A Novel User Interface Approach for Personal and Semantic Knowledge Management

Heiko Haller and Andreas Abecker
(FZI Forschungszentrum Informatik, Karlsruhe
haller@fzi.de, abecker@fzi.de)

Abstract: We present iMapping, a diagrammatic zooming and nesting based approach for visually structuring information objects on the desktop with a special focus on personal knowledge management. It was designed based on a set of requirements gathered from the analysis of existing knowledge mapping techniques from a cognitive science point of view. We also briefly introduce a prototypical implementation.
Key Words: personal knowledge management, knowledge maps, semantic desktop
Category: H.5.2, H.4.1, I.2.1

1 Motivation

Many knowledge management systems, especially those who rely on highly structured information and meta data being entered and maintained by the users, fail because users do not make this additional effort if they do not have to. This may be one of the reasons why semantic technologies have not found widespread use so far, although semantic meta data would undoubtedly improve findability, interoperability and, in general, automated processing of information and knowledge items. For these technologies to be widely used, they have to provide immediate benefit to the user and it is crucial that they are very easy to use and do not constrain the user in her daily work. This immediate benefit is more likely to be experienced when users manage their everyday knowledge resources like personal notes, files, bookmarks etc. like in the setting of a semantic desktop environment (cp. [SKSB09]).

When semantically formalised knowledge structures are used, content is typically fine grained and highly structured. Such content structures are typically more complex than plain text or classical hypertext structures. Even with the relatively simple structures in classical hypermedia, where we just have interlinked information objects on the granularity level of whole pages or documents, hypertext research has shown that users often get “lost in hyperspace” when browsing without additional navigational help [Ter02]. This stresses the need for user interfaces that facilitate navigation and authoring of such structures without losing orientation. Using graphical environments for structuring externalized knowledge enables the user to use his highly efficient sense of spatial orientation in his *knowledge space*.

Allowing users to spatially arrange information items may enhance the link between their mental and external models because it enables the use of diagrammatic depictions whose obvious structure corresponds more closely to the structure of the content. This helps the user to intuitively grasp an overview of the subject matter. Unlike text, diagrammatic knowledge representations carry a structural analogy to the content they represent [Sch02].

Research in cognitive and instructional psychology has shown that visual mapping techniques provide easy ways to rather intuitively structure fine-grained information objects—for an overview see [Hal03]. Based on this analysis, related literature, and some user-studies, we derived a set of requirements described in [HA09]. Based on these, iMapping was designed as a new visual mapping approach that tries to unite the strengths of established mapping techniques. It combines these strengths with modern IT approaches like deep zooming and semantic technologies.

iMapping supports the whole range from easy informal note-taking to formalized knowledge engineering in the same powerful, yet easy-to-use environment. The basic metaphor of an iMap is a large pin-board where information items can be spatially arranged, enabling users to gain a visual overview over collections of items at once. These items can represent bits of text as well as (in the near future) any kind of external resources like files, web pages, pictures or other maps. They can also be nested into each other and interlinked in various ways. Besides browsing by links, users can navigate an iMap by zooming through it.

2 iMapping Design and Functionalities

The iMapping design is based on the requirements described in [HA09] (in this volume). It combines functionalities from several of the related tool categories sketched in Sec. 4, none of which cover all of these functionalities themselves. These are in particular: (i) visual knowledge representations with structural analogy to content; (ii) easy hierarchical overall topology; (iii) providing overview by integrating context and detail through zooming; (iv) facility for graph-based relation mapping; (v) allowing constructive ambiguity; (vi) support for formal semantic statements and (vii) and querying these semantic structures at the various levels of formality they have (mainly through the QuiKey extension mentioned below). To be more concrete, here are the basic functionalities of iMapping:

Basic Hierarchy of Information Items. The basis of iMapping is a large two-dimensional surface where items can be freely placed. Usually (e.g., for personal note-taking or idea management), these items will be short text passages. The size can vary from just a keyword to a short note or whole paragraphs including rich text marked up by using a wiki syntax provided by the CDS back end. These items can contain other items such that one can use microcontent rather than long, unstructured text passages. Longer texts are then a sub-map containing a sequence of smaller text-items. The iMap hierarchy goes down into deeply nested nodes which can be zoomed into (see Fig. 1). There can be multiple visual instances of one and the same information object, because it may be relevant in different contexts.

Adding and Creating Content. Adding content to an iMap is done by clicking anywhere in the map and typing some text. It is always possible to add vague and unstructured content. Although allowing for semantic knowledge management, it has been a fundamental design decision, to never force the user to specify any semantics. Content can later be refined and formalised incrementally.

Levels of Detail. Some information objects—especially text—are rather hard to recognise when they are scaled down to thumbnail size. So, composite items have two possible states: open and closed(expanded/collapsed).

Establishing Link Structures. There are three different ways of interrelating items in an iMap: (a) *linking* on an item level (stating a relation between two objects). Each of these can be mere navigational links or carry formal semantics, if specified. (b) *nesting items* into another (c) relating them by short semantic statements with QuiKey

To avoid the confusing "spaghetti syndrome", links (arrows) are not shown by default, but only made visible on demand, depending on user settings—e.g. on mouse-over (see Fig. 1). Explicitly drawn links stay visible permanently. In the same way, items can be semantically interrelated by drawing links between them which can then be typed. If this is done using auto-completion, reuse of existing relation types is fostered.

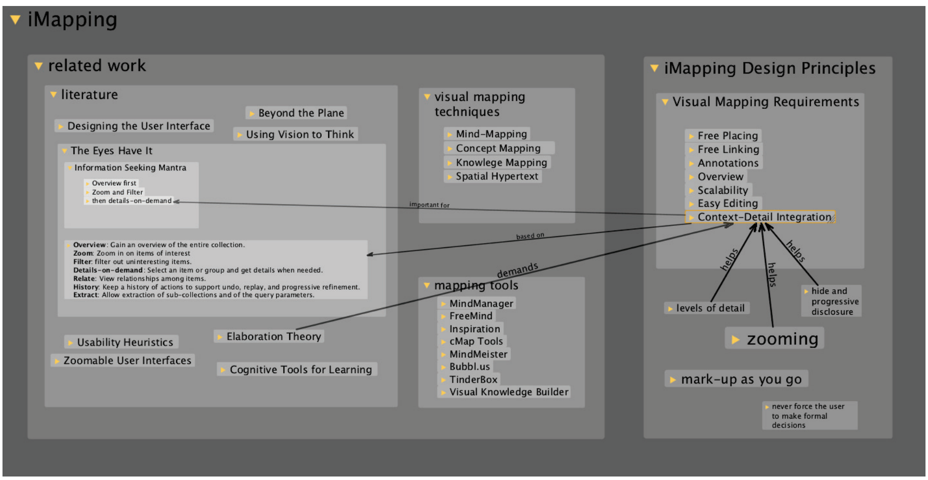

Figure 1: Sample iMap with link-structure of one item made visible.

3 iMapping Implementation

The current prototypical implementation of iMapping is based on java and makes use of the zooming user interface framework piccolo (See Sec. 4). It is complemented by QuiKey[1], a kind of smart semantic command-line that focuses on highest interaction-efficiency to browse, query and author semantic knowledge bases in a step-by-step manner. The data model and back-end used by both iMapping and QuiKey is called "Conceptual Data Structures" (CDS):

[1] For further information on QuiKey see `http://quikey.info/`.

3.1 Conceptual Data Structures

CDS is a lightweight top-level ontology about relations that naturally occur in common knowledge artefacts. It is designed to bridge the gap between unstructured content like informal notes and formal semantics like ontologies by allowing the use of vague semantics and by subsuming arbitrary relation types under more general ones. By that it is suitable for representing knowledge in various degrees of formalisation in a uniform fashion, allowing gradual elaboration. CDS serves two purposes here:

First, as a flexible semantic data model providing a set of crucial structural primitives that can be extended by the user. The core top-level relations in CDS e.g. are *order* (before, after, etc.), *hierarchy* (corresponding to item nesting in iMapping and subsuming semantic relations like *is_a* and *part_of*), *linking* (subsuming hyperlinks as well as any other freely specified relation carrying formal semantics or not) and *annotation* (subsuming free-form notes as well as tags and types).

Second, the java-based CDS-API serves as a back-end for both iMapping and QuiKey, providing basic querying and reasoning functionality. Furthermore, like that the data entered in either of these tools can also be accessed and edited by other CDS-based tools like the browser-based CDS workbench HKW[2].

For more information about CDS see [VH06].

4 Related Work

As a rather novel GUI approach for personal and semantic knowledge management, iMapping has a unique position and there is no directly comparable approach known to the authors. But, of course, there are many related fields of work which contributed ideas to the design and realization of iMapping:

Visual Mapping Techniques: are methods to graphically represent knowledge structures. Most of them were developed as paper-based techniques for outlining, brainstorming, learning facilitation or for eliciting knowledge structures. Most of these techniques are related to one of the following basic approaches:

Mind-Maps [BB96] provide an easy-to-understand, tree-like structure for outlining a topic or for sorting items. But they are not suitable for relational structures between items because they are constrained to hierarchical models.

Concept Maps [NG84], on the contrary, have a graph-based structure which emphasizes these relations. But they are not as easy to handle because explicitly specifying all these relations is too laborious e. g. for simple note taking or brain storming.

Spatial Hypertext is for viewing a self-contained hypertext from an overview perspective, by spatially arranging single pages. However, the Spatial Hypertext paradigm expressly abandons the concept of explicitly stating relations between objects and uses spatial positioning as the basic structure. To fuzzily relate two objects, they are simply placed near to each other, but maybe not quite as near as to a third object. This allows for so-called “constructive ambiguity” [SM99] and is an intuitive way to deal with vague relations and orders. While Spatial

[2] `http://semanticweb.org/wiki/HKW`

Hypertext in its pure form is not suitable to author formal knowledge structures needed for semantic knowledge management, the general approach may well be used as a User Interface basis. For an overview over visual mapping approaches and the cognitive psychology underlying them, see [Hal03].

Zooming User Interfaces: Pad and its successor Pad++, both developed in Maryland [3], were the seminal developments in this area. It has been used in various applications and also as a web browser capable of showing the viewed web pages and their link-structure from a bird eye's view. The work on Pad++ finally led to "Piccolo"[4], a toolkit in Java and .Net that supports the development of 2D structured graphics programs, in general, and, in particular, Zoomable User Interfaces like iMapping, whose current implementation is based on it.

Semantic Desktops *DeepaMehta* [RVH05] was one of the first semantic desktop systems. It lets the user freely specify semantic relations between typed information items on a topic maps basis. It provides a graph-based UI in a thin client. Once an item (or relation) has been specified (in a topic map), DeepaMehta keeps it in a background repository on the server, independent from whether they are still part of an actual topic map. This separation between the structural model and visual model is also adopted in iMapping: it allows multiple (visual) instances of an item to be used in different contexts or locations—much like hard links in a Unix file system. Also, through the use of CDS, the same structured data can be used in different tools.

Several iMapping Prototypes and also most of the current implementation has been developed as part of the *Social Semantic Desktop* project *"nepomuk"*[5]. For more information on semantic desktop systems in general, see [6].

5 Discussion and Future Work

We have used repeated formative evaluation [Scr91] to guide our design decisions. Further evaluations are being carried out. First evaluation results where an iMapping prototype has been tested against the market leading mind-mapping application "MindManager", seem to encourage the iMapping design, as interaction speeds did not differ significantly while users preferred the iMapping application. The exact results of this study are still in the publication process and can be obtained soon from the website mentioned below.

Limitations of the current iMapping implementation that are subject to ongoing and future work include the following: Currently, there are only text-items. In the future, it should be possible use external content in iMaps, such as pictures, documents and web pages. The layout of items in a map could be parsed to extract implicit structures, like sequence, and proximity. Formal semantic refinements of content structures could be unobtrusively proposed during informal interaction ("mark-up as you go").

Semantic knowledge management technologies might in the future be able to significantly increase interoperability. However, if knowledge management does

[3] `http://www.cs.umd.edu/hcil/pad++/`
[4] `http://www.cs.umd.edu/hcil/piccolo/`
[5] `http://nepomuk.semanticdesktop.org`
[6] `http://semanticdesktop.org`

not start on a personal level, providing immediate benefit to the single knowledge worker, it is questionable whether semantic knowledge management systems will ever spread very far. Focusing on user interaction and cognitive ergonomics will be an important point there, especially because in knowledge-intensive tasks, a user can barely afford to sacrifice much of his limited cognitive capacity[Mil56] to dealing with the tool instead with the content itself. iMapping-based user interfaces could be a step to make highly structured knowledge management in different levels of formalisation easy and intuitive.

For more information on iMapping and to download the latest implementation, see `http://imapping.info/`.

Acknowledgemants

This work has been partially funded by the EU projects NEPOMUK (IST-FP6-027705) and MATURE (IST-FP7-216356).

References

[BB96] Buzan, T. and Buzan, B.: *The Mind Map Book: How to Use Radiant Thinking to Maximize Your Brain's Untapped Potential* Plume, 1996.

[HA09] Haller, H. and Abecker, A.: Requirements for Diagrammatic Knowledge Mapping Techniques; In *Proceedings of the IKNOW Conference*, 2009.

[Hal03] Haller, H.: Mappingverfahren zur Wissensorganisation;, 2003 Knowledge Board Europe.

[Mil56] Miller, G. A.: The magical number seven, plus or minus two: Some limits on our capacity for processing information; *Psychological Review*, 63:81–97, 1956.

[NG84] Novak, J. D. and Gowin, D. B.: *Learning how to learn* Cambridge University Press, New York, 1984 For a crisp overview on "The Theory Underlying Concept Maps and How To Construct Them" by J.D. Novak, see `http://cmap.coginst.uwf.edu/info/`.

[RVH05] Richter, J., Völkel, M., and Haller, H.: Deepamehta - a semantic desktop; In Decker, S., Park, J., Quan, D., and Sauermann, L., editors, *Proceedings of the 1st Workshop on The Semantic Desktop. 4th International Semantic Web Conference (Galway, Ireland)*, volume 175. CEUR-WS, NOV 2005.

[Sch02] Schnotz, W.: *Information und Lernen mit Multimedia und Internet*, chapter Wissenserwerb mit Texten, Bildern und Diagrammen Beltz, PVU, Weinheim, third, completely revised edition, 2002.

[Scr91] Scriven, M.: Prose and cons about goal-free evaluation; *American Journal of Evaluation*, 12:55–62, 1991.

[SKSB09] Sauermann, L., Kiesel, M., Schumacher, K., and Bernardi, A.: Semantic Desktop; In Blumauer, A. and Pellegrini, T., editors, *Social Semantic Web*, pages 337–362. Springer-Verlag, X.media.press, 2009.

[SM99] Shipman, F. and Marshall, C.: Spatial hypertext: An alternative to navigational and semantic links.; *ACM Comp. Surveys*, 31(4), 1999.

[Ter02] Tergan, S.-O.: Hypertext und multimedia: Konzeption, lernmöglichkeiten, lernprobleme und perspektiven; In Issing, L. J. and Klimsa, P., editors, *Information und Lernen mit Multimedia und Internet*. Beltz, PVU, third, completely revised edition, 2002.

[VH06] Völkel, M. and Haller, H.: Conceptual data structures (cds) – towards an ontology for semi-formal articulation of personal knowledge; In *Proc. of the 14th International Conference on Conceptual Structures 2006*, Aalborg University - Denmark, JUL 2006.