

Synchronous and Asynchronous Collaborative Information Space Analysis Tools

Ke-Thia Yao, Robert Neches, In-Young Ko, Ragy Eleish, Sameer Abhinkar
University of Southern California Information Sciences Institute (USC ISI)

Abstract-- The DASHER Project at USC/ISI has focused upon helping organizations with rapid-response mission requirements by providing information analysis tools that help make sense of sets of data sources in an intranet or internet: characterizing them, partitioning them sorting and filtering them. This paper focuses on a subset of these tools that help individuals in the organization to collaboratively, both synchronously and asynchronously, form task-oriented information repositories. Also, the paper discusses planned extension of the tools to support collaboration in a mobile computing environment.

Index Terms-- Computer supported co-operative work, Web-based cooperative systems, location based computing

I. INTRODUCTION

The focus of USC ISI's DASHER Project is the rapid formation and utilization of *topic-* and *task-oriented* information repositories in order to help organizations with rapid-response mission requirements. Such organizations need to be able to quickly organize rapid response teams, backed by the information, materiel, and support services they need to do their job. To do so, the team and its organizers need to find and assess sources of those services and work with them to define the specific nature of the services provided. Prototypical examples include an engineering team formed to take an IFF (Identification Friend-or-Foe) device from design to deployment in weeks, a team coordinating technical, contractual and transport requirements for an emergency purchase, or a disaster relief mission trying to assess locally available services.

DASHER provides a set of information search, retrieval, analysis, visualization and organization tools within a collaborative and platform-independent environment. In contrast to many other efforts in information search and retrieval, which emphasize retrieval and are focused on finding a specific document, the DASHER information space analysis tools focus on making sense of *sets* of data sources: characterizing them, combining them, sorting them, filtering and reorganizing them [1]. The tools include an interactive hierarchical category editor to capture, and organize sets of documents; a document set analysis tool that take a category structure of documents extract useful noun phrases, suggest alternative structures, and visualize/compare the structures; and a category analysis tool that analyze and import external category structures, like Yahoo [2] and GTE Superpages [3].

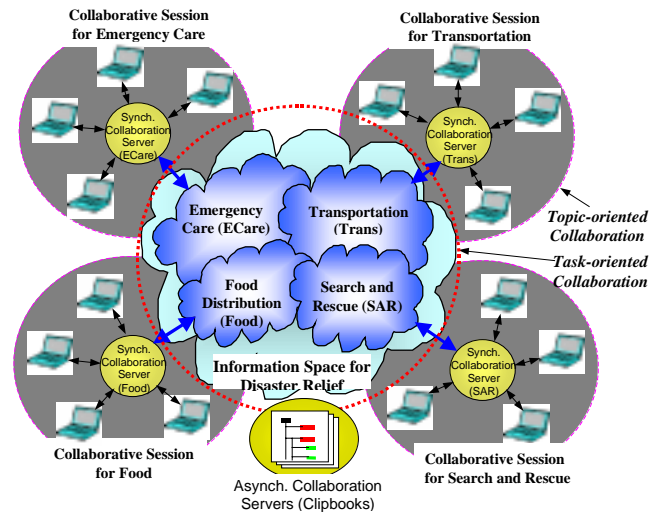


Figure 1. Collaboration in a Sample Application

These tools exist under a collaborative environment. This environment helps support collaboration among increasingly mobile information seekers. The functionality provided includes:

- *Dynamic session formation:* creation of shared, multi-user sessions based on shared-interest groups. Users can join or leave a session at any time.
- *Information exchange:* information and service sharing from remote locations either synchronously or asynchronously. This is used to support the following two functions.
- *Collaborative browsing and searching.* DASHER provides information retrieval tools that search the Web and analyze the results. Within an interactive session, users can show each other what they have found.
- *Collaborative repository editing.* Within a session users can collaboratively organize information found during a search into hierarchical structures that best suit the task requirements. Changes that one user makes to the repository are instantaneously propagated and viewed by the others.

Typically, collaborative interactions are classified based on a taxonomy that distinguishes four categories: same time/same place, different times/same place, same time/different places, and different times/different places [4]. The latter two categories are known as synchronous and asynchronous distributed interaction. Both topic- and task-oriented collaboration in DASHER are supported by a

combination of these modes. Synchronous tools are implemented using NCSA's Java™ based Habanero collaborative tools and tool development framework [5], which provides *application sharing*. Under this framework the individual applications are linked to a Habanero server by an event sharing mechanism. This way a change (an event being raised) in one application is reflected in all other applications. For asynchronous collaboration, we provide an information exchange mechanism using distributed clipbooks servers. Clipbooks allow users to store and retrieve information in a structured manner. Currently, the Habanero and the clipbook servers are implemented as global servers, which can be accessed anywhere. In section V we discuss extensions and modifications to localize these servers to support mobile computing.

In section II of this paper, we provide an overview of the collaboration process in DASHER. We use information repository formation during a disaster relief mission as an example. Section III describes our approach to synchronous repository formation based on the Habanero framework. Section IV describes our approach to asynchronous information exchanging using distributed clipbook servers. Section V describes planned extension of DASHER collaborative tools in the context of mobile computing. Section VI discusses related work. Section VII discusses conclusions and future work.

II. OVERVIEW OF COLLABORATION USING DASHER

As a demonstration of DASHER repository formation capabilities, ISI is building the GeoWorlds system for DARPA (the US Defense Advanced Research Projects Agency). GeoWorlds integrates, tests, and evaluates technology from DASHER with Digital Library and Geographical Information System technology from a number of other sources (including the University of Arizona, the University of Illinois at Urbana-Champaign, the University of California at Berkeley, and the University of California at Santa Barbara). Using Humanitarian Assistance / Disaster Relief as a testbed application, GeoWorlds is demonstrating feasibility of relating geographical information to a corpus of "other" information in documents. The function of the system is to help a user understand facts and events in relation to space and time. It allows users to take a set of documents, relate them to places and times relevant to their contents, and provide a visual environment for presenting and exploring those relationships.

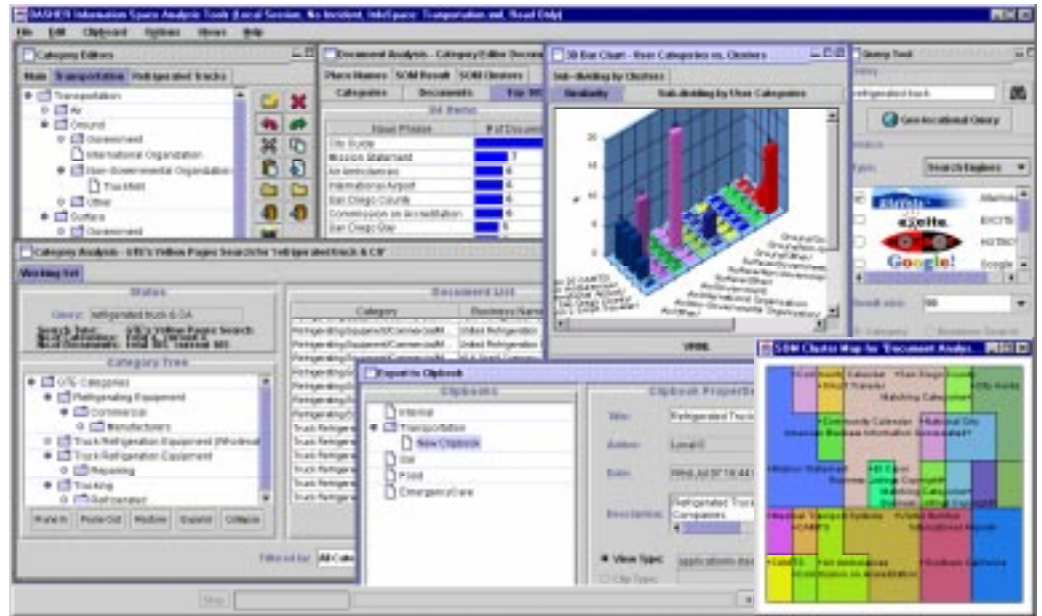


Figure 2. DASHER Information Space Analysis Tools

Using the disaster relief scenario as an example, this section describes topic-oriented and task-oriented collaboration in a sample operational scenario. This scenario demonstrates DASHER's unique combination of browsing/editing and information processing tools in an interactive and collaborative (both synchronous and asynchronous) environment. The environment is implement in Java, which makes it platform-independent and widely accessible. Also, unlike typical asynchronous exchange mechanism, DASHER's mechanism allows the exchange of active objects, not just passive data.

To describe the operational steps related to collaboratively organizing an information space using DASHER services, let us consider the case of a disaster relief task force comprised of several different groups working on different topic areas such as emergency care, transportation, food distribution, and search and rescue. Users in a group collaborate with each other to organize an information space related to their topic area. Topics usually correlate with tasks, but they often need to collaborate with the users outside their topic groups in order to get relevant information or coordinate information.

Figure 1 shows the collaboration overview using DASHER collaboration tools. A *DASHER session* is a collaborative environment where a group of users can work together to organize an information space for a specific topic area. DASHER's persistence server manages predefined sessions (e.g. *ECare*, *Trans*, *Food* and *SAR*). A user can instantiate a session by logging into a session. Users can join or leave a DASHER session dynamically and an instance of a session will be destroyed when the last user disjoins the session.

The *topic-oriented collaboration* is the collaboration mechanism within a DASHER session that provides sharable views of an information space by which users can manage the information space synchronously and tools to exchange information space data asynchronously among

users. Using these, multiple users in a session can work in parallel to retrieve data from different information sources, to analyze the data retrieved, and to add the analyzed result into the information space of the session. The synchronous collaboration server maintains consistent views of the information space among the users.

The system updates shared views of the information space, so multiple users can contribute different pieces to a puzzle. For example, the medical group of the *Trans* session activates an evacuation plan to transport patients and perishable medical supplies to medical centers in a safe area. A user in the *Trans* session retrieves transportation categories from Yahoo™ Web search engine and performs a filtering operation to collect only the categories related to the geographical region of the incidence. At the same time, another user in the same session retrieves the trucking companies near the incident area from an on-line yellow page, such as GTE SuperPages™ and classifies those companies according to the type of trucks using document clustering and category editing tools. Then, the first user sees the result view of the second user and decides to combine the result with the existing trucking company category in the information space. All the users in the session will see the updated information space, which contains the Yahoo's transportation categories along with the trucking companies from GTE yellow page.

The *task-oriented collaboration* is the collaboration mechanism that provides an efficient tool to exchange information space data among users in different DASHER sessions. Compared to topic-oriented collaborations, these collaboration activities are infrequent and sporadic, so the data exchange mechanism should support asynchronous exchange modes. Using this collaboration support, users can correlate parts of their information space with related DASHER sessions or retrieve relevant data from other sessions.

For example, the *ECare* session determines that it does not have enough refrigerated trucks to move all the perishable medical supplies. Using the synchronous collaboration mode, *ECare* joins the *Trans* session to acquire refrigerated trucks. Users in the two sessions jointly browse the transportation category structure, information prepared in advance in anticipation of emergency evacuations, but they discover that the need for refrigerated trucks was not anticipated. Using DASHER, the users create a refrigerated truck category under *Trans* category structure, and they collaboratively search information sources to populate the category. The results organized in the *Trans* session can be transferred to the *ECare* session using the task-oriented collaboration mechanism. The *Trans* session correlate the subcategory 'Refrigerated Trucks' in its information space with categories in a common asynchronous collaboration working space (Clipbooks; see section IV for details), and *ECare* can take the subcategory and include it into the *ECare* information space. Besides exchanging categories, entire views can be exchanged, for example, *Trans* can send the entire GTE Category Analysis frame window to *ECare*. *ECare* does not receive a GIF image of the frame,

but an *active object* including the internal state of the frame. This allows *ECare* to manipulate the contents of the frame and perform additional analysis as needed.

The sample information space for the disaster relief task organized and visualized using DASHER by the *Trans* session is shown in Figure 2. It includes the main category structure for the topic area, the top-100 noun phrases related to the topic, a clustering map of the working set of documents, a bar-graph to compare the user defined categories and the clustering results, a working set of documents retrieved from a yellow page, and the interface of the clipbooks for the asynchronous collaboration.

Both of the synchronous and asynchronous collaboration modes can be used for topic-oriented and task-oriented group work. Sections III and IV describes these two modes.

III. SYNCHRONOUS COLLABORATION

The ability to effectively share information is the foundation of any group activity. Traditional data repositories, such as databases, go to great length to insulate users from each other [4]. DASHER provides tools that not only share information, but also help users to collaboratively form the repository. The tools allow geographically dispersed users to collaborate in real-time over the Internet to jointly search, retrieve, filter, partition, and organize the information available on the Web. Users choose to collaborate by joining a DASHER session, where each session is organized around a particular topic of interest. Users can create sessions at run-time as the needs arise. For example, in a disaster consequence management situation the users may create sessions according to the relief efforts, such as emergency care, transportation, food, and search and rescue. Within each session users collaboratively explore the Web information space by pooling together their expertise and know-how.

Let us examine one of the major DASHER collaborative tools that build repositories: the *category editor*. Many of its functions resemble the bookmark facilities found in a Web browser, in that it is able to communicate with the browser in real time, bookmark the current URL into categories, copy and paste, etc. In addition, however, it is also able to perform extra operations on the user-defined categories: importing all the external links in a Web page at once under a category, merging different categories together, classifying a set of documents or a sub-category under multiple categories by creating symbolic links, etc. In collaborative mode, the operation performed by one user is shared by all other users in the same session. For example, if one user clicks on a category tree node to expand it to view its subcategories, then all the users get to view the expanded subcategories. Or, if one user creates new category tree node, then this new node is created for all the users. Using the category editor, one user might alert the other users of a particularly relevant, but obscure, Web site that he found, or another user might suggest an effective and compact organization of the retrieved information.

DASHER's synchronous collaboration capabilities are created using the NCSA's Habanero collaborative tool development framework [5]. We chose to use Habanero, because it provides a set of built-in tools that support meetings, where persons exchange views, such as whiteboards and audio chat. More importantly, it provides a platform independent environment for sharing applications. A single-user application can be changed into multi-user application with only minor modifications. Habanero shares information across application in the form of *actions*. An action is an executable data object, which stores information ranging from mouse clicks to key presses to any user-defined objects (such as tree expansion events and Yahoo search results). First, the client application generates an action. Then, this action is sent to a Habanero arbitrator, which forwards it to all other client applications in the same session. Finally, each event handler of the client applications receives, and executes the action. Augmenting DASHER with synchronous collaboration capabilities involves defining actions to share information along with event handlers to process the actions.

The major actions defined for the category editor can be divided into three types: GUI actions, manipulation actions, and processing actions. GUI actions change how the hierarchical category structure is displayed. Examples include a "tree-expand" action to view subcategories and a "tree-collapse" action to hide them. Manipulation actions operate on the DASHER category data structure ,e.g., actions that add categories, cut categories, copy categories, paste categories, undo, and redo actions. Processing actions send all or part of the category structure to other tools for processing. For example, the "open URL" action sends the URL of a DASHER category entry to a Web browser to display, and the "document analysis" action sends a set of categories to a document analyzer that will extract noun phrases and cluster related documents within the set.

With respect to mobile computing concerns, another way to divide the actions is to partition them into *request* actions and *result* actions. A request action calls upon the client receiving the action to perform the action locally on their computer. A result action contains the result of some computation. In this case the client only needs to display the result. Because of the nature of their functionality, some actions must be implemented as request actions, such as GUI actions and the open URL action where the browser must be local to the client. Currently, for actions that are not necessarily request actions, we choose to implement them as result actions if they potentially can take more than a minute to complete. Clearly, the proper choice of implementation for these actions strongly depends on the *computation time* and the *network communication time*. In the context of mobile computing, where the computation power of a machine may vary due the desire to conserve battery power and the communication bandwidth may vary due to roaming, it would be desirable for the system to decide at run-time the type of implementation to invoke depending on the operating conditions.

IV. ASYNCHRONOUS COLLABORATION

The second mode of collaboration supported by DASHER is the asynchronous collaboration mode. In contrast to the synchronous collaboration mode, users have full control over their own applications. They collaborate, as needed, by exchanging information, possibly over a long period of time, plus they need not to co-exist. As mentioned earlier this mode of collaboration supports both topic and task-oriented collaboration.

The basic infrastructure for this mode of collaboration is a *distributed clipbook* system. This system is composed of several clipbooks distributed across the Internet. We define a *clipbook* as a hierarchical tree organization of other clipbooks. Each tree node is given a name and has the capability to store information and description about that information. We define a hypothetical root, whose direct child nodes are root nodes in their own trees. Each root node is contained in a clipbook server, which is given the name of that node. The hierarchical organization of clipbooks is intended to organize information and to facilitate the information exchange, because it is a categorization of the information being exchanged.

When a clipbook server starts, it registers itself with a directory server, which holds an index to all the available clipbook servers and it holds setting information for DASHER clients. At the beginning of a DASHER session, a client queries the directory server for all the settings, including the available clipbook servers. In this way each client uses the settings suitable for the network it is connected to and selects the clipbook server(s) that suits its goal(s). Using a directory server solves the problem of finding the suitable network settings for the network the user is visiting. But there remains the problem of finding the directory server. In our current implementation, we set up directory server addresses manually. Our approach to this issue in a mobile setting is discussed in section V.

Attached to each clipbook node is a clipboard that is its main storage unit. The clipboard is used to store any type of data plus a description about that data. Its main MIME [6,7] type, plus any other MIME type(s) that the data could be converted to, identify the data stored. We have defined special MIME types for DASHER category structures and view window frames. This allows the exchange of active objects.

A user can request to receive the data with the MIME type that is most suitable for his needs and/or that his software, hardware, and network configuration can support. This makes the system suitable for mobile computing, where software, hardware and network environments can vary widely.

Data can be stored as a copy or as a reference (i.e. a Java RMI proxy). In the copy case, a complete copy of the data is transmitted to and stored into the clipboard (until overwritten by newer data) and then copied to the participant who pastes that data. In the reference case, only a reference to the data is created and stored in the clipboard. When a participant wants to paste the data he has the choice

of pasting either a copy of or a link to the original data. If he pastes a copy then a copy of the data is constructed on his machine. On the other hand if he pastes a link then a copy of the reference is copied to his machine, which creates a link to the original data. In this case if the original data is modified, the pasted data is updated. In our current implementation the data owner cannot close his session while other participants are referencing his data.

V. DASHER IN A MOBILE COMPUTING ENVIRONMENT

Ellis, Gibbs and Rein [4] present the spatial component of their time space taxonomy as a dichotomy, same place vs. different place. But, under mobile computing, with its notions of location-based computing and varying quality of service (QoS), the dichotomy becomes a scale, based on geographical proximity and communication bandwidth. While these mobile considerations do introduce added complexity to collaborative tool design, we think that they also provide benefits. Geographical proximity can be used as a context to discover relevant collaborators and information. QoS can be made explicit, then used to select the appropriate level of interactive collaboration.

Currently, we assume that users are part of social groups that allow them to interact outside of the system to develop foci of interests, around which DASHER sessions are then created. We plan to bring this interaction within the DASHER system. Below we describe a scenario where this interaction involves interchanges of topics of interest among a group of people (and information agents) in a geographical region.

For example, as rescue workers approach a building in an incident area, their personal digital assistants (PDAs) detect the presence of each other and form a DASHER session so that the workers can exchange information about the incident. The PDAs detect the building has an information server. The server is asked to join the session to provide information about the building, such as floor plans and fire extinguishing systems. Late arriving rescue workers who join the session automatically receive all this information.

Our design of the mobile DASHER is based on two interacting agents, the *information traffic expeditor* and the *information gatekeeper*, per computing client.

The information traffic expeditor is the main interface to the network. One of its main functions is to identify available servers and other clients on the network, and to interact with them to move information. For this identification process our plan is to have the expeditor query the available directory servers. Using JiniTM's [8] lookup service will solve the initial problem of finding the directory servers. The creation of the expeditor layer above Jini is to add a degree of freedom to the network interface. This allows the addition or change of services without disrupting the upper layers of the DASHER system.

The information gatekeeper in the DASHER system provides a mechanism to selectively join appropriate collaborative sessions identified by the information

expeditor. The gatekeeper decides based on user's current information needs and priorities whether to ignore it or to selectively join it (e.g., a rescue worker entering a store building during an incidence, wants the asynchronous collaboration server on sale items ignored). This is driven by matching the meta-data of information spaces of the collaboration servers with user's requirements, and any instructions explicitly provided by the user.

Also, the expeditor deals with the problem of variation in the network QoS as mobile computers roam. For the DASHER system to work seamlessly over different network QoS, we need a caching system and an intelligent data format selector. The caching system is used to speed up the data access and to allow the access to the data even in case of a disconnection (offline mode). The main purpose of the intelligent data format selector is to choose, based on network QoS conditions, the appropriate data (MIME) type to exchange information. So, in case of high network QoS, we select rich data types, while in the case of low bandwidth we select data types with less description (e.g. graphics vs. text HTML pages).

Maintaining consistent information space views and data between collaboration servers and PDAs in collaboration sessions is one of the most important issues in the caching mechanism. Consistency may be lost due the disconnection of the participants from sessions either intentionally or unintentionally (i.e. crossing a no network area). This disconnection will affect both modes of collaboration.

We plan to explore various alternatives to maintaining consistency in *synchronous collaboration mode*. One approach is to make use of the event play back feature of Habanero. When a participant reconnects after a disconnection, Habanero plays back all the dropped events to allow the participants to catch up. The major problem we foresee with this approach is that it assumes the computational environment is static. For example, if the played back event is query Yahoo and Yahoo just happens to change its contents, then the participants would be out of sync. Or, if the played back event is modify a shared repository, then this may clobber modification to that repository done by others. Another approach is to maintain policies and constraints among various types of data structure, and develop a smarter resynchronization mechanism. For example, the mechanism might reason based on the write policy of a particular repository that all incremental updates can be safely ignored. Various avenues of this synchronization problem need to be investigated. This is one of the issues where we hope to benefit from the interactions at this workshop.

Losing connection in asynchronous collaboration mode causes two problems, *dangling links* and *data synchronization*. There are two issues related to linked data. The first is concerned with referencing links during disconnection, and the second is concerned with reestablishing the link after reconnection. To deal with the first issue, we will use caching on the side of the data reference, to have data present incase of a network disconnection. For the second issue we will use Jini's

distributed leasing mechanism [9] to deal with the reestablishment of the link. Since we allow local caching and multiple users, general conflict resolution in data synchronization is difficult. However, we plan to provide mechanism users to refresh their cache, and to upload changes if the data on the persistence server is unchanged.

VI. RELATED WORK

With respect to computer supported cooperative work (CSCW) the DASHER Information Space Analysis Tools intersect with two of its areas: collaborative authoring and collaborative filtering. Collaborative authoring systems support a group of authors working on a shared document. Various authoring systems have been developed. The Collaborative Editing System (CES) [10] focuses on database management issues, such as access control, concurrency control, and long-lived transactions. The BSCW (Basic Support for Cooperative Work) Shared Workspace system [11] focuses on providing integrated, multi-platform, widely accessible authoring tools based on Web technologies (HTML, browser, CGI, HTTP). WebDAV (Web Distributed Authoring and Versioning) [12] is an IETF standard that extends HTTP to include version management, advanced collection representation for documents and access control. Database management issues is not our research focus. Currently, we are taking a laissez faire attitude providing few data consistency mechanisms, such as concurrency control and transaction. We do plan to incorporate these mechanisms in the future. In spirit we are much closer to BSCW in that we want to provide an integrated, multi-platform, widely accessible system. However, instead of HTML we chose Java as the basis of implementation. Java is not only widely available and platform independent, but also allows for more flexible user interfaces and better interactivity. DASHER supports synchronous collaboration, which none of the above three systems support. Also, it supports the exchange of active objects, not just passive documents.

In CSCW collaborative filtering usually implies an indirect process where a social group collectively recommends and filters information resource, see the March 1997 issue of Communication of the ACM for an overview of several of these systems [13]. Although we do provide this kind social filtering service (see [14]), here we present a much more direct interaction, where users synchronous decide on the organization and content of an infospace.

VII. CONCLUSION AND FUTURE WORK

We have presented DASHER's collaborative tools for forming task-oriented repositories. A Repository typically contains foci of activities that develop based on shared user interests. Our topic-oriented collaboration tools support these tightly knit user groups by giving them the dynamic ability to form sessions to be tailored to their interests, and real-time editing capabilities to collaboratively impose hierarchical structure on the repository. Sometimes the foci intersect, i.e. user group interests overlap. Our task-oriented

collaboration tools provide capabilities for user groups to exchange information, both passive data and active objects, using distributed clipbook servers.

There are several directions to extend this work; this paper has focused particularly on those which support mobile collaboration. This include context-based detection of available collaborators, level of collaboration based on quality of services, data caching and integration, and distributed and batch processing. We also discussed the notion of result and request actions as a way to balance computation and communication time. We want to add a third "server" action, in which a client makes a request a server to perform the computation, then the server sends the result to all the clients. This is useful in mobile computing situations, where all the clients have minimal computation power. Another issue is access control and floor control. Currently, we rely on human social conventions to determine who may access what data and when can it be accessed. Any controls that we do add to the system must be flexible enough to support rapid formation of repositories during emergency situations. We do not want excessive or rigid data protection scheme that may hinder the formation process.

VIII. ACKNOWLEDGEMENT

We would like to thank the Habanero team at the National Center for Supercomputing Applications at University of Illinois for their help. We specially like to thank Al Woss for making the initial Habanero modifications to DASHER.

IX. REFERENCES

- [1] Neches, R., Abhinkar, S., Hu, F., Eleish, R., Ko, I., Yao, K., Zhu, Q., and Will, P., Collaborative Information Space Analysis Tools. *D-Lib Magazine*, Vol 4, No 9, October 1998. [<http://www.dlib.org/dlib/october98/dasher/10dasher.html>]
- [2] Yahoo. [<http://www.yahoo.com/>].
- [3] GTE SuperPages. [<http://www.superpages.com/>].
- [4] Ellis, C.A., Gibbs, S.J., and Rein, G.L. Groupware. *Communications of the ACM*, Vol. 34, No. 1, January 1991.
- [5] Chabert, A., Grossman, E., Jackson, L., Pietrowicz, S., and Sequin, C. Java Object-Sharing in Habanero. *Communications of the ACM*, Vol. 41, No. 6, June 1998.
- [6] IETF, "Multipurpose Internet Mail Extensions (MIME) Part One: Format of Internet Message Bodies". RFC 2045, 1996.
- [7] IETF, "Multipurpose Internet Mail Extensions (MIME) Part Two: Media Types". RFC 2046, 1996.
- [8] "Jini Technology Architectural Overview", white paper [<http://sun.com/jini/>].
- [9] "Jini Distributed Leasing Specification", [<http://sun.com/jini/>].
- [10] Grief, I, and Sarin, S. Data Sharing in Group Work. In *Computer-Supported Cooperative Work: A book of Readings*, edited by Irene Greif. San Mateo, California: Morgan Kaufmann Publishers.
- [11] Bentley, R., et al. Basic Support for Cooperative Work on the World Wide Web. *International Journal of Human Computer Studies: Special issue on Novel Applications of the WWW*, 1997.
- [12] Whitehead, J., and Wiggins, M. WEBDAV: IETF Standard for Collaborative Authoring on the Web. *IEEE Internet Computing*, September/October, 1998.
- [13] Resnick, P, and Varian, H. Recommender Systems. *Communications of the ACM*, Vol. 40, No. 3, March 1997.
- [14] Quan Zhu et al., Searching for Parts and Services on the Web, In *Proceedings of International Symposium on Research, Development, and Practice in Digital Libraries*, Nov. 18-21, 1997, Tsukuba, Japan.