Managing Knowledge for Organization-Wide Ad Hoc Committees

Alok Sharma  Les Miller  Sree Nilakanta
Department of Computer Science  Department of Computer Science  Department of Supply Chain and Information Systems
Iowa State University  Iowa State University  Iowa State University
Ames, IA 50011  Ames, IA 50011  Ames, IA 50011
lmiller@cs.iastate.edu  nilakant@mail.iastate.edu

Abstract

Organization and industry-wide ad hoc committees are often formed to develop solutions to crises that are ill-defined. Knowledge available to such committees is at best scattered and possibly nonexistent within the organization. The present work describes an environment created to provide the committee members with the tools to collect, organize, and utilize the knowledge for them to complete their task. The environment makes use of topic maps as the knowledge structure. A distributed system has been implemented to support the proposed environment and demonstrate feasibility.

1. Introduction

Many rapid decision making situations arise when natural calamities, industrial accidents, or other unexpected incidents occur. In order to tackle the situation, teams of subject matter experts (SME) are often assembled on short notice from a wide range of disciplines, organizations, and locations. While such assemblage of teams are common in these high visibility problems, similar teams may be corralled to focus on software development, ad hoc committees, research teams, new product development, etc. The basic notion that binds these examples is that exploring and exploiting the domain specific knowledge of SMEs and facilitating the sharing of knowledge among the SMEs leads to success. Erden et al [6] notes that group tacit knowledge with high quality is a resource that organizations can rely on when confronted with unexpected and unfamiliar situations which require intuition and spontaneous collective action. Though information technology advances have been widely adopted, there is still a significant gap in knowledge related to organizing, and using widely disbursed tacit and codified knowledge, especially in crisis management and humanitarian assistance and disaster relief [5,35]. Crisis and disaster management could gain from the use of topic map implementations of transactive memory, organizational memory, and tacit memory.

Transactive memory principles can be an appropriate vehicle for invoking group tacit knowledge. Transactive memory is a shared system for encoding, storing, and retrieving information [32,33,34] that can connect different decision makers such that each has an understanding of who knows what (a directory) and constantly updates this directory through communication among the individuals. Managing the web of knowledge space requires tools that not only allow dynamic structuring of the SME’s domain knowledge but also create linkages among them. The distributed decision making task would require a scheme to organize, navigate, search, and repurpose an incessantly expanding web of knowledge and information.

2. Background

As stated earlier, group tacit knowledge is useful in handling crisis, scientific discoveries, and in routine team work. Using Mitroff’s five stage crisis framework and Earl’s taxonomy on knowledge management (KM) strategies Wang [22] presented a knowledge centric adaptation showing that different stages of a crisis require appropriate knowledge management strategies. In another study, Chua, Kaynak, and Foo [5] compared published reports of two major catastrophes (hurricanes Katrina and Rita) in the US using a KM framework based on KM processes of knowledge creation, knowledge transfer, and knowledge reuse. As these two studies show, distributed cognition among the participants is facilitated through the use of dynamic organizational memories. Collaboration in encoding knowledge,
2.1. Topic maps

Topic maps provide a subject based classification of resources where each resource represents a real world object or tool. In topic maps, three constructs are provided for describing the subjects represented by the topics: names, occurrences, and associations. These describe the names, properties, and relationships of subjects, respectively. A name may be assigned to more than one topic and a topic may have more than one name. Also by defining scope and types a name can become rich and complex. An occurrence links to one or more real knowledge sources. An occurrence, however, is not part of the topic map. Associations describe relationships among topics and are independent of the real knowledge objects. Associations add value through the relationships. The ISO standard ISO/IEC 13250 provides a standardized notation for interchangeably representing information about the structure of information resources used to define topics, and the relationships between topics. A set of one or more interrelated documents that employs the notation defined by this International Standard is called a topic map. The standard further states that a topic map defines a multidimensional topic space — a space in which the locations are topics, and in which the distances between topics are measurable in terms of the number of intervening topics which must be visited in order to get from one topic to another, and the kinds of relationships that define the path from one topic to another, if any, through the intervening topics, if any.

The Topic Map architecture is designed to help merging topic maps without needing the merged topic maps to be copied or modified. This feature makes it suitable for use in organizational memory systems. As Steiner et al. [19] state, constructing large, complex organizational memories takes place over extended periods of time, growing incrementally both in size and structure. The ISO standard describes how topic maps may be used in situations such as that may be found in OM systems. For example, a topic map can be employed “to structure unstructured information objects, or to help create topic-oriented user interfaces that provide the effect of merging unstructured information bases with structured ones [15]. The overlay mechanism of topic maps can be considered as an external markup mechanism, in that an arbitrary structure is imposed on the information without changing its original form.” Knowledge management applications can also benefit from topic maps, especially when combined with ontologies [20]. Korthaus et al. [11] describe a topic grid architecture that enables a client application to view distributed topic maps of organizational knowledge as a combined virtual topic map. Each distributed topic map represents a different view of the underlying information.

Smolnik and Erdman [17] state that topic maps provide strong paradigms and concepts for the semantic structuring of link networks and therefore, they are a considerable solution for organizing and navigating large and, continuously growing organizational memories. More recently, Smolnik [16] discussed topic maps in comparison to other semantic and traditional search mechanisms. Miller et al. [12] developed a model of a topic map based OM knowledge management system to represent and use organizational memory artifacts. Miller et al. [13] looked at potential models for process and group topic maps. Newman et al. [14] make use of topic maps to visualize document collections.

In the following sections we describe a model of topic map environment that addresses the collaborative aspect of crisis management and committee support.

3. Architecture

The current model is built on the architecture presented in Miller et al. [12] and the topic map model described in Miller et al. [13].

The goal of any organization is to make the best use of the business knowledge that it has accumulated over the years. To do this, it is necessary to determine the sources of business knowledge and provide members of the organization access to all sources in a timely matter. Here we
focus on the latter issue of providing access through retrieval, analysis, merging, modifying, and learning tools. As with our original model, we also assume that business knowledge sources are employee knowledge, transaction data, data warehouse data, memory artifacts (internal and external), and the knowledge inferred from these sources. Note that in crisis management, teams may be assembled from within and outside. Also, sources of business knowledge span organizational boundaries.

3.1. System architecture

An organization’s memory can be seen as consisting of a space that includes knowledge of the expertise of current employees, artifacts, and the tools required to locate and/or interpret available information. The memory space is distributed across the organization and beyond.

Any organizational memory must be able to deal with the distributed nature of the memory space and be able to fully integrate the memory’s information with the necessary tools. To deal with this problem, we introduce a distributed organizational knowledge management environment.

A block diagram of the architecture is shown in Figure 3.1. Artifacts and information about the expertise of current employees can either be stored in the data sources or in the Internal Object-Oriented Data Store and Toolkit. Artifacts in the data sources can be stored in any format to take advantage of existing software or compression techniques. To simplify communication between components and provide for standardization within the model, object views [23] are used to define data types. Data from the data sources are converted into object view instances using local interface views [23].

![Figure 3.1. Block diagram of the distributed organizational memory](image)

Topic maps are presented to the user via either the Web Interface or the Java Client Web Service depending on the access mode chosen by the user. Regardless of the user’s access mode the visual structure and use of the topic map interface remains unchanged. The primary difference between the modes is in the details of their respective implementations.

3.2 Topic maps

A topic map provides users with visual access to both search and analysis. This interesting blend of semantic search and analysis motivates its use in the proposed model.

We define a topic map to be the directed acyclic graph \( T = (N,E) \), where \( N \) is the set of topic map terms, data types, or search/analysis tools. The directed edges in \( E \) are of the type \((n_1,n_2)\), where \( n_1 \) is a topic map term or a data type and \( n_2 \) is a topic map term, a data type, or a search/analysis tool. An implicit node, called the root, points to the topic map terms, data types and/or tools that make up the first level of the topic map. Nodes with out degree zero are said to be leaves of the topic map. Leaves point to the information in the organization’s memory and carry any search terms and/or data types accumulated on the path from the root of the topic map to the leaves. Note that leaves that are either search terms or data types make use of an implicit search tool or a list of resources. Also each non-leaf level of the topic map has a search tool that can be used to initiate a search based on the topic map terms that have been traversed to get to the current topic map level.

A visual topic map over a topic map \( T = (N,E) \) is a directed acyclic graph \( V = (S,A) \), where \( S \) is a set of screens and \( A \) is a set of directed edges that connect the screens via a node context. A screen \( s \in S \) is defined as 2-dimensional representation of a set of nodes \( S_N \) such that each node \( m \) in \( S_N \) is in \( N \) and there is a node in \( n \in N \) such that \((n,m) \in E \) for every \( m \) in \( S_N \). An edge exists in \( A \) whenever a non-leaf topic node on one screen points (in \( T \)) to the nodes on the second screen. The root node described in the topic map definition above is replaced by the screen representing the nodes in \( T \) that are adjacent to the implicit root node. Each edge in \( A \) has a node context. This node context implies selecting a non-leaf node on a screen results in following the link to the screen associated with the node. Figure 3.2 illustrates a screen from the current prototype. A dynamic visual topic map \( D = (V,O) \) is a visual topic map \( V \) and a set of operations \( O \) that allow the users to modify the content of the visual topic map.
3.3. Committee topic maps

A Committee Topic Map is defined by the 4-tuple \(< D, C, O, R >\), where \(D\) is a dynamic visual topic map, \(C\) is the committee of users, and \(R\) is the set of roles relevant to the committee’s mission.

The visual topic map in a Committee Topic Map has the additional property of being dynamic. Beyond being dynamic, the basic definition of a visual topic map remains relatively unchanged. One additional difference is that the notion of a path label is weakened in this version of the visual topic map. The motivation for this difference is that it gives the model the opportunity of making use of the users’ understanding of their areas of work and skills. It gives them the ability to place nodes on screens that they feel will provide the best information for other members of the committee. To this point, we have assumed that everyone in the committee can use all of the operations in \(O\), but our prototype could easily be extended to restrict who can make the changes and limit the kinds of operations that individual users can perform.

The initial visual topic map would be up to the committee. It could simply be an empty initial screen, a topic map that exists based on prior efforts of the committee, or a topic map that is created by members of the committee using the operations available in the operation set. The choice would be up to the committee and/or the organization at large.

In the remainder of this section, we describe the components of the model.

3.3.2.1. Committee. At this point the committee is simply represented by the ids of the members of the committee. It could easily be extended to include more information on the committee members, e.g., their roles within the organization.

3.3.2.2. Operations. The operations (\(O\)) defined for the dynamic topic map in the Committee Topic Map provide the members of the committee with the ability to build a topic map that contains knowledge relevant to the task the committee has been charged with. The basic operations available to members of the committee are:

AddANode – The operation allows committee members to copy nodes from existing visual topic maps to the Committee Topic Map. The operation copies the node and its descendents to the Committee Topic Map.

CreateANode – This operation is used to add either newly created or synthesized knowledge to the topic map. The new knowledge is introduced into the topic map by creating a new topic node and the resources associated with it or by creating access to a new or existing tool via a new tool node.

GeneralizeANode – This operation is designed to
allow users to partition the knowledge space indicated by the semantics of a node. It is used when the user sees the need for more general information on a topic. The operation creates a new node that is connected to a set of more general resources which makes it more valuable to all of the members of the committee.

SpecifyANode(Node, Role) – Whenever a user copies a node from a process-based topic map, the node is associated with knowledge that is focused at least to some extent on the underlying process. To provide a broader interpretation, the operation takes a role and creates a new node that reflects the underlying concepts of the original node while incorporating the role. For example, the system can use the existing knowledge indicated by the original node combined with the definition of the new node to search the knowledge space. The results of the search would reflect knowledge more skewed to the new role. The motivation for this operation is to allow users to obtain access to information that is related to the node topic, but is focused on the role that the committee member plays on the committee.

ModifyANode – This operation allows the members of the committee to modify properties of a node. In our current model, this could result in changes in the value of the topic indicated on the screen, changes in the path label associated with the node, or modify the resources associated with the node.

MoveANode – The motivation is to allow users to move nodes from one screen to another to enhance the use of the node by the committee.

DuplicateANode – This operation allows a node to be duplicated from one screen to another within the Committee Topic Map. Note that the new node is only a shallow copy of the original node and takes with it any screen links and path labels.

DeleteANode – As nodes become irrelevant or dated, this operation allows users to delete extraneous nodes.

3.3.2.3. Roles. The roles in R are the roles of the members of the committee that are relevant to the task that the committee is charged with. The set restricts the way some of the operations (e.g., SpecifyANode) work and also adds information to potential searches required for operations on individual nodes.

4. Implementation

Our prototype runs Java classes on the client side and communicates with the server using the Apache Axis 2 platform. Apache AXIS 2 is an implementation of SOAP [18] and REST [2].

4.1 Server side

The Web Service Archive handles and services the Client request. The Server Java Code which provides the interface to the Internal Object Oriented Data Store (Hibernate Database [7]) is used as the input to the Axis2 Service Archiver to generate the Web Service Archive. Once this has been completed, the server is started to provide the services offered by the server.

The server also provides access to the Information Retrieval tool. The Information Retrieval tool is also AXIS2 based. The Information Retrieval tool is used to search the Organizational Repository and fetch documents, presentations, emails etc corresponding to the keywords provided by the client. The resources fetched from the Information Retrieval tool are displayed to the client in the form of JavaScript enabled web page. These web pages are created dynamically corresponding to the current request. The content of the various documents that are fetched can be contracted and expanded according to the user requirement. This information retrieval tool [8] was developed by our research group and was integrated and modified by me to meet our requirements.

4.2 Client side

The RPCServiceClient object provides client access to the service. The result is that the user operates as if he/she is operating locally, but the software has access to the content available on the server and the user is able to interact directly with the content. The screen shots shown in the remainder of this section were generated on the client.
4.3 Topic maps

All of the visual topic maps are stored on the server in the Internal Object-Oriented Store as a set of object instances. Node instances are created using the Node class. The Tool class and the Topic class are subclasses of the Node class. Screens are created as object instances of the Screen class. The Screen object instances point to the node objects that define the nodes on the screen.

Two types of visual topic maps are supported by the current prototype, namely, static visual topic maps and Committee Topic Maps. Committee Topic Maps by comparison are dynamic and are built by the committee members. All operations on the Committee Topic Map operate on the object instances supplied by the server.

4.4 User Interface and topic map operations

To provide more detail on the user interface, the implementation and the operations supported by the current prototype, the most interesting operations are presented through snapshots.

4.4.1 CreateANode. The CreateANode operation allows a committee member to introduce either a tool or a topic to a screen of the Committee Topic Map. If the user adds a Tool, the system returns the list of tools that are currently active and supported by the server (Figure 4.1). The current prototype allows the user to select a tool from the list of the tools currently supported by the system. A web tool has been created to allow the user to incorporate new tools.

To create a new topic is a similar operation. The user chooses the name of the new topic and can choose between loading the resource pointers from existing resources (Figure 4.2), uploading new resources from a file, or using the IRTool (a keyword-based document retrieval tool based on PROMIS [8] with a set of keywords to search the document files in the Internal Object-Oriented Store or data sources for relevant resource documents.

4.4.2 ModifyANode. The ModifyANode operation provides three ways to modify an existing node in the Committee Topic Map, namely, change the node name, change the path label, or modify the resources associated with the node (Figure 4.3). The first two are handled by providing the user with a textbox so that the member can modify or replace the existing name or path label. The third option, modify resources, is more interesting. If the node chosen represents a tool, the user has the ability to change the tool to one of the tools currently supported by the system. Again any new tool can be added to the system by using the web tool developed to manage the resources associated with topic maps. If the user chooses to modify an existing topic, the operation uses the same system components that were used to define the CreateANode operation. The node modified will be converted to a leaf node with both
the newly added resources and the resources already associated with the node available to users of the topic map. If the existing node is a leaf node the newly added resources are simply unioned with the existing resources to form the new set of resources that are associated with the modified node. If the node was originally not a leaf node, the system searches the sub-topic map that has the node as its root for resources. The resource set that is obtained from the search is unioned with the new set of resources to form the modified node as a leaf node.
4.4.3 GeneralizeANode

The GeneralizeANode operation provides the user with the ability to generalize a topic node by supplying the system with a set of additional keywords (Figure 4.4). The operation uses the path label of the existing node combined with the new keywords entered by the user as the parameters to the IRTool. The IRTool then searches the documents stored on either the Internal Object-Oriented Store or on the data sources attached to the server (Figure 3.1). The matching resources (documents) are then unioned with the resources found by searching the sub-topic map that has the original node as its root. The set of resources are then associated with a new node on the same screen as the original node with the new node’s name appearing with the term “generalized” added to the original topic name. Figure 4.5 shows the new screen with the new topic node labeled as “generalized transcripts”.

4.4.4 SpecifyANode. The SpecifyANode operation acts in a similar manner to the GeneralizeANode operation in that it creates a new topic node on the
same screen as the original node. The new node has the term “specified” added to the name of the original node. The difference is that instead of the user giving a set of keywords designed to generalize the type of resources available, the user provides the system with a role. The operation then uses the metadata associated with the role and the topic node’s path label information to target the resources associated with the newly created leaf node to a specific role. For example, a topic node that targets resources specific to software developers could be used as the basis of a new node that specifies that the resources should be targeted at managers. The operation searches the relevant documents to provide a set of resources that match the primary topic, but retrieves resources that focus on managers.

4.4.5 Remaining operations. While the remaining operations seem more straight-forward, it should be noted that all of the operations described in Section 3.3 provided some interesting challenges. The operation AddANode is straight-forward in what it does, but relies on the system supporting multiple visual topic maps that can be considered active at the same time by the web service. Also the service has to be able to realize that at least one of the visual topic maps is a Committee Topic Map and other visual topic maps are likely to be static and cannot be edited.

Even the simple operation DeleteANode has important consequences. Since the server maintains the definitions of the screens and nodes via the classes Screen and Node, the existence of a topic or tool node on a screen in a visual topic map indicates that there is an object of type Node stored on the server. The delete operation has to determine whether it has to delete the node on the server or just modify the instance of the Screen object that the user is working with.

5. Conclusions and future directions

We presented an implementation of a group topic map that enables ad hoc team members collaborate to resolve ill-defined tasks. The system addresses the need to retrieve role specific information and create and modify cross-functional knowledge. Our system provided several unique operations to make the group topic map dynamic. We plan to evaluate the prototype using data compiled from primarily from secondary sources (e.g., news reports and other publicly available sources) of events or crises (e.g., the BP oil spill).

Though the implementation of the topic map addressed the needs of ad hoc committees, it is equally applicable to the larger crisis and disaster management scenarios. Topics of issues management, planning and prevention tasks, understanding the full extent of the crisis or disaster, and taking post event decisions can rely on the topic map implementation.

6. References


