Semantic Multicast: Intelligently Sharing Collaborative Sessions

Son K. Dao, Brad Perry[*], Eddie C. Shek, Asha Vellaikal
Information Sciences Laboratory
Hughes Research Laboratories
Malibu, CA 90265

Richard R. Muntz, Lixia Zhang, Miodrag Potkonjak
Computer Science Department
University of California, Los Angeles
Los Angeles, CA 90095

Ouri Wolfson
Department of Electrical Engineering and Computer Science
University of Illinois, Chicago
Chicago, IL 60607

Abstract

We present novel methods for efficiently sharing the content produced during collaborative interactions among internetworked users. We introduce the concept of semantic multicast to implement a large-scale shared interaction infrastructure providing mechanisms for collecting, indexing, and disseminating the information produced in collaborative sessions. This infrastructure captures the interactions between users (as video, text, audio, and data streams) and promotes a philosophy of filtering, archiving, and correlating collaborative sessions in user and context sensitive subgroupings. The semantic multicast service efficiently disseminate every piece of potentially relevant information to every user engaged in the collaborative session, making the aggregated streams of the collaborative session available to the correct users at the right amount of detail. Given a collaborative session of many overlapping streams, semantic multicast helps decide which streams should be propagated to and archived for which semantic interest groups. This contextual focus is accomplished by introducing proxy servers to gather, annotate, and filter the streams appropriate for specific interest groups. Users are subscribed to appropriate proxies, based on their profiles, and the collaborative session becomes a multi-level multicast of data from sources through proxies and to user interest groups.

---

[*] Currently at MCC, Austin, TX

1 Introduction

Current networked collaborative activities usually consist of an "interaction stream" broadcast over a single network channel – if users have an interest in the session, they must participate in the entire event and process all broadcast information. Furthermore, if a collaborative session involves multiple interactions from interrelated working groups, a user must participate in the full broadcast from all groups to learn the interrelationship. This model of collaboration only supports two modes of operation: a user either actively participates in the session or the user does not participate at all. In actuality though, a collaborative session is one in which users participate in varying levels at varying times and multiple working groups, or interaction streams, concurrently overlap. Minimal support, if any, exists to decompose collaborative sessions among related working groups, filter and share information between groups, efficiently recall specific discussions, support occasionally disconnected users, or augment the interaction streams with "hooks" to online information repositories.

In this paper, we introduce the concept of *semantic multicast* as a means to realize a large-scale shared interaction infrastructure for the seamless collection, indexing, and dissemination of the information produced in collaborative sessions. This infrastructure captures the interactions between users (as video, audio, text, and data streams) and promotes a philosophy of filtering, archiving, and correlating collaborative sessions in user and context sensitive subgroupings. The rest of the paper is organized as follows. Section 2 outlines the semantic multicast architecture and some of the most important research issues in realizing such a collaborative infrastructure. Sections 3 to 6 address issues on semantic multicast graph construction, collaborative stream archival and access, collaboration stream annotation and correlation, and realization of semantic multicast service over IP multicast, respectively. Section 7 compares semantic multicast with related work. Finally, Section 8 conclude the paper with a look at how a number of applications can benefit from semantic multicast.

2 Architecture and Infrastructure

The underlying basis behind creating a multicast service for collaborative sessions is to efficiently disseminate every piece of potentially relevant information to every user engaged in the collaborative session. The basis of semantic multicast is to introduce filtering, archiving, and contextual focus along the dissemination paths so as to reduce and specialize the information to particular groups of user needs and interests.

We consider a collaborative session to consist of two primary collections of objects: (1) the many streams of interaction arising in the session that need to be disseminated to all users in various forms; (2) the actual users themselves with their specialized interests, degree of interest, and abilities to participate in the streams of the collaborative session. The goal of semantic multicast to is create a logical dissemination, filtering, and archiving structure for making the streams of the collaborative session available to the correct users at the right amount of detail and in an efficient manner. Given a collaborative session of many overlapping streams, a *semantic multicast graph* establishes the flow of collaborative streams between interest groups. Each node of the graph represents a range and quality of semantic topics present in the collaborative streams and/or user profiles, and is allocated to a semantic proxy server in the network that performs data gathering, archiving, and filtering for the semantic coverage of the node. The goal of the graph is to define a semantically focused, yet network optimized, allocation of streams of the collaborative session to proxies and user groups. Users are subscribed to appropriate proxies, based on their profiles, and the collaborative session becomes a multi-level multicast of data from sources through proxies and to user interest groups.

Each proxy gathers the streams related to the one or more semantic topic areas defined by nodes in the semantic multicast graph. For each stream the proxy archives the stream, filters it for the specific semantic topic, merges it with similar streams, and disseminates the stream to all users subscribed to the proxy. As the proxy archives the stream, it performs a more detailed and offline analysis to provide additional semantic structuring for subsequent retrieval and feedback to the semantic multicast graph.

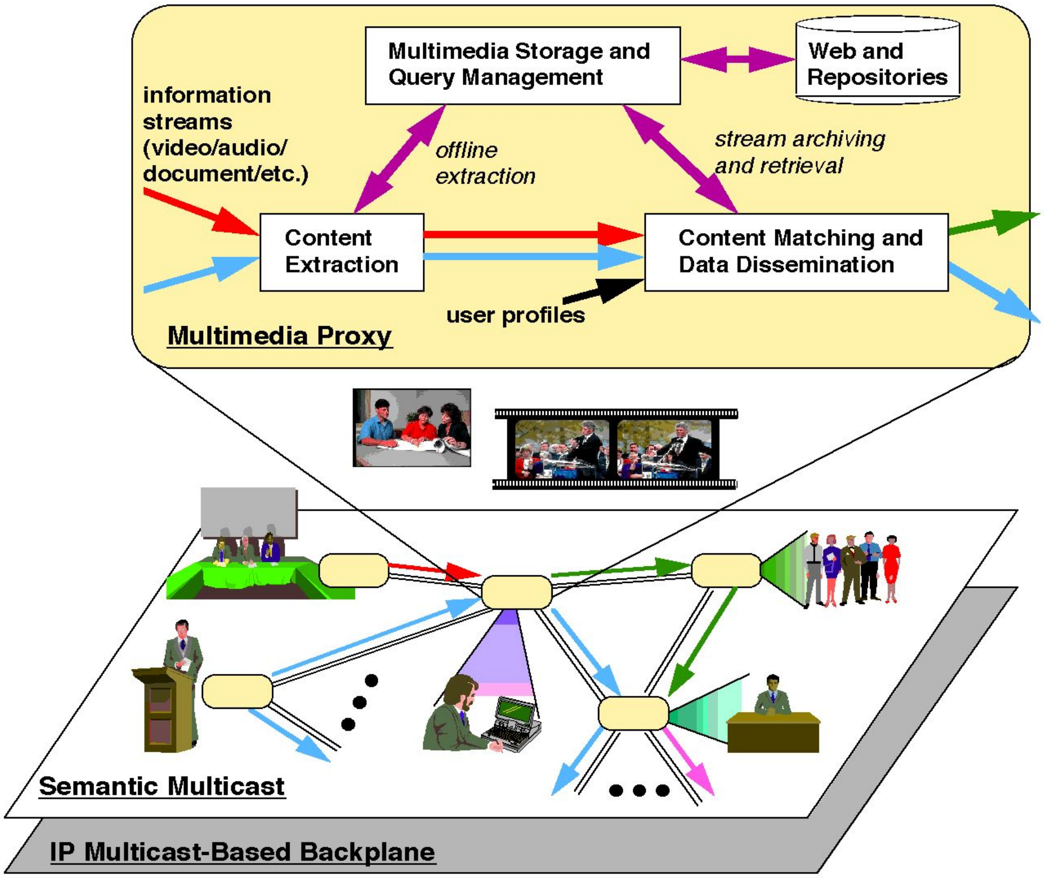

Figure 1: Semantic Multicast Architecture

Figure 1 depicts the general components comprising the semantic multicast process. Specifically, an information stream enters the network at any node of the semantic multicast graph. Each such stream is subjected to a "quick" content extraction process to approximate the general semantics (i.e., subject and topic areas) exhibited by the stream. Based on this content summary, the stream is matched against the semantic coverage of nodes in the graph and forwarded (i.e., multicasted) to the appropriate proxies in the network. Upon receiving an annotated information stream, a proxy archives the stream and filters it into a level of detail appropriate for disseminating to its user groups. Furthermore, a proxy may choose to merge two or more incoming information streams and disseminate (and archive) the fused data product. The proxy archives and performs detailed offline content analysis and indexing of the stream. As new detailed semantics are uncovered in the stream, this information is fed back into the content matching process to see if additional proxies are "interested" in the content discovered. At any time, a user may interact with a proxy and request specialized "stream summaries" be created and sent to the user. To reiterate, Figure 1 demonstrates an advanced architecture for filtering, archiving, and disseminating collaborative sessions as a combination of realtime and non-realtime as well as annotated and summarized information streams. The ultimate goal being a scalable and reactive model for distributing streams of collaborative sessions based on the underlying semantics in the streams and profiles of the users.

Semantic multicast adds a level of "semantic indirection" to the IP multicast model. As a result, the group communication model of IP multicast forms a natural basis for dissemination of information amongst the various logical entities in the semantic graph. The distribution of information from proxies to user groups and other proxies, of control information among proxies, and of new user announcements all have an efficient realization in standard IP multicast techniques.

3 Dyanmic Semantic Multicast Graph Configuration

The proper construction and maintanence of semantic multicast graphs amongst a set of proxy servers and an evolving user base is central to the semantic multicast service. We consider attacking the semantic multicast graph configuration problem with the following constructs:

- Every stream has a set of metadata summarizing the scope of content in the stream. This metadata is an abstract summary of the stream, the proxies will further refine and extract content from each stream as needed.
- Every user has a profile summarizing his/her topics of interest and the "level of interest" attached to each topic.
- The stream metadata and user profiles are constructed from a common set of fields for describing different aspects of the content to be found in the collaborative session.
- Various fields are associated with weighted similarity ontologies; that is, structures that gather instance values and ranges into hierarchical similarity classes. Given a set of profiles (or stream metadata) we can compute the most specific generalization between elements of the set by using the similarity ontologies. These ontologies can be provided by domain experts (such as a thesaurus) or automatically learned from samples of commonly co-occurring values.

Each node in a semantic multicast graph is managed by a proxy in the network and is associated with a profile for the semantic coverage and quality of coverage of information archived in the node. The coverage defines the semantics maintained in the proxy and the content filtering it performs on streams it receives and archives. When constructing a semantic multicast graph, we create a node in the graph if: (a) a large set of user profiles are interested in the node; and (b) at least one stream in the collaborative session is anticipated to be semantically covered by the node. We use the similarity ontologies to find common generalizations and evaluate the semantic intersection between user profiles and stream metadata. This generalization process is an extension of the profile aggregation process described in [5] and summarized below.

We compute the "intersection" between profiles by using the similarity ontologies to generalize the interests described in each (intersecting stream metadata is a similar process). If two profiles are annotated with the same topic, then their intersection is the topic itself. Alternatively, if two profiles are annotated with different topics that have a common parent in the similarity ontology, then their intersection is the parent topic with an error proportional to the weights between the topics and the parent in the ontology. Finally, if two profiles are annotated with some common topics and some uncommon topics, then their intersection is the set of topics generalizing the common topics. In this manner we can compute the intersection between any set of profiles as the weighted generalization of the profiles' fields.

The construction of the semantic multicast graph is a heuristic optimization process that tries to minimize the number of nodes in the graph, maximize the number of users interested in each node's coverage, minimize the amount of generalization applied to any profile intersection, maximize the number of streams covered by a node, and minimize the amount of generalization applied to an stream intersection. Each node in the semantic multicast graph represents a point of archiving and dissemination to subsets of users in the network. In most cases the number and location of the semantic proxies are fairly static since each proxy encompasses a complex archiving subsystem. But, we are not prohibited from mapping multiple nodes in the semantic multicast graph to the same physical proxy server. Therefore, the semantic multicast graph has the potential to introduce a lot of overlapping traffic in the network. For this reason, we augment the algorithms for constructing the semantic multicast graphs to use "geographic proximity" as an additional criteria for deciding which users, or semantics, to aggregate and allocate to proxies in the network.

The semantic multicast graph described above provides a sound framework for defining semantic-based distribution schemes for the data streams comprising a collaborative session. To construct and maintain such graphs in a dynamic network of proxies and users, we assume the following items about the semantic multicast environment: (1) all proxies are connected and communicate on a common information channel; and (2) users "enter" the system by sending a profile to any one proxy currently in the network. The first assumption can be met by establishing all proxies to exist in a common network multicast group and new proxies can be established by joining and announcing them to this group. Given these assumptions, we propose the following approach to dynamically construct a semantic multicast graph.

1. Every proxy announces itself in the network, thus each proxy knows of the existence and geographic location of other proxies. Given the data storage and processing complexities of the proxies, it is safe to assume the size and dynamics of this group is relatively stable.

2. Assume each proxy knows the set of user profiles and anticipated stream metadata for the

collaborative session. Then, each proxy independently computes the semantic multicast graph optimizing this user population and stream metadata.

3. Using a distributed allocation, or election, scheme, the proxies allocate themselves to nodes in the semantic multicast graph. All proxies inform each other of the nodes they are assuming responsibility for.
4. Each proxy informs the users covered by its nodes in the semantic multicast graph that it is the proxy providing content for the user in the collaborative session.
5. When a new data stream begins in the collaborative session, the content summary of the stream is announced to the proxy group. All proxies that manage nodes in the semantic multicast graph that relate to the stream summary join the multicast channel carrying this stream. If a proxy creates a new information product (i.e., merging several streams), it submits this to the network as a new data stream.

This general sequence describes a distributed, yet practical, approach for creating semantic multicast graphs for a group of users in a collaborative session. In order to maintain this graph amidst a dynamic user population, we note the following approach for configuring new users:

1. A new user announces its profile to any proxy in the network. This proxy classifies the user in the existing semantic multicast graph and forwards the user's profile to the appropriate proxy, or proxies, best covering the user's interests – this is the set of "adopting" proxies for the user
2. Each adopting proxy informs the user that it will be its (or one of its) content providers for the collaborative session. If the user's profile is only an approximate match to the proxy's semantic coverage, then the proxy increments its number of "marginally qualifying" users.
3. Whenever a proxy reaches a threshold of marginally qualified users, it requests a reorganization of the semantic multicast graph (by announcing this desire on the channel shared by the proxies). If the request is accepted then the previous algorithm for creating and allocating the graph is performed – thus reorganizing the coverage and subscription of semantics in the proxies.

4 Representation, Capture, and Access of Collaboration Streams

Our goal is to support not only real-time collaborative activities but the capture of collaborative work sessions in a distributed multimedia database for non-real-time offline access and analysis. We utilize a formal model of collaboration [14, 6] which defines the roles of participants, modes of interaction, and protocols for joining and leaving. This model serves not only as the basis for the collaborative work middleware control of a session but also as a part of the database schema for storing data captured during a session. Additional schema information is required to describe the semantic tags that are generated to support flexible, intelligent access to the data. Basically the collaborative work model is synonymous with the database schema for the raw video, audio, text, and data streams associated with the session. Where necessary, control information (e.g., floor activity) will be captured also to support playback.

The establishment of a session requires selection of an existing schema for collaboration (from the data base) or creation of a new one. This schema is expanded to include the semantic information/views which are materialized by the analysis tools that are to be invoked on the session streams. These tools to analyze and generate derived data will be described in a later section will support both online (realtime) and offline processing, with typically the later improving the quality of the metadata since there is no time constraint.

The analysis of the session raw data results in "indexes" to portions of the data and also views on the data. One of the main goals is support of summary or catchup presentations that allow a participant to get up to speed, or quickly review a past session. There are several approaches to this: one is to compress data (e.g., video stream is compressed to a sequence of key frames) and a second is to "project out" information that is not germane to the current purpose. The latter exploits the user profile to determine topics of particular interest.

The schema for captured session streams and associated information is a global schema. For any particular instantiation of this schema (actual session based on the schema) the data is captured across the distributed repositories that are associated with proxies. The schema for a session is used by the collaborative session groupware to control and monitor the session as it progresses. Proxies also have roles in the session and can join and communicate with the session in essentially the same way as other participants. Proxies are also responsible for interfacing with with an associated multimedia repository to record that portion of the session relevant to its semantic context.

Users can dynamically retrieve previously stored session data (e.g. to get a recap of a missed meeting). Playback can require synchronized access to captured streams and derived data at distributed repositories. The nature and timing of component streams which are part of the same session but stored at different repositories may have to be "linked" to support coordinated playback of information that spans epositories.The combination of all proxy data repositories is a distributed multimedia database. A session schema is implemented as a "global view" that spans multiple proxy repositories and playback capabilities allow synchronization of related component streams across distributed repositories.

4.1 Support for Disconnected Users

Wireless computing is becoming increasingly more popular because it naturally supports itinerant users that have a desire to remain "wired" even while on the move. However, one of the biggest drawbacks limiting the adaptation of wireless and mobile computing in online information systems is the unreliability of connection. Specifically, access to information over a wireless network may become unavailable when a mobile user is out-of-range. Semantic multicast supports the notion of "collaboration on the move" by means of mechanisms that support mobile users, who may be frequently disconnected, in a seamless manner. This goal is realized, transparent to users, by having each proxy managing a node in a semantic graph keep track of the connection status of its group members, automatically record the time when a user becomes disconnected and resynchronize the user's view of the session when reconnected. We adopt this proxy-centric approach because proxies are at a better position to keep track of the connectedness of mobile users. In many wireless data service, servers receive "negative" acknowledge of user disconnection. Another mechanism that has been proposed for networks which do not support such negative acknowledgment protocol is for the server to poll clients (possibly at a low frequency that is inversely proportional to the number of clients to control the control overhead) [12]. When a disconnected user reconnects, the "catch-up" capability of semantic multicast proxies allows "fast-forward" presentation of the session (based on user-profile, e.g., profiles to contain relative importance so that the less interesting topics can be filtered out when users are connected at "catch-up" mode) to allow users to quickly resynchronize with subsequent information.

Note that the support of disconnected users is similar to the following archived data retrieval scenarios: (1) a new user wants to catch up with the status of the session to establish a context, (2) a connected user want to "replay" a previous portion of the session. Unlike disconnected operation support, which is transparent to the user, these access patterns require a different data request mechanism that involves the user explicitly asking for an interval of data predicated on an interest profile. The proxies provide support for such "explicit interval" queries; but, for disconnected user support, it is also the task of the proxy to derive the correct interval to server to specific users.

5 Content Extraction and Correlation

Collaborative sessions will generate various types of data streams, primarily raw audio, video and graphics data, along with application-specific data types. For example, a collaborative session involving computer-aided design techniques can output video/audio streams as part of a teleconference as well as CAD-specific data such as SPICE files. As part of its operation, semantic multicast aims to filter, archive, fuse and disseminate these data streams. As described in earlier sections, the semantic content of the data streams will be an important role in being able to support the above functionalities. In its raw form, data types such as video and audio are not amenable to automated semantic interpretation and typically have to be enhanced with other features, which are either manually created/attached or are extracted by analyzing the raw data. Some of the enhancements which are required to support the above functionalities include the following:

1. Tagging annotations to the data streams so as to assist filtering at the proxies.
2. Creating indices so as to allow fast retrieval of desired data based on user interests.
3. Generating summaries so as to provide synchronization and review support to the occasional disconnected user.

These automated or semi-automated enhancements are either created at the data source or/and at the mid-way proxies in the network, and can either form an additional synchronized stream along with the data streams so as to enable real-time filtering or be statically located at the proxies where the data is archived for later recall. These multimedia data stream annotations, indices and summaries should be based on the collaborative information model or schema as described in the earlier section and be closely matched or tied to end-user interests, which is of course expressed in terms of a user profile. Depending on the sophistication of the underlying processing operations, these enhancements are produced in real-time, near real-time or non real-time with the subsequent increase in processing complexity typically resulting in greater precision in the operation involved. Our overall system requires both on-line (real-time) processing (for example, at the sources as "live" data are generated) and off-line processing (for example, at the proxies where the data are archived) techniques. For example, a live videoconferencing session can use a real-time speech understanding system at the source, which in general will possess relatively low accuracy, in order to create a rough annotation so as to allow instant filtering whereas a non real-time higher accuracy system can be utilized at the proxies where the sessions are archived.

As mentioned earlier, one of the important module in order to provide the different functionalities to semantic multicast is the creation of an annotation stream to tag the video and audio sections. Several kinds of annotations can prove to be useful - keywords, video scene change tags, representative sample frame etc. This annotation can be of two kinds: manual or automated. An example of a manual annotation is that of a person taking notes during a collaborative session. If these notes are to correlated with the audio and video streams in conjunction with a common timestamp, it provides a powerful means of annotation [13]. In automated generation of annotations, it is necessary to process the raw data streams as the original representations are typically just digitized, compressed and packetized representations of the original analog signals. Examples of such common tools for collaboration involving video and audio over the Internet are vic [12] and vat [10] respectively.

An important form of annotating video/audio data is by the use of keywords which describe the content of the video at different levels of temporal resolution, i.e. frames, scenes, collection of scenes etc. Speech understanding systems which can transcribe the audio stream in order to create a text of the spoken words can be utilized towards this end. While speaker-independent, unlimited-vocabulary, connected speech-recognition systems are still in its infancy, it has been shown that performance greatly improves by the use of domain knowledge, where the context has been reasonably restricted. CMU's SPHINX-II speech understanding system has shown to recognize with 90-percent accuracy in benchmark evaluations, speaker-independent continuously spoken speech with a vocabulary of 60,000 words [9]. Such systems will allow the creation of a time-aligned transcript of the spoken words contained in the audio stream. In a similar application, the above mentioned system has been utilized effectively by the CMU Informedia digital library project to index their video archive [17]. Compared to audio, automated semantic interpretation of video is far beyond current capabilities, though improvements in optical character recognition techniques should allow extraction of text which appears on the video screen, which can then augment the audio-based transcript. At the next level, natural language processing techniques can be applied to correct and summarize the transcript as well as to identify keywords which will describe logical subunits of the entire session as defined by the video segmentation operation. For example, a conference can consist of different sessions, with each session covering different subtopics. A hierarchical structuring method should be able to segment the video stream as shown in Figure 2 using both video and audio processing techniques.

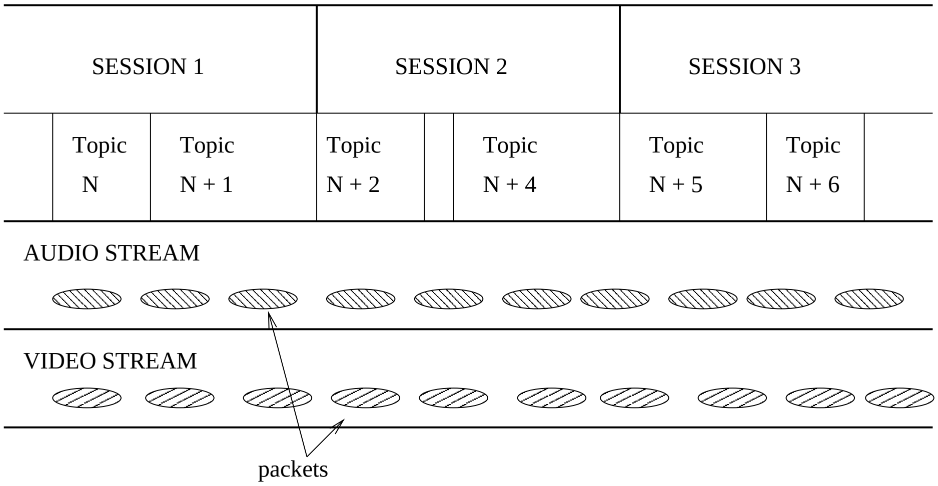

Figure 2: Semantic-based structuring of collaborative video and audio data

Video structuring operations require the segmentation of video into logical subunits comprising of a series of related scenes with common content [15]. Cues from both the video and audio portion of the data can be utilized for this purpose, whereby video processing techniques in conjunction with speech signal analysis are used to automatically locate boundaries of shots, scenes and conversations. While most of the currently available scene-change detection algorithms have been developed for video data files which are static, these techniques have to be adapted to a real-time video transmission scenario, where the video has been compressed and packetized. Of particular importance are methods which operate on the bits in the compressed domain [22, 16]. Apart from such scene change tags, other types of annotations which can be derived from the processing of video data are low-level image features such as spatial-color representations, texture measures etc., which can be important features to search for other scenes which are "visually" similar. Another example of a useful annotation is a keyframe which is a representative sample from a particular scene. Model-based techniques can also be utilized to characterize a particular scene as being of a specific type (e.g. conference table, facial close-up, viewgraph etc.) in a shot classification scenario.

In addition to annotation, another important functionality is the indexing of the video in a given archive. The annotations which are generated as described above can be utilized for indexing. Similarly, techniques to extract effective video summaries or abstracts which are concise versions of the video at different temporal resolutions in a real-time environment will also be part of the semantic multicast system.

6 Semantic and IP Multicast Coordination

We need a robust and scalable communication substrate to support semantic multicast. As stated earlier, one of the basic goals of semantic multicast is to make supplies meet demands. Demands come from clients of semantic multicast sessions. Data supplies come from both original data sources and proxy servers. As matchmakers between raw data and heterogeneous clients, proxy servers need to know the profiles of existing data sources, as well as that of the client demands. Some sources may also have the flexibility of supplying data in different forms and thus behaving as proxies as well.

Our design uses IP multicast as a basic building block. IP multicast serves two distinguished functions, one being the most efficient way to deliver the same data to multiple receivers, the other being an information discovery vehicle – a client can multicast a query to a relevant group when it does not know exactly whom to contact for the service. Our communication substrate design makes use of both features. We multicast both clients queries and proxy services so that all participating entities in the semantic multicast easily find the information needed without any manual configuration. We also multicast all data so that it can be most efficiently disseminated to (potentially large) interested parties. More specifically, we propose to bootstrap the auto-configuration of semantic multicast by use of a well-known multicast address to serve as a "radio channel" to broadcast information to all participating entities. To the first degree of approximation, we envision that all sources and proxy servers send announcement to the "radio channel" their profiles and the corresponding multicast addresses (that their data is being sent to). Thus when a new client comes on-line, it can easily find information about all existing data services and join the multicast group(s) which provide the desired service, or it may start sending its profile to the information channel if the desired service is not available, hoping that some proxy server will learn its request and offer the needed service (by providing the tailored service from either the origin source data, or data from existing proxy servers).

A single information channel can provide adequate services to semantic multicast sessions with a moderate size. When the session size goes up (due to more data sources, more proxy servers, more clients, and/or a larger number of diverse service requests), however, everyone's announcement transmission rate must slow down in order to keep the overhead within a comfortable limit (in a way similar to SDR's and RTP's approach to scalability). As a result, after a new client tuned to the information channel, it will have to wait for a certain amount of time to hear all the announcements. To gain both scalability and responsiveness of this "information channel" approach, we propose to automatically build a hierarchy of announcement channels by applying semantic aggregation. Any entity in the session may propose a new multicast channel to group all the announcements that fall within a certain category and move them to that separate channel, and then multicasts to the main channel only the binding of the subject and the corresponding multicast address. For example, one may group all video data announcements into a video channel. The same approach can be applied recursively when those subchannels become large.

In addition to auto-configuration of semantic multicast structure based on semantics, we also propose to make the structure automatically adjust itself according to the participants distribution, the observed traffic load, the performance required, and the changes in topology and membership. Our earlier results from Scalable, Reliable Multicast (SRM) work [7] are readily applicable here. For example, if the clients of the same proxy server learned that they are densely populated around two metropolitan areas that are far apart, then those around the same can request a new proxy server with their area. Furthermore, if one proxy server finds that it no longer has any clients, it can stop the current service and change to a new role based on the observed requests. In general, we note that the translation between the "logical" information channels in the semantic multicast graph and "physical" communication channels in IP multicast is not a simple one-to-one mapping. Instead, the physical communication substrate must be hierarchically organized to handle the number and volume of channels introduced by the logical organization. Furthermore, the physical communication substrate can provide feedback to the logical organization and allocation of proxies to better utilize network resources. Therefore, the coordination between semantic multicast and IP multicast will be a library that efficiently allocates the wealth of logical channels to an efficient structure in the IP multicast substrate.

7 Related Work

Semantic multicast extends various other approaches for dynamic information sharing, including Internet multicast, information dissemination techniques, multimedia representation and storage, and multimedia content extraction.

Internet multicast defines an efficient mechanism for distributing entire data streams to dynamic user groups. The multicast is based on the idea of constructing "routing trees" that intelligently interconnect all listening users in an attempt to minimize network traffic and load. The benefits of Internet multicast are its scalability and decentralized model of subscription and receipt. Unfortunately, this network-level multicast protocol enforces a single level of indirection for delivering group data – either receive all data on the multicast stream or receive none. Internet multicast clearly defines an essential layer of information delivery in collaborative applications; but without integrated application support to partition and filter the information flows, multicast will never emerge to truly interconnect users at varying levels of detail and interest.

Our "information channel" design closely follows the basic approaches taken in SDR, and substantially extends them to semantic multicast applications. As SDR, we use a well-known multicast address for initial information dissemination. Unlike SDR, however, after using the well-known channel for bootstrapping, our design can then automatically configure a hierarchy of information channels by the demand. Thus our design is expected to scale well with large numbers of media types, proxy servers, and clients. Similarly, our proposal for automatic readjustment of semantic multicast groups closely follows the basic approaches taken in RTP [1] and SRM [7] however we also substantially extend the basic approaches to utilize unique features and flexibilities offered by the application level semantic multicast. RTP uses feedback from receivers to adjust the sender transmission, SRM uses measurement to dynamically adjust the request and retransmission timers. In semantic multicast, proxy servers can adapt the service they offer, and the locations where they offer services by observed user interest and location distributions.

As alluded to above, we base our strategy for constructing semantic multicast graphs on a sound body of work in "cooperative query answering" [4]and textual information dissemination techniques [19]. Cooperative query answering provides an advanced framework for computing "neighborhoods" of information relevant, an varying relaxation thresholds, to user queries. The relaxation performed in this work is based on hierarchical nearness structures that are automatically created from analyzing the correlation in the databases themselves. The main-stream cooperative query systems, though, primarily operate atomically on each individual user query – little work has been done to extend the techniques for establishing generalized neighborhoods for classes of queries, or user profiles, as proposed herein. The SIFT tool from Stanford [YM95] represents the pinnacle of textual dissemination techniques and is being used as the basis of many commercial service on the Internet, such as those offered by Reference (http://www.reference.com/) and the LA Times (http://www.latimes.com/). The basis of SIFT is to represent users by profiles of the keywords they are most interested in being informed about. As articles enter the database, SIFT offers an efficient indexing and matching technique to identify those user profiles that closely match the article's content. The SIFT work provides a good basis for aggregating and matching articles to profiles, yet needs to be extended for the work proposed herein because it is currently a syntactic, exact-match, and centralized search process specially tailored to textual documents.

There are a number of existing systems for supporting collaborative work but few provide persistent storage of session records. None provide a unified session model to support (a) session protocols, (b) database schema for storage, search and playback and, (c) distributed implementation of persistence for efficient storage and access. Some recent work has concentrated on formal models for defining the nature of roles and communication between participants, i.e., session protocols such as [6]. In the multimedia database community there has been significant work on data models for storing and accessing multimedia presentations [e.g., [3] which are typically constructed with authoring tools as opposed to being recorded version of real activities. Our unified model will enable simpler mapping of recorded session data to the database and playback of versions of that data. Previous systems that have considered indexing and persistent storage of a collaboration have also utilized centralized databases systems. They have not dealt with distribution of the audio/visual stream data for a session into local repositories while maintaining a global view of the data for playback. Synchronization and coordination of play out from distributed multimedia database systems are still an active area of research.

The concept of proxies as "on-the-fly" data stream transcoders has been investigated as part of the BARWAN project [11]. The purpose of such proxies is to bridge the capabilities gap between varying network delivery links. For example, a proxy is often situated on the cross from wireline to wireless delivery links with the proxy transcoding information, on the fly, to use lower bandwidth as it crosses from wireline to wireless delivery. The BARWAN concept of "proxy" is only concerned with monitoring and adjusting the network resources used by data streams crossing connectivity boundaries, minimal effort has been made to propagate the semantics of the stream or the end-users into the transcoding decisions. Furthermore, the placement of proxies within BARWAN is largely a manual process – no facilities are developed to discover where such proxies should be placed based on the users, streams, or network characteristics in the overall session.

View maintenance under disconnect operation in the context of wireless and mobile computing is a particularly relevant area of research. Studies [8, 18] have addressed issues in allocation and caching algorithms for mobile computers that aim to minimize the cache-refresh communication between client and server. Another approach to guarantee data delivery under disconnection is to allow failed data delivery to be repeated when connection is re-established. It is supported in wireless data services such as RAM Mobile Data, which stores a message in a mailbox within the network when the messages cannot be delivered to the addressee immediately, and forward automatically when the addressee re-registers with the network. However, existing techniques do not take into account the real-time aspect of data delivery and the synchronization of data that we propose to address in the "catch-up" session delivery capability.

Speech understanding systems which aim to recognize speaker-independent, unlimited vocabulary, continuously spoken speech are currently being studied by several research groups [21]. Such systems have been utilized for indexing archived video in a video-on-demand application by the CMU Informedia project [17].Scene change detection techniques to identify the boundary locations of different types of shots in both compressed and uncompressed video are also currently under investigation [2] and methods to create skims or summaries for different types of video data have also been topics of recent research [20]. Most of the above work has been based on static data files. In comparison, the semantic multicast effort tries to extend these functionalities to data which has been compressed and packetized as well as corrupted due to losses while traversing a network. Other issues to consider are session archiving and time-constrained filtering of the collaboration session data.

8 Conclusions

In this paper, we have described the design of the semantic multicast service. Sementic multicast's support for effective dissemination of collaborative sessions over space and time (i.e., real-time and non-real-time) and among a diverse user community promises to bring significant benefit to applications including, but not limited to, distance learning and disaster assessment.

Distance learning has been defined as learning that takes place when the student and instructor are separated by space or time. This type of learning is typically conducted through electronic means and is often characterized by high-bandwidth transmission of text, image, audio and video signals. There are several aspects to distance learning which are closely related to our concept of semantic multicast. A typical learning scenario is a multi-level one-to-many process wherein there is a single source (teacher) and multiple recipients (students) who often form "study groups" based on common interests and geographical location. This type of group communication between geographically separated entities can be ideally supported by IP multicast whereby a single copy of the course data can be effectively distributed over a network. But, the concept of study groups means that certain groups of students will be furthering the standard course distribution with background material and exercises. The application is rich in semantics as the students and/or study groups can have diverse well-defined interests with definite overlaps, which clearly fits our paradigm. Finally, distance learning allows students to retrieve material at a later time thereby requiring the materials to be archived, which also closely corresponds to our overall design of interaction. In fact the indexing and summarizing facilities of our system will naturally enhance current distance learning methods. For example, students can easily get summaries of previous lectures and be pointed to other material related to their interests that have been generated elsewhere.

Disaster assessment is concerned with bringing medical, monetary, and/or safety reinforcement to areas devastated by natural (or man-made) events, such as earthquakes, floods, and civil unrest. In typical scenarios, reinforcement teams are sent to independently administer relief to isolated points of the disaster area – these teams independently interact but minimal collaboration occurs across the teams or upward to higher levels of decision making. It is usually not until after the disaster has been settled that the amount of redundant effort and shared lessons/information is identified. Semantic multicast allows different teams to be connected to the same overall collaborative session so that the close interactions among members of one team are continually analyzed, abstracted, summarized, and disseminated to other teams encountering similar circumstances and needs. In this manner, we see that semantic multicast is an ideal environment bringing together many interrelated collaboration streams into an overall collaborative session that is intelligently archived and disseminated through a shared interaction metaphor.

References

[1] *RFC1889: RTP: A transport protocol for real-time applications.*

[2] G. Ahanger and T.D.C. Little. A survey of technologies ofr parsing and indexing digital video. *Journal of Visual Communication and Image Representation*, 7(1):28–43, 1996.

[3] R. Botafogo and D. Mosse. The morena model for hypermedia authoring and browsing. *IEEE Multimedia*, pages 42–49, 1995.

[4] W.W. Chu, H. Yang, K. Chiang, M. Minock, and et. al. Cobase: a scalable and extensible cooperative information system. *Journal of Intelligent Information Systems: Integrating Artificial Intelligence and Database Technologies*, 6(2-3), June 1996.

[5] S. Dao and B. Perry. Information dissemination in hybrid satellite/terrestrial networks. *Data Engineering Bulletin*, 19(3), September 1996.

[6] H.P. Dommel and J.J. Garcia-Luna. Floor control for multimedia conferenceing and collaboration. *ACM Multimedia*, 6(1), 1997.

[7] S. Floyd, V. Jacobson, C.Liu, S. McCanne, and L. Zhang. A reliable multicast framework for lightweight session and application layer framing. In *Proceedings of ACM SIGCOMM*, Aug. 1995.

[8] Y. Huang, P. Sistla, and O. Wolfson. Data replication for mobile computers. In *Proceedings of ACM SIGMOD*, Minneapolis, May 1994.

[9] M. Hwang, R. Rosenfeld, E. Theyer, R. Mosur, and et.al. Improving speech recognition performance via phone-dependent vq codebooks and adaptive language models in sphinx-ii. In *Proceedings of IEEE Intl. Conf. Acoustics, Speech and Signal Processing*, pages 549–552, 1994.

[10] V. Jacobson and S. McCanne. Visual audio tool. Technical report, Lawrence Berkeley Laboratory, ftp://ftp.ee.lbl.gov/conferencing/vat.

[11] R. Katz and et.al. The bay area research wireless access network (barwan). In *Spring COMPCON Conference*, 1996.

[12] S. McCanne and V. Jacobson. Vic: A flexible framework for packet video. *ACM Multimedia*, November 1995.

[13] S. Minneman, S. Harrison, B. Janssen, G. Kurtenbach, and et.al. A confederation of tools for capturing and accessing collaborative activity. In *ACM Multimedia*, pages 523–534, San Francisco, CA, 1995.

[14] S. Rajan, P.V. Rangan, and H.M. Vin. A formal basis for structured multimedia collaboration. In *Proc. IEEE Multimedia*, May 1995.

[15] Y. Tonomura, A. Akutsu, Y. Taniguchi, and G. Suzuki. Structured video computing. *IEEE Multimedia*, 1(3):34–43, 1994.

[16] A. Vellaikal and C.C. Kuo. Joint spatial-spectral indexing of jpeg compressed data for image retrieval. In *Proceedings of IEEE Intl. Conf. Image Processing*, pages 867–870, 1996.

[17] H.D. Wactlar, T. Kanade, M.A. Smith, and S.M. Stevens. Intelligent access to digital video: Informedia project. *IEEE Computer*, 29(5):46–52, 1996.

[18] O. Wolfson, P. Sistla, S. Dao, and et.al. View maintenance in mobile computing. *SIGMOD RECORD*, Dec. 1995. invited article.

[19] T.W. Yan and H. Garcia-Molina. Sift – a tool for wide-area information dissemination. In *Proceedings of 1995 USENIX Technical Conference*, 1995.

[20] M.M. Yeung and B.L. Yeo. Video content characterization and compaction for digital library applications. In *SPIE Storage and Retrieval for Image and Video Databases V*, pages 45–58, 1997.

[21] S. Young. A review of large-vocabulary, continuous-speech recognition. *IEEE Signal Processing*, pages 45–57, 1996.

[22] H.J. Zhang, C.Y. Low, and S.W. Smoliar. Video parsing and browsing using compressed data. *Multimedia Tools and Applications*, 1995.