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

WildDocs – Emerging Metainformation Support

Claus Atzenbeck
Aalborg University Esbjerg, Denmark
atzenbeck@cs.aaue.dk

Peter J. Nürnberg
Aalborg University Esbjerg, Denmark
pnuern@cs.aaue.dk

Abstract: In general, metainformation plays an important role in knowledge management for finding information. However, adding metainformation usually takes additional time. We work on *WildDocs*, a spatial-based knowledge management system. One of its main tasks is providing implicit metainformation that was added automatically during the structuring process without requiring a high cognitive load. This is supported by a detailed and less abstract structure model as well as by real world-based simulation of behavior. In this paper, we focus on emerging rotation, fixed sized documents, and binding mechanisms. We conclude that spatial structure techniques that consider physical limits and emerging structures may add important metainformation without significantly increased cognitive load on the user. We further point to our current and future implementation and the development of a special input device that supports efficient navigation on *WildDocs*'s space.

Key Words: knowledge management, graphical representation, information structuring, information browsing, emerging rotation, binding mechanisms, real world simulation, limitation, cognitive load, reduction of complexity, structural computing, *WildDocs*

Category: H.1.2, H.5.2

1 Introduction

Human resources are important for task and user oriented views in knowledge management. Humans use computers as tools to solve problems. A very important aspect today are user interfaces, because they provide the connection between machine and user. In knowledge management, this is the link between the computer application and the knowledge worker.

A good knowledge management tool can be any tool that supports the user's way of thinking by augmenting the human intellect [Engelbart, 1962]. Many different kinds of applications were created that try to achieve this. Already in 1945, Vannevar Bush described *Memex*, a machine that can be used to build trails of information that can be followed at any time later. The idea behind this was using a machine in a similar way to how humans think: building associations. Twenty years later, some researchers referred to Bush's idea by developing hypertext applications, systems that support creation and representation of associations among fragmented pieces of information.

In the late 1980s, people created tools – known as spatial hypertext applications [1] – that enabled knowledge workers to apply visual cues to information that is represented on a 2D space [Marshall et al., 1991]. They were successors of other knowledge management applications, such as *NoteCards* [Halasz, 1987], a hypertext tool that was able to represent its node–link structures graphically.

The step from explicit representation of relations, as the node–link model has, to implicit structures with visual and spatial cues on a 2D space, was an important one. The new approach followed a card metaphor. Pieces of information may be moved on a 2D space similar to cards that can be arranged on a table. So-called collection objects were implemented that allow users to collect pieces of information in sub-spaces [Marshall et al., 1994]. They follow a drawer metaphor in that different items can be placed to be grouped together.

Any graphical representation of knowledge is generated either by a machine or a human. In our research, we focus on the aforementioned, and tools that are used by humans to structure knowledge.

Recently, [Atzenbeck and Nürnberg, 2004b], we argued that there is a significant difference between spatial structure applications and paper-based real world structures in their abstraction levels. Paper-based structures show more sloppiness in their alignment as well as have more physical limitations than comparable computer applications. The questions that arise are:

1. Is the reduction or prevention of sloppiness in spatial based knowledge management applications positive due to reduced danger of cognitive overload, or do additional structure details carry useful information?

2. Is the lack of simulated physical limitations in computer applications positive, or does simulation of limits help knowledge workers directly or indirectly?

To answer those questions, we implement a test environment that lets us switch on and off certain aspects of emerging sloppiness as well as limitations that are well known from the real world. Our application is called *WildDocs* [2] and is based on *Piccolo*, a 2D zoomable graphical user interface [Bederson et al., 2004].

2 Aspects of Sloppiness

Most of the current spatial based knowledge management applications, e. g., the *Visual Knowledge Builder (VKB)* [Shipman et al., 2001] or *Tinderbox* [Bernstein, 2003], do not allow objects to be rotated. In fact, some applications even snap objects into a grid to get them aligned easily. This takes away rich spatial structure details and increases the level of abstraction.

[1] In the following, we use the term “spatial structure application” for any knowledge management tool that is used to structure knowledge spatially.

[2] *WildDocs* is named after “wild documents,” a term that refers to the ability to apply sloppiness to its objects while they are being moved.

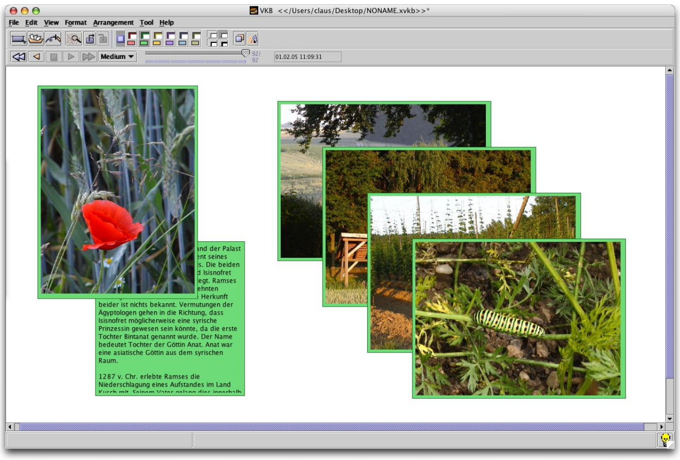

Figure 1: *VKB* Screenshot

For our discussion, we consider the following use case. John Doe, a knowledge worker at a large company, structures pieces of information on a 2D space provided by a spatial structure tool, such as *VKB* (see Fig. 1.) After using this tool for a long period of time, the application holds several thousand objects. He has created piles and uses collection objects to group similar documents. Due to the fact that the spatial structure tool aligns objects on a grid and does not have rotation implemented, piles as well as other visual structures look very similar to each other throughout the space.

For an upcoming presentation, John needs to gather certain information. Because he does not know exactly what he is looking for, he cannot use any information retrieval system effectively. Instead, he is *browsing* the spatial structure he has built. Due to the human ability to recognize and remember shapes, John remembers that the information he is looking for is part of a specific pile. The problem he faces now is that all piles on his 2D space look almost the same. This makes it hard to differentiate between the pile he has in mind and other piles. Even though aligned objects look more appealing to many people, the drawback of having a higher abstract level of structure is obvious.

How might a possible solution look that enables John to find the required information more quickly by having the shape and visual appearance of the appropriate pile in mind?

We have analyzed real world structures, created with paper. One significant difference to many computer applications is that paper cannot be moved without constantly changing its angle. This sloppiness creates piles and other spatial structure types that have individual characteristics applied.

One notion of those characteristics is metainformation. For example, paper that is structured in a hurry usually appears to be sloppier or messier than documents that are

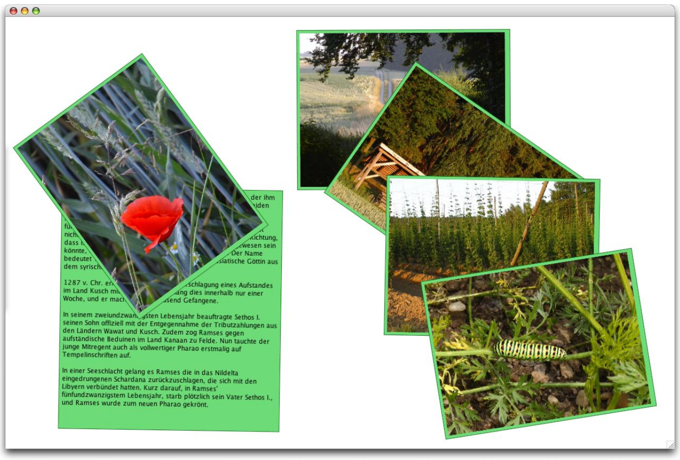

Figure 2: *WildDocs v1* Screenshot

carefully put together. Besides a sloppy offset, rotation plays an important role. A computer agent that aligns objects on a virtual desk deletes this implicit metainformation. Recent research has shown that this kind of information (e. g., changing the color of e-books according to the frequency of use to make them look older) is well recognized and understood by the user [Card et al., 2004].

WildDocs has two different types of rotation implemented. One kind is applied consciously and directly by the user, the other is applied automatically *while* the user is structuring objects by moving them around. As with paper on a table, the manipulated object rotates slightly. This rotation depends on speed of movement, position relative to the user, and a random factor. The algorithm is based on observations we have made with paper on a desk. Because emerging rotation happens without being explicitly called by the user, it does not cost any cognitive load.

Instead of *VKB*, we assume now that John was using a tool that applies sloppiness in offset and rotation automatically, such as *WildDocs* as shown in Fig. 2. John may remember that the paper he is looking for his presentation was collected in a hurry and therefore it is part of a wide spread and very messy looking pile. It is likely that he will find this pile faster, because it has graphically represented metainformation that John remembers.

3 Aspects of Limitation

Most computer application aim to avoid limitations, such as a small maximum number of objects that are possible inside a collection. Examples are collection objects in spatial structure tools that can hold a large number of objects including other collection objects, or the directory structures of file systems. One single directory can hold a very large number of files as well as sub-directories. In the real world, bindings are much more restricted. For instance, a binder cannot hold other binders and it can hold only a rather small number of single pages, caused by its limited space.

Another important difference between computer applications and paper organization is that a file on a computer is not equivalent to a single sheet of paper. Furthermore, there is no generic function that allows a user to guess the number of printed pages from the file's size in bytes. For example, a PDF file may consist of hundreds of single pages, but it may consist of less bytes than a high quality picture that is supposed to be printed on one single page.

On modern computer desktops, such as *Mac OS Finder*, *KDE*, *Gnome*, or *Microsoft Windows Desktop*, numerical representations of file sizes can be switched on, but there are no representations that show the size of a file according to its equivalence in printed pages. It can be assumed that the user remembers the size in pages rather than the size in bytes. Therefore, a representation of a document's size according to its number of printed pages is more useful for our proposed kind of knowledge management applications than a representation of the file size in bytes

Assume that John Doe is looking for a specific e-book that he has put on the 2D space some time ago. The only thing he remembers is that the book is quite thick and has around 800 pages. It might be relatively easy to find this book on a physical desk, because of its size. However, in John's *VKB*-like spatial structure application this book is represented as a single visual object that has scroll bars. John does not remember how big the visible section of this object is. This makes it very hard to find it.

In *WildDocs* we consider fixed size documents. These are predefined sizes a document can have (e. g., ISO A4 or ISO A5) that cannot be changed after the document is created. The document does not have scroll bars. Whenever there is more content that would fit on a fixed size object, a new instance is created. Even though this behavior is a limitation on the one hand, it automatically applies visual represented metainformation about the amount of data. A book in *WildDocs* can be found according to its thickness, equivalent to print copies.

Because this behavior forces the user to deal with a large number of single pages, we also introduce bindings in *WildDocs*. There are different types of bindings, such as binder or book. A book can be composed, but pages cannot be added to it after that, whereas the content of a binder may be changed frequently. The notion of bindings in *WildDocs* is significantly different compared to collection objects in spatial structure applications or directories in file systems. Bindings have a fairly low number of objects that they can hold. This number is taken from experiences based on the real world, e. g., a maximum number of 500 single pages per binder or 1000 single pages per book.

This limitation forces the user to restructure information constantly according to the binding's attributes. An important advantage is that the number of objects within a binding stays relatively small. This makes it easier and more efficient than browsing

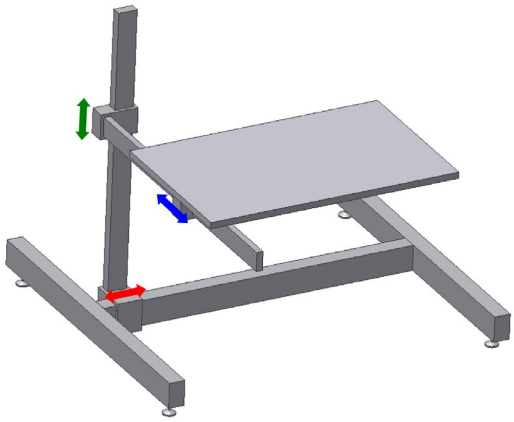

Figure 3: Early Sketch of *WildDocs*'s Input Device

through a large number of collection items that are represented at one single level. It also gives a better overview of how many pages a binding holds.

Assuming that John would have used *WildDocs*, he literally would have seen the book that has 800 pages because of its thickness. He can use the metainformation that he remembers and find the book based on that.

4 Conclusion and Future Work

We conclude that spatial structure techniques including the implementation of physical limits (such as the maximum number of documents within a binding) and emerging structures (such as changing rotation of objects during movement) may add important metainformation without significantly increased cognitive load on the user. Humans are familiar with paper as information carrier for thousands of years. Implementing some of the limitations may help knowledge workers to use computers more efficiently as knowledge management tools. Nevertheless, we do not claim that detailed implementation of a real world metaphor is positive per se. With *WildDocs*, as a first step, we aim to increase the amount of metainformation applied to spatial structures without having the user doing this explicitly. As a second step we evaluate whether implemented features are useful for increasing efficiency and effectiveness of knowledge workers.

Currently, *WildDocs v1* has fully implemented zooming and both kinds of rotation, emerging and purposeful, as well as partly implemented bindings. The aspects of fixed size documents as well as more details for bindings will be implemented next [Atzenbeck and Nürnberg, 2004b]. To be more flexible and to make use of the advantages of modern system architectures, we will port *WildDocs v2* to serve as structure services and frontend in *Construct* [Wiil et al., 2003], a structural computing environment [Nürnberg et al., 2004].

Our goal is to test to what extend *WildDocs*'s features support the user in faster information organization as well as retrieval. There will be different subject groups that have single features on or off. All will be given the same task. We will measure the time that it takes to solve the given problems. We expect to gain first evaluation results later this year. This approach follows our research in structure interoperability and rich structures [Atzenbeck and Nürnberg, 2004a], that we have presented at the I-KNOW'04 conference. The results about the effect of emerging metainformation will also affect other structure types, such as navigational, argumentation, or taxonomic structures.

We are also involved in developing a special input device for our application that supports quick and effective navigation and zooming and is easy to use. Figure 3 shows an early sketch of our development: A touch screen can be moved on a x and y axes to navigate on a 2D space. Movement on the z axis is used for zooming in and out. The goal is to provide a similar navigation and focus experience to the user that he/she is used to have with papers on a desk. In this respect, our research is related to tabletop workspaces that use tables as displays, e. g., *The Escritoire* [Ashdown, 2004], that projects digital documents on a real desk and lets them move around or annotate.

Our overall goal is to improve spatial-based knowledge management tools for knowledge workers. In combination with modern IR systems and semi-intelligent agents, the user will benefit from rich metainformation that does not require him or her to spend any time to create them.

References

[Ashdown, 2004] Ashdown, M. S. D. (2004). Personal projected displays. Technical Report 585, University of Cambridge Computer Laboratory.

[Atzenbeck and Nürnberg, 2004a] Atzenbeck, C. and Nürnberg, P. J. (2004a). Approaching structure interoperability. In *Procedings of I-KNOW '04. 4th International Conference on Knowledge Management*, J.UCS Conference Proceedings, pages 269–278.

[Atzenbeck and Nürnberg, 2004b] Atzenbeck, C. and Nürnberg, P. J. (2004b). Looking beyond computer applications: Investigating rich structures. Proceedings of the MIS'04 Symposium, Salzburg, Austria (in print).

[Bederson et al., 2004] Bederson, B. B., Grosjean, J., and Meyer, J. (2004). Toolkit design for interactive structured graphics. *IEEE Transactions on Software Engineering*, 30(8):535–546.

[Bernstein, 2003] Bernstein, M. (2003). Collage, composites, construction. In *Proceedings of the 14th ACM Conference on Hypertext and Hypermedia*, pages 122–123. ACM Press.

[Card et al., 2004] Card, S. K., Hong, L., Mackinlay, J. D., and Chi, E. H. (2004). 3book: a scalable 3d virtual book. In *Extended abstracts of the 2004 conference on Human factors and computing systems*, pages 1095–1098. ACM Press.

[Engelbart, 1962] Engelbart, D. C. (1962). Augmenting human intellect: A conceptual framework. Summary Report AFOSR-3233, Standford Research Institute.

[Halasz, 1987] Halasz, F. G. (1987). Reflections on notecards: seven issues for the next generation of hypermedia systems. In *Proceeding of the ACM Conference on Hypertext*, pages 345–365. ACM Press.

[Marshall et al., 1991] Marshall, C. C., Halasz, F. G., Rogers, R. A., and Janssen, W. C. (1991). Aquanet: a hypertext tool to hold your knowledge in place. In *Proceedings of the 3rd Annual ACM Conference on Hypertext*, pages 261–275. ACM, ACM Press.

[Marshall et al., 1994] Marshall, C. C., Shipman, F. M., and Coombs, J. H. (1994). Viki: spatial hypertext supporting emergent structure. In *Proceedings of the 1994 ACM European Conference on Hypermedia Technology*, pages 13–23. ACM Press.

[Nürnberg et al., 2004] Nürnberg, P. J., Wiil, U. K., and Hicks, D. L. (2004). Rethinking structural computing infrastructures. In *Proceedings of the 15th ACM Conference on Hypertext and Hypermedia*, pages 239–246. ACM Press.

[Shipman et al., 2001] Shipman, F., Airhart, R., Hsieh, H., Maloor, P., Moore, J. M., and Shah, D. (2001). Visual and spatial communication and task organization using the Visual Knowledge Builder. In *Proceedings of the 2001 International ACM SIGGROUP Conference on Supporting Group Work*, pages 260–269. ACM Press.

[Wiil et al., 2003] Wiil, U. K., Tata, S., and Hicks, D. L. (2003). Cooperation services in the Construct structural computing environment. *Journal of Network and Computer Applications*, 26(1):115–137.