Concept Maps: Theory, Methodology, Technology
Proc. of the Second Int. Conference on Concept Mapping
San José, Costa Rica, 2006

USING FORMALIZED CONCEPT MAPS TO MODEL ROLE-BASED WORKFLOWS

Alexander Feinman, Jonathan Oexner, and Subrata K. Das
Charles River Analytics, USA

Abstract. Concept maps have traditionally been used as a method for eliciting, representing, and storing information about domains of knowledge. In this paper, we explore an additional use of Concept Maps, to model patterns of activity in a fashion that is useful for both human understanding and computer-based deduction. Via formalization of Concept Maps the resulting information structures can be parsed automatically and used to support collaboration. The goal is to create a formalized representation of information that retains the crucial aspect of human understandability, while allowing a wide range of additional uses via adaptive system support. In this paper, we propose a process for formalizing concept maps describing a specific workflow by identifying roles, tasks, and information within the domain. This approach is used to create and store information models required by TREK, an ongoing project to support spatially-dispersed, dynamically-formed teams of workers.

1 Introduction

Concept mapping is a powerful tool for eliciting and communicating knowledge about a complex domain (Novak, 1998; Novak & Gowin, 1984). By graphically representing concepts and the relations between them in a standard, easy-to-understand format, Concept maps enable novices and experts alike to understand and share knowledge bases. Their flexibility and intuitive layout simplifies their creation and promotes adoption by experts, rather than requiring an elicitation expert to achieve usable results. However, the flexibility of concept maps restricts their use in intelligent systems. The freedom of language allowed, and the flexibility of concept linking, turns the problem of parsing concept maps into one of natural language processing devoid of structural grammar. But to deny users this flexibility removes many of the features of concept maps that make them so useful: their intuitive creation procedure, and their ability to represent any domain of knowledge. To allow concept maps to be used by both computer and human, a balance needs to be struck between ease of creation and ease of parsing. In this paper, we will detail a method for creating semi-structured concept maps that retain readability and flexibility while providing some clues for automatic parsers.

We are engaged in the construction of a Toolkit for Role-aware Exchange of Knowledge (TREK), a project designed to create a toolkit for building groupware that supports team formation and collaboration. Based on a knowledge-intensive approach to modeling coordination, TREK systems utilize models of a task, the roles within a task, and the information requirements for those roles to direct collaboration and deliver task-appropriate information. Currently, concept maps are used to elicit this complex body of knowledge from experts, which are then manually converted into machine-readable models. As the project moves forward, TREK systems will use formalized concept maps to store and reason about these models. The proposed method for formalizing concept maps is a key element to this strategy, allowing end users to encode information in a formalized yet intuitive fashion usable by both computer and human. Recognizing the difficulty of achieving these goals in the general case, the TREK system currently restricts concept mapping to a narrow range of topics. Specifically, we model one work process at a time, eliciting the roles and information within that process. By focusing knowledge elicitation along these lines, we can improve the probability that the formalized concept maps created as a part of the knowledge elicitation process contain the information required for proper functioning of the TREK system in a format understandable to the computer but expressible by a non-expert user.

The method we propose for formalizing standard concept maps produces a set of related concept maps, connected via common concepts, which contain a strong framework for the information needed within TREK systems. A process map, centered around an ordered list of the actions performed by people fulfilling roles within the system, reveals underlying workflow within the interaction. Another view of the interaction reveals the propagation of information from representation to representation, guiding elicitation of further information about these processes. Finally, a role-relation diagram can be constructed, showing the interrelations and dependencies between roles in the system. This allows a designer to determine how and via what media a pair of roles should be interacting, and whether support for that interaction is a priority.

1.1 Formalized Concept Maps

Various formalizations for concept mapping have been proposed. One notable effort, the Cooperative Ontology Environment, has investigated the incorporation of a complete ontological language into a concept mapping tool (Hayes et al., 2005a; Hayes et al., 2005b). Formalization can be viewed as migrating interaction further to the right on the scale shown in Figure 1. At the far left, we have completely informal graphs of knowledge: unstructured collections of concepts, with or without relations joining them. Concept mapping introduces a certain level of structure: relations should link a pair of concepts into a logically consistent preposition; concepts should not appear multiple times within a map; and so forth. These formalisms limit the flexibility of the representation, but without great impact on the power of its representation.

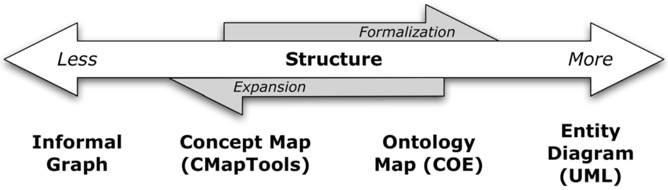


Figure 1. Sliding scale of formalism in knowledge representation.

Adding yet more structure increases the possibility of automatic parsing of the map. Ontology maps such as those proposed by Hayes et al. utilize auxiliary data, structural and stylistic information, and a limited vocabulary to improve automatic parsing of maps, while allowing flexibility in the content of nodes and in the pattern of relations. This increased utility comes at the cost of increased effort to construct a well-formed map, and a decrease in the intuitiveness of the representation. Further along this axis we encounter completely formalized representation such as the Unified Modeling Language (UML), created to express complex processes in an unambiguous, communicable format; it has been used to model complex processes from computer programs to business structures (Object Management Group, 2006; Kogut et al., 2002; Eriksson & Penker, 2000). This fully-specified language is appropriate in situations such as programming where ambiguity can be catastrophic; however, its lack of flexibility and complex syntax make it inappropriate for use by domain experts who are unable or unwilling to express ideas in the formal fashion required to satisfy UML syntax.

The solution we propose lies somewhere in the middle of this sliding scale. By walking the user through the steps to formalize an existing process-centric concept map, the Gulf of Expression (Norman, 1988) is reduced; the user is not forced to learn an esoteric syntax such as in UML or even in the ontological representations of COE. While the resulting map is neither provably correct nor complete, it provides enough parsable information for a computer to bootstrap the process of domain model creation.

2 Supporting Collaboration

2.1 Role-Based Collaboration

To understand the basic ideas behind our formalized concept maps, we must first examine the domain that we are addressing: role-based collaboration. The TREK project, funded by the Air Force Small Business Innovative Research program, seeks to create a toolkit for building role-based collaborative software. By explicitly modeling information flow between roles within a task, a TREK-based system will aid a commander in gathering a skilled team in real-time, and will distribute filtered, customized information to each user based on their current set of roles.

In the view of activity used for TREK, a *task* is a series of *actions* executed by *roles* on *objects* (or on other roles) to achieve or maintain a set of *goal states*. This view of activity is built from concepts within workflow analysis (Kammer et al., 2000; Bardram, 1997) and activity theory (Kaptelinen & Nardi, 1997; Kuutti, 1996; Leont'ev, 1978). An overview of the interrelation between concepts within TREK is shown in Figure 2. Users are modeled based on their current and expected future *availability*, set of relevant *skills*, and information/action *clearance* levels. Roles are modeled based on their *responsibility* to achieve or maintain a certain goal state, their *relation* to other roles, and the *capabilities* for action granted by that role. Finally, tasks are modeled as a set of goal states, and the set of roles that are required for achieving or maintaining those.

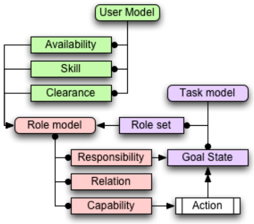


Figure 2. Interrelation of key concepts within TREK.

2.2 *Using the models*

Within the TREK system, these models are used to suggest membership for on-the-fly team construction, to maintain peripheral awareness of other team members, and for filtering and delivery of task information. Specifically designed for military operations, where well-defined skill sets and explicit security measures are present, the TREK model aims to filter information according to suitability and clearance. This two-sided process is shown in Figure 3. On the left, the process of selecting team members for a specific task is outlined: the set of all users is filtered according to status, skills, and requirements. On the right side, the process of filtering and delivering information in a secure, appropriate, and tailored representation is outlined; information sources, tagged with meta-data to indicate content, structure, and sensitivity, are filtered according to user authorization, role-based knowledge requirements, and the potential actions a role is responsible for taking.

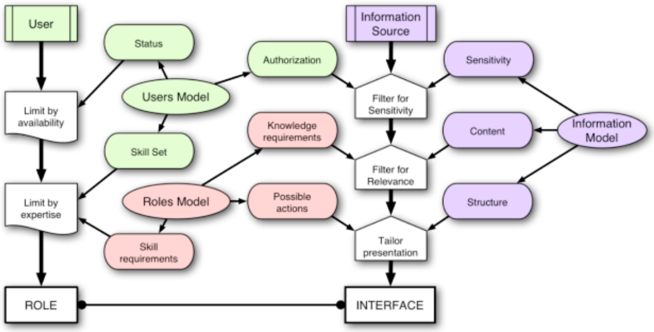
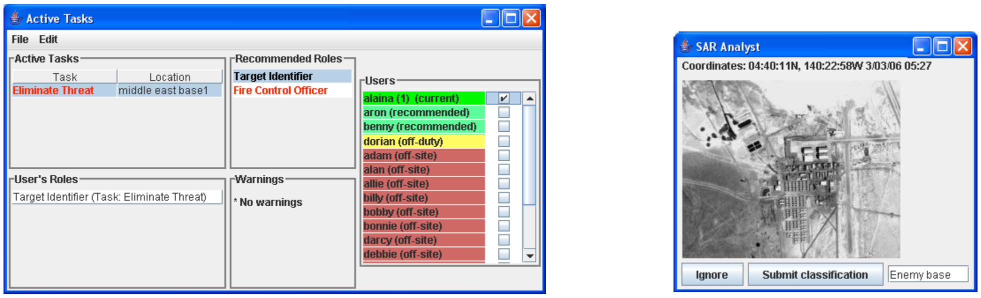


Figure 3. Models are used within TREK to filter availability of personnel, and guide information delivery

This process has been implemented in a preliminary Java-based simulation. Specifically crafted for the Time-Sensitive Targeting (TST) domain with the Air Operations Center (AOC) of the Air Force, this system allows a commander to gather a team of physically-disparate team members in real-time, selected according to the parameters outlined above, and delivers information based on task requirements. A screenshot of the basic TREK team-building interface appears in Figure 4a. Here, the team commander is presented with a list of active tasks (upper left); tasks are highlighted if roles within them remain unassigned. The roles for the highlighted task are presented beside the list of tasks; in this case, “Fire Control Officer” remains unfulfilled. At the bottom, the team leader is presented with an ordered list of users, sorted by their suitability for the role in question; further details on each user can be brought up by selecting a user, which will list any warnings regarding assigning that user (e.g., “off-duty”, “off-site”, “not skilled”), and the user’s current list of roles. The leader can then select appropriate users to assign to roles; the users are notified of their role assignments, and can accept or decline in real-time. When a team is assembled, individual team members receive tailored, interconnected, task-oriented interfaces.



Active Tasks

Task	Location
Eliminate Threat	middle east base1

Recommended Roles

Target Identifier
Fire Control Officer

Users

alaina (1) (current) ☑
aron (recommended) ☐
benny (recommended) ☐
dorian (off-duty) ☐
adam (off-site) ☐
alan (off-site) ☐
allie (off-site) ☐
billy (off-site) ☐
bobby (off-site) ☐
bonnie (off-site) ☐
darcy (off-site) ☐
debbie (off-site) ☐

User's Roles

Target Identifier (Task: Eliminate Threat)

Warnings

* No warnings

Coordinates: 04:40:11N, 140:22:58W 3/03/06 05:27

Enemy base

Figure 4a and 4b. Team-building interface, and a role-centric interface, for a TREK system.

This second portion of the TREK interface is shown in Figure 4b. It presents a user with role-centric information; an alternate console (not shown) allows peripheral awareness of other team members. In this case, an analyst has been asked to visually confirm the identity of a potentially hostile target, and send that information back to decision-makers (in this case the Fire Control Officer). A tailored interface for this task is presented; the result of the task is automatically passed to the role(s) that need it. Further details on the TREK system currently under development are available in (Das & Feinman, 2006).

3 Modeling Workflows using Concept Maps

The above system is strongly data-driven, requiring accurate and complete models of task, role, knowledge, and users. However, populating these models can be problematic, which is where the flexibility and intuitive nature of concept mapping comes into play. By utilizing the intuitive nature of concept mapping, it is possible to elicit information of superior quality from domain experts than interviews or other such techniques.

Building concept maps suitable for use as a TREK knowledge model requires identifying roles, actions, and objects within a concept map, and determining the nature of the relations between them. We propose an interactive, iterative process wherein the user is queried to disambiguate the nature of concepts within a concept map to transform it into a process-oriented map suitable for use within a TREK system.

A sample, non-formalized concept map is shown in Figure 5. This map shows the basic concepts and relations in a general domain: prescription drug fulfillment. A doctor diagnoses a patient, writes a quick note to his assistant, who generates an appropriate prescription to be handed to a pharmacist by the patient; the pharmacist prepares the drug and gives it to the patient. In this simple example many complexities have been trimmed out; however, the sample retains enough complexity to show the steps taken to convert this map to a formalized map of a workflow.

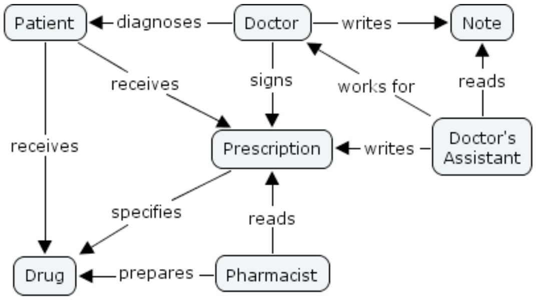
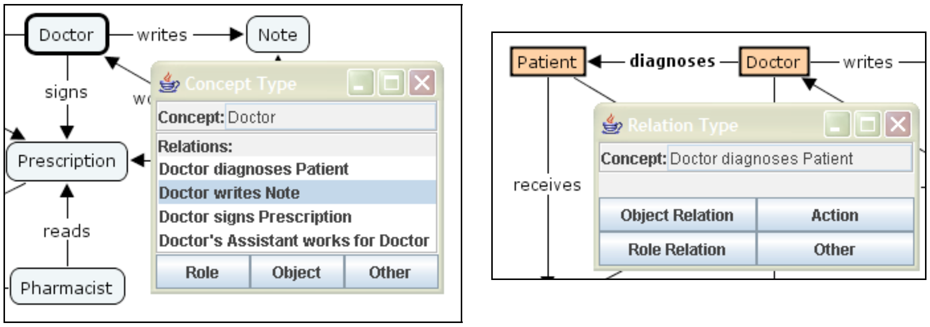


Figure 5. Concept map for a simple process: prescription drug fulfillment.

The first step in formalizing this concept map is to interactively query the user regarding the nature of each concept within the map. This is a straightforward classification process: each concept is identified as a Role, an Object, or Other. While some difficulties lie in complications such as Roles played by faceless entities like corporations, in general this distinction was well-understood by our test users. For each concept in the map, the user is prompted to enter a simple classification as shown in Figure 6a: Role, Object, or Other.



Concept: Doctor

Relations:
Doctor diagnoses Patient
Doctor writes Note
Doctor signs Prescription
Doctor's Assistant works for Doctor

Concept: Doctor diagnoses Patient

Figure 6a and 6b. The system prompts the user for the type of each concept and relation.

Given this level of information, the system can automatically propose tentative assignments of the types of relations between concepts, which are confirmed by the user via a similar mechanism as for concepts (Figure 6b). Links from a role to an object are generally an *action*; links from one object to another (such as the "Prescription specifies Drug" relation) are *object relations*, describing ontological connections between these objects; links from one role to another are *role relations*, describing pertinent interactions between roles. However, these are only defaults; in this map, for example, the relation between Patient and Doctor ("Doctor diagnoses Patient") is actually an action.

Figure 7 shows the final results of this classification. Styles are used to give users feedback as to which concepts have been identified, and what category they are assigned to. Here, concepts such as "Patient" and "Doctor" have been identified as roles, while "Note", "Prescription", and "Drug" are objects. Styles serve to maintain a consistent and understandable format, while giving consistent information suitable for parsing by the TREK system. At this point in the process, the TREK system has a list of actions, including the actor and object for each. These actions are presented to the user in list format, as predicates, for ordering. Currently, only linear ordering ("'Doctor signs prescription' is step five") is envisioned; allowing for partial orderings (e.g., "'Doctor signs prescription' must go before 'Pharmacist prepares prescription'") is a potentially useful expansion. Because the map is restricted to examining a single process, it is sufficient to prepend sequential numbers to the actions to order them in the eyes of the user; a separate representation of the step ordering is maintained by the computer.

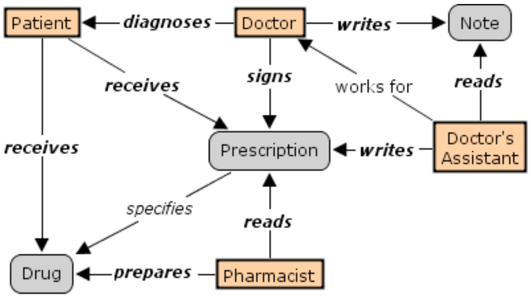


Figure 7. Concept map after relation types are specified by TREK and verified/altered by the user: "Patient", "Doctor", etc. are in the *Role* style, while "Prescription", "Drug" and "Note" have been marked with the *Object* style.

This ordering allows reformatting of the map into a form of limited Petri Net, useful for process flow analysis (van der Aalst, 1998; Petri, 1962). This requires a violation of some of the basic constraints of concept mapping; a potential visualization that breaks the "one node per concept" maxim appears in Figure 8. Here, auxiliary nodes appear for roles involved in non-contiguous steps, to avoid visually-confusing crosses or circuitous relationship routes; these auxiliary nodes are shown with dashed outlines to provide visual feedback. It is important to emphasize that this representation can be automatically generated from the above data. Although layout is a little tricky, the simple method used in this layout (to position roles on the left, actions on their right, and action targets on the far right, and to order actions vertically) is amenable to translation into algorithmic form.

This alternate representation greatly simplifies understanding the ordered nature of this work process. The coupling of the time-ordered actions with the ability to show cross-links between concepts (e.g., the 'specifies' links) make this representation superior to either a pure concept map or a table of relationship 'triples'. In a TREK system, this knowledge is used to understand what sorts of information are used by a specific role, and what communications link roles. These determine the nature of the resulting interface. For example, the Pharmacist role would be given an interface which presents Prescription information, and which allows execution of the Drug delivery action.

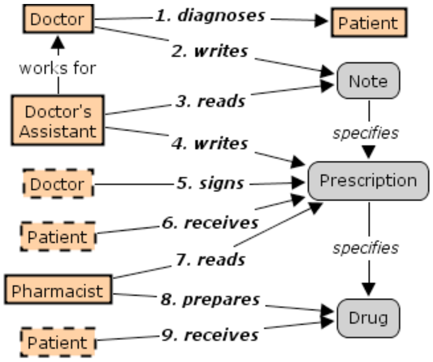


Figure 8. Re-representation of the process embedded in previous concept maps to a linear format.

Two notable features of this representation are: (1) the way pairs of roles are connected via *boundary objects* (Star, 1990), objects (physical or conceptual) which encapsulate information in a format suitable for understanding by an external work group; and (2) the transcription of information from representation to representation, mediated by various actions (from "Note" to "Prescription" to "Drug"). Both features indicate the need to incorporate concepts from Distributed Cognition (Hutchins & Klausen, 1996; Hutchins, 1995; Rogers & Ellis, 1994) into our analysis.

This view of information flow can be used to rearrange the concept map further, providing an alternate perspective for the user to fill in additional information regarding this information flow. An example of this is shown in Figure 9, which contains a rearranged view of the data, plus the result of further knowledge elicitation.

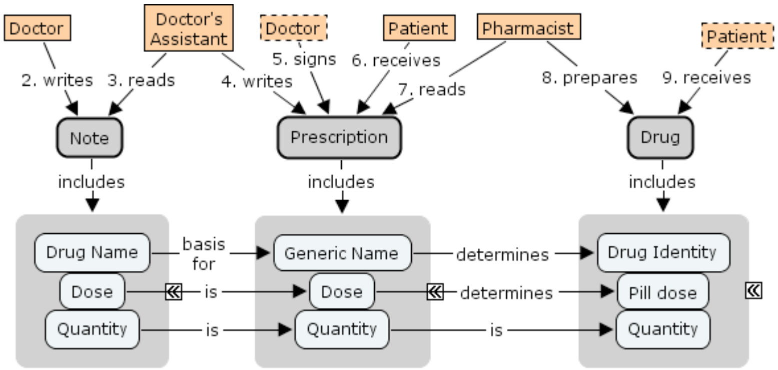


Figure 9. Reformatting to show information propagation reveals important features and simplifies further elicitation.

In an example of what Hutchins and Klausen (op. cit.) refer to as propagation of information across representations, the drug information propagates (in various forms) from the doctor's internal representation (in his mind), to the note, via the doctor's assistant to the prescription, to the pharmacist's internal representation, where it is expressed as a particular choice of drug. This view, which again can be automatically generated from the above knowledge base, presents a user with an intuitive, understandable format for showing how task participants use a particular set of information. Armed with knowledge of this flow of information, the expert designing a TREK system can produce appropriate representations utilizing this map as to who accesses this information, in what order, and for what purpose.

Another formalized concept map that we extract from the root data is shown in Figure 10. This map shows the relations between roles, as automatically identified and summarized from the above information. Direct relationships between roles are copied over ("Doctor's Assistant works for Doctor"), as are role-role actions ("Doctor diagnoses Patient"). In addition, linkages between roles via objects are extracted by examining ordered pairs of actions with common elements. For example, the pair of relations "Doctor writes Note" and "Doctor's Assistant reads Note" are used to conclude that the object Note serves as a connection between Doctor and Doctor's Assistant. These derived relations can also be presented to the user for culling, in the case of nonsensical or duplicate derivations.

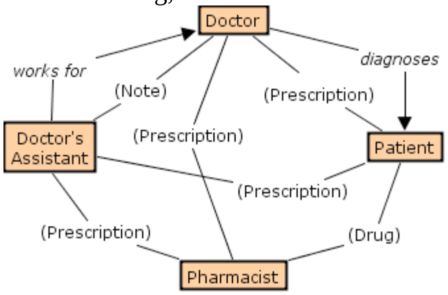


Figure 10. Direct and mediated role relations.

This concept map allows further examination of the ways that roles really interact. For example, it may not be apparent to the expert that the doctor and pharmacist are effectively communicating via the prescription slip, nor what the information path between the two should contain.

The approach outlined for the TREK system makes use of IHMC's CmapTools package to generate and store concept maps. The major form of communication used for receiving data from and publishing data to CmapTools is XML, via that program's built-in XML export/import routines. The resulting tagged set of concepts and predicates are stored internally for querying within the TREK system. In the above example, the list of Roles for the Task would be presented to the expert during task specification; the list of actions taken by each role then becomes a starting point for devising a task model. In specifying the "doctor" role identified above, for example, the user would step through each action ("diagnoses", "writes", "signs"), and begin elicitation of the skills and capabilities required for performing these actions.

4 Summary

In this paper we have presented a novel approach to formalizing concept maps into a format that retains the intuitive and flexible nature of non-formalized concept maps, while allowing automatic parsing via intelligent algorithms. We have presented a system that could use such technology to support collaboration. TREK-based systems are based on thorough models of tasks and roles within tasks, requiring extensive knowledge elicitation of the sort made possible via concept mapping. Finally, we have presented a new variant on concept maps, centered around a work process, that retain the understandability of concept maps while allowing a computer to support that work process. These maps end up looking a lot like Petri Nets, an alternate formalism for activity, and thereby allow leveraging of work in that area for modeling and improving activity flow.

TREK uses concept maps as a persistent, human- and computer-readable storage medium for user and task models. As envisioned, users would create concept maps for various processes and formalize them using the procedure outlined above. These would be tagged according to the roles and tasks they detail, and stored as a part of a larger database of knowledge necessary for a functioning system. As explained above, the semi-structured format of the stored information allows the intelligent back-end to use the data for reasoning, while retaining the beneficial features of concept map storage. If the user at a later date needs to go back and update the knowledge base, the information stored within the TREK system can be re-expressed as a concept map for ease of updating.

Our approach of incrementally formalizing concept maps in order to allow machine parsing has wide application. We have focused here on process-oriented concept maps due to project constraints; however, these ideas could be used to provide input for expert systems in health diagnosis domains, or to aid in creation of template-based concept maps for operational order representation (Hoffman & Shattuck, 2006). Striking a proper balance between user effort and ease of parsing is difficult, but an iterative process seems to enable a wide range of flexibility. Future work for this approach involves examining the problematic edge conditions, such as concepts that serve as both role and object, or concept or relations classifications which are outside of the specified set (Role, Object, Other). Creation of a Java 'wizard' for concept map formalization, potentially integrated with the CmapTools environment, is another area for examination. Finally, a thorough investigation of the ability of intelligent systems to automatically suggest mappings for concepts and relations would improve the user experience when tagging large maps; given a classifier based on automatic part-of-speech tagging that could discriminate Roles from Objects with high accuracy, the task of the user could simply become concept classification verification.

5 Acknowledgements

This work was supported by Air Force SBIR Contract #AFA0650-05-M-6615, under the oversight of Dr. June Skelly, whom we would like to thank for her guidance and support during this project. We also would like to thank IHMC for use of their CmapTools environment, and Dr. Robert Hoffman for his input and tutelage.

References

Bardram, J. (1997). Plans As Situated Action: an Activity Theory Approach to Workflow. In *Proceedings of European Conference on Computer-Supported Collaborative Work (ECSCW '97)*.

Das, S. & Feinman, A. (2006). *Toolkit for Role-Aware Exchange of Knowledge.* (Rep. No. Final Report No. R05271). Cambridge, MA: Charles River Analytics.

Eriksson, H. & Penker, M. (2000). *Business Modeling With UML: Business Patterns at Work*. Indianapolis, IN: Wiley.

Hayes, P., Eskridge, T. C., Mehrotra, A., Bobrovnikoff, D., Reichherzer, T., & Saavedra, R. (2005a). COE: Tools for Collaborative Ontology Development and Reuse. In *Proceedings of Knowledge Capture Conference (K-CAP 2005)*.

Hayes, P., Eskridge, T. C., Saavedra, R., Reichherzer, T., Mehrotra, M., & Bobrovnikoff, D. (2005b). Collaborative Knowledge Capture in Ontologies. In *Proceedings of K-CAP '05*. Banff, Canada.

Hoffman, R. & Shattuck, L. (2006). Should We Rethink How We Do OPORDs? *Military Review,* 2006(March-April), 100-107.

Hutchins, E. (1995). *Cognition in the Wild*. Cambridge, MA.: MIT Press.

Hutchins, E. & Klausen, T. (1996). Distributed Cognition in an Airline Cockpit. In D. Middleton & Y. Engestrom (Eds.), *Communication and Cognition at Work* (pp. 15-34).

Kammer, P., Boicer, G., Taylor, R., Hitomi, A., & Bergman, M. (2000). Techniques for Supporting Dynamic and Adaptive Workflow. *Computer Supported Cooperative Work,* 9(3-4).

Kaptelinen, V. & Nardi, B. (1997). Activity Theory: Basic Concepts and Applications. In *Proceedings of CHI 1997 Electronic Publications: Extended Abstracts,* E. Francik & J. Larson (Eds.).

Kogut, P., Cranefield, S., Hart, L., Dutra, M., Baclawskit, K., Kokar, M. et al. (2002). UML for Ontology Development. *Knowledge Engineering Review Journal Special Issue on Ontologies in Agent Systems*, 17.

Kuutti, K. (1996). Activity Theory As a Potential Framework for Human-Computer Interaction Research. In B. Nardi (Ed.), *Context and Consciousness: Activity Theory and Human-Computer Interaction*. Cambridge, MA: MIT Press.

Leont'ev, A. (1978). *Activity, Consciousness, and Personality*. Englewood Cliffs, NJ: Prentice-Hall.

Norman, D. A. (1988). *The Design of Everyday Things*. Basic Books.

Novak, J. & Gowin, D. B. (1984). *Learning How to Learn*. New York, New York: Cambridge University Press.

Novak, J. D. (1998). *Learning, Creating, and Using Knowledge: Concept Maps As Facilitative Tools in Schools and Corporations*. Mahweh, NJ: Lawrence Erlbaum Associates.

Object Management Group, I. (2006). Unified Modeling Language (UML), Version 2.0. http://www.omg.org/technology/documents/formal/uml.htm [On-line].

Petri, C. A. (1962). *Kommunikation Mit Automaten.* Institut fur instrumentelle Mathematik, Bonn, Germany.

Rogers, Y. & Ellis, J. (1994). Distributed Cognition: an Alternative Framework for Analysing and Explaining Collaborative Working. *Journal of Information Technology,* 9(2), 119-128.

Star, S. L. (1990). The Structure of Ill-Structured Solutions: Boundary Objects and Heterogeneous Distributed Problem Solving. In *Morgan Kaufmann Series In Research Notes In Artificial Intelligence. Distributed Artificial Intelligence.* (2 ed.), (pp. 37-54).

van der Aalst, W. (1998). The Application of Petri Nets to Workflow Management. *The Journal of Circuits, Systems, and Computers,* 8(1), 22-66.