

# Lookmark: A 2.5D Web Information Visualization System

Christian Breiteneder, Horst Eidenberger<sup>1</sup>, Geert Fiedler and Markus Raab<sup>2</sup>

Vienna University of Technology, Institute of Software Technology and Interactive Systems,  
Favoritenstrasse 9-11 – 188/2, A-1040 Vienna, Austria

<sup>1</sup>{breiteneder, eidenberger}@ims.tuwien.ac.at

<sup>2</sup>{a9505913, a9505911}@unet.univie.ac.at

**Abstract.** Lookmarks are thumbnails of existing web pages that can be arranged within a 2.5-dimensional space, just like documents can be arranged on a normal desk. The Lookmark system offers the user the opportunity of taking individual web pages and structuring and managing them within a 2.5-dimensional space. The paper discusses relevant related work in the field of Information Visualization and Interaction, design issues and the implementation of Lookmark in Java and Java3D.

## 1 Introduction

Due to the enormous growth of the World Wide Web more and more users have access to a steadily increasing number of documents. All developers of the various web browsers offer web document management solutions based on tree/directory structures. Examples are the Favorites of Microsoft Internet Explorer or the Bookmarks used by Netscape Navigator. The clear advantage of such a solution is the easy integration into browsers and the simple handling (the user just uses the mouse to pull the link over the designated folder). A further criterion is the low demand on hardware. However, the fact that such a system is not always conducive to finding documents that have thus been marked, is an important reason for the development of alternative forms of representation. A tree structure presented in text format can easily seem overloaded and confusing.

A better solution is to address the inherent human ability to think spatially. Just like a user would normally arrange files and folders on his real desk, he should be able to arrange electronic documents and web pages on his electronic desk. In the same context these electronic documents (e.g. an image of a website) should be used physically, and not only its description or links. Documents that are related in content can then be managed in spatial groups, whereby the depth of arrangement displays the hierarchical position of individual documents.

Lookmark, the visualization of bookmarks, is a prototype implemented in the project 'Bhutan – Fortress of the Gods', a virtual exhibition on the religion, history, art and culture of the kingdom of Bhutan [2].

The rest of this paper is organized as follows: Section 2 points out relevant related work, Section 3 describes the idea and design concepts behind Lookmark and Section 4 is dedicated to implementation issues.

## **2 Related work**

### **2.1 Motivation of Information Visualization**

The authors of [3] argue that the growth of the internet, the computerization of business and defense and the deployment of data warehouses have created a widespread need and emerging appreciation for Information Visualization (IV) techniques. The media of visual computing and display are quite new, and we do not understand well their advantages and disadvantages. For the authors, the key problem is to discover new visual metaphors and understand what analytical tasks they support [3]. Generally, IV aims at the 2D or 3D arrangement of information that lacks inherent 2D or 3D semantics [5]. The goal of spatial IV methods is to offload part of the burden of conscious information processing to the human perceptual system [6].

The motivation behind IV is that visualization provides an interface between two powerful information processing systems: the human mind and the modern computer [3, 5]. It allows real 'browsing in information'. Documents have content and history that can both be visualized [3]. Apart from that, humans gain more joy from using visual methods. This is relevant, because effectiveness (of retrieval), ease of use and joy are significantly correlated with each other [6].

IV systems can be symbolic (information items are represented by text) or diagrammatic (information items are represented by images or graphics, [8]). The major reasons for representation of information items by images are that images can transport more information and are cognitively direct: seeing is – in comparison to reading – a non-abstract process [3, 8]. On the other hand, text represents information more clearly and images are to some extent culturally dependent [3]. Practical arguments against symbolic representation are that in the average only about 100 data items can be displayed at one time [5]. Visualization offers more capacity, is faster and leads to better results ([5, 8] based on usability evaluations).

### **2.2 Information Visualization methods**

Spatial IV methods can be split in two groups: 2D and 3D methods. State-of-the-art 2D IV methods include:

- SeeSoft. This system represents the features of source code (e.g. line length, last modification, frequency of use) in diagrammatic form. Each line of code is displayed as one row (line). Colored line segments indicate features (e.g. red lines for recently modified code [3]).
- Tree map visualizes hierarchical information in diagrammatic and symbolic form. A 2D display – similar to a map – allows to fill in available space [3].

Prominent 3D IV approaches are:

- Cone Tree (developed by XEROX PARC). The Cone Tree was one of the first attempts to represent information symbolically in a conic tree structure [3].

- WebBook, Web Forager and DataMountain. These systems allow the diagrammatic organization and manipulation of web pages and other documents. Web Forager embeds WebBook and other objects in a hierarchical workspace. DataMountain is a generalization of the IV concept of WebForager [3, 7].

Subsequently, we point out the major advantages of 3D IV systems over 2D methods. 3D views take advantage of human spatial memory [7], allow displaying more information without incurring additional cognitive load because of pre-attentive processing of perspective views [7] and lead to better retrieval results in user studies (reaction time, number of incorrect retrievals, failed trials, [7]). Additionally, they allow displaying more information items because of scaling and a better global view [8]. Finally, there is experimental evidence that 3D displays enhance subjects' spatial performances [8]. The major open problem of 3D systems is the development of 3D user interaction techniques [7, 5].

### **3 Lookmark design**

This Section gives an overview on the Lookmark design: we will explain the idea behind Lookmark, the user interface layout and the interaction methods.

#### **3.1 Idea**

Traditionally, annotated web-links are stored in symbolic tree structures. In this paper, we use the term 'Bookmark' for this concept. As pointed out above, this method has some serious drawbacks: (1) Tree-structured graphs are limited and easily cluttered, tend to become confusing and often unusable [3]. (2) Symbolic approaches lead to worse retrieval results than diagrammatic representations.

Lookmark is basically a diagrammatic 3D IV approach for websites. The new idea in Lookmark is that we give up one dimension of movement resp. interaction in 3D space to gain advantage in ease of use. Information items are represented as thumbnails and are always displayed in parallel to the screen (image plane). The thumbnails are positioned on the X-Z-plane. Movement and interaction are restricted to the horizontal (left-right) and depth (front-back) dimension. Because this method uses a perspective 3D view and 2D interaction, we call it 2.5 dimensional (2.5D). The limited movement and interaction has two advantages: movement is less confusing than in 3D spaces and can be easily mapped to 2D interaction metaphors (e.g. mouse).

In the existing implementation, Lookmark offers the user the opportunity of taking each of the individual pages of the virtual exhibition and then structuring and managing them within a three-dimensional space, just like documents can be arranged on a normal desk. The name Lookmark should more than hint at the fact that here use is made of a visual filing system. The user decides on the selection of the web pages, how they will be arranged in space and how they are to be grouped together. Grouping describes a content-based linking that can be freely named. The item's hierarchical order is reflected by its designated spatial depth.

#### **3.2 User interface design**

The user interface (see Figure 1 for a screenshot) consists of two panels (the 2.5D

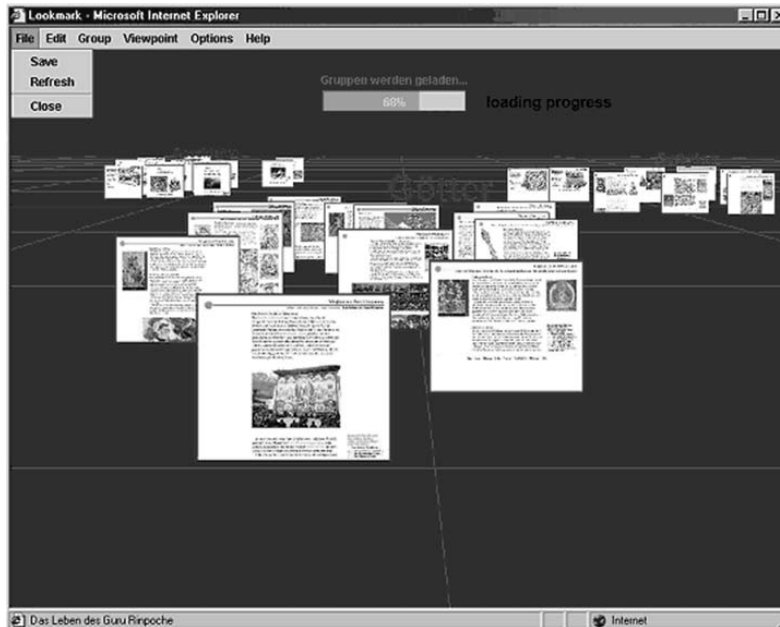


Figure 1. Lookmark user interface screenshot. The user interface consists of a 2D control panel that contains a menu and a media loading progress bar and a 2.5D visualization and interaction panel.

manipulation panel and a 2D control panel) and a status bar. The manipulation panel displays the thumbnail representations of web pages and represents the main work area. The chosen design and perspective shrinking of the thumbnails provide a 3D look-and-feel. The control panel consists of a pull down menu and a status bar for the loading progress of the thumbnails. The name of the current selected object is displayed in the status bar. For technical reasons (see Subsection 4.1 for details) the control panel must be extended as far downwards as the longest pull down menu. Because the background colors of both areas are the same, this trick is barely noticed.

After the user interface has been built up, the thumbnails, groups and views are loaded from a database and the progress of the download is displayed in the status bar, which disappears as soon as the download has finished. The download from the database is done in a separate thread, which allows the user to interact with the applet during the download. Groups have associated labels that are shown on the panel as text. The panels are independent from each other and interact through events.

### 3.3 User interaction

As pointed out above, one of the major advantages of a 2.5D view is the intuitive

handling with 2D interaction devices. In Lookmark both standard devices, mouse and keyboard, can be used. Mouse handling is limited to single click, double click and drag-and-drop. The meaning of mouse events depends on the context of the object that receives the event:

- Thumbnails. With a single mouse click one can add resp. delete the clicked thumbnail to resp. from the current selection. At the same time the name of the corresponding web page is displayed in the status bar of the user interface. By double clicking on a specific thumbnail, one can open the corresponding web page in the associated Web-browser. By using drag-and-drop one can move a thumbnail along the horizontal and depth dimension (X- and Z-axis).
- Groups of thumbnails. Thumbnails in a group are handled like single thumbnails. Groups of thumbnails can be moved simultaneously by dragging the group label.
- Selections. Dragging a thumbnail of a selection moves all selected thumbnails. A selection can be dissolved by clicking somewhere in the manipulation panel.

By using the menus a selection of thumbnails can either be grouped together, deleted or copied. When thumbnails are grouped, the user is required to enter a group name. This group label is then displayed in the manipulation area centered above the selected thumbnails. The color of the selection is replaced by that of the group. By using the menu any grouping can be dissolved. A thumbnail can be member of only one group at a time. However, copies are allowed to participate in other groups. When a thumbnail, selection or group is copied, an exact copy is created, which is displayed as a new object offset behind the selected object.

A fundamental part of Lookmark is the possibility to change and save the camera position. Each time Lookmark is initiated, it uses a default camera position (central view on the center of the given objects). The position of the camera can be changed with the cursor keys on the X- and Z-axis. For the user this movement of the camera causes the impression as if he were flying over the thumbnails. Additionally, the camera can also be rotated along the X- and Y-axis. With the use of the menu the present camera position can be saved and restored.

## **4 Implementation**

Subsequently, we describe the software development environment and the implementation of the visual part (based on scene graphs) of Lookmark.

### **4.1 Software development environment**

The Lookmark prototype, although developed for the special application of a virtual museum, should be open and extendible. It should be exclusively based on free software, equipped with a well-documented API and consist of independent components (panels). To satisfy these demands, we selected Java and the JavaSDK as the basis for the prototype. The control panel is based on Java Swing components. For reasons of compatibility and the support of 3D modeling with scene graphs, Java3D was chosen for the 2.5D panel implementation. The Java3D API packet represents a good base to quickly create a virtual world with simple interaction. Many aspects of

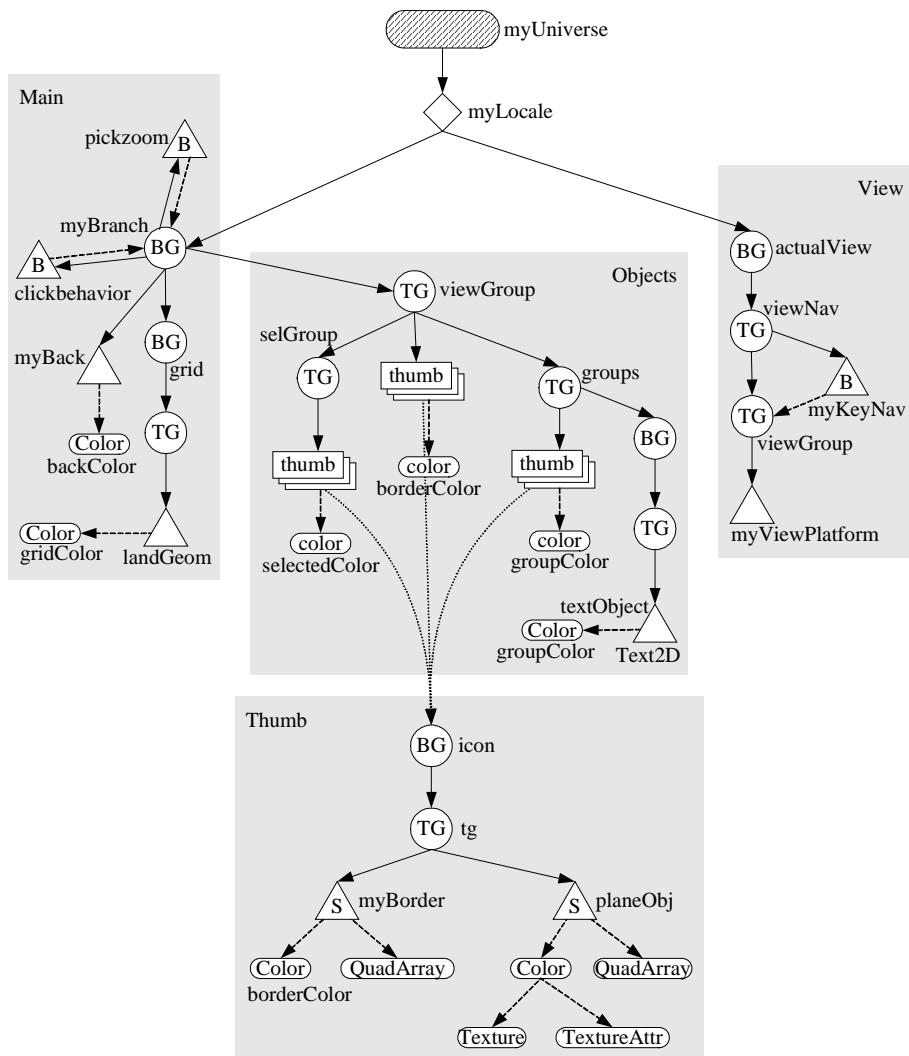


Figure 2. Lookmark scene graph. Thumb is a self-defined object that is used in the branches of transform group viewGroup.

3D computer graphics, such as the calculation of transformations, shadings, collision tests, textures etc. are taken over by Java3D. On the other hand, Java3D has the disadvantage that 3D areas are always set to the foreground automatically. Intersecting user interface components (like the entries in the pull down menus and the status bar

of the Lookmark control panel) would basically disappear behind the 3D area. This problem was solved with the trick described in Subsection 3.2. To handle the Lookmark thumbnails the development of special Java3D interaction classes was necessary.

In the Bhutan project, web pages consist of page building blocks stored in a database and are generated dynamically. PHP and the macro processor m4 are used for web page generation. The Lookmark thumbnails in the database are constructed with the same method. The database is accessed through JDBC and can be any relational database. We used Oracle 8i for the prototype. To keep the working performance as high as possible, small changes in the data (e.g. new groups, movements, etc.) are not saved immediately, but only when the Lookmark user interface is closed. The user can force saving through the control panel menu.

#### **4.2 Visual interaction modeling**

The modeling of visual display and user interaction in the 2.5D panel is based on Java3D scene graphs (see [4] for details). The scene graph for the whole application is depicted in Figure 2. The nodes are named after the resources used in the implementation classes.

*myUni* and *myLocale* are standard objects of the VirtualUniverse and Locale Classes that are elements of each Java3D world. The *myBranch* branch group node forms the root for all objects that are used in the 3D environment. For this node and all its children, both the behaviors *pickzoom* and *clickbehavior* apply. *pickzoom* is responsible for dragging visual objects. The handling of single and double mouse clicks are defined in *clickbehavior*. *myBack* is a background type leaf node, which uses the *backColor* attribute to define the background color of the 3D environment. Another branch group node (*grid*) is the base node for the panel grid. This allows the grid to be made visible or invisible easily. The form of the grid is defined in the leaf node *landGeom* (a line array node) and the color is defined in the color attribute *gridColor*.

In the *viewGroup* layer of the scene graph the differences between the three types of visual objects (thumbnails, groups and selections) are visualized. The transform group node *viewGroup* forms the basis for all these different representations. In this node a  $0.25rad$  rotation of the original coordinate system along the X-axis is implemented. This rotation leads to the perspective used in Lookmark. Thumbnails are directly inserted as children of *viewGroup*. The frame color of thumbnails is determined by the attribute *borderColor*. Selections are connected by the transform group *selGroup*. If properties of the selection change (e.g. movement) not all thumbnails in the selection have to be changed since only *selGroup* needs to be transformed. *selectionColor* defines the frame color of the selected thumbnails. The behavior of grouped thumbnails is very similar to selections. Unfortunately, it is necessary to predefine all the transform group nodes of the *groups*, since transform group nodes cannot be added to the scene during runtime. Additionally, each group

has a group label (represented as a leaf node *textObject*), which is a Text2D element.

*icon* is a self-defined object derived from a branch group object, and therefore allows the trouble-free inclusion or deletion of thumbnails in a scene during runtime. If thumbnails are moved, the transformation of the transform group node *tg* is adjusted. An *icon* consists of the leaf node *planeObj*, which is a Shape3D type node and consists of a simple square provided with the texture of the relevant image. The effect of a frame is created by drawing a slightly bigger square, which has the correct frame color (*borderColor*) assigned, behind the *icon* square. Since all thumbnails are descendants of the transform group node *viewGroup*, they have to be rotated by  $-0.25rad$  along the X-axis to make them appear parallel to the view plane.

Finally, the transform group nodes *viewNav* and *viewGroup* and the behavior node *myKeyNav* are used to display the parallel movement of the camera in relation to the grid plane. Java3D-specific rendering constructs were omitted from the scene graph.

## 5 Conclusion

Lookmark is a prototype for visual information organization originally developed for the virtual exhibition 'Bhutan – Fortress of the Gods'. Its purpose is to visualize web pages as thumbnails in a 3D environment that following the desktop metaphor and to allow the user quick and comfortable interaction and browsing. Lookmark is a novel approach that integrates many ideas from other spatial Information Visualization systems and goes one step further by using a 2.5D display and reducing interaction to two dimensions. Thus, Lookmark is easy-to-use and less confusing than 3D systems. Users that navigated with Lookmark in the Bhutan virtual exhibition gave the feedback that it allows creative interaction with the topic and is after all great joy.

## References

1. Breiteneder, C., Hitz, M., Platzer, H.: Bhutan – Fortress of Gods (2000). <http://www.bhutan.at>
2. Breiteneder, C., Hitz, M., Platzer, H.: A Reusable Software Framework for Authoring and Managing Web Exhibitions. In: Bearman, D., Trant J. (eds.): Museums an the Web Proceedings. Archives & Museum Informatics (2001)
3. Gershon, N., Eick, S.G., Card, S.: Information visualization. ACM interactions. 5/2 (1998) 9-15
4. Java3D website: <http://java.sun.com/products/java-media/3D/index.html>
5. Keim, D.A.: Visual exploration of large data sets. ACM Comm. 44/8 (2001) 38-44
6. Modjeska, D., Waterworth, J.: Effects of desktop 3D world design on user navigation and search performance. In: Proceedings IEEE International Conference on Information Visualization. Salt Lake City UT (2000) 215-220
7. Robertson, G., Czerwinski, M., Larson, K.: Data Mountain: Using Spatial Memory for Document Management. In: Proceedings of ACM Symposium on User Interface Software and Technology. San Francisco CA (1997)
8. Tavanti, M., Lind, M.: 2D vs. 3D, implications on spatial memory. In: Proc. IEEE Symp. on Information Visualization. San Diego CA (2001) 139-145