Managing, Mapping, and Manipulating Conceptual Knowledge*

Alberto Cañas
Institute for Human and Machine Cognition
University of West Florida
11000 University Parkway
Pensacola, FL 32514
acanas@ai.uwf.edu

David B. Leake and David C. Wilson
Computer Science Department
Lindley Hall, Indiana University
150 S. Woodlawn Ave
Bloomington, IN 47405
{leake,davwils}@cs.indiana.edu

Abstract

Effective knowledge management maintains the knowledge assets of an organization by identifying and capturing useful information in a usable form, and by supporting refinement and reuse of that information in service of the organization's goals. A particularly important asset is the "internal" knowledge embodied in the experiences of task experts that may be lost with shifts in projects and personnel. Concept Mapping provides a framework for making this internal knowledge explicit in a visual form that can easily be examined and shared. However, it does not address how relevant concept maps can be retrieved or adapted to new problems. CBR is playing an increasing role in knowledge retrieval and reuse for corporate memories, and its capabilities are appealing to augment the concept mapping process. This paper describes ongoing research on a combined CBR/CMap framework for managing aerospace design knowledge. Its approach emphasizes interactive capture, access, and application of knowledge representing different experts' perspectives, and unobtrusive learning as knowledge is reused.

Introduction

Managing the knowledge assets of an organization requires capturing and retaining useful knowledge and making it available in a usable form when it is needed in the future. This process is complicated by difficulties in acquiring and representing knowledge, in accessing relevant knowledge, and in reapplying prior lessons to new situations. These issues are particularly acute in capturing and utilizing "internal" knowledge assets embodied in the experiences of task experts.

Different technologies offer different benefits for addressing these problems. Concept Maps (CMaps) (Novak & Gowin 1984) provide a framework for capturing experts' internal knowledge and making it explicit in a visual, graphical form that can be easily examined and shared. Concept mapping has been used for knowledge acquisition during the development of expert systems (Ford *et al.* 1991), as the basis for the explanation component of expert systems (Ford, Cañas, & Adams-Webber 1992), and for knowledge preservation at NASA (Coffey, Moreman, & Dyer 1999).

Procedures have been developed help guide CMap generation (e.g., (Jonassen, Beissner, & Yacci 1993)), and interactive tools have been developed to facilitate generation and manipulation of concept maps and map sharing over the Internet (Cañas *et al.* 1995). These tools support knowledge access through map browsing but not automatic retrieval of relevant maps from map archives or support for adaptation of retrieved CMaps. Such adaptation is important, for example, when a concept map represents a design that must be modified to fit new constraints. Case-based reasoning (CBR) is increasingly investigated as a knowledge management technique to support the retrieval and adaptation of prior cases (Becerra-Fernandez & Aha 1999; Klahr 1997), and retrieval and adaptation methods from CBR are promising to extend existing concept mapping tools. Conversely, using the concept mapping process to capture cases may help CBR by facilitating case engineering.

Thus concept maps and case-based reasoning each address complementary parts of the knowledge management problem. This paper describes ongoing research on combining concept mapping and CBR to leverage off their respective strengths. CBR provides support for the retrieval and application of CMaps, while CMaps and CMap tools provide mechanisms for capture and representation of hierarchical cases, browsing through the case organization to find alternative cases, and case examination. The combined framework provides interactive knowledge capture and access, support for multiple conceptualizations of knowledge, and unobtrusive learning as stored knowledge is

*This research is supported in part by NASA under award No NCC 2-1035. Authors Leake and Wilson are currently at the Intelligent Information Laboratory at Northwestern University, on leave from Indiana University. They thank the Laboratory and the Northwestern Computer Science Department for their support. They also gratefully acknowledge the many contributions to this project by Mary Livingston and the ADTT team at NASA Ames and by James Newkirk at Indiana University.

applied to new situations.

Concept Mapping for Knowledge Capture and Sharing

Concept maps represent meaningful relationships between concepts in the form of propositions. Propositions are two or more concepts linked by words to form a semantic unit. In its simplest form, a concept map would contain just two concepts connected by a linking word to form a single proposition. For example, "Centaur is a rocket" would represent a simple map forming a valid proposition about the concepts "Centaur" and "Rocket." A concept acquires additional meaning as more propositions include the concept. Thus, that the Centaur is a rocket, Centaur is powered by a turbopump, Centaur's role is as an upper stage, and so on, all expand the meaning of the concept Centaur. In this sense, concept maps represent meaning in a framework of embedded propositions. (Semantic nets are a form of concept map, but concept maps also include less constrained network representations.)

Different content and structure are contained in concept maps depending on the contexts for which they are generated. Consequently, maps having similar concepts can vary from one context to another and can be highly idiosyncratic. The strength of concept maps lies in their ability to express a particular person's knowledge about a given topic in a specific context. Concept maps thus provide an elegant, easily understood representation of an expert's domain knowledge.

The Institute for Human and Machine Cognition at the University of West Florida has developed software tools that extend the use of concept maps beyond knowledge capture and examination, to serve as the browsing interface to a corporate memory of hierarchical concept maps and associated information resources. For example, the tools are currently being used in a NASA Lewis Center project to capture and preserve Senior Engineers' design expertise knowledge of launch vehicle systems integration for the Centaur/RL-10 rocket system (Coffey, Moreman, & Dyer 1999). As part of this research project, a prototype browsable, multimedia model of the experts' domain knowledge was built, as illustrated in Figure 1. The CMap tools allow icons to be associated with concepts, providing links to other concept maps or other explanatory media (video, text, images, simulations, WWW pages, etc.), which may be distributed throughout the Internet.

The tools also support ways in which the knowledge encoded in concept maps can be selectively shared among a community of users. Concept maps are hierarchical and may link to other maps over the Internet, enabling distributed teams to collectively develop and access complex maps. In addition, during the construction of concept maps, the tools allow users to designate sentences or propositions selected from CMaps for "publication" or sharing with other users. These propositions, or "claims," are stored in a shared server, composing a "knowledge soup" of assertions from multiple sources. The system extracts from this information the claims that other users chose to share and that are relevant to the claims the user published, providing information that may aid the user in constructing his or her own concept maps. The system also provides a process for commenting on or questioning these shared claims, querying other users about aspects that the user does not understand. This process facilitates distributed discussion, refinement and use of concept maps, and the technology is currently being enhanced to develop a collaborative knowledge sharing environment for the NASA Astrobiology Institute.

Combining CBR and CMaps

We are investigating the combination of CMaps and CBR for knowledge management to support aerospace design at NASA. Aerospace design is a complex task area in which "knowledge loss" as projects are discontinued or engineers retire is a profound problem. Previous efforts have been made at NASA to store and access textual reports of important lessons using standard commercial CBR tools (Bagg 1997). However, even when textual design records have been captured they may be hard to understand and reuse because different experts conceptualize designs very differently. This has resulted in a push to capture design knowledge in the form of CMaps.

Our interactive design support framework, DRAMA (Design Retrieval and Adaptation Mechanisms for Aerospace), is being developed in cooperation with the Advanced Design Technologies Testbed project at NASA Ames Research Center. The goal is both to develop useful tools for aerospace design and to establish a general "knowledge-light" (Wilke *et al.* 1997) framework for interactive case-based design support systems.

Motivations for combining CBR and CMaps: The integration of CBR with interactive CMap tools provides leverage for both the CBR and CMap systems. Existing CMap tools provide an interactive medium for representing and examining designs, but their framework does not provide search facilities to find relevant stored CMaps or advice on how to navigate hierarchical CMap structures. Likewise, although the tools provide capabilities for interactively defining new CMaps and manipulating their structure by adding, deleting, or substituting components, the tools provide little support for the decision-making that underlies the adaptation process. Consequently, their usefulness can be extended by the addition of automatic facilities for retrieving relevant CMaps, automated aids to navigating CMaps and finding relevant information therein, and by aids to the reuse of existing CMaps.

Conversely, case-based reasoning can leverage off the interactive case definition and revision capabilities of the CMap tools. CMaps can be used as a browsable structure for indexing cases, either simply, according

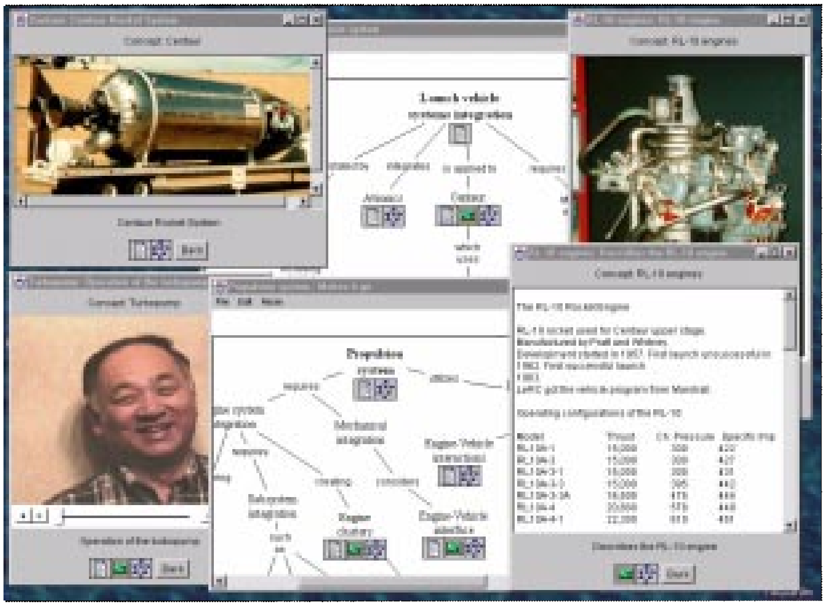

Figure 1: Expert's Domain Knowledge Model of the Centaur/RL-10 rocket system.

to the nodes under which they are placed, or contextually, according to the user's perspective, reflected by the path taken to reach them. For example, different design cases indexed under "Boeing 777" might be appropriate to present to users depending on whether they reached that node by following links for hydraulic systems or links for avionics.

Also, CMap tools provide a convenient method for entering case information in a middle form between textual descriptions (which are easy to input but hard to reason about) and rich structured representations (which are hard to input but support complex reasoning). In our domain, the push to use concept mapping to understand the design process means that such cases will be available at low cost as "seed cases" for the CBR system. In addition, the CMap tools already provide crucial functions for interactively generating, examining, and navigating the hierarchical structure of these cases.

Using the CMap/CBR combination to support knowledge access, reuse, and capture during design: In DRAMA, CMaps are used to represent two types of information. First, they represent hierarchies of aircraft and part types. Second, CMaps represent specific information about particular designs such as their components and component relationships.

The system treats the design process as generating a CMap to describe each new design. Retrieval and adaptation of relevant prior CMaps is an integral part of this process. A designer starts the design process by selecting a similar prior design as a starting point. The user may select this design either by using a traditional CBR retrieval tool for stored CMaps, or by interactively navigating through a set of concept maps providing alternative "views" of aircraft and aircraft component types, used to organize CMaps for specific aircraft. For example, suppose the designer is considering alternatives for increasing the fuel efficiency of a large airliner. The designer first navigates through the types of aircraft to select an aircraft, and selects a particular case—represented as a concept map—to adapt into the new design. The designer may adapt the specific design or may request that it be abstracted into a fill-in design template.

Adaptation of design CMaps is supported by providing suggestions of relevant prior designs and enabling the user to browse CMaps to gather information to support the adaptation process. In our example, to revise the engine to increase fuel efficiency, the designer selects the engine node of the current aircraft as the part to adapt. If no CMap is already present for the component selected (e.g., the designer wishes to fill in a sketchy design by specifying its engine), the designer can use the interactive CMap tools to create a new CMap from scratch, or browse the CMaps for designs, import a design, and then adapt as desired. To help support adaptations—e.g., to find a more fuel-efficient engine—the designer may initiate a retrieval focused either on similar components (e.g., CMaps that show aircraft using similar engines), or similar contexts for the current type of component (e.g., CMaps that show the engines of similar aircraft). The result of the process is automatically saved as a new CMap for future use. Thus each design augments the corporate memory and provides additional starting points for future knowledge reuse.

Significance of the approach

CMaps as a medium for capture and representation of experiences: Structured representations have been extensively studied within CBR. They provide much power but may require significant "case engineering" effort (Aha & Breslow 1997; Simoudis, Ford, & Cañas 1992). Work in textual case-based reasoning (Lenz & Ashley 1998) applies CBR to information already stored in textual form, but textual cases may be difficult to use. CMap representations are at a middle point between these alternatives: they include structural information and are intended to concisely represent key concept properties, but may not use a standardized semantics. This makes them more difficult to manipulate autonomously than standardized representations but also easier to acquire when domain experts are called upon to encode their knowledge. DRAMA alleviates problems of differing representations in two ways. First, when a user draws a map and is about to fill in a new link or node, it presents the user with menu of alternatives from previous maps. The user is not required to use links from this list, but when appropriate links are on the list this helps build a set of standard types over time. The second is simply the "retrieve and adapt" process itself: When new designs are generated by adaptation, significant portions of old representations are brought to new tasks, resulting in representations with similar structure.

Concept mapping as a form of design rationale capture: Many projects have applied rule-based or model-based approaches to design rationale capture, but encoding and updating the needed information can be prohibitively expensive. Because CMap design cases already capture an entire design as context, we believe that useful rationale capture can be achieved with fairly limited additional information: an annotation about why the designer chose a particular component, given the implicit context of the previous components chosen. DRAMA enables designers to provide this information as a form of "weak explanation" of the type advocated by Gruber and Russell (1992), providing just enough information to guide a designer's own reasoning process.

CMaps/CBR as interactive retrieval: The ability to browse through the CMap indexing structures provides a convenient way for users to interactively search for cases. This is in the spirit of conversational case-based reasoning (CCBR) systems, which guide the retrieval process through an interactive dialogue of questions (Aha & Breslow 1997), but here the user directly examines and traverses hierarchical organizational structures.

Conclusion

Concept mapping is useful for knowledge management as a vehicle for externalizing "internal" expert knowledge, to allow that knowledge to be examined, refined, and reused. CBR is useful for knowledge management in providing an easy-to-understand knowledge representation—records of specific reasoning episodes—and methods for accessing relevant information and building up a corporate memory of experiences. The synergy of the two technologies provides a promising approach for addressing corporate "knowledge loss" by supporting the capture and reuse of expert design experiences, helping to manage and maintain an important component of organizational knowledge assets.

References

Aha, D., and Breslow, L. 1997. Refining conversational case libraries. In *Proceedings of the Second International Conference on Case-Based Reasoning*, 267–278. Berlin: Springer Verlag.

Bagg, T. 1997. RECALL: Reusable experience with case-based reasoning for automating lessons learned. http://hope.gsfc.nasa.gov/RECALL/homepg/recall.htm.

Becerra-Fernandez, I., and Aha, D. 1999. Case-based problem solving for knowledge management systems. In *Proceedings of the Twelfth Annual Florida Artificial Intelligence Research Symposium.* Menlo Park: AAAI. In press.

Cañas, A.; Ford, K.; Brennan, J.; Reichherzer, T.; and Hayes, P. 1995. Knowledge construction and sharing in quorum. In *World Conference on Artificial Intelligence in Education.*

Coffey, J.; Moreman, D.; and Dyer, J. 1999. Institutional memory preservation at NASA Lewis Research Center. In *Proceedings of the HBCU/OMU Research Conference.*

Ford, K.; Cañas, A. J.; Jones, J.; Stahl, H.; Novak, J.; and Adams-Webber, J. 1991. ICONKAT: an integrated constructivist knowledge acquisition tool. *Knowledge Acquisition* 3.

Ford, K. M.; Cañas, A. J.; and Adams-Webber, J. 1992. Participatory explanation: A new paradigm? In *Proceedings of the Tenth European Conference on Artificial Intelligence Workshop on Expert Judgement, Human Error, and Intelligent Systems*, 146–155.

Gruber, T., and Russell, D. 1992. Generative design rationale: Beyond the record and replay paradigm. Knowledge Systems Laboratory KSL 92-59, Computer Science Department, Stanford University.

Jonassen, D.; Beissner, K.; and Yacci, M. 1993. *Explicit methods for conveying structural knowledge through concept maps.* Hillsdale, NJ: Erlbaum. chapter 15, 155.

Klahr, P. 1997. Knowledge management on a global scale. In Gaines, B.; Musen, M.; and Uthurusamy, R., eds., *Proceedings of the 1997 Spring Symposium on Artificial Intelligence in Knowledge Management*, 82–85. Stanford, CA: AAAI.

Lenz, M., and Ashley, K., eds. 1998. *Proceedings of the AAAI-98 workshop on textual case-based reasoning.* Menlo Park, CA: AAAI Press.

Novak, J., and Gowin, D. 1984. *Learning How to Learn.* New York: Cambridge University Press.

Simoudis, E.; Ford, K.; and Cañas, A. 1992. Knowledge acquisition in case-based reasoning: "...and then a miracle happens". In Dankel, D., ed., *Proceedings of the 1992 Florida AI Research Symposium.* FLAIRS.

Wilke, W.; Vollrath, I.; Althoff, K.-D.; and Bergmann, R. 1997. A framework for learning adaptation knowedge based on knowledge light approaches. In *Proceedings of the Fifth German Workshop on Case-Based Reasoning.*