

# Semantic Structure in a Federated Collaboration Environment

John Buford and Venkatesh Krishnaswamy

IP Communications Dept.  
Avaya Labs Research  
Basking Ridge, NJ 07920, USA

**Abstract**—ConnectedSpaces is a novel federated collaboration environment that integrates different collaboration tools and paradigms for enterprise use. Collaborations are organized into containers called spaces. Recently we have extended ConnectedSpaces to enable spaces and their contents to have semantic properties. These properties can be queried, and queries can be embedded in a space or can be invoked by client applications. Different collaboration models are supported with different semantic structure. The design and implementation is described, and example illustrations are provided. Comparison is made with related approaches.

**Keywords**- *semantic structure; multi-paradigm; federation; collaboration tools; cloud-based computing; enterprise communications; application federation*

## I. INTRODUCTION

Many enterprise collaborations start informally, such as an idea communicated via a phone call, instant message or email. Then the idea is discussed further, perhaps through additional emails. Documents might then be created and shared between the collaborators. When the goals and issues are clearer, other collaborators might become involved, through email, collaborative document authoring, or web conferences.

Throughout this process, today's enterprise users work with multiple collaboration and communication tools from different vendors. First, collaborations may involve outside vendors and customers, who may have different tools. Additionally there is currently no single-vendor collaboration platform that adequately incorporates all types of modes (VoIP, IM, blogging, email, document sharing, wikis, web conferences, etc.). Enterprises consequently use a variety of tools. Content created in one tool is not accessible from another tool. Over time, the discussions, blogs, documents, emails, and messages grow into islands of content. An integrated view of all the dimensions of a collaboration is important to the collaboration process but has to be manually managed by the user.



Figure 1 Example timeline of a collaboration

The variety and complexity of tools and collaboration processes make collaboration more difficult than necessary. The exploitation of semantic structure in a collaboration environment is one approach we are pursuing to automate collaboration tasks and elevate the level of interaction for the user. The challenges include: a) specifying a flexible and general purpose model, b) populating the model with content without placing undue burden on the user, c) making the inference processing computationally tractable, d) finding those areas where there is significant value added.

The new results presented here include:

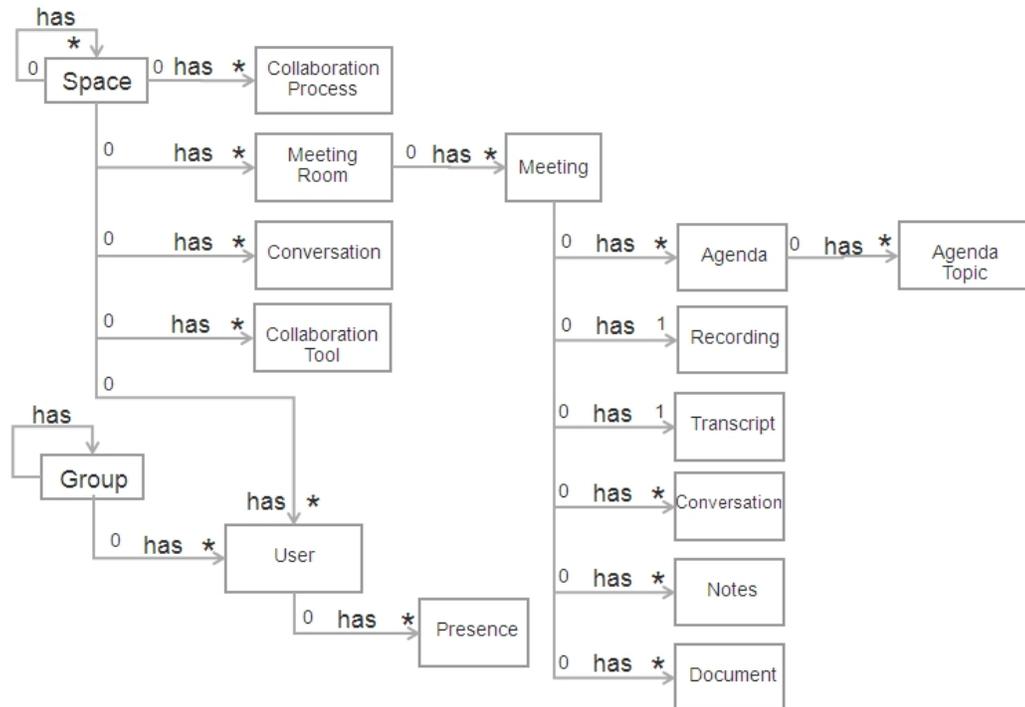
- A semantic model of entities and their properties that are important in enterprise collaboration
- A detailed discussion of categories of queries against such a model, with a number of example using the semantic wiki query language
- A discussion of the system implementation and experimental work

The next section summarizes related work. Section III presents the problem statement. Section IV describes the semantic collaboration model. Section V presents and discusses a variety of queries. Section VI describes implementation work. Section VII summarizes experiments with the system. Section VIII concludes the paper.

## II. RELATED WORK

Collaboration tools enable groups of enterprise users to work as a team, sharing information and communicating as needed, without being co-located. There are many types of tools today, provided by different vendors. For a survey of collaboration platforms, see [3]. We previously discussed types of enterprise collaboration tools in [1]. Here we review work in integrating semantic models into collaboration tools and process.

The semantic web research direction has spawned several directions in adding semantic processing to collaboration tools, most notably the Semantic Wiki such as [4] and the Semantic Desktop [5][6]. A Semantic Wiki combines a formal knowledge modeling syntax with the wiki, such that semantic properties can be explicitly denoted, stored, and queried. Examples of semantic wikis include OntoWiki [7] and Semantic Wiki [8]. SMW+ [9] used in this paper is an extension of Semantic Wiki. SMW+ organizes knowledge according to categories and their properties. Each wiki page can be assigned one or more categories, and can have properties defined. The collection of the categories and



**Figure 2 Semantic model**

properties are the knowledge base, which, depending on implementation can be stored in either a database or triple store for query purposes. SMW+ supports a simplified query language ASK, and work is underway to provide complete support for SPARQL[10] query language. SMW+ also currently supports export of the knowledgebase as RDF.

Work in developing ontologies related to collaboration processes include The CoAKTinG project [11], the Open Semantic Collaboration Architecture (OSCA) Foundation [12], NEPOMUK (Networked Environment for Personal Ontology-based Management of Unified Knowledge) [13], and selected DAML ontologies such as DAML agenda [14].

In comparison to this work, other work either focuses on single paradigm collaboration tools (i.e., semantic wiki), or not specifically on the collaboration semantics (i.e., semantic desktop), or on specialized subsets of the collaboration problem (CoAKTinG, DAML agenda).

### III. PROBLEM STATEMENT

#### A. Overview

Enterprise collaboration often involves large teams working on parallel sub-problems over long periods of time. The collaborations involve users in a rich set of roles. Complex documents and other types of collaborative artifacts are produced and manipulated by groups of users. Increasing efficiency in the collaboration process is important for achieving successful results.

Today's tools require users to determine significance and inter-relationships between activities and the document content.

Relationships between entities that might be important are hidden in different incompatible tools and have to be manually integrated by a knowledgeable user. Identifying expertise needed in a collaboration process depends on social relationships. Efficiency of the collaboration process is difficult to measure.

#### B. Importance of semantics

Semantic models are needed to make key facts about collaboration processes, roles, objects, etc. explicit so that they can be computationally evaluated. Such models need to be integrated into the collaboration tool(s) so that the model can be immediately populated during the related process step, and so that population of semantic models can be automated as much as possible.

Further, semantic processing of the models can be used to inform participants of ways to improve the collaboration, and to understand the state of a collaboration.

### IV. MODELS

This section presents the key models for ConnectedSpaces using the Category/Property formulation of semantic wiki. All entities have implicit properties such as creation date, version, owner, and creator. In the following section we will use these models to illustrate a variety of queries over the collaboration structure. Figure 2 shows the entities and the key containment relationships used in the model.

### A. Space

A space is a persistent container for collaboration. The collaborators consist of a set of owners and members. A space has access rights controlled by the owner(s). Large collaborations can be organized into a hierarchy of spaces. Spaces contain content produced by the collaborators.

A collaboration may follow a collaboration process, such as project management process, customer engagement process, etc. This process may restrict the structure of the space and the roles of the members. (Roles are defined in the user model).

Spaces support both asynchronous and synchronous collaboration. Synchronous collaborations are held in meeting rooms. Large collaborations may organize collections of meeting rooms into conference centers. Asynchronous collaborations are called conversations.

Space			
Property	Req/Opt	Range	Type
Title	R	1 ... 1	String
Owner	R	1 ... *	User
Member	O	0 ... *	User
Content	O	0 ... *	Media
Has space	O	0 ... *	Space
In space	O	0 ... *	Space
Has collaboration process	O	0 ... *	Collaboration Process
Has meeting rooms	O	0 ... *	Meeting Room
Has conference center	O	0 ... *	Conference Center
Has conversation	O	0 ... *	Conversation
Created	R	1 ... 1	Date
Last Updated	R	1 ... 1	Date

### B. Meeting

A meeting

The meeting participants are the moderator, a non-empty subset of the invitees, and zero or more guests. A meeting has a specific start date and continues to the end date. A meeting is held in a location, possibly virtual such as a web conference or virtual world conference room. Side discussions during a meeting are represented by conversations.

A Meeting ID is a globally unique identifier, for example, as provided by MS Outlook Global Appointment ID. A meeting may be associated with a space, for example when a space represents a project and a meeting is held for that project.

A meeting may have an agenda. A meeting may be recorded. There may be a transcript for the meeting recording.

There may be meeting notes for a meeting provided by different scribes.

Meeting			
Property	Req/Opt	Range	Type
Moderator	R	1 ... 1	User
Invitee	R	1 ... *	User
Attendee	R	1 ... *	User
Name	O	0 ... 1	String
Description	O	0 ... 1	String
Start date	R	1 ... 1	Date
End Date	R	1 ... 1	Date
Location	R	1 ... 1	Meeting Room
Meeting ID	R	0 ... *	Hash
In space	O	0 ... *	Space
Has agenda	O	0 ... 1	Agenda
Has recording	O	0 ... 1	Stream
Has transcript	O	0 ... 1	Transcript
Has notes	O	0 ... *	Meeting Notes
Has conversation	O	0 ... *	Conversation

### C. Meeting Agenda

A meeting may have an agenda. An agenda consists of one or more Agenda Topics. The agenda may have keywords, a summary, action items, notes, and associated materials.

Possible agenda types include brainstorming, decision making, planning, status, problem-solving, kickoff, staff meeting, interview, sales pitch, data collection, seminar, conference.

Meeting Agenda			
Property	Req/Opt	Range	Type
Type	R	1 ... 1	Enum
InMeeting	R	1 ... 1	Meeting
Agenda Topic	R	1 ... *	Agenda Topic
Keywords	O	0 ... *	String
Summary	O	0 ... 1	String
Action Items	O	0 ... 1	String
Notes	O	0 ... 1	Text
Materials	O	0 ... 1	Document Store

Has vote	O	0 ... *	Vote
----------	---	---------	------

An Agenda Topic is an identified sub-part of a meeting with a specific set of speakers. There may be an explicit start date and stop date. The Agenda Topic may have sub-topics. Each Agenda Topic may have a summary, action items, notes, and meeting materials. There may be one or more decisions carried out by polling or voting among the participants.

In addition to recording the entire meeting, specific agenda items may be selectively recorded.

Agenda Topic			
Property	Req/Opt	Range	Type
Title	R	1 ... 1	String
Speaker	R	1 ... *	User
ID	R	1 ... 1	Hash
In Agenda	R	1 ... 1	Meeting Agenda
Start date	O	0 ... 1	Date
End date	O	0 ... 1	Date
Sub-topic	O	0 ... *	Agenda Topic
Has vote	O	0 ... *	Vote
Summary	O	0 ... 1	String
Action Items	O	0 ... *	Action item
Keywords	O	0 ... *	String
Notes	O	0 ... 1	String
Materials	O	0 ... 1	Document Store
Has recording	O	0 ... 1	Recording

#### D. Conversation

A conversation is a dialogue. Types of conversations include voice and video calls, instant messages, SMS, email, shared white board, and discussion forum threads.

Conversation			
Property	Req/Opt	Range	Type
Conversation ID	R	1 ... 1	Hash
Type	R	1 ... 1	Enum
Participants	R	1 ... *	User
Has transcript	O	0 ... 1	Transcript

Has recording	O	0 ... 1	Recording
Posts	O	0 ... 1	String
Start date	R	1 ... 1	Date
End Date	R	1 ... 1	Date
Location	R	1 ... 1	Meeting Room
In space	O	0 ... *	Space

#### E. Meeting Room

A meeting room is used to collect a series of related meetings. It may correspond to a physical location. It includes a set of tools for supporting collaboration. It may be contained in a space.

Meeting Room			
Property	Req/Opt	Range	Type
Meeting Room ID	R	1 ... 1	Hash
Meeting Room Name	O	0 ... 1	String
Organizer	R	0 ... *	User
Meetings	R	0 ... *	Meeting
In Space	O	0 ... *	Space
Physical Location	O	0 ... *	Address
Tools	R	1 ... *	Collaboration Tools
Last usage	O	0 ... 1	Date
Size	O	0 ... 1	Number

#### F. Meeting Recording

A recording is a set of content streams that can be used to play back the meeting or conversation.

Recording			
Property	Req/Opt	Range	Type
Recording ID	R	1 ... 1	Hash
Meeting ID	O	0 ... 1	Hash
Agenda Topic ID	O	0 ... 1	Hash
Conversation ID	O	0 ... 1	Hash
Streams	R	1 ... *	Media
Has transcript	O	0 ... *	Transcript

#### G. Transcript

Labelled timeline includes timestamp, speaker, and text. A separate timeline may include keyword tags with timestamps for rapid indexing of the recording.

Transcript			
Property	Req/Opt	Range	Type
Transcript ID	R	1 ... 1	Hash
Recording ID	O	0 ... 1	Hash
Voice transcription	O	1 ... *	Labelled timeline
Keyword	O	1 ... *	Labelled timeline

#### H. Group Scheduling and Calendars

A group calendar is used to schedule group activities. It is associated with a space and has a set of events related to the group.

Group Calendar			
Property	Req/Opt	Range	Type
Calendar ID	R	1 ... 1	Hash
Space	R	0 ... 1	Hash
Events	R	1 ... *	Event

#### I. Vote

A vote allows a set of voters to select from choice. The result is in the tally. A vote can be associated with a meeting or agenda topic. Each vote has a unique id.

Vote			
Property	Req/Opt	Range	Type
Vote ID	R	1 ... 1	Hash
Meeting ID	O	0 ... 1	Hash
Agenda Topic ID	O	0 ... 1	Hash
Question	R	1 ... 1	String
Choices	R	2 ... *	String
Tally	R	2 ... *	Number
Deadline	R	0 ... 1	Date

#### J. Presence

Typical notion of rich presence is augmented to provide status of collaboration state in the system.

Presence			
Property	Req/Opt	Range	Type
User	R	1 ... 1	User
Status	R	1 ... 1	Enum
Current task	O	0 ... 1	String
Current space	O	0 ... 1	Space

Current meeting room	O	0 ... 1	Meeting room
Current meeting	O	0 ... 1	Meeting
Location	O	0 ... 1	Latt - Long
Timestamp	R	1 ... 1	Date

#### K. Social connectivity

The user model provides information to determine relationships between the user and other users, customers, projects, collaborators, and their professional social network. In addition, the user may have different roles in different collaborations. Further, a user's expertise may be a factor in automatic recommendations for collaboration team.

User			
Property	Req/Opt	Range	Type
Name	R	1 ... 1	String
Phone number	R	1 ... *	Phone
Email	R	1 ... *	Email
Location	O	0 ... 1	Latt - Long
Job Title	R	1 ... 1	String
Role	O	0 ... *	Role
Following	O	0 ... *	User
Works for	R	1 ... 1	User
Collaborates with	R	0 ... *	User
Customer connections	R	0 ... *	Customer
Project connections	R	0 ... *	Projects
Professional connections	R	0 ... *	Contact
Expertise	R	1 ... *	String
Presence	R	1 ... 1	Presence

#### V. SAMPLE QUERIES

In this section we illustrate the variety of queries using the model presented in the previous section. The queries are formatted using the semantic wiki query language [15].

These queries can be embedded in the content portion of a space, or can be sent by clients to the server via the API, with the results available.

A formal analysis of the expressivity of the SMW query language is found in [16]. According to this analysis, "several frequently needed features like transitive properties, inverse properties, and class negations are currently not supported." For example, the following queries can not be expressed in the query language:

- Meetings with no agenda
- Meetings with no meeting materials
- Meetings with low attendance

#### A. Data Queries

These queries show the ability to return lists of objects satisfying some property.

- Spaces owned by user John, i.e., objects of category Space with property SpaceOwner equal to User:John. Show the result in ascending order.

```
{#ask: [[Category:Space]] [[SpaceOwner::User:John]]
| ?SpaceOwner | format=table | order=ascending |
source=wiki | merge=false }}
```

When embedded in a space, this query returns the following:



When invoked through the API, the query returns:

```
<?xml version="1.0"?>
<api>
  <spaces result="Success">
    <owner owner="User:John " />
    <count count="35" />
    <spaces>
      <option>AAC7</option>
      <option>ConnectedSpaces Brainstorming</option>
      <option>ConnectedSpaces Development</option>
      <option>ConnectedSpaces Extensions</option>
    </spaces>
  </spaces>
</api>
```

For brevity of presentation, these return results will be omitted in the remaining queries.

- Spaces that user John is a member of:

```
{#ask: [[Category:Space]] [[SpaceMember::User:John]]
| ?SpaceMember | format=table | source=wiki | merge=false
}}
```
- List of meeting rooms

```
{#ask: [[Category:MeetingRoom]]
| ?Organizer | ?Size | ?Creation date| ?Last usage| ?Number
of meetings held | format=table | order=ascending |
source=wiki | merge=false }}
```
- Number of meetings organized by Tom in the past week

```
{#ask: [[Category:Meeting]][[Moderator::User:Tom]] [[start
date::> Aug 3, 2012]]
| format=count }}
```

#### B. Semantic Queries

- Meeting notes that include a keyword

This query is valid provided the exact keyword appears

```
{#ask: [[ Keywords::keyword ]] [[In Meeting::
SpecifiedMeeting ]] | ... }}
```

This query is invalid since substring matching, for example by including a regular expression, is not supported.

```
{#ask: [[Notes::*keyword*]] [[In Meeting::SpecifiedMeeting
]] | ... }}
```

- Action items assigned to John in a specified meeting

This query includes 2 sub-queries indicated by <q> </q>. Within the Agenda Topic set, find Action Items where the Assigned To value is set to User:John, where the Agenda Topic is in an Agenda which has the property InMeeting set to the specified meeting.

```
{#ask: [[Category:AgendaTopic]] [[Action Item:<q>
[[Assigned To::User:John]] </q>]] [[Has Agenda:: <q>
[[Category:Agenda]] [[InMeeting:: SpecifiedMeeting]] </q> ]]
| ... }}
```

- Votes with participation < 50%

To do this, we must add a participation rate field to the Vote object, whose value is computed by the voting tool that implements the vote mechanism, since the query language can not perform general arithmetic over fields.

```
{#ask: [[Category:Vote]] [[Participation Rate::<50]] | ...
}}
```

- Times in meeting where Mary is speaker

```
{#ask: [[Category:Agenda Topic]] [[Speaker::User:Mary]]
[[In Agenda:: <q> [[Category: Agenda]] [[InMeeting::
SpecifiedMeeting]] </q>]] | ?Start date ... }}
```

- Times in transcript where discussion exceeded 3 minutes

Automatic identification of discussion segments in a meeting transcript could be done by determining density of speaker changes in a given time window. Speaker identification with timestamp is part of most transcripts. The frequency of speaker changes would have to be computed outside of the query language.

- Teams that haven't met within 2 weeks

All meetings in the past 2 weeks can be queried. However negative results can not be queried in the current query language.

- Start the meeting when 60% or more of invitees have presence status equal to available

A query can be formulated which returns the count of users with status equal to available. Computing over these two counts must be done outside the query language. The total number of invitees can also be queried.

### C. Social Network

The user entity is defined so that social relationships important in enterprises can be described. Given this, queries such as the following can be supported:

Which employees outside of the team have expertise or experience with a particular domain, product, technology, customer, etc.

Who are the most active contributors in each space.

### D. Collaboration Process

Collaborations for activities such as project management, requirements gathering, software development, and so on can follow specific business process models. A space can have an associated collaboration process. The management of the process might be mediated by another application. Queries which could be made then include:

Which spaces are following the Scrum process

Which teams have completed more than 10 scrums

Which teams have held at least 80% of scrum meetings

Which project deadlines are behind schedule

### E. Notifications and real-time sources

It is useful to set up queries which can trigger notifications, such as in a publish-subscribe mechanism. Notifications related to collaborations include:

Agenda topic has exceeded time limit

Discussion has exceeded time limit

Voting period will close in 1 hour

Unlike complex event processing engines such as Esper, queries in semantic wiki run only when explicitly invoked by the application. However various real-time sources could be used to trigger queries to run or to perform an action if a query is successful, such as:

Using presence, determining if there is a quorum to start the meeting

Routing RSS Feeds to spaces by matching keywords

Forwarding emails and/or attachments to spaces based on keywords and/or recipient matching

### F. Application Domain Queries

The previous examples showed illustrated queries that are independent of the application domain. Semantics related to domains such as sales process, education, remote health care could be include in collaboration systems adapted to those applications. For example:

- Which customers have not received a briefing on the new product plan
- Show customer comments on new product plan briefings
- Which Fortune 500 customers have indicated an interest in product X at a customer briefing in the past 60 days

## VI. IMPLEMENTATION

ConnectedSpaces is implemented with both server side and client side components. The server is a set of extensions to the semantic wiki bundle produced by Ontoprise which is in turn based on the semantic wiki extension to mediawiki. The ConnectedSpaces extensions include:

- Click to call and click to conference from within a space
- Implementation of the model presented in section 3
- Integrated collaboration tools, including sharepoint, skype, Avaya IXC, Avaya Flare, Avaya web conferencing, Avaya Web.Alive, threaded discussion environment, distributed white-board, chat, and short messaging
- Import of email and RSS feeds into a space with automatic keyword generation and semantic processing
- Meeting recording into a space
- Presence indicator when user is active in ConnectedSpaces
- Integration of the user model and authentication with enterprise directory and social network accounts
- API extensions for clients

Client access is currently via either web browser, an MS Outlook plug-in (Figure 3), and an iPad application.



Figure 3 Outlook plug-in

## VII. EXPERIMENTS

### A. Business Collaboration Scenarios

A variety of demonstration spaces, meeting rooms, and meetings have been created to illustrate team collaboration around typical business uses such as:

- Customer care
- New product competitive technology review
- Operations review
- Product architecture and IPR
- Project review
- Revenue forecast and budget

Figure 4, Figure 5, and Figure 6 show screen shots of these spaces.

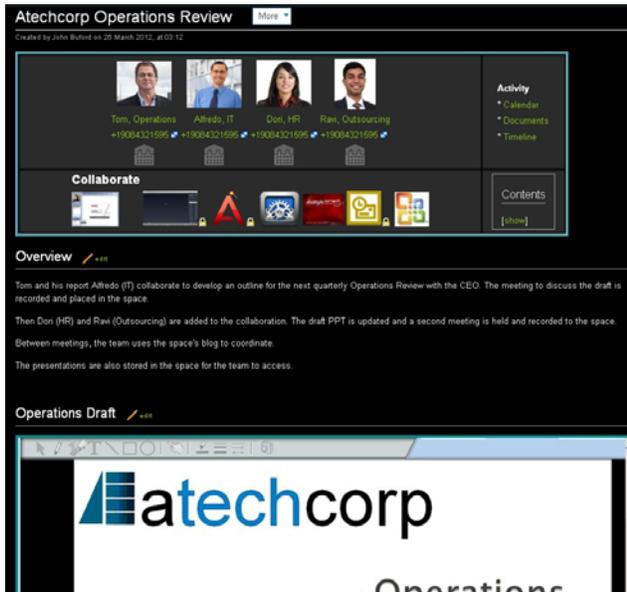


Figure 4 A collaboration space showing team, collaboration apps, and a meeting recording

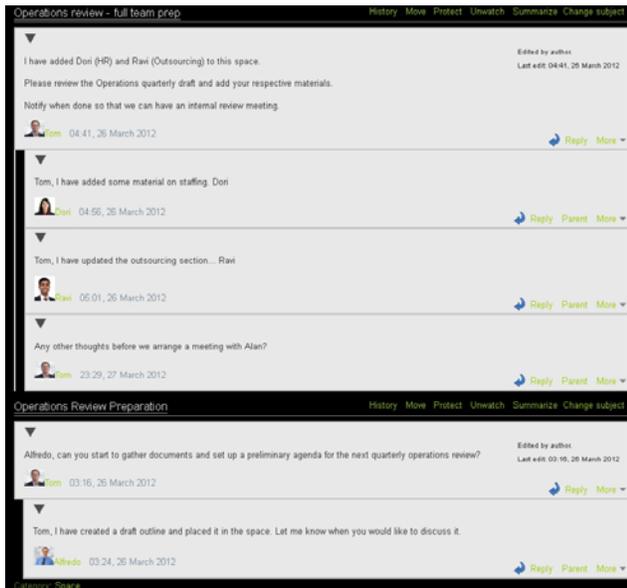


Figure 5 A discussion forum in the space shown in Figure 4

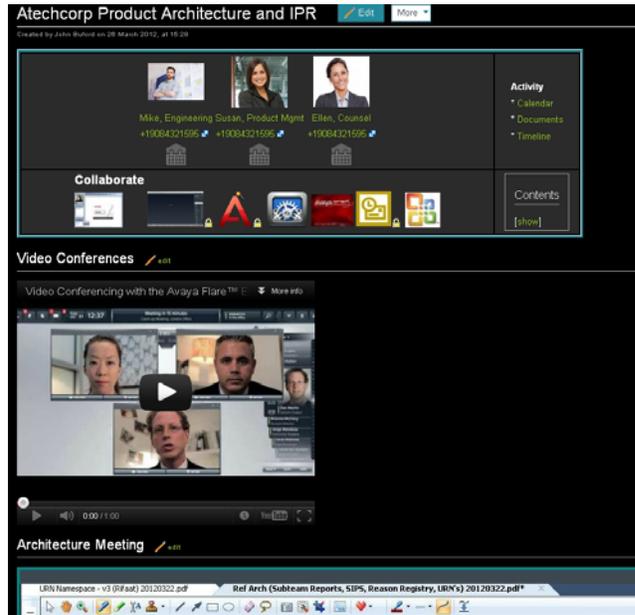


Figure 6 A collaboration space with a recording of a video conference meeting

### B. Research Team Collaboration

Four research projects lasting 10 weeks were run in parallel using ConnectedSpaces. Each team was between 2 and 4 members. Figure 7, Figure 8, and Figure 9 show some sample screen shots for one space “iPad Client for ConnectedSpaces”.

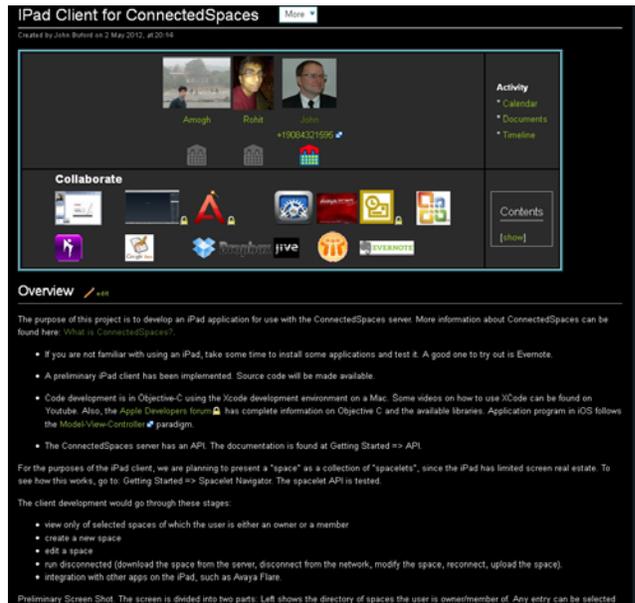


Figure 7 Space used by project team to develop an iPad client

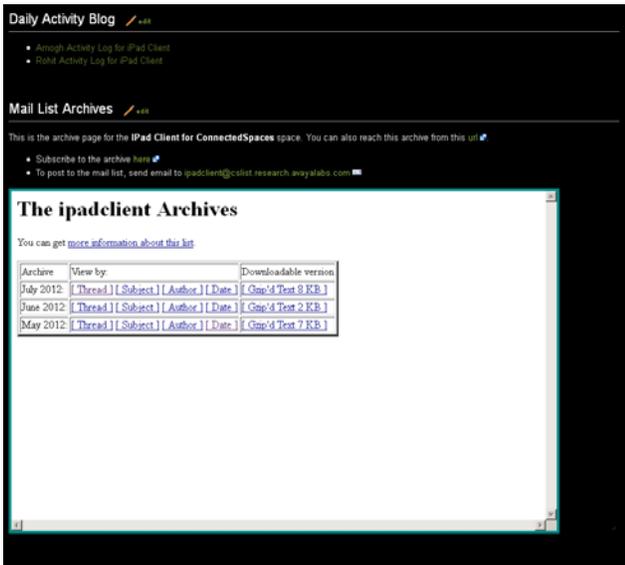


Figure 8 Mail archive used by the space

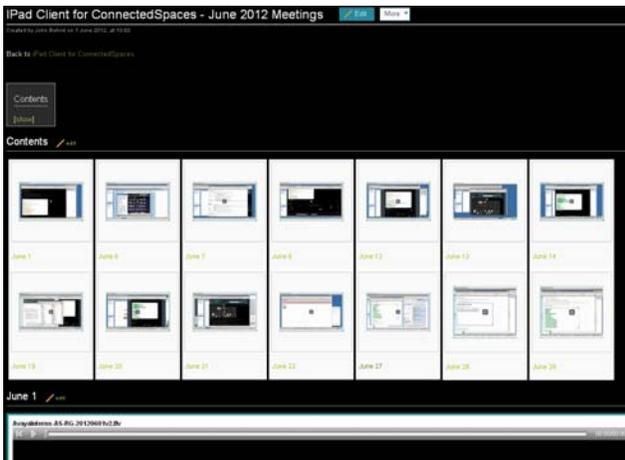


Figure 9 Gallery of recorded meetings

An existing project team of 50 members has run its collaboration using sharepoint, a wiki, and the Avaya web conferencing application. ConnectedSpaces is used to integrate these existing collaboration tools and to capture online meetings.

### C. Semantic Query Examples

Semantic models are entered in to the system using the built-in Semantic Wiki data explorer. Queries are embedded in pages and executed when the page is displayed, or can be invoked via the server API with response sent to the client in XML. For example, Figure 10 shows a timeline output of a semantic query for Meeting objects.

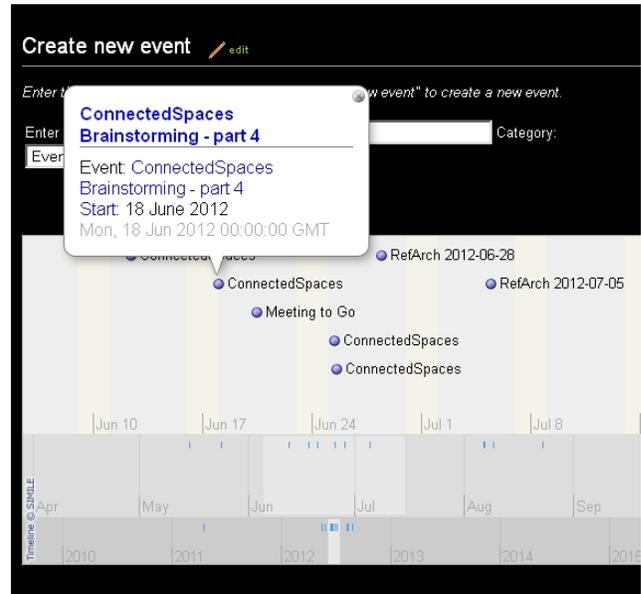


Figure 10 Query-driven meeting timeline

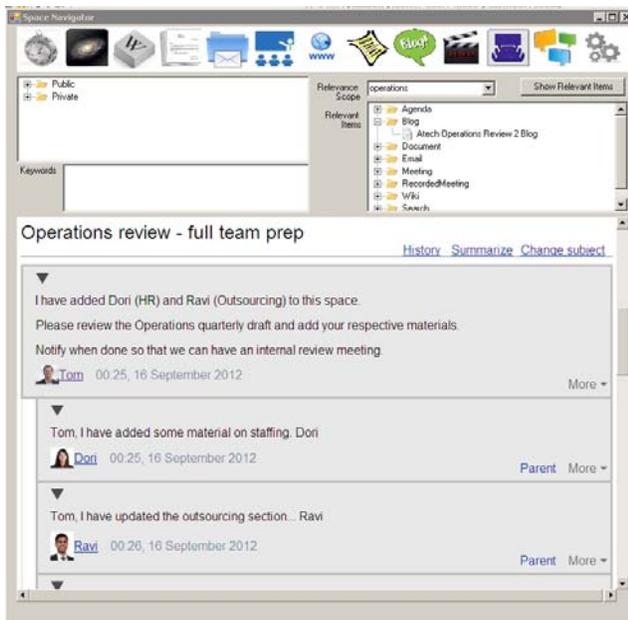
As another example, Figure 11 shows a meeting room query displayed in page as a table. Whenever the page is accessed (or a client invokes the api), the query will be executed and return the current results.

#	Moderator	Size	Creation date	Last usage	Number of meetings held
AA07 Meeting Room	John Buford	100	25 June 2012 22:47:58		0
MR123	John Buford	100	29 July 2012 03:33:56		
MR12345	John Buford	99	29 July 2012 03:34:53		
MRTest1213	John Buford	123	1 August 2012 19:42:13		0
Meeting Room 123	John Buford	57	29 July 2012 03:45:05		
Meeting Room ABC	John Buford	100	29 July 2012 14:41:07		0
RefArch Meeting Room	Obunson	100	26 June 2012 23:34:36	21 June 2012 11:00:00	4
Venkye Meeting Room	Venkye	100	3 August 2012 17:02:27		0

Figure 11 Meeting room query result



Figure 12 Space Navigator



**Figure 13 Displaying relevant objects**

The Space Navigator (Figure 12, Figure 13) performs queries for “Relevant Objects” by the categories shown in Figure 2. Each object may have one or more keyword properties. These properties can be generated automatically for text or manually for other media. Any object which has matching keywords is then added to the set of relevant objects displayed in the space navigator. For example, Figure 13 shows relevant objects for the “operations” keyword. This application uses both the ASK query language and full-text search engine to identify relevant objects.

### VIII. CONCLUSION

The variety and complexity of tools and collaboration processes make collaboration more difficult than necessary. The exploitation of semantic structure in a collaboration environment is one approach we are pursuing to automate collaboration tasks and elevate the level of interaction for the user. The challenges include: a) specifying a flexible and general purpose model, b) populating the model with content without placing undue burden on the user, c) making the inference processing computationally tractable, d) finding those areas where there is significant value added.

The new results presented here include:

- A semantic model of entities and their properties that are important in enterprise collaboration
- A detailed discussion of categories of queries against such a model, with a number of example using the semantic wiki query language
- A discussion of the system implementation and experimental work

### REFERENCES

- [1] J. Buford, K. Dhara, V. Krishnaswamy, X. Wu, M. Kolberg. Work in Progress: A Communications-Enabled Collaboration Platform. IPTComm 2010. July 2011.
- [2] J. Buford, K. Mahajan, V. Krishnaswamy. Federated Enterprise and Cloud-based Collaboration Services. IEEE IMSAA 2011. Dec. 2011.
- [3] J. Rama, J. Bishop. Survey and Comparison of CSCW Groupware Applications. Proceedings of SAICSIT 2006
- [4] Semantic Media Wiki. semantic-mediawiki.org
- [5] SemDesk 2006 Proceedings sunsite.informatik.rwth-aachen.de/Publications/CEUR-WS/Vol-202
- [6] SemDesk 2005 Proceedings
- [7] S. Auer, S. Tramp, J. Lehmann, and T. Riechert: OntoWiki: A Tool for Social, Semantic Collaboration In: Proceedings of the Workshop on Social and Collaborative Construction of Structured Knowledge CKC 2007 at the 16th International World Wide Web Conference WWW2007 Banff, Canada, May 8, 2007
- [8] M Völkel, M Krötzsch, D Vrandeic . Semantic MediaWiki, Proc. of the 15th international conference on World Wide Web, p. 585, 2006
- [9] SMW+. www.smwplus.net
- [10] W3C. SPARQL Query Language for RDF. 2008. Retrieved from <http://www.w3.org/TR/rdf-sparql-query/>
- [11] CoAktinG. (Collaborative Advanced Knowledge Technologies in the Grid. www.aktors.org/coacting
- [12] Open Semantic Collaboratin Architecture Foundation. www.oscaf.org
- [13] NEPOMUK. Nepomuk.kde.org
- [14] DAML Agenda. [www.daml.org/2001/agenda](http://www.daml.org/2001/agenda)
- [15] Semantic Wiki. Semantic search. Retrieved from: [http://semantic-mediawiki.org/wiki/Help:Semantic\\_search](http://semantic-mediawiki.org/wiki/Help:Semantic_search)
- [16] J. Bao, L. Ding, J. Hendler. Knowledge Representation and Query in Semantic MediaWiki: A Formal Study. RPI Technical Report. [http://www.cs.rpi.edu/~baojie/pub/2009-06-02\\_iswc-bao\\_tr.pdf](http://www.cs.rpi.edu/~baojie/pub/2009-06-02_iswc-bao_tr.pdf)