Proceedings of I-KNOW '05
Graz, Austria, June 29 - July 1, 2005

A Visualization Tool for the Sitemap of a Knowledge Portal and the Concept Map of Group Knowledge

Yi-Li Lin
(Department of Computer Science and Information Engineering,
National Chiao-Tung University, Hsinchu, Taiwan
yllin@csie.nctu.edu.tw)

Jyh-Da Wei, Gen-Cher Lee* and D. T. Lee*
(Institute of Information Science, Academia Sinica, Taipei, Taiwan
{jdwei, gc, dtlee}@iis.sinica.edu.tw)

Abstract: Contents of a website can be constructed by two means. Conventional websites usually have a sitemap initially. Using this map as a blueprint, the website designers develop the pages of the content and add links to the content of a website, such as the I-Know'05 homepage. For the website known as a knowledge portal, a sitemap cannot be prepared to outline the contents in advance. A knowledge portal site allows its members to create their own knowledge objects. The website therefore grows itself unlimitedly and unexpectedly just like a coral. Interestingly, these knowledge objects are not only represented in webpages but also associated with conceptual elements defined within the ontology of the knowledge portal. In this regard, we develop a visualization tool to display the sitemap of a knowledge portal and observe the concept map of group knowledge. As a practice platform of our visualization tool, we chose a locally developed knowledge portal, called Open Computational Problem Solving (OpenCPS) Knowledge Portal.

Keywords: website, sitemap, concept map, knowledge portal, group knowledge, problem-solving environment
Categories: H.1.1, H.3.1, H.3.5, H.5.2

1 Introduction

Knowledge management has been an attractive research topic for over three decades. The study of knowledge management gets increasing attention recently, due to the exponential growth of knowledge and the mature technologies of the World Wide Web (WWW). Web technology brings about a kind of website known as "knowledge portal", which supports various kinds of content types and presents the knowledge objects as webpages. Knowledge portals may also be equipped with further capabilities, such as data sharing, software warehousing, access control, and workflow management. Many companies and research organizations began to create development platforms for building knowledge portals, for example, Microsoft

---

* Also with the Department of Computer Science and Information Engineering, National Taiwan University, Taipei, Taiwan

Sharepoint [Sharepoint 05], the Problem-Knowledge Couplers (PKC) [PKC 05], and the Zope application server [Zope 05].

In comparison with conventional websites, knowledge portals have two features: (1) the contents are contributed by all the portal members, not just by the web design community. Accordingly, the contents of a knowledge portal can be regarded as group knowledge, which aggregates all the intelligent assets from the user base. (2) Except an axiom ontology given by the portal designer, no detailed sitemaps or blueprints are prepared to outline the contents in advance, i.e., the contents of the website grow in an unplanned and unpredictable manner. In consideration of the above two aspects, we believe the quality of a knowledge portal can be improved if it can incorporate a visualization tool to monitor the sitemap after the contents are built. This sitemap allows the members to examine the current status of knowledge structure by visualization. Thereby, the group knowledge of this knowledge portal can not only serve as an aggregation of content for learning purposes, but also get enhanced from users' feedback.

In this work, we attempt to generate this sitemap of a knowledge portal by adopting the notion of a concept map. That is, the arcs connecting related knowledge objects are labeled with relations and the inner properties of knowledge objects can also be unfolded. Hereafter, this map is referred to as the Site-And-Concept (SAC) map, mainly used for group knowledge visualization on a knowledge portal. We chose a locally developed knowledge portal, dubbed Open Computational Problem Solving (OpenCPS) (http://www.opencps.org), as the platform for current experiments. The SAC map generator is potentially useful in other domains, for we can extend its backend search engine to suit any particular environment.

2 Knowledge Management for Computational Problem Solving

We have implemented a knowledge management platform for Computational Problem Solving. In solving computational problems, we need to know first if the problems under study are open, i.e., if there exist any solutions. If so, we need to know whether the solutions have been implemented, and what the associated time complexities are. If the solutions are not available, what are the relations with other similar problems that already have solutions? In this context, the knowledge management is regarded as management of relations among problem, solution, and implementation spaces [Lee et al. 03], where problem space objects consist of uniquely identifiable computational problems, solution space objects consist of algorithmic solutions, and implementation space objects consist of actual code for the solutions. An abstraction of our knowledge structure is shown in Fig. 1. Equivalent problems are grouped together, and sub-problems, super-problems, and variant problems are indicated. (A problem is considered a sub-problem of the other problem when the solution for the latter can be transformed into a solution for the former.) There may exist different solutions to the same target problem with different time complexities, and different implementations for the respective solutions. The knowledge objects in these spaces are associated with proper attributes. For example, an algorithmic solution possesses attributes such as solution name, target problem, description, pseudo code, complexity, existing implementations, and related publications. Additional details can be found in [Lee et al. 03].

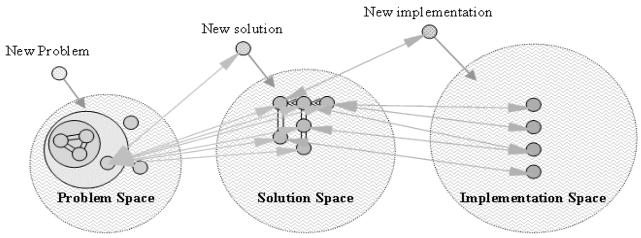

Figure 1: Knowledge Management for Computational Problem Solving. The knowledge management is regarded as managing the relations among problem, solution, and implementation spaces.

The Open Computational Problem Solving (OpenCPS) Knowledge Portal has been established. The OpenCPS website is now available on the URL http://www.opencps.org, and will be used as the test bed when we develop the SAC map generator.

3 Development of the Sitemap Generator

We attempt to develop a site-and-concept (SAC) map generator for knowledge portals and use the OpenCPS website as a practice platform. In this section, we introduce concept maps, and present how we design the SAC map generator.

3.1 Introduction to Concept Maps

Based upon David Ausubel's theories about the psychology of learning in the 1960s [Dürsteler 04], Joseph D. Novak of Cornell University invented concept maps for organizing and representing knowledge [Novak 00, Novak 84, Milam et al. 01]. Concept maps are defined as a collection of concepts or propositions, represented by circles or boxes of some types, and relationships between concepts or propositions, represented by connecting lines between two concepts or propositions [McAleese 98]. Concept maps have been broadly used to analyze learning and education processes. Our work is to generate sitemaps in terms of concept maps to display the formation of group knowledge.

3.2 Map Generator

The OpenCPS website is built upon Plone/Zope open source products [Plone 05], which support Python language [Python 05] to develop plug-in packages. Based on its easy plug-in property, we have developed a backend search engine to retrieve inner attributes and relation information of given knowledge objects. After that, we also create a frontend SAC map monitor using Java language. This frontend monitor retrieves search results from an application programming interface (API) of the backend search engine and then draws up the sitemap. The SAC map generator is a combination of both the search engine and the SAC map monitor.

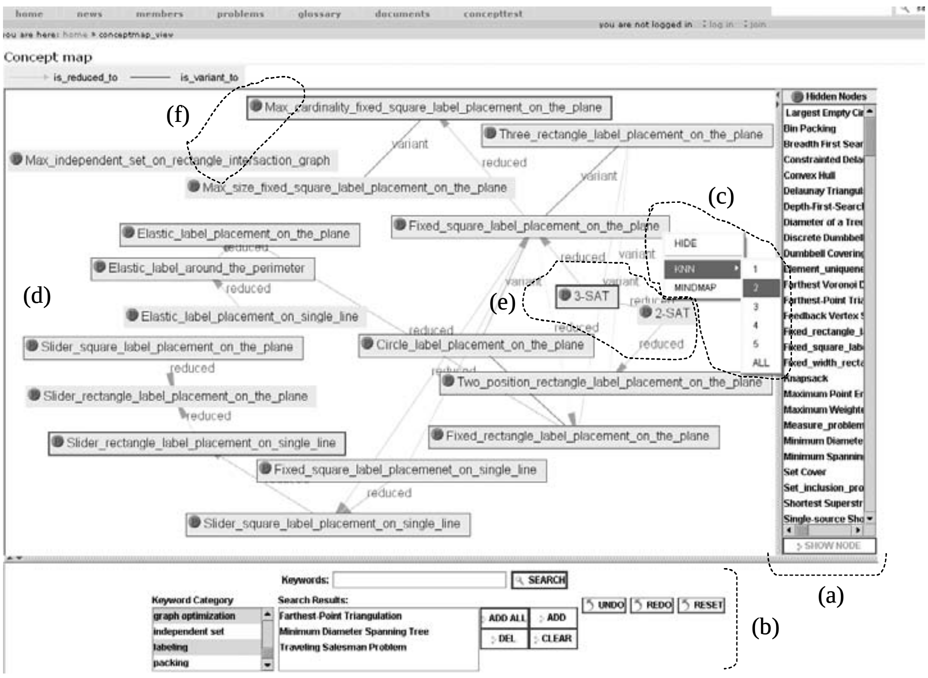

Figure 2: Snapshot of the SAC Map Generator. Four modules and two showcases are shown. System modules are (a) a full list, (b) an advanced search panel, (c) a set of right-click functions, and (d) the drawing panel. Showcases are (e) learner-mode and (f) expert-mode operations.

Figure 2 is a snapshot of our system, where four modules are shown, i.e., (a) a full list of knowledge objects, (b) an advanced search panel, (c) a set of right-click functions, and (d) the drawing panel. To save space, two showcases – (e) learner-mode and (f) expert-mode operations are shown in the same figure and are described in the next section.

Module (a) of Fig. 2 is a full list holding all the current problem objects. Users may select one or more target items and click a button to show them as nodes on the drawing panel. Module (b) is an advanced search panel, which provides an alternative method for selecting target problems efficiently. This search panel supports searching by keywords and predefined keyword categories. The search results appear in another multiselect list for users to add into the drawing panel. Nodes on the drawing panel are clickable to open the relative webpages; on the other hand, module (c) provides additional right-click functions. The "Hide" function returns the target node to the full list. The "k-NN" function brings all the k nearest neighbors of the target node into the drawing panel. Finally, the "Mindmap" function zooms into the target node and prints another map as shown in Fig 3. This map is drawn by the Freemind freeware [Freemind 05]. The central node is the problem object of interest and is linked to outer related objects on the left and inner properties on the right.

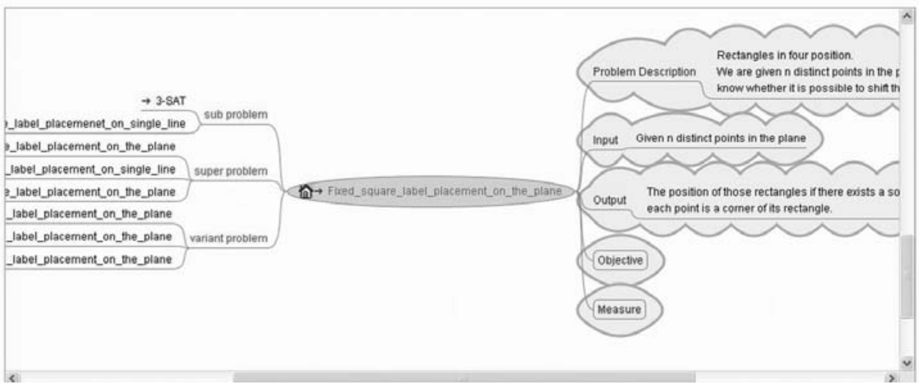

Figure 3: Mindmap of a target Computational Problem. The central node is the concerned problem linked with inner properties and outer related objects.

3.3 Site-And-Concept Map

Module (d) of Fig. 2 is the drawing panel that shows the Site-And-Concept Map. As a sitemap, it provides a global view of connection topology among the web contents. It is also clickable for access to the webpages. As a concept map, it links the knowledge objects and labels their relation, such as “reduced-to”, “equivalent”, “variant” and so on. For each knowledge object, the backend search engine keeps a record of its properties and the adjacency list of related objects. Therefore, the links and labels can be refreshed quickly when a node is newly added into or removed from the map. The nodes can be manually dragged to anywhere on this map, as long as we feel the layout is clear to present the topology of knowledge connection. We also provide undo and redo functions for convenience of the user.

4 Group Knowledge Visualization via the SAC Map

For computational problem solving, several efforts have been focused on formalizing and indexing computational problems. The NP optimization problem compendium [Crescenzi and Kann 95] is a noteworthy electronic book full of useful information on this particular class of problems. The U.S. National Institute of Standards and Technology has also established a “Dictionary of Algorithms and Data Structures” [NIST 05] that formally defines over 1,000 computational problems. However, both projects lack adequate knowledge structures to display well-organized concept maps.

The OpenCPS knowledge portal has a promising knowledge structure (Fig. 1) to arrange member-contributed contents. It also applies a rigid workflow control [Lee et al. 03], i.e., knowledge objects must be accepted by a reviewer before they can be disseminated. These knowledge management measures are likely to ensure the correctness of group knowledge. On the other hand, the SAC map generator we proposed is an efficient artifact to make the group knowledge useful for education and expandable from the user’s feedback.

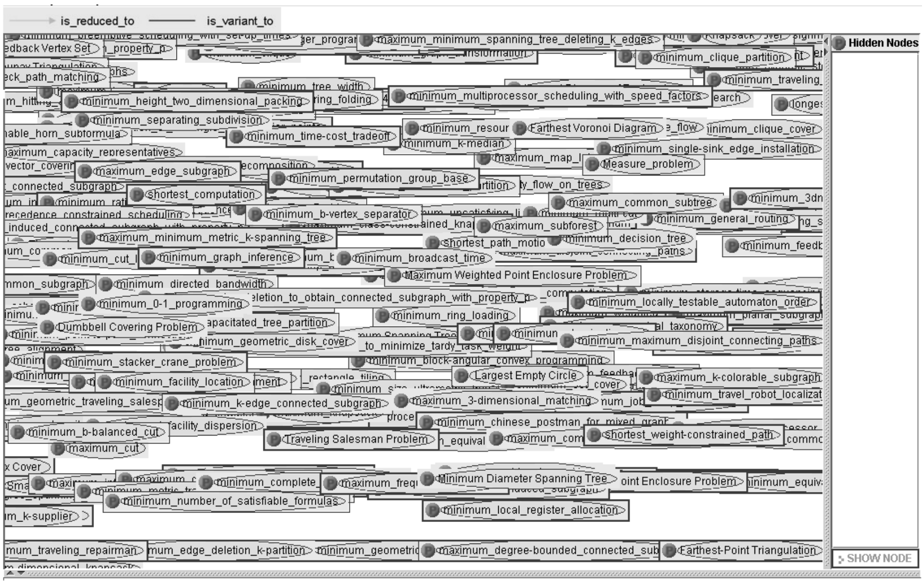

Figure 4: Complete map of current problem objects. We do not apply any filtering anc layout process so that this picture may simulate the image of human cognition withou concept map supports.

The typical operations of the SAC map generator can be categorized into three classes, i.e., the basic, learner-mode, and expert-mode operations:

1) Basic operation: The basic search operation is to use keywords to search the target objects. For example, users may type “map labeling” as the keyword and thus draw a SAC map as shown in Fig. 2(d), except parts (e) and (f).
2) Learner-mode operation: For an aggressive learner, to show the knowledge objects associated with given keywords is not enough. For example, the 3-SAT and 2-SAT problems cannot be found when we search for keyword “map labeling”, but these two problems are important for a student to get a thorough understanding of how difficult map labeling problems are, because they can be polynomially reduced to the “Fixed square label placement on the plane” problem. The right-click k-NN function helps display k nearest neighbors of a target object, as Fig 2(e) shows.
3) Expert-mode operation: Expert users are usually interested in verifying the completeness of other members’ knowledge. The SAC map generator is suitable for the experts to make contributions. If they are very certain that one problem object is related to another and this relation cannot be found by the k-NN function, they can use the keyword search again. The search result will imply that they either create such a new problem, or add a “relation object” for the target problems. For example, an expert user firmly believes that the “Max cardinality fixed square label placement on the plane” problem is a variant of the “Max independent set on rectangle intersection graph” problem, and a follow-up search also reveals that the latter problem object is an isolated node as shown in Fig 2(f).

Accordingly, this expert user could then create such a "relation object" in OpenCPS, making the knowledge structure more complete.

The SAC map generator is currently a monitor to visualize the formation of group knowledge, so we do not provide any function to generate relation objects automatically. Nevertheless, we will create a new function in the near future to check "transitive closure", i.e., if problem A is a variant of problem B, and problem B is a variant of problem C, then problem A is considered a variant of problem C. We will furthermore create a checkbox to enable or disable this new function.

The OpenCPS portal members have contributed more than 200 problem objects to the website for the present. Figure 4 is the complete map of these knowledge concepts, where we do not apply any filtering or layout processes. This picture simulates the image of human cognition without concept map supports. This shows a rather messy situation. We can hardly make out any of the concept relations, much less to examine or learn from this knowledge portal. In comparison with the above showcases, we believe that the SAC map generator can properly enhance the performance of a knowledge portal.

5 Conclusion

In previous works, researchers have used concept maps to extract expert knowledge [Basso 04, Coffey et al. 02], share cognitive space for group learning [Lee 02], and develop a website in a given conceptual framework [Gaines and Shaw 95]. However, there are few efforts dedicated to presenting the sitemap or concept map corresponding to a knowledge portal.

The contents of a knowledge portal are contributed by all the members, and thus become group knowledge. As current knowledge portals are in the form of a website, we develop a site-and-concept (SAC) map generator to visualize the formation of group knowledge. The SAC map generator supports both the learner- and expert-modes to allow them to interact with the knowledge portal more efficiently. Users not only can learn from but also can contribute to the group knowledge through the SAC map.

The SAC map generator for OpenCPS knowledge portal is comparatively easy to develop, because the computational problem solving environment is characterized by (1) all the knowledge objects are attributed into specific spaces; and (2) the relations are constructed based on rigorous mathematical proofs. The SAC map generator can be of practical value in general because its frontend is developed using Java language. Meanwhile, the backend search engine can be extended to include information retrieval and data mining techniques for particular application environments.

Acknowledgements

This work was supported in part by the National Science Council under the Grants NSC-93-2213-E-001-013, NSC-93-2422-H-001-0001, and NSC-93-2752-E-002-005-PAE

References

[Basso 04] Stefano Basso: “Teaching by Doing with Concept Maps: Integrating Plone and CMAPTools”, Proceedings of the First International Conference on Concept Mapping, Pamplona, Spain 2004.

[Crescenzi and Kann 95] Crescenzi, P. and Kann, V.: “A compendium of NP optimization problems”, Technical report SI/RR-95/02, Dipartimento di Scienze dell'Informazione, Università di Roma “La Sapienza”, 1995

[Coffey et al. 02] Coffey, J., Hoffman, R., Canas, A. and Ford, K.: “A concept map-based knowledge modeling approach to expert knowledge sharing”, Proceedings of IASTED International Conference on Information and Knowledge Sharing(IKS’02) , VI, USA, November (2002), 212-217.

[Dürsteler 04] Dürsteler, J. C.: “Conceptual Maps”, The Digital Magazine of Infovis.net, num 141, 2004, Available: http://www.infovis.net/printMag.php?num=141&lang=2

[Freemind 05] Freemind – free mind mapping software, Available: http://freemind.sourceforge.net/wiki/index.php/Main_Page

[Gaines and Shaw 95]Gaines, B. R., and Shaw, M. L. G.: “WebMap: Concept Mapping on the Web”, Proceedings of WWW4: Fourth International World Wide Web Conference. Boston, December 1995, Available: http://ksi.cpsc.ucalgary.ca/articles/WWW/WWW4WM

[Lee et al. 03] Lee, D. T., Lee, G. C., and Huang, Y. W.: “Knowledge Management for Computational Problem Solving”, J.UCS (Journal for Universal Computer Science), 9, 6, 2003, 563-570, http://www.opencps.org.

[Lee 02] Lee, C. S.: “Shared cognitive space: an enabling tool for conceptual frameworks”, Proceedings of International Conference on Computers in Education. Auckland, New Zealand. 3-6 December 2002, 813-817.

[McAleese 98] McAleese Ray, “The Knowledge Arena as an Extension to the Concept Map: Reflection in Action”, Interactive Learning Environment 6, x, 1998, 1-22.

[Milam et al. 01] Milam, J. H., Santo, S. A. and Heaton, L. A.: “Concept Maps for Web-Based Application”, ERIC Technical Report (Educational Resources Information Center), 2001, Available: http://highered.org/docs/milam-conceptmaps.PDF

[NIST 05]NIST (National Institute of Standards and Technology): Dictionary of Algorithms and Data Structures, http://www.nist.gov/dads/.

[Novak 00] Novak, J. D.: “The Theory Underlying Concept Maps and To Construct Them”, 2000, Available: http://cmap.coginst.uwf.edu/info/printer.html

[Novak and Gowin 84] Novak, J. D. and Gowin, D. B.: “Learning How to Learn”, Ithaca/NY: Cornell University Press, 1984.

[PKC 05]PKC Corporation Home (Problem-Knowledge Couplers), http://www.pkc.com/.

[Plone 05] Plone: A user friendly and powerful open source Content Management System, Available: http://www.plone.org

[Python 05] Python Programming Language, Available: http://www.python.org

[Sharepoint 05] Microsoft Sharepoint Products and Technologies, Available: http://www.microsoft.com/sharepoint/.

[Zope 05] Zope.org, Available:http://www.zope.org/.