer-aided information corpus browsing, natural language-based search, automated information flow and collaboration, user-specified ontology management, freeform annotations, among others.

Related Work: Research on personalized information tools can be seen in the early days of the Hypertext research including Bush's Memex[2] and Engelbart's AUGMENT[3]. Another example is the Lifestreams project[4], which is predominantly based on the storage of documents in a temporal context; in contrast, Haystack allows for classification and retrieval based on other document features.

The Semantic Web project at the World Wide Web Consortium (W3C) pioneered the use RDF for addressing the issues of interchangeability. The focus of the Semantic Web effort is to proliferate RDF-formatted metadata throughout the Internet in much the same fashion that HTML has been proliferated by the popularity of web browsers. Haystack is designed to work within the framework of the Semantic Web. However, the focus is on aggregating data from users' lives as well as from the Semantic Web into a personalized repository.

Challenges: Providing a unified platform for creating and retrieving information of all forms presents a number of interesting research challenges. At the bottommost layer, we are investigating the use of Semantic Web technologies, including RDF, to provide a common representation and to unify information of different schemata together. As a result, we are forced to deal with the issue of how to present general semi-structured information. For example, what happens when the UI designer of an address book application can no longer rely on a fixed schema? Also, how does one navigate a corpus once the requirement of a strict hierarchy has been lifted from the system?

Another problem that arises is that of ontology management. We cannot assume the general user to be a skilled database administrator; consequently, inputting information into Haystack must be achieved with the help of intelligent agents that massage loosely-structured information into RDF. Also, the classic Semantic Web problem of ontology sharing has to be addressed if Haystacks are ever to foster collaboration, as it is unlikely that every user will agree to one standardized, overarching ontology.

Other interesting challenges exist in the area of automation, one of the many general goals of the Semantic Web. Agents in Haystack are given a unified abstraction for accessing and sharing data. What are the general patterns of data access employed by such agents? How do we resolve conflicts in the data produced by agents? For example, when two agents are tasked with extracting titles from documents using two different methodologies, which agent should be believed when they produce different results? Furthermore, how does a user indicate some event-triggered action to take place, such as the automatic disposal of e-mail messages from a specific author?

Future Work: Currently support has been added mainly for viewing data that has been added to the users' corpora (through one or more agents). We plan to allow the user to edit his or her own data and to annotate the information appropriately. We also aim to support collaboration among multiple Haystacks.

Research Support: This research was supported by the MIT-NTT collaboration, the MIT Oxygen project, a Packard Foundation fellowship, and IBM.

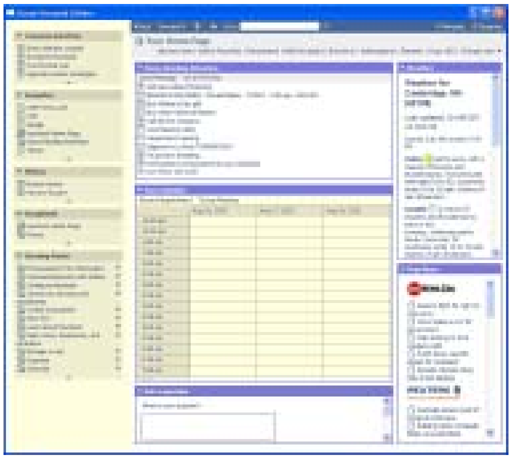

Figure 1: Haystack's User Interface

References:

[1] Tim Berners-Lee, James Hendler, and Ora Lassila. The semantic web. *Scientific American*, May 2001.

[2] Vannevar Bush. Artificial intelligence. *Atlantic Monthly*, 176(1):641–649, January 1945.

[3] Douglas C. Engelbart. Augmenting human intellect: A conceptual framework. Technical report, Stanford Research Institute, Menlo Park, CA, October 1962.

[4] Eric T. Freeman and Scott J. Fertig. Lifestreams: Organizing your electronic life. In *AAAI Fall Symposium: AI Applications in Knowledge Navigation and Retrieval*, Cambridge, MA, November 1995.