An Object-Oriented Hypermedia Application System

Julia C. Lee
Argonne National Laboratory
9700 S. Cass Ave. DIS/900
Argonne IL 60439-4832
lee@dis.anl.gov

Michele McCusker-Whiting
Argonne National Laboratory
9700 S. Cass Ave. DIS/900
Argonne IL 60439-4832
mccusker@dis.anl.gov

Craig E. Swietlik
Argonne National Laboratory
9700 S. Cass Ave. DIS/900
Argonne IL 60439-4832
swietlik@dis.anl.gov

ABSTRACT

We are to present a hypertext/hypermedia application system design. We show how the system confirms the Dexter Reference Model on the system level. We also introduce an object-oriented approach used in our design of the actual hypertext system application domain. Some future improvements are suggested.

Keywords: Hypermedia/hypertext, Dexter reference model, Object-oriented approach, application domain.

1. Introduction

Hypermedia or hypertext systems have become one of the major types of information systems to provide convenient access to different information sources. Mighty hypermedia sources merge into an information ocean. Hyperlinks bring users to voyage through the vast sea of information on the network. The diversity and flexibility of the hypermedia system...
through the vast sea of information on the network. The diversity and flexibility of the hypermedia system provides not only access information but also creates new information by allowing users to compose their own information from existing "pieces".

Research is very active in hypermedia/hypertext area. Some of the research contributed to reference model for hypertext system [HAL94][HAR94][LEG94]. Some of the research contributed to design methodologies [ISA95][SCHW95]. Some studied the design process of hypermedia system and contributed principles and/or pointed out human factors of the design process [GIN95][NAN95]. Others presenting their experience on design and implementing hypermedia system by applying a specific model or methodology [GRO94][BMY95][GRO94][GRO942]. Specific issues such as evaluation issue, distributing and collaborative hypermedia system are studied by some researchers [FUR95][GAR95][SCH95]. Pioneer researchers in hypermedia area presented some future characteristics that hypermedia systems can promise [ENG95][NEL95].

The two sides of some controversial issues such as flexibility-vs. structuralization are studied parallel by researchers. On one hand researchers try to generalize methodology for design hypermedia system or hypermedia application system and point out the importance of structuralization or organization "to avoid being drowned in this sea of information" [GIN95]. On the other hand, researchers try to seek alternative principles or models which will allow even more flexibility in a hypermedia system [MAR95]. These different research trends reflect the different rolls played by hyper media systems.

We have not yet had any general research results in hypermedia/hypertext area, but we have created a hypermedia system for storing, querying and composing information on environment, safety and health related documents. It is called the Facility Profile Information Management System (FPIMS). We would like to share our experience on the design and implementation of this system with interested readers.

In section 2 we give an overview of the system and compare the system with the Dexter Reference Model. Section 3 describes detailed design use Object-Oriented approach. Section 4 describe features to be implemented into the system in the future. Section 5 is the closing remark.

2. Overview of the system

Many researchers point out that the design of a hypertext management system is different from the design of the hypertext application system. The former involves the design or implementation of a general model for all the hypertext management systems (some times called a depository or a shell). The later involves the design of the specific domain of hypertext and its possible application forms.
Our system is a hypertext application system. We use off-the-shelf tools. However, since off-the-shelf tools can not provide all the features we needed for our system, we had to extend our design and implementation process of an application hypertext system to include the design and implementation of some features of "hypermedia management" system.

The structural overview of our system is depicted in Figure 1 below. We tried to confirm the design with the Dexter Reference model [HAL94]. Each layer is compared with the corresponding layers of Dexter Model. WPL (Window Personal Librarian) is an off-the-shelf information storage and retrieval engine that we integrated into our system.

According to Dexter Model "the fundamental concept in the run-time layer is the instantiation of a component. An instantiation is a presentation of the component to the user." Part 1 in the figure is the Window interface (GUI) laying above WPL which invokes the WPL. This is part of the "Run-time layer" in the Dexter Model. Part 2 is the part of the Window interface which are parallel to the WPL interface. This is also part of the "Run-time layer" in Dexter Model term. Part 3 is the interface provided by WPL. This is also part of the "Run-time layer" in the Dexter Model. The run-time layer features different types of graphical utilities such as windows, figures, dialog boxes, pull-down buttons, etc.

<table>
<thead>
<tr>
<th>Part 1</th>
<th>Part 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Part 3</td>
<td>Part 4</td>
</tr>
<tr>
<td>Part 5</td>
<td></td>
</tr>
<tr>
<td>Part 6</td>
<td></td>
</tr>
<tr>
<td>Part 7</td>
<td></td>
</tr>
<tr>
<td>Part 8</td>
<td></td>
</tr>
</tbody>
</table>

Part 1: Window Interface
Part 2: WPL Interface
Part 3: Window Interface
Part 4: WPL Retrieval Functions
Part 5: Customized Retrieval Functions
Part 6: Record Tags, Links
Part 7: Link Marks, WPL Search Engine

Part 8: FPIMS Text Records

**Figure 1** Layers of FPIMS Hypermedia System

Part 4 and Part 5 form the "Presentation specification" of Dexter Model. In real life Part 4 is provided by WPL internally and is hidden from both the user and the designer. Part 5 is the program codes which support the Window Interface in Part 1 and Part 2.

Part 3, Part 4 provides a variety of search and display methods such as searching using operators, natural language queries, concept search, field-restricted searching, feedback searching, etc. The search result can be displayed in different window and forms such as bar-charts, pie-charts, etc. Part 2 and Part 5 provide the specific link structures used to access the components in the system. Part 1, 2, 3, 4, 5 together provide a friendly interface which allows users to view the hypermedia information in both context and structural ways.

Part 6 corresponds to the "Storage Layer" of the Dexter Model. It contains links, fields, navigable table of contents, etc. The basic (or atomic) component is records. A record contains a piece of text data. Its logical (or semantic) correspondence could be a paragraph, a subsection or a section, etc. Related records are put into one storage unit called WPL database. The logical correspondence of WPL databases are books. Each record in a WPL database has a unique ID within that database. Records are linked by links to form subsections, sections, documents and books of documents. Each record has a set of attributes called fields which define the characteristics of the record such as title, document type, etc. Record structure and record attributes are searchable through links by the layers above. This part is created by the designer based on the requirements of the users. They are created by a set of tools developed locally. Note that the definition and/or existence of this layer is actually embedded in Part 8 below which is the actual text records.

Part 7 corresponds to the "Anchoring" layer in Dexter Model. According to Dexter Model anchoring is "a mechanism for specifying substructure within components" with which the links are spanned. "An anchor has two parts: an anchor id and an anchor value. The anchor value is an arbitrary value that specifies some location, region, item, or substructure within a component."[HAL94]. In our system the "anchoring" is just the way the system used to implement and to access the location where a link mark is placed. Since in our system the links are also used for access the image files and executable binary files, the "anchoring" layer (or concepts) also includes the mechanism used to access these different types of media. This part is provided by the WPL engine and is hidden from the users.
Part 8 is the records of the FPIMS data which corresponds to the "Within-Components Layer" of the Dexter Model. The dividing of the records from the original documents is also done by a set of locally develop tool set. As we mentioned in Part 6 above that records are organized into different WPL databases physically for ease of access. Each record has an ID within the database. The same ID number can be used for different records in different databases. This allows for large number of records to be put into the system using relatively small ID numbers.

As one can see that the system is basically conformed with the Dexter Reference Model although the realization of each layer is much more conceptual than physical. One thing seems worthwhile to be pointed out is that the Dexter Model put some emphasize on different link types with respect to their direction (e.g. FROM, TO, BIDIRECT, NONE) [HAL94]. In our system there is only one type of link. That is TO link. The major attributes of links in our system is resident address, point-to address and point-to media type (e.g. text, image or executable binary code, etc.). Bidirectional links can be implemented using two links. One resides on each directed address.

3. Object-oriented approach

Isakowitz et al [ISA95] introduced a hypermedia application data model called Relationship Management Data Model RMDM and a hypermedia design methodology - Relationship Management Methodology (RMM). In RMDM application information domain is generalized as two different types of primitives. E-R Domain Primitives including entities, attributes, one-one and one-many associative relationships. RMD Domain Primitives include slices which represents grouped attributes for some entities. Access links and structures are generalized as Access Primitives which includes unidirectional link, bidirectional link, grouping, etc. In Object-Oriented Hypermedia Design model (OOHDM) [SCHW95] information domain, navigation structure and interface design are all generalized using Object-oriented approach.

Our design is more close to the OOHDM approach. In the following subsections we will explain the specific design approaches using some examples.

3.1 The domain definition.

The contents of the hypermedia in our system are mainly documents. Documents are of different types and are from different sources. The sources of documents are different offices and institutes.

The major classes associated with documents are: The Book-case Class, The Book-shelf Class, The Book Class, The Document class, The record class and The WPL-database Class. The major classes associated with sites are: The Head-quarter Class, the Institution
Class, the Site class and The facility class. Some of these classes have hierarchical relation, some of these classes have parallel and interactive relations. Each class contains some specific attributes. The attributes of the superclasses are also the attributes of the subclass by inheritance characteristics of Object-Oriented approach. The relationships of the major classes are depicted in Figure 2.

As one can see that the Book-case, Book-shelf, Book, Document and Record Classes are in the order of class hierarchy, That is each later class has a "belong to" relationship with the previous class, Or in an "Object-Oriented words" the later one is a subclass of the previous one. The same relationship applies to the order of the Department, Institute, Site and facility classes. Other relations between classes are marked in the figure.

Figure 2 Classes and their relations of FPIMS

The "belong to" relation is a many-one relation, the "issued" and about "relation" are many-many relations. Note that Subject class could be one of the attribute of the Document class. We separate it as individual class not only because different subjects can be grouped into different fields but also because "Subject" itself has some attributes that need to be take into consideration. For example a subject of "environmental impact" could have attributes like "social", "historical", "health and safety", "wildlife", etc. different attributes each of them form a important part of the subject. Separating these
types of attributes into individual classes could provide flexibility for navigation design. On the other hand, there is no need to separate document title from the document class.

Note that the above classes and their relations include both logical concepts as well as physical concepts. For example, the classes WPL Database and Record are physical concepts classes. They represent the actual data structures implemented in the system. Since the actual implementation supports the logical structure or concepts, they interact (or have relations) with the classes which represent logical concepts.

The figure above contains only the major classes and relations, due to space limitation.

3.2 The navigation definition

When talking about navigation definitions most articles emphasize on link design and access. If we consider different searching capabilities as also part of the navigation design, our system contains two categories of navigation capabilities. One category is the context searching capabilities. The other is the structural searching capabilities.

The context searching capabilities are basically provided by the WPL software although we still need to provide application domain associated information to the system. This type of searching capabilities is based mainly on statistical results on the context of the hypertext database contents.

- Boolean Search uses operators such as AND, OR and ADJ. For example when user types in a line: radioactive AND waste AND level, any record in the current WPL database that contain the three terms would satisfy the condition of the search. If the user types radioactive ADJ waste, then all the records containing the two word term "radioactive waste" would satisfy the condition of later search.

- Natural Language Query (or Search) allows the users to use normal conversational syntax. This type of search is fast. However, while precise, it requires strict interpretation that can often exclude information that is relevant to the interests of the users.

- Concept Search allows the user to enter an area of interest. The system first generate a list of terms that are statistically related to the words in the description of the area (or in the query). Then it performs a conventional search using the original query words as well as the related terms.

This type of context search is suitable for browsing and for finding statistical type of information. It hardly gives structural type of information about the hypermedia database. Large number of records are retrieved into the system buffers. If user needs to find some precise information within the logical structure of the hypermedia, they need to look at all (or most) the records.
Structural search can lead the user quickly to the point of interesting through the links and/or fields (attribute values). The links in our system are of three types: text link, image link and command link. Text links can navigate the user within a record from one place to another place, between several records or searching a query within a WPL database. Image links invoke and display image files. command links can run an external application from our system. Field (or attribute) search targeted on the field values. One can issue a query on certain field(s) defined by the field name(s) (or tag(s)). One also search for the records whose field value equal to a specific value or fall into a range of the value domain. The link classes and their relations are depicted in Figure 3. above. The basic difference among the different link classes are the destination attribute. All the links are embedded [DAV95] in content records in the current implementation which is a inheritance of the off-the-shelf software.

From the application domain perspective, the link structure of our system is designed following the logical structures of the documents stored. Each document contains a Navigable Table of Contents record which contains not only the titles of all the sections and subsections of the document but also the links to each of the section and subsection. Each section and/or subsection contains links to its direct or indirect logical "parent" records. This type of link structure can provide a clean view of the document logical structure.
structure while the user follows the links through the hypermedia context.

3.3 The interface design

FPIMS resides on both PC LAN and WWW. The interface design of our system on PC LAN is based mainly on Microsoft Windows concepts. Each window can contain single screen, multiple screens and/or dialog boxes. Each screen contains one or more of the following components: MenuBar, ToolBar, HelpBar, Selection ListBox, StatusBar, Bitmap or Image, Buttons or Group of Buttons. The screen components on WWW include ToolBar, Option List, Bitmap or Image and Icons.

Some of the classes depicted in Figure 2. are actually perceptible interface objects. The Book-Case, Book-shelf and Book classes are interface metaphors. When they are mapped to the domain concepts, they are different collections of documents. The Headquarter, Institute, Site and facility are interface classes. When they are mapped to the domain concepts, they are just part of the attributes of a document. Note that we did put "spacial metaphors" [DIE95] as well as "Visual cues for Local and Global Coherence" [KAH95] into our interface and navigation design as pointed out.

One point worth making is that the interface objects are also navigation objects. The classification of documents into books, book-shelves, book-cases is not only for a perception purpose but also for the purpose of navigation. When a book-case is chosen, a collection of documents is chosen for navigation. The same situation applies to the Headquarter, Institute, etc. classes. Although they are for "user friendly interface", they are also point to a specific navigation direction. This raises a interesting question on the distinction of interface objects and navigation objects. It seems that in many cases the interface design includes part of the navigation design. For example, a menubar is a interface component, but it is also a navigation tool. It would be an interesting research topic on the principle distinction of interface components/design and the navigation components/design in hypermedia application systems. This subject is certainly not the focus of this article.

3.4 The Implementation environment

As we mentioned in Section 2, we integrate the off-shelf-software into our system. We also use off-the-shelf software such as Visual C++, etc. for developing of our system. However, many of the development task can not rely only on the off-the-shelf software packages, a set of tools are developed for the hypermedia application system implementation. These tools include link creation tools, attribute insertion tools, text reformat tools, record splitting tools, etc.

We are to integrate these tools into a hypermedia development environment. This effort is similar but not necessarily the same as the development environment suggested
by [SAL94] [RIV94] and [WII95]. We are not going to discuss this environment in detail in this article.

4. Improvements can be made in the future

The major purpose of the FPIMS hypermedia system is not for composition of new documents or articles. It is mainly an information source for readers. Therefore, our experience tells us more about structuralization rather than flexibility. We feel that a more structuralized presentation of hypermedia is more beneficial for the readers with a definite searching purpose. In the case of document hypermedia system, most readers are motivated by specific searching subjects. The improvements we thought about are more toward structuralization. It may, in fact, also provide more flexibilities.

One bold idea is to make the links addressable by programs. This will provide a large possibility of structuralization as well as flexibility. Link structures can be displayed to the users and/or to the developers. Links can be manipulated through programming to form different navigation structures or query forms. This needs to be done by the cooperation of the off-the-shelf software developers. In fact this could be a general idea for a hypermedia shell software. This idea has been mentioned briefly by [DAV95].

Another idea is to separate the attributes of a record from its content. Currently the attributes are in the same file as the content. This will provide easier manipulating of the attributes and hence more diversified searching tools.

One of the hypermedia research pioneers [ENG95] pointed out an important idea in future hypermedia system - view specification. "A structured, mixed-object hyperdocument may be displayed with a flexible choice of viewing options:...... Links may specify views so traversal retrieves the destination object with a specified presentation view (......). View specification becomes a natural and constantly employed part of a user's vocabulary." We copied this brilliant prediction here to share with our readers in case they have not read. View specification is a common relational database vocabulary, but it will take some thought and effort to apply it to the hypermedia systems.

Composition capability could be an important feature to implement into the system in the future. This feature will be aimed to those users who write similar documents and need to constantly refer to and/or excerpt from the existing documents.

The above ideas are in a high level description. The detailed concepts and/or design will take much more study and research.

5. Remarks
We have presented a hypermedia application system which conforms with the Dexter Model from system level of view. We adopted Object-Oriented approach in the application domain analysis and navigation design. We also described some features that can be implemented into the system in the future. As we pointed out in Section 1, this article is not meant to contribute some general ideas to the hypermedia research area. We merely describe the works we have done to implement this system. We feel that hypermedia research field is relatively new and very much "application-oriented". The experiences on application designs and implementations could inspire more general research topics and ideas. That is why we did not hesitate to present our limited experience here.

Besides implementing the features we have described in Section 4, we are also interested in working on some general research topics in this area such as general system model and application model and methodologies. As we pointed out in Section 4 some of the thoughts from our experience could be a general research topic. Much work has been done by the pioneers and researchers in these areas, but still when technology develops and application develops much more work also develops.

This work was supported by the U.S. Department of Energy under contract W-31-109-Eng-38.

References

[DAV95] H. Davis;
To Embed or Not to Embed?
Communications of the ACM

[DIE95] A. Dieberger & J. Bolter;
On the Design of Hyper"spaces";
Communications of the ACM;

[ENG95] D. Engelbart;
Toward Augmenting the Human Intellect and Boosting Our Collective IQ;
Communications of the ACM;

[FUR95] R. Furuta & P.D. Stotts;
Dynamic Hyperdocuments Authoring Replaces Programming;
Communications of the ACM; Page: 111-112; 1995

[GAR95] Garzotto, L. Mainetti & P. Paolini;
Hypermedia Design, Analysis, and Evaluation Issues;

[GIN95] A. Ginige, D.B. Lowe & J. Robertson;
Hypermedia Authoring;

Cooperative Hypermedia Systems A Dexter-based Architecture
Communications of the ACM; Page: 64-74; 1994.

[GRO942] K. Gronbeak & R.H. Trigg;
Design Issue for a Dexter-Based Hypermedia System

[GRO943] K. Gronbeak & R.H. Trigg;
Hypermedia System Design Applying the Dexter Model;

[HAL94] F. Halasz & M. Schwartz;
The Dexter Hypertext Reference Model;

[HAR94] L. Hardman, D.C.A. Bulterman & G. van Ross
The Amsterdam Hypermedia Model
Adding Time and Context to the Dexter Model;
Communications of the ACM; Page: 50-62; 1994

[ISA95] T. Isakowitz, E. Stohr & P. Balasubramnnm
RMM: A Methodology for Structured Hypermedia Design;
Communications of the ACM; Page: 34-44; 1995.

[KAH95] P. Kahn;
Visual Cues for Local and Global Coherence in the WWW;
Viewing Dexter with Open Eyes;
Communications of the ACM;

Spacial Hypertext
Designing for Change;
Communications of the ACM; Page: 88-97; 1995

[NAN95] J. Nanard & M. Nanard;
Hypertext Design Environment and the Hypertext Design Process;
Communications of the ACM;

[NEL95] T. Nelson;
The Heart of Connection
Hypermedia Unified by Transclusion;
Communications of the ACM; Page: 31-33; 1995.

[RIV94] E. Rivlin, R. Botafogo & B. Shneiderman;
Navigating in Hyperspace Designing A Structure-Based Toolbox;

[SAL94] G. Salton, J. Allan & C. Buckley;
Automatic Structuring and Retrieval of Large Text Files;
Communications of the ACM;

The StudySpace Project
Collaborative Hypermedia in Nomatic Computing Environments;

[SCHW95] D. Schwabe, G. Rossi;
The Object-Oriented Hypermedia Design Model;
Communications of the ACM;

[WII95] U. Kock Wiil;
Hyperform Rapid Prototyping of Hypermedia Services;
Communications of the ACM; Page: 109-111; 1995
DISCLAIMER

This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency thereof, nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.