

Structuring Facilities in Digital Libraries

Peter J. Nürnberg¹, Uffe K. Wiil², and John J. Leggett³

¹ Department of Computer Science, Aarhus University,
Ny Munkegade 116, Bldg 540, DK-8000 Århus C, Denmark
`pnuern@daimi.aau.dk`

² Department of Computer Science, Aalborg University Esbjerg,
Niels Bohrs Vej 8, DK-6700 Esbjerg, Denmark
`ukwiil@auc.auc.dk`

³ Center for the Study of Digital Libraries, Texas A&M University,
College Station, TX, 77843-3112, USA
`leggett@cSDL.tamu.edu`

Abstract. Digital libraries offer much promise for patrons and many challenges for system designers and implementers. One important issue that faces digital library system designers is the type of support provided to patrons for intellectual work. Although many researchers have noted the desirability of robust hypermedia structuring facilities in digital library systems, this research has tended to focus on navigational hypermedia (primarily used for associative storage and retrieval) only. Many other types of hypermedia, such as spatial, issue-based, and taxonomic, have been ignored. We briefly review some of our experiences with building digital library systems and discuss some of the lessons we learned from our initial prototypes. We then present a scenario of digital library work that illustrates many of the kinds of tasks we have observed users of our systems perform. We use this scenario to suggest a potential area of improvement for current hypermedia support in digital library systems and discuss some of our initial work in this area. Finally, we present some directions of future work and some concluding remarks.

1 Introduction

Libraries are much more than physical collections of data. They are institutions that filter and select information based on space, cost, and quality criteria; professionals that generate metadata about their collections and offer support to their patrons; social organizations that provide the basis for collaborative work; and, much more. The degree to which *digital libraries* can and will assume these and other roles has been and continues to be the subject of much research [8, 15, 16, 25, 29, 34]. Above all, it is clear that digital libraries are much more than electronic collections of data – they are virtual analogs to the complex organizations that are physical libraries [21, 28, 46].

We can call the software and hardware parts of a digital library a *digital library system*. In this paper, we examine one technology – hypermedia – that has received much attention in discussions about digital library systems [3, 7, 39,

43, 45, 47]. Despite the recognition that hypermedia has received to date in this area, current work has tended to focus heavily on only one particular type of hypermedia structure, namely, navigational. Navigational hypermedia structures are mostly suited for supporting associative storage and retrieval. Several other types of hypermedia structures suited to supporting other problem domains have been reported upon, e.g.: spatial [23], which focuses on supporting information analysis tasks; issue-based [27, 44], which focuses on supporting argumentation and capture of design rationale; and, taxonomic [33, 35, 36], which focuses on supporting classification tasks. All of these types of structure share certain basic notions, although each also has its own specialized and tailored abstractions. Despite the potential usefulness of these structures for digital library patrons, little work has been done to date on providing this kind of support within digital library systems. We argue that digital library system designers should extend the navigational hypermedia facilities in many digital library systems to include tailored structural abstractions such as those found in spatial, issue-based, taxonomic, and other kinds of hypermedia.

The remainder of this paper is structured as follows. Firstly, we consider the current state of hypermedia support in digital library systems. Secondly, we discuss some of our own past work in this area, and describe some of the lessons we have learned from our experiences. Thirdly, we present a scenario of digital library use, distilled from our experiences with users of our systems. This scenario illustrates a recurrent need for the kinds of flexible structure management provided by various kinds of hypermedia systems. Fourthly, we consider the notion of *structural computing*, or generalized hypermedia, and describe its implications for digital library system design. Finally, we present some of our work in applying structural computing principles to digital library system design, describe directions of future work, and provide some concluding remarks.

2 Hypermedia and Digital Libraries

In this section, we look at the role hypermedia currently plays in digital library system design. We first examine the type of hypermedia most often provided in a digital library setting and the problem domain it was designed to address. We then consider open hypermedia systems, which represents one of the most commonly accepted ways in which to provide this hypermedia functionality.

2.1 Traditional Hypermedia Problem Domain

The first hypermedia researcher is generally taken to be Vannevar Bush, who, in 1945, described the problem of information overload and one possible method for coping with it [4]. He proposed an information structuring system called the Memex (“memory extender”) that would allow its users to associate arbitrary pieces of information and navigate over these associations. Bush reasoned that since people used associations to store and retrieve information in and from

their own minds, a machine that provided this ability would be useful for storing and retrieving information in and from external sources. He realized that alongside organization schemes built on community conventions (such as alphabetization, Library of Congress classification, etc.), people often retrieved (or found) information by navigating through information spaces using arbitrary and idiosyncratic associations. The Memex would allow such idiosyncratic associative structures to be created, stored, and then retrieved at a later time during navigation of the information space. These developments later led to many *navigational hypermedia* systems, starting in the early 1960's (e.g., NLS [6]) and continuing through to today (e.g., the WWW [2]). All navigational hypermedia systems primarily focus on addressing the problem of *associative storage and retrieval*.

The case for powerful navigational hypermedia in digital library systems has been made by a number of researchers [3, 7, 16, 39, 43, 45]. Most of these researchers have noted the usefulness of associative structures in locating tasks performed by patrons. Such structures can help patrons find information they might otherwise have missed. They are also useful as a kind of metadata, since they are a form of implicit commentary or annotation on the associated data that can help patrons understand this data more fully. In most hypermedia systems designed for supporting digital libraries, structure is stored separately from data, allowing patrons to make private associations or selectively view the associations provided by the library.

2.2 Traditional Open Hypermedia Systems

One approach to providing navigational hypermedia within a digital library is to integrate a (traditional) open hypermedia system (OHS) into the digital library system. An OHS is one that provides navigational hypermedia facilities to an open set of (often third-party) clients orthogonal to their data storage mechanisms [5]. The advantages of providing navigational hypermedia through an open system as opposed to within a closed, monolithic one are well documented [26, 37]. One of these advantages is that an OHS does not modify the data over which it defines its structure – data may and often are managed by entities other than the OHS. Instead, the OHS publishes an interface that allows arbitrary clients that manipulate arbitrary data to make use of structuring facilities, including persistent structure storage. Note that the term *open* in this context has a quite specific meaning. A hypermedia system may be open in the traditional sense of the word if it allows arbitrary clients to interact with it through a published interface. However, it may not be an open *hypermedia* system if it does not provide structuring facilities over arbitrary data types handled by the client. An example of such a system is the WWW. Although a WWW server may interact with an open set of browsers, such browsers may only take advantage of the structure inherent in the markup of HTML files. Traditional WWW servers cannot define structures over arbitrary data types. For example, it is not possible for a WWW server to store a link between frames in two mpeg movies. Thus, the WWW is an open system, but not an OHS. Building a digital library system with integrated

open hypermedia support allows new clients that handle arbitrary data types to be added to the digital library system and still take advantage of existing hypermedia functionality. Some examples of traditional OHS's are Chimera [1], DHM [10], Microcosm [5], and SP3/HB3 [19]. All of these systems have integrated numerous clients, both third party and specially built. Additionally, many of these systems have been integrated (in different ways and to different degrees) with the WWW, providing external structure storage to WWW browsers in a way that does not modify the data (unlike traditional HTML markup). Some of these experiments have been reported in [9, 13]. Fig. 1 illustrates a common generic OHS architecture.

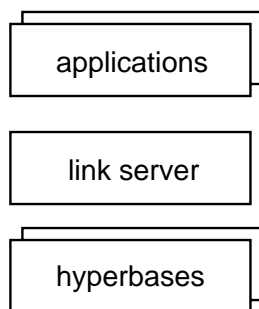


Fig. 1. A generic open hypermedia system architecture. The *link server* provides navigational hypermedia services to an open set of clients, called *applications*. In turn, these link servers use the services of an open set of persistent structure stores, called *hyperbases* (hypermedia databases [41, 42, 49]).

3 Experiences

In this section, we briefly review some of our own experiences with building digital library systems, especially focusing on the provision of hypermedia functionality within these systems. We begin by introducing the kinds of collections we have used to populate our digital library testbeds. We continue by providing an overview the SP3/HB3 environment and its applications to supporting our digital library efforts. We conclude by pointing out some of the key lessons we learned from our experience. More complete treatments of our prototypes and experiences to date are available in [32, 33, 47].

3.1 Botanical Collections

For several years, we have worked together with botanical taxonomists to build a digital library of herbarium collection data. This work has gone on in cooperation with collection managers and botanical taxonomists at the Texas A&M

University herbarium, which contains over 250,000 dried plant specimens, and with members of the Flora of Texas Consortium, who are seeking to distribute a collection of nearly 6000 taxonomic records defined over 1,000,000 specimens.

For reference, we present a short description of the botanical taxonomic problem. The object of botanical taxonomic classification is the taxonomy. A taxonomy consists of taxa, which themselves consist of other taxa or plant specimens. Taxa are composed in a hierarchic fashion. That is, the taxonomy itself may be viewed as a tree, with specimens at the leaves. Taxa at different levels in the tree have different names, such as family, genus, species, etc.

We have noticed a number of interesting characteristics of the botanical classification work that occurred over our digital library collections and over botanical collections in general. We mention three of these here, concentrating on the implications for any system that is designed to support such work.

1. *Different groups of taxonomists impart different characteristics to identical taxa.* Even if two groups of taxonomists agree on the definition of a taxon in terms of its subtaxa, supertaxon, and component specimens, they may disagree on other attributes of the taxon, such as its name. Such debates are viewed as important intellectual work by taxonomists. This implies the need for treating taxa as first-class entities in the system, allowing them to carry arbitrary attribute/value pairs, be versioned, locked for collaborative work sessions, etc.
2. *Different groups of taxonomists produce different taxonomies, even if the specimen set examined is identical.* Groups in which particular specialists work on a given taxon may show more detail in the expansion of that taxon, or different groups may use different measures of similarity when composing taxa, weighting various types of evidence differently. This implies the need for versioning (with respect to taxonomic authority) for the taxonomies generated over the given specimen data.
3. *The products of botanical classification work are often full taxonomies, not simply revisions to existing taxonomies.* Whether updates or new revisions, products are viewed as closed and well-defined entities, representing the opinion of a particular group at a specific time. These static snapshots of work belie the complexity of botanical classification work, however. New evidence, analysis methods, and interpretations are constantly being introduced. The practice of generating these static snapshots comes from the limitations of non-electronic media. However, such closed products are useful for consumption by others. Ideally, authors should view their collective product as open, while readers should be able to view some closed subset of this work. This implies the need for computation of structure that can generate such closed products dynamically.

3.2 SP3/HB3

SP3/HB3 [19] is an OHS developed at Texas A&M University in the mid 1990's. Roughly speaking, its architecture corresponds to the one shown in Fig. 1, with

the addition of another type of architectural entity called a metadata manager. Metadata managers are architectural peers of the link server, mediating interactions between applications and hyperbases. Like the link server, they also provide specialized and tailored abstractions to their clients. Unlike the link server, however, they derive their specialized abstractions from the data abstractions provided by the hyperbase. The link server, on the other hand, builds upon the structural abstractions of the hyperbase.

HB3 is the name of the hyperbase in the SP3/HB3 environment. It is implemented on top of a relational database management system (DBMS). The latest instance of HB3 was built on top of the Illustra DBMS [14], although previous versions resided on top of other DBMS's. HB3 provides its clients with generic data and navigational structural abstractions. These abstractions may be tagged with arbitrary attribute/value pairs. It also provides basic concurrency, access, and version control mechanisms for both data and structure objects.

The Link Services Manager (LSM) is the SP3/HB3 link server. It provides basic link authoring and browsing functionality to an open set of clients through a simple API. Applications wishing to use these services must register with an instance of the LSM and communicate (and respond to) various requests through a proprietary protocol.

Several metadata managers for SP3/HB3 have been built. Of particular importance to this discussion is the TaxMan, which is a metadata manager that serves botanical taxonomic abstractions described above such as taxon and specimen. Since these abstractions are derived from the generic data abstractions, they are first-class system entities that may carry arbitrary attributes, be versioned, locked, etc. The TaxMan also implements various computations over its abstractions, such as generating the closed sets of taxa as described above.

As with metadata managers, several applications have been implemented or integrated into SP3/HB3. One of these is the TaxEd, which allows the authoring and browsing of taxonomies, specimen records, etc. TaxEd uses the facilities of the TaxMan to provide its users with taxonomic abstractions. Additionally, TaxEd participates in the LSM link services protocol, allowing arbitrary associative linking among taxonomic objects handled by different TaxEd instance and even arbitrary objects handled by any other SP3 application.

3.3 Lessons

The three implications of botanical classification work practices for system design we discussed above evolved over time. Our first prototypes did not provide sufficient flexibility. As our understanding of the problem domain increased, TaxMan became more complex. It became increasingly apparent that TaxMan resembled the LSM in many ways. Both provided a type of first-class structure over a data set (taxa and links, respectively). Both used partitioning and grouping mechanisms to effect closure of their structure spaces (taxonomies and contexts). Both used computations over structure to generate new structures dynamically or to perform other tasks (such as finding all taxa without specimen records in a taxonomy or all dangling links in a context). Most importantly,

both shared “structure” specific problems, such as keeping consistent pointers to versioned data, providing structure-specific access control in the form of reference permissions, etc. However, there was very little reuse between TaxMan and LSM.

One reason for this is the location of the tailoring of abstractions for these two entities. As stated above, HB3 is implemented on top of a DBMS, which means that ultimately, HB3 views all persistent abstractions as database records. It provides a generic data abstraction derived from these records, which it then serves to clients such as the TaxMan. It also provides a closed set of specializations of this generic data abstraction in the form of navigational hypermedia abstractions, which are then used by the LSM. HB3 provides versioning, concurrency, and access control for both data and structure abstractions. Many of the policies for providing this advanced functionality for structure abstractions were geared to navigational structure, and thus not reusable in general for the taxonomic structure served by TaxMan.

In order to maximize the amount of reuse, we need to introduce the concept of “generic structure”. Generic structure is a specialization of generic data that can be further refined into abstractions suitable for other problem domains, such as associative storage and recall or classification. We discuss this idea further in Sec. 5.

4 Scenario of Digital Library Use

In this section, we consider a scenario of digital library use. We base this scenario on our experiences with the users of our digital library and hypermedia system prototypes, many of whom are botanical taxonomists. In this scenario, we describe another similar kind of classification work called linguistic reconstruction. In this task, linguists try to deduce the grammar and vocabulary of a language by examining its descendant languages. They often do this by comparing these descendants and interpolating the characteristics of a common ancestor. For example, if one examines the vocabularies of English, German, Danish, and Swedish, one notices a number of similarities. From these similarities, it is possible to construct the vocabulary of some common ancestor, which in this case might be called Proto-Germanic. In addition to using strictly linguistic data to perform reconstructions, historical and archeological data can often be helpful as well. In the example above, we can confidently posit that Proto-Germanic existed, since we can find archeological evidence for a single group of tribes that settled (many of) the regions in which English, German, Danish, and Swedish are spoken today. Although we can compare Danish and Chinese and reconstruct a common ancestor, this makes little sense based on historical evidence. Reconstructions can be applied to any sets of languages, even to reconstructed languages themselves. The resulting language “family tree” looks very much like the taxonomies discussed above built by botanical and other taxonomists.

Introduction. *Marge, Maggie, and Lisa are linguists. They are attempting to reconstruct Proto-Indo-European (PIE) from Latin, Greek,*

and Sanskrit. One key issue they need to resolve is how these languages are related. One possibility is that all three languages diverged from PIE at about the same time. Another possibility is that some pair of these languages (say, for example, Latin and Greek) have an intermediate common ancestor which should first be reconstructed and then compared to the remaining language (in this case, Sanskrit) to generate PIE. Marge, Maggie, and Lisa decide they can only resolve this issue by learning more about the period of history during which PIE is posited to have existed (c. 3000 BC).

Associative Storage and Retrieval. *They begin by consulting various historical and archeological texts. They can search not only the texts themselves, but also the associations between various facts built up by previous researchers. By browsing these associations, they are able to find more material and understand more connections than would have been possible in the same time using only the texts themselves. They annotate the historical and archeological (digital or digitized) texts with their observations and make additional associations, all within their own workspace. Because of the remoteness of this period of history, no clear picture emerges as to the history of the PIE and its speakers. Instead, they learn of many different hypotheses concerning this period of history.*

Information Analysis. *Marge, Maggie, and Lisa continue by considering the data they gathered from their initial investigations. At first, they do not have a clear overview or understanding of the problem space. They use the information analysis tools provided by the digital library system to organize their thinking on the relevant historical data. Over time, they develop an initial understanding of the problem space.*

Argumentation Support. *Once they understand enough of the problem space to begin their work, Marge, Maggie, and Lisa use on-line tools provided by their digital library system to argue the merits of the various positions presented in these works. The digital library system allows them to capture the issues and evidence presented during their analysis and helps them structure their reasoning. After analyzing the evidence, they reach the conclusion that one hypothesis is most likely, one is plausible, and the others may be discounted.*

Classification. *Instead of building a PIE language family tree from scratch on paper, Marge, Maggie, and Lisa download the PIE language family tree maintained by the library that represents the most current thinking on the history of Latin, Greek, and Sanskrit. They make various updates to this community taxonomy in their private workspace. They create another version of this workspace to illustrate how their results would differ under the historical hypothesis they found plausible but less likely.*

5 Structural Computing and Digital Libraries

In this section, we introduce the notion of structural computing, or generalized hypermedia, and its possible role in future digital library system design, analogous to our analysis of the connections between navigational hypermedia and current digital library system design above. We first consider a number of problem domains in addition to associative storage and retrieval that are mentioned in the scenario above, and review some of the different types of hypermedia abstractions and systems proposed to support tasks in these domains. We then analyze component-based open hypermedia systems in terms of their abilities to provide this wide range of abstractions to applications.

5.1 Additional Hypermedia Problem Domains

The first part of the scenario above describes the use of structures found in navigational hypermedia systems that are used for associative storage and retrieval tasks. As discussed above, the usefulness of supporting navigational structures in digital library systems is well-established. Our scenario continues by describing several other types of tasks carried out by the characters, each of which calls for the support of other types of structural abstractions. Nürnberg et al. [31] have used the term *structural computing* to denote a generalized approach to applying hypermedia structuring concepts to an open set of problem domains. Below, we look at three problem domains that characterize some of the other types of tasks presented in the scenario and describe the structural computing abstractions (or, in these cases, non-navigational hypermedia abstractions) that have been proposed to help users perform tasks in these domains.

Information Analysis and Spatial Hypermedia. Marshall and Shipman [23] note that *information analysts* faced with the task of organizing and understanding large amounts of data develop structures over this data over time. As their understanding of the information space changes, the structures they use to characterize the space also change. Systems designed for such analysts are required to support emerging, dynamic structures that avoid the problems associated with premature organization and formalization, as discussed, e.g., by Halasz [12] and Marshall et al. [24]. Marshall and Shipman have proposed *spatial hypermedia* to meet these requirements. Spatial hypermedia systems allow users to represent pieces of information as visual “icons”. Analysts can represent relationships among objects implicitly by varying certain various visual attributes (color, size, shape) of the icons and by arranging the icons in arbitrary ways in a 2.5 dimensional space. A spatial parser can then recognize the spatial patterns formed by these icons. Examples of such spatial structures might be lists of red rectangles that contain text or piles of blue ovals that contain images. Both the user and the system can use the structures recognized by this parser to support the task of analysis. For example, the system may recognize some particular type of structure as occurring frequently and conclude that it represents a meaningful abstraction to the analyst. It may then prompt the analyst to recognize this

type of structure formally, perhaps by naming it. This formal recognition may allow additional functionality to be provided, such as searches for instances of the structure in the space or replacement of the structure with a new visual symbol that may be “expanded” into its constituent parts.

The second part of the scenario above describes the use of more structures built by the patrons themselves over the data in the digital library. In this case, we could imagine a spatial hypermedia information analysis tool as acting as a kind of analog to large working table in a reading room of a physical library on which patrons can organize and arrange the material of the library, except that the materials may remain available to other patrons and their analysis can be supported by computer-based algorithms and tools.

Argumentation Support and Issue-Based Hypermedia. McCall et al. [27] describe community *argumentation support* systems in the context of capturing design rationale. In contrast to the information analysis domain described above, which must support personalization, here the focus is on a unified community understanding of an information space. Systems designed to support participants in a joint decision process or an argument must support simultaneous structure and data creation operations. Argumentation spaces consist of typed data nodes that represent issues to be discussed, positions with respect to issues, evidence that argues for or against a position, and other such entities. Many *issue-based hypermedia* systems provide tools that allow formal reasoning operations to be performed over the structures built, such as finding circular arguments [27]. Argumentation structures are like spatial ones in that they often serve the role of helping users develop an understanding of a space. However, unlike spatial structures, issue-based structures must be persistent and have first-class status.

In the third part of the scenario above, the patrons move their “reading room” material to a more formal system, in which structure is built up explicitly between items instead of implicitly through spatial arrangement. An issue-based hypermedia system would provide patrons not only with a tool for organizing library materials, but “value-added” formal reasoning tools as well. As above, this can be done without affecting the availability of these materials.

Classification and Taxonomic Hypermedia. We described one instance of the *classification* problem (namely, botanical classification) above. As is the case for information analysts, taxonomists must be able to build and express their own idiosyncratic understandings of the information space. This must be done in parallel with the development of a communal understanding among several people, as is the case in argumentation support systems. Systems designed to support classification tasks have many of the same requirements as spatial hypermedia systems. Additionally, they require the ability to tailor different views of the space to match the different understandings represented by different users and community conventions. Parunak [35,36] and Nürnberg et al. [32,33] have proposed *taxonomic hypermedia* to meet these requirements. Generic

taxonomic hypermedia is built over a set of samples, each containing, for example, text descriptions, images, and movies of a plant for botanical taxonomists, or grammar and vocabulary descriptions for linguists. These samples can be grouped into taxa, which themselves may be further grouped into supertaxa. Unlike spatial hypermedia structures, which are usually non-persistent and not treated as first-class objects by the system, taxa must be persistent and able to be named, described, tagged with various attributes, versioned. Taxonomic hypermedia systems must support the addition of new samples; the ability to comment on samples, taxa, or other comments; and, the ability to redefine the taxa to which samples and other taxa belong. Additionally, they must support the ability to change the “view” of the taxonomic space, which may entail re-parenting records, changing the visibility characteristics of certain comments, and/or other actions.

In the final part of the scenario, the patrons use a taxonomic hypermedia system to formulate and express their findings. This taxonomic system provides versioning and collaborative work support, both of which have been described as necessary components of digital library systems in general [11, 19, 38, 50] and of taxonomic systems in particular [32, 33]. Of course, the percentage of patrons performing linguistic reconstructions may not be very high, but classification is a very common task, even for people whose “normal” intellectual products are not expressed as taxonomies. Much intellectual work in many fields concerns dividing problem spaces, classifying problem instances, and describing prototypic examples of problems, solutions, or data.

Summary. Of course, many non-hypermedia systems have also been designed to address the work practices mentioned above. One advantage of addressing them within the context of hypermedia systems, however, is that they all share a need for flexible structuring mechanisms and policies, with the result that they all can benefit from the kinds of support that hyperbase systems and structural computing environments can provide, as we learned with our early TaxMan experiences described above. In the next section, we describe some of these kinds of infrastructure and how they can support systems like those mentioned above.

5.2 Component-Based Open Hypermedia Systems

A hypermedia system enables its users to build and manipulate structure over some set of data. If it is to support a wide range (or open set) of problem domains, however, it is not sufficient for the system to provide only some sufficiently “powerful” or “expressive” set of structural abstractions to developers of new applications. Instead, the system must allow some basic set of core abstractions to be tailored to abstractions well-suited to particular problem domains. Spatial hypermedia application designers, for example, should not be forced to develop their systems in an environment that provides only navigational structural abstractions such as “link” and “node”. Instead, they should be able to extend the basic structural abstractions provided by the environment into ones suitable for spatial hypermedia applications, such as “icon”, “space”, and “arrangement”.

In our previous work (and in any traditional OHS), facilities for managing navigational structure cannot be easily leveraged to manage non-navigational structure. In order to support new structural abstractions within an OHS environment, one is forced essentially to “reinvent” basic structure management facilities for each new set of abstractions added (i.e., for each new problem domain addressed). Although it is certainly possible to design and implement navigational, spatial, issue-based, taxonomic and other kinds of hypermedia systems independently, doing so requires much wasted effort.

A component-based OHS (CB-OHS) is essentially an OHS that contains an extensible set of structure servers [30]. Each structure server can be viewed independently as a kind of OHS, in that it provides its structuring facilities to an open set of clients (over arbitrary data types). However, the difference between a CB-OHS and a traditional OHS is that the former serves an open set of structure facilities. Contemporary CB-OHS’s provide generic structure management (structural computing) functionality that can be specialized to provide support for specific kinds of structure (such as navigational or spatial). This allows both re-use of common functionality and an extensible platform to which new kinds of tailored structure servers may be added. A generic CB-OHS architecture is shown in Fig. 2.

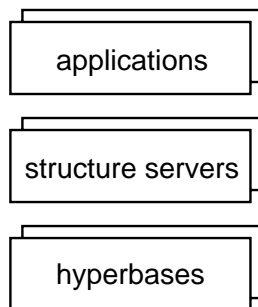


Fig. 2. A generic component-based open hypermedia system architecture. Note the replacement of the link server of a traditional OHS with a new open layer of structure servers.

Two examples of CB-OHS’s are HOSS [33] and HyperDisco [48]. As with the traditional OHS’s, these systems have been integrated with third party and specially built clients. Additionally, new structure servers have been implemented alongside traditional link servers. For example, HOSS has implemented servers of navigational, spatial, and taxonomic structure. In addition to both of these systems, the Open Hypermedia Systems Working Group (OHSWG) (see <http://www.csdl.tamu.edu/ohs/>) has begun definition of interface standards for different types of structure servers. Many of the most well-known OHS’s and both CB-OHS’s mentioned above are represented in the OHSWG by researchers who have committed to implementing or conforming to applicable

OHSWG component-based interfaces for providing structure facilities. Additionally, the OHSWG plans to deliver the first of these specifications as proposals to the World Wide Web Consortium (W3C) and the Internet Engineering Task Force (IETF) by the end of 1998.

6 Current Work

In this section, we discuss Construct, a new public domain CB-OHS that is natively compliant with OHSWG (de facto) standards, reference models, and reference architectures. Construct builds upon previous work done at Texas A&M University (e.g., [17, 19, 33, 40, 41]), Aalborg University (e.g., [48, 49, 51]), and Aarhus University (e.g., [9–11]). Various components of Construct are currently under development at Aarhus University and Aalborg University Esbjerg in the Coconut and Fasit projects of the Danish National Center for IT Research. In the remainder of this section, we first provide a brief overview of the project and its history. Then, we examine each architectural layer more closely.

6.1 Overview

Construct consists of a number of distributed hyperbases and structure servers. Applications (both legacy and custom made) access the different types of hypermedia services provided by the different types of structure servers. Both hyperbases (called hyperstores in Construct) and structure service components are comprised of different layers of functionality (see Fig. 3).

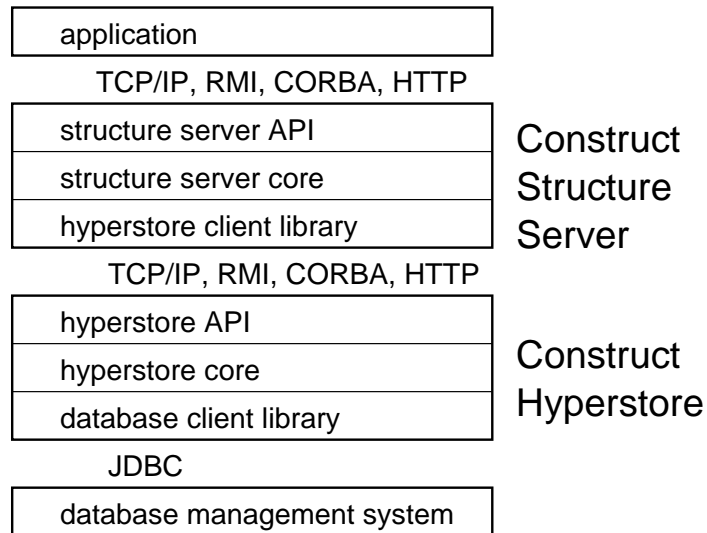


Fig. 3. Architectural layers of Construct.

Communication between components occurs in potentially many different ways. Structure service components can receive hypermedia requests from applications (or from encapsulating components acting on behalf of applications) encoded using different technologies and (de facto) standards (TCP/IP, Java RMI, CORBA, and HTTP). Likewise, communication between structure services and hyperstores can occur using the same technologies and (de facto) standards. As of May 1998, Construct supports communication using ASCII streams over TCP/IP, Java RMI, and tunnelling through HTTP. Support for CORBA and potentially other technologies and (de facto) standards will be provided in future versions.

6.2 Hyperstore

A hyperstore provides persistent structure (and optionally data) storage to other entities in the environment. Each hyperstore is implemented on top of a database management system (DBMS) that is accessed from the Database Client Library through JDBC drivers. The Construct Hyperstore server is currently running on top of Oracle. We are in the process of evaluating various public domain persistent stores with the intent of porting the hyperstore to such a store in early 1999. The Database Client Library implements basic storage and retrieval services to be used by the Hyperstore Core. The Hyperstore Core implements core hypermedia services (such as objects, attributes, behaviors and relations), and core collaboration services (such as nested transactions, concurrency control, notification control, access control, and version control) using the services of the underlying DBMS. The hyperstore services are provided through the hyperstore API.

6.3 Structure Servers

A structure service is layered in the same manner as the hyperstore. The services of the hyperstore are provided through a client library. The core provides the specific functionality of the structure service. That is, the core tailors and extends the generic hyperstore services to provide a new set of structure services supporting a specific domain (such as associative storage and retrieval or information analysis). The API provides the interface to the domain specific structure services. An application can access the hypermedia services of one or more domain specific structure services through standard APIs. We currently have implemented an OHSWG standards compliant navigational hypermedia structure server. We are also porting structure servers for spatial and taxonomic hypermedia from HOSS that will be modified to comply with the appropriate OHSWG standards when these standards are released (planned for end of 1998). It should be noted that the Construct and HOSS (i.e., the CB-OHS) implementations of TaxMan are considerably simpler than the previous SP3/HB3 (traditional OHS) implementation, because of the generic structure facilities available in the Construct and HOSS backends.

6.4 Applications

A number of applications have been integrated into or specially written for HOSS and HyperDisco and are now available for Construct. A complete description of these applications is available in the sources cited above. Some of the applications we have found most useful are the integrations of XEmacs and the Netscape and Microsoft WWW browsers. These are general purpose tools useful both in a digital library setting and elsewhere. With respect to applications more specifically aimed at digital libraries, we are porting the HOSS TaxEd client described above to Construct. We plan to continue to use the botanical collections over which TaxEd operates as testbeds for our work, but now concentrate on orchestrated, integrated delivery of multiple types of structural computing services alongside the taxonomic and navigational services reported in the source cited above.

7 Future Work

One obvious area for future work is to develop more structure servers that serve structural abstractions tailored to new hypermedia problem domains, as well as applications that take advantage of these new abstractions. As with all problem domains discussed above for which hypermedia systems and/or structure servers have been developed, this is a problem rooted in observing work practices. From a hypermedia perspective, this involves identifying the structural abstractions used by people performing work (in this case, in the digital library). We expect to find opportunities for defining new structure servers as our base of experience with digital library patrons grows. Clearly, at least structure servers that manage bibliographic metadata should be built. Although such metadata can be (and usually is) managed by non-hypermedia systems, developing metadata servers as instances of structure servers offers proven and integrated ways to handle integrity, versioning, concurrency, and similar problems familiar from other hypermedia domains.

Another interesting area for future work concerns hypermedia awareness on the part of the operating system. Though “hypermedia operating system” work is still immature, there have been some preliminary investigations reported in [33]. Hypermedia operating systems are potentially of interest to digital library system designers. The motivation for considering structure awareness at the operating system arises primarily from the observation of the changing nature of the way in which application-level computing environments are built. Many traditional operating system designs arose from observations of the computing environments of three and four decades ago. However, modern computing environments have much different characteristics. As just one example, consider that it is increasingly more common for code and data to be physically distributed and fetched on demand. In such an environment, does virtual locality provide a good measure of semantic locality? Estimating semantic locality is key to efficient memory management algorithms. Previous observations based on contiguous code and data spaces of centralized, monolithic processes do not apply as straightforwardly in modern computing environments. Consider that an

operating system that was aware of hypermedia structures between data items would have more available information to deduce semantic distances than simply information about proximity in the virtual address space. Nürnberg et al. [33] consider other possible advantages of structure awareness at the operating system that concern functionality other than memory management such as access control, network management, and the operating system user interface. Some related work has been carried out in the operating systems field concerning the use of semantic locality for file replication [18, 22], although this work does not use explicit hypermedia structure as its measure of locality. The presence of this work, however, does indicate room for improvement of the policies of current operating systems based on increasingly less valid characterizations of modern operating environments. If a digital library system is to provide comprehensive structuring facilities to its users, and if such structures represent common paths and patterns of access, it seems logical to investigate the possibility of using this information wherever possible in the system to provide a more efficient environment for the digital library patron.

8 Conclusions

Digital libraries will be more than simply electronic stores of information. They will be powerful resources for knowledge workers in the next century, acting as virtual places in which patrons will gather to use the digital library materials and interact with one another to carry out intellectual work. Critical to these digital libraries will be digital library systems that will enable users to work effectively and efficiently. In this paper, we have considered a scenario of work carried out in a digital library to demonstrate the usefulness of comprehensive structuring facilities provided by several different types of hypermedia. We then considered the implications for digital library system architectures, arguing that component-based open hypermedia systems that can provide these comprehensive facilities more effectively than traditional open hypermedia systems. We also described some of our current work in this area and directions for future work.

Digital library research has long been acknowledged as an inherently interdisciplinary undertaking. In this paper alone, in which we have focused only on hypermedia structuring facilities, we have highlighted the need to synthesize the results from fields as diverse as human-computer interaction, computer-supported collaborative work, hypermedia, and operating systems in order to address the issues surrounding the design and implementation digital library systems. We believe any successful attempt to address digital library systems will require the broad kind of interdisciplinary cooperation we have described here, and that component-based open hypermedia systems provide a good platform for developing and experimenting with more powerful, flexible, and useful digital library systems.

Acknowledgements

The work presented in this paper was partially funded by the Danish National Center for IT Research project 123 (Coconut).

References

- [1] Anderson, K. M., Taylor, R. N., and Whitehead, E. J. 1994. Chimera: Hypertext for heterogeneous software environments. *Proceedings of the European Conference on Hypertext '94* (Edinburgh, Scotland, Sep), 94-107.
- [2] Berners-Lee, T., Cailliau, R., Luotonen A., Nielsen, H. F., and Secret A. 1994. The World Wide Web. *Communications of the ACM* 37 (8) (Aug), 76-82.
- [3] Bieber, M., and Vitali, F. 1997. Toward support for hypermedia on the World Wide Web. *IEEE Computer* 30 (1) (Jan), 62-70.
- [4] Bush, V. 1945. As we may think. *Atlantic Monthly* 176 (1) (Jul), 101-108.
- [5] Davis, H., Hall, W., Heath, I., Hill, G., and Wilkins, R. 1992. Towards an integrated information environment with open hypermedia systems. *Proceedings of the 1992 European Conference on Hypertext (ECHT '92)* (Milano, Italy, Dec), 181-190.
- [6] Engelbart, D. C. 1962. Augmenting human intellect: A conceptual framework. Stanford Research Institute Technical Report AFOSR-3223, Contract AF 49(638)-1024, (Oct) Palo Alto, CA.
- [7] Fox, E. A., Akscyn, R. M., Furuta, R. K., and Leggett, J. J. 1995. Introduction to special issue on digital libraries. *Communications of the ACM* 38 (4) (Apr), 23-28.
- [8] Graham, P. 1995. The digital research library: Tasks and commitments. *Proceedings of the Second Symposium on Digital Libraries (DL '95)* (Austin, TX, Jun), 49-56.
- [9] Grønbaek, K., Bouvin, N. O., and Sloth, L. 1997. Designing Dexter-based hypermedia services for the World Wide Web. *Proceedings of the Eighth International Hypertext Conference (Hypertext '97)* (Southampton, UK, Apr), 146-156.
- [10] Grønbaek, K., and Trigg, R. H. 1994. Design issues for a Dexter-based hypermedia system. *Communications of the ACM* 37 (2) (Feb), 40-49.
- [11] Grønbaek, K., Hem, J. A., Madsen, O. L., and Sloth, L. 1994. Designing Dexter-based cooperative hypermedia systems. *Communications of the ACM* 37 (2) (Feb), 64-74.
- [12] Halasz, F. 1988. Reflections on NoteCards: Seven issues for the next generation of hypermedia systems. *Communications of the ACM* 31 (7) (Jul), 836-852.
- [13] Hall, W., Carr, L., and De Roure, D. 1995. Linking the World Wide Web and Microcosm, Presented at the BCS workshop on New Directions in Software Development, (Wolverhampton, UK, Mar).
- [14] Illustra Information Technologies, Inc. 1995. *Illustra User's Guide*. Oakland, CA: Illustra Information Technologies, Inc.
- [15] Kacmar, C., Jue, D., Stage, D., and Koontz, C. 1995. Automatic creation and maintenance of an organizational spatial metadata and document digital library. *Proceedings of the Second Symposium on Digital Libraries (DL '95)* (Austin, TX, Jun), 97-105.
- [16] Kacmar, C., Hruska, S., Lacher, C., Jue, D., Koontz, C., Gluck, M., and Weibel, S. 1994. An architecture and operation model for a spatial digital library. *Proceedings of the Symposium on Digital Libraries (DL '94)* (College Station, TX, Jun), 156-162.
- [17] Kacmar, C. J., and Leggett, J. J. 1991. PROXHY: A Process-Oriented Extensible Hypertext Architecture. *ACM Transactions on Information Systems* 9 (4) (Oct), 399-419.
- [18] Kuenning, G. and Popek, G. 1997. Automated hoarding for mobile computers. In *Proceedings of the Sixteenth ACM Symposium on Operating System Principles (SOSP-16)* (Saint Malo, France, Oct).
- [19] Leggett, J. J., and Schnase, J. L. 1994. Viewing Dexter with open eyes. *Communications of the ACM* 37 (2) (Feb), 76-86.

- [20] Levy, D. M. 1995. Cataloging in the digital order. *Proceedings of the Second Symposium on Digital Libraries (DL '95)* (Austin, TX, Jun), 31-37.
- [21] Levy, D. M., and Marshall, C. C. 1994. What color was George Washington's white horse? *Proceedings of the Symposium on Digital Libraries (DL '94)* (College Station, TX, Jun), 163-169.
- [22] Matthews, J., Roselli, D., Costello, A., Wang, R., and Anderson, T. 1997. Improving the performance of log-structured file systems with adaptive methods. *Proceedings of the Sixteenth ACM Symposium on Operating System Principles (SOSP-16)* (Saint Malo, France, Oct).
- [23] Marshall, C. and Shipman, F. 1995. Spatial hypertext: designing for change. *Communications of the ACM* 38 (8), 88-97.
- [24] Marshall, C. C., Shipman, F. M., and Coombs, J. H. 1994a. VIKI: spatial hypertext supporting emergent structure. *Proceedings of the ECHT '94 European Conference on Hypermedia Technologies* (Edinburgh, Scotland, Sep), 13-23.
- [25] Marshall, C. C., Shipman, F. M., and McCall, R. J. 1994b. Putting digital libraries to work: Issues from experience with community memories. *Proceedings of the Symposium on Digital Libraries (DL '94)* (College Station, TX, Jun), 126-133.
- [26] Meyrowitz, N. 1989. The missing link: Why we're all doing hypertext wrong. *The Society of Text*, The MIT Press, 107-114.
- [27] McCall, R., Bennett, P., D'Oronzio, P., Ostwald, J., Shipman, F., and Wallace, N. 1990. PHIDIAS: Integrating CAD graphics into dynamic hypertext. *Proceedings of the First European Conference on Hypertext (ECHT '90)* (Versailles, France, Nov), 152-165.
- [28] McKnight, C. 1995. Digital library research at Loughborough: The Last fifteen years. *Proceedings of the Second Symposium on Digital Libraries (DL '95)* (Austin, TX, Jun), 65-70.
- [29] Miksa, F. L., and Doty, P. 1994. Intellectual realities and the digital library. *Proceedings of the Symposium on Digital Libraries (DL '94)* (College Station, TX, Jun), 1-5.
- [30] Nürnberg, P. J., Leggett, J. L., and Wiil, U. K. 1998. An agenda for open hypermedia research. To appear in *Proceedings of the Ninth ACM Conference on Hypertext (HT '98)* (Pittsburgh, PA, Jun).
- [31] Nürnberg, P. J., Leggett, J. J., Schneider, E. R. 1997. As we should have thought. *Proceedings of the Eighth ACM Conference on Hypertext (HT '97)* (Southampton, UK, Apr), 96-101.
- [32] Nürnberg, P. J., Schneider, E. R., and Leggett, J. J. 1996a. Designing digital libraries for the hyper-literate age. *Journal of Universal Computer Science* 2 (9).
- [33] Nürnberg, P. J., Leggett, J. J., Schneider, E. R., and Schnase, J. L. 1996b. HOSS: a new paradigm for computing. *Proceedings of the Seventh ACM Conference on Hypertext (HT '96)*, (Washington, DC, Mar), 194-202.
- [34] Nürnberg, P. J., Furuta, R. K., Leggett, J. J., Marshall, C. C., and Shipman, F. M. 1995. Digital libraries: Issues and architectures. *Proceedings of the Second Symposium on Digital Libraries (DL '95)*, (Austin, TX, Jun), 147-153.
- [35] Parunak, H. 1993. Hypercubes grow on trees (and other observations from the land of hypersets). *Proceedings of the Fifth ACM Conference on Hypertext (HT '93)* (Seattle, WA, Nov) 73-81.
- [36] Parunak, H. 1991. Don't link me in: Set based hypermedia for taxonomic reasoning. *Proceedings of the Third ACM Conference on Hypertext (HT '91)* (San Antonio, TX, Dec), 233-242.

- [37] Pearl, A. 1989. Sun's link service: A protocol for open linking. *Proceedings of the Second ACM Conference on Hypertext (Hypertext '89)* (Pittsburgh, PA, Nov), 137-146.
- [38] Pettengill, R., and Arango, G. 1995. Four lessons learned from managing World Wide Web digital libraries. *Proceedings of the Second Symposium on Digital Libraries (DL '95)* (Austin, TX, Jun), 177-180.
- [39] Schnase, J. L., Leggett, J. J., Metcalfe, T., Morin, N. R., Cunnius, E. L., Turner, J. S., Furuta, R. K., Ellis, L., Pilant, M. S., Ewing, R. E., Hassan, S. W., and Frisse, M. E. 1994a. The CoLib project: Enabling digital botany for the 21st century. *Proceedings of Symposium on Digital Libraries (DL '94)* (College Station, TX, Jun), 108-118.
- [40] Schnase, J. L., Leggett, J. J., Hicks, D. L., Nürnberg, P. J., and Sánchez, J. A. 1994b. Open architectures for integrated, hypermedia-based information systems. *Proceedings of HICCS '94* (Maui, HI, Jan).
- [41] Schnase, J. L., Leggett, J. J., Hicks, D. L., Nürnberg, P. J., and Sánchez, J. A. 1993. HB1: Design and Implementation of a Hyperbase Management system. *Electronic Publishing - Origination, Dissemination* 6 (1) (Mar), 35-63.
- [42] Schütt, H. A., and Streitz, N. A. 1990. HyperBase: A hypermedia engine based on a relational database management system. *Hypertext: Concepts, Systems, and Applications. Proceedings of the European Conference on Hypertext* (France, Nov.), A. Rizk, N. Streitz, and J. Andre, Eds. Cambridge University Press, Cambridge, UK, 95-108.
- [43] SIGLINK. 1995. Special Issue on Digital Libraries. C. Kacmar, Ed. *SIGLINK Newsletter* 4 (2) (Sep), 2-28.
- [44] Smolensky, P., Fox, B., King, R., and Lewis, C. 1988. Computer-aided reasoned discourse or, how to argue with a computer. *Cognitive Science and its Application for Human-Computer Interaction*, R. Guindon, Ed. Ablex, Norwood, NJ, 109-162.
- [45] Stotts, D., Smith, J., Dewan, P., Jeffay, K., Smith, F. D., Smith, D., Weiss, S., Coggins, J., and Oliver, W. 1994. A patterned injury digital library for collaborative forensic medicine. *Proceedings of the Symposium on Digital Libraries (DL '94)* (College Station, TX, Jun), 25-33.
- [46] Sugimoto, S., Gotou, S., Zhao, Y., Sakaguchi, T., and Tabata, K. 1995. Enhancing usability of network-based library information system - experimental studies of a user interface for OPAC and of a collaboration tool for library services. *Proceedings of the Second Symposium on Digital Libraries (DL '95)* (Austin, TX, Jun), 115-122.
- [47] Wiil, U. K. 1998. Evaluating HyperDisco as an infrastructure for digital libraries. *Proceedings of ACM Symposium on Applied Computing '98 (SAC '98)* (Atlanta, GA, Feb).
- [48] Wiil, U. K., and Leggett, J. J. 1997. Workspaces: The HyperDisco Approach to Internet Distribution. *Proceedings of the Eighth ACM Conference on Hypertext (HT '97)* (Southampton, UK, Apr), 13-23.
- [49] Wiil, U. K. 1993. Experiences with HyperBase: A Multiuser Hypertext Database. *SIGMOD RECORD* 22 (4) (Dec), 19-25.
- [50] Wiil, U. K., and Leggett, J. J. 1993. Concurrency control in collaborative hypertext systems. *Proceedings of the Fifth ACM Conference on Hypertext (HT '93)* (Seattle, WA, Nov), 14-24.
- [51] Wiil, U. K., and Leggett, J. J. 1992. Hyperform: Using Extensibility to Develop Dynamic, Open and Distributed Hypertext Systems. *Proceedings of the 1992 European Conference on Hypertext (EHT '92)* (Milan, Italy, Nov), 251-261.