Browsing Interface of Private Virtual Library PVL

6R−5

Mayumi Ueda, Tomohiro Yoshida, Shinichi Ueshima
Graduate School of Informatics, Kansai University

1 Introduction

We have been proposing Private Virtual Library PVL that utilizes information resources over the Internet from user's purposes and/or preferences[1]. In viewing Internet contents, user interface, as well as many other aspects, gives an opportunity for personalizing systems.

In this paper, we propose a *Virtual Book*, a 3D browsing interface, employing the metaphor of 'real book'. We believe we have the possibility to obtain *naturalness* in *reading* virtual data by virtue of being digital.

2 PVL Overview

PVL is a user-oriented system, realizing PVL paradigm as in Fig.1. The left circle gives functions given as PVL Paradigm, while the right circle shows the requirements on the software system. Technical features can be summarized as follows. PVL employs

- Mediator-Wrapper model and access agents for information gathering and handling,
- data warehousing model, which gives materialized views, for the realization of BookShelf, and
- client-side personalization mechanism of agent scoring by user interactive operations[2].

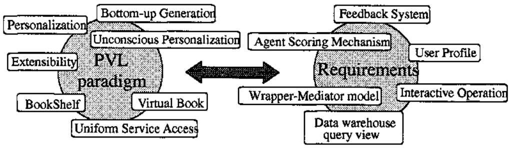


Figure 1: PVL Paradigm

In other words, PVL gives a private window open towards the Internet that stores a lots of books, articles, media that a user might want.

3 Virtual Book : 3D Browsing Interface

In this section, we describe the structure of Virtual Book, sensors to detect anomalies, and appearance/attitude control to realize real book metaphor.

3.1 Motivation

Our goal is to build-in the advantages and naturalness of real book into the virtual book. In viewing Internet contents with PC's, we have to use scroll-bars and follow hyper-links. In turn, we've realized it as turning over leaves, and as rotating the virtual book to change its attitude.

3.2 The Structure

Scenegraph

In Fig.2, we describe the hierarchical structure of Virtual Book. We've introduced the same concepts of 'Book','Leaf', and 'Page' as real book into virtual book.

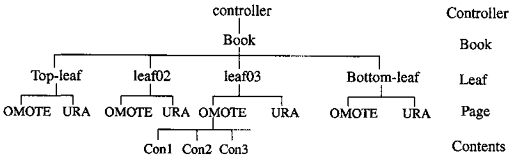


Figure 2: Architecture of Virtual Book

The correspondence between Book and Leaf is one-to-many, Leaf and Page is one-to-two. Top of a Leaf is called OMOTE, while bottom is URA. Page and Contents is one-to-many. Contents and Page is many-to-one, as con1,con2,con3, as in the figure. And Controller and Book is one-to-one.

Leaf

Each page of Virtual Book is defined by using *IndexedFaceSet*. OMOTE and URA are grouped to one as a Leaf. We first define a node of OMOTE, and reference it from the URA in a leaf. We put a small space between OMOTE and URA, which enables sensors of both sides work properly. We put angle information to each leaf.

Backbone(Spine) of Book

We make 'backbone' of virtual book, and let it work as a rotation shaft of leaves. By attaching all leaves to this rotation shaft, a user can turn pages by mouse operations, clicking and dragging.

Further, we put sensors for avoiding collision among pages. The sensors read angle information of each leaf, and kick out action that prevents the page from moving over. Without them, each leaf may pass through the other.

Multimedia Data Handling

Virtual Book uses various types of information, by virtue of being digital, such as (1)text type, (2)image, such as GIF and JPEG type, (3)animation, such as MPEG type, (4)sound, such as WAV and MIDI type.

Table 1: Mixing Virtual and Real Books

Virtual Book	Real Book
Searching capability	Ruffle the pages
Zoom-in/zoom-out	Meet unexpected favorites
Change font size and font type	Guess the amount of data from thickness
Multimedia contents and HyperLink Navigation	Read in free position and angle
Internet-based contents	Mobile

Hyperlink Navigation
We, of course, define Hyperlink by setting *Anchor node* between *Transform node* and *Shape node.*

3.3 Move and Rotation of Virtual Book

When we read a book, we usually adjust our position and/or angle in the posture. In the same way, when we use Virtual Book, we can read it in the posture, which is easy for reading.

We've realized three types of motions. We use TouchSenser for movement of these motions, and realize by computing the increase and decrease of coordinate values. We also use SphereSenser for rotation.

3.4 Features of Virtual Book

Our Virtual Book tries to take the advantages that real book might afford us. By mixing both advantages, we believe we have digital prestige and naturalness. We compare advantages in Tab.1 can be adopted.

In Fig.3, we show *turning-over* pages. Users ruffle pages naturally, as for real books, and might have a chance to encounter unexpected favorites.

The amount of data, which shows the amount of retrieved date from the Internet, can be thickness of the Virtual Book.

With more sophisticated interface like *wearables*, the user, who is inexperienced in operating a computer, can handle virtual books intuitively.

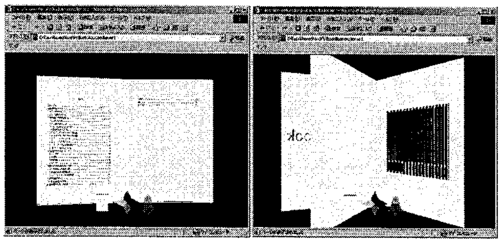
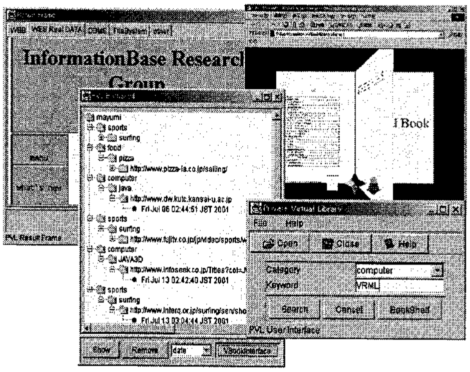


Figure 3: Turning-over pages of Virtual Book

4 Multiple GUI's in PVL

The goal of PVL is to use the Internet from personal viewpoints. We've implemented personalization mechanism in many aspects, including information


WEB
WEB Real DATA
DBMS
FileSystem
other
InformationBase Research Group
PVL Result Frame
mayumi
sports
surfing
food
pizza
http://www.pizza-la.co.jp/sailing/
computer
java
http://www.dw.kutc.kansai-u.ac.jp
Fri Jul 06 02:44:51 JST 2001
sports
surfing
http://www.fujitv.co.jp/jp/video/sports/w
computer
JAVA3D
http://www.infoseek.co.jp/Titles?col=
Fri Jul 13 02:42:40 JST 2001
sports
surfing
http://www.interq.or.jp/surfing/sen/sho
Fri Jul 13 03:04:44 JST 2001
Show
Remove
date
VbookInterface
Book
File
Help
Open
Close
Help
Search
Cancel
BookShelf
PVL User Interface

Category: computer
Keyword: VRML

Figure 4: Multiple Graphical User Interface in PVL

gathering by utilizing agents and their scores. In Virtual Book, we personalize the user interface according to user's purposes and preferences, by user's interactive operations.

In Fig.4, we show a screen shot of PVL with Virtual Book interface and standard GUI. When retrieving the Internet, we put retrieval condition, such as keywords and categories, into the box of the bottom right pane of the figure, as usual. The data, which is selected by user's interactive operations, can be saved and stored in the BookShelf (bottom left pane). In case a user selects one in the BookShelf, the system shows the real contents in the frame of PVL (upper left pane) or Virtual Book (upper right pane).

In our Virtual Book approach, we can extend its capabilities in the following ways. We can (1) guess the amount of retrieved data by changing thickness, (2)change distance from a user in accordance to their relevance, (3) organize and display books in the BookShelf in the order of preferences.

5 Concluding Remarks

In this paper, we've described a structure of Virtual Book, and discussed using it for PVL. We can handle the data intuitively without using scroll and hyperlink, by using Virtual Book.

As a part of our future work, we are discussing on interactive operations, such as methods of sealing bookmarks and post-it functions, etc.

References

[1] Ueda, M., Ueshima, S., Yajima, S., Nakamura, T., Nakayama, M., and Nakashima, E.,:*Bottom-up Generation of Private Virtual Library PVL over Advanced Information Network*, Proceedings of INFORMS-KORMS Seoul 2000(CD Proceedings)(2000)

[2] Ueda, M., Ueshima, S., :*Client-side Personalization Based on Agent Scoring in Private Virtual Library PVL"*,APSITT2001 (2001)