Next Generation Hypermedia Authoring Systems

Athula Ginige, David Lowe
School of Electrical Engineering,
University of Technology, Sydney
E-mail: {athula,dbl}@ee.uts.edu.au

Abstract
At present the cost of producing large hypermedia information systems is very high and the process is
time consuming. Also the way related information is linked together is prone to errors resulting in
lower quality applications.

The process of creating hypermedia information systems is known as authoring. The way authoring is
performed today is analogous to how cars were produced before the introduction of mass production
techniques and assembly lines when every part had to be handcrafted. Similarly today many
authoring tools require the author to handcraft every screen and manually find and link related
information together. This makes these techniques unsuitable for production of large hypermedia
information systems. Thus we need a new generation of authoring tools that can support rapid
production of higher quality large hypermedia applications in a cost effective way.

To get a deeper understanding of the issues that need to be addressed when developing next
generation authoring systems we have developed two experimental systems; one for text (Hypermedia
Authoring Research Tool) and another for images (HyperImages).

Based on our experience we concluded that to develop higher quality hypermedia applications in a
commercially viable way we need to develop a structured authoring methodology and tools that can
provide intelligent assistance to the author during the authoring process. Also the process should be
such that we should be able to re-use relevant information that has been created before for some other
application, and readily maintain and update existing applications, to reduce the overall cost of
production and maintenance.

Evolution of Hypermedia Information
Systems
We develop information systems to organise and
obtain efficient access to information. The
information that we want to access can be related
to an organisation, about how to perform a task,
information archived in a library, someone’s view
point on a particular subject matter or information
associated with our thought processes. In some
cases we may be looking for an answer to a very
specific question, other times we may only have a
vague idea of the information that we are looking
for.

Information systems may be classified in different
ways depending on the intended use of the
information, and the way in which it is accessed
and manipulated. One useful classification is
based on the function provided by the information
system:

• Information systems that support organisational
activities. These include such systems as large
business-oriented database applications, and
treat the information solely as a set of data to be
manipulated.

• Information systems to retrieve specific isolated
information. These include such systems as
information archives and library catalogues,
and are focussed on information access which is
achieved through various indexing schemes.

• Information systems to augment the human
brain. These include systems which support
our ability to interpret, utilise and structure the
information in ways which are most meaningful
to us, such as interlinked annotations.

Early information systems supported only a subset
of these three functionalities. As information
systems grew in size and sophistication, it became
necessary to access and manipulate the information
in more complex fashions. We needed to develop systems which are flexible enough to support a broader range of functionalities. Pure database systems do not allow us to access the information in complex interactive fashions. Information retrieval systems do not permit complex interlinking of information.

### Hypermedia Information Systems

Human memory access information through association. For example we will associate a telephone number with a name of a person. With the explosion of information it became necessary to develop external systems to augment the human memory. Dr. Vannevar Bush, a former Director of the Office of Scientific Research and Development in USA, first describe a such a system in his article “As we May Think” which appeared in The Atlantic Monthly, July 1945 [Bush 1945 ]. He called his system Memex.

The first such system was developed by Douglas C. Engelbart at Stanford Research Institute in 1968. This system was called Augment. Theodor Holm Nelson coined the words “hypertext and hypermedia” in 1965. He also envisioned “a repository of everything that anybody has ever written and thereby of a truly universal hypertext system” which he called Xanadu [Nelson 80 ].

Hypermedia systems provide us with a technology which can be used to allow access to information, represented as various media forms (text, images, video, sound), in ways better suited to our natural modes of working. Hypermedia systems have the potential to be able to combine the benefits of a large variety of existing information systems - information retrieval, database systems, etc.

Hypermedia allows us to link information in complex fashions. We can easily browse through the interlinked concepts. This ability to browse information makes hypermedia a very powerful and effective way of presenting various information such as training material, education material, reference material , travel information, news, text books etc.

### Future Information Systems

Now we see the emergence of a new generation of information systems that draw upon the strengths of above mentioned systems. The rapid growth of the World Wide Web has established beyond doubt the power of browsing when searching for information. Various indexing schemes and search algorithms can be used to narrow down the information space to reduce the time required to find the required information. Object oriented database technologies provide an efficient way of managing different types of data.

As a result of the continuing evolution and improvement of information systems, the size, scale, and complexity will continue to rapidly expand. In order to support this evolution, and ensure that their quality, relevance and cost effectiveness increases accordingly, we need to develop tools which support their development. A major aspect of this is the development of new authoring methodologies to be used during the creation of this new generation of information systems.

### Basic Structure of Hypermedia Systems

The basic components of hypermedia systems can be considered to consist of nodes, key concepts, index terms, anchor points and links [Halasz 94].
These may be collectively referred to ashyper-components.

A consistent chunk of information is known as a node. For example this can be a paragraph within a text document or an image. Within these nodes one can define anchor points. These anchor points are connected to other nodes via links as shown in figure 1. A user can click on an anchor point and this will take the user to the node that anchor point is connected to via a link. This process is known as navigation. We can mark key concepts and assign index terms to nodes that will best describe the information contained within that node. When searching for information we can use these key concepts and / or index terms to access the information.

**Authoring Process**

A critical aspect of developing hypermedia applications is being able to identify the interlinking within the information, and structure it in such a way that enhances accessibility. This is a major part of the process which is referred to as authoring. The other being combining different media and designing the screen layout for enhanced presentation of information. Early authoring tools provided a mechanism to traverse from one piece of information to another. HyperCard [Goodman 87 ] and Hyperties [Shneider 89 ] are good examples of this type of information systems. The authoring systems that followed such as KMS [Akscyn 88 ] used a hierarchical structure to organise information. A good review of early authoring systems can be found in [Conklin 87 ].

When we look at how hypermedia systems have been authored so far we can identify three main approaches:

- Programming Language based approach
- Screen based approach
- Information centred approach

**Programming Language based approach**

Early hypermedia applications were coded from scratch using a programming language such as C or C++. This approach is very time consuming and is suitable only for very specific high performance applications.

**Screen based approach**

In this approach every screen is handcrafted and manually linked together to build a hypermedia application. High emphasis is placed on the screen layout. Many commercial authoring systems such as Icon Author, Authorware Professional, Macromind Director, Hypercard, Linkway Live use this approach to develop hypermedia applications.

Each screen tend to convey a single theme (a chunk of information) such as a particular topic or a sub topic. Some systems use fixed size screens while others allow scrolling. These chunks of information are known as nodes in hypertext, cards in NoteCard and HyperCard, frames in KMS, documents in Augment and Intermedia, or articles in Hyperties [Halasz 94 ]. These screens are linked together based on the structure that was developed during the design phase (structural links) and also if there are related information on different screens that are not linked by the structural links then additional links are created (associative links).

This approach is very time consuming as each screen has to be individually designed and is not suitable for rapid development of large systems.

**Information centred approach**

In this approach the required content is obtained from existing information if available. Otherwise it is created. This information is then structured and stored in a database. The structuring involves dividing information into nodes and identifying key concepts. It is important to preserve the original structure of the information as this reflects the viewpoint of the original author of that information. When searching for specific information the key concepts (or key words if it is text) that were identified during structuring of information can be used to locate relevant information nodes. The next step is to link related concepts. Then how the information should be presented on the computer screen is specified.

User needs a presentation system to view this information. The user can access this information based on the original structure or use the links that were created during the linking phase to access related concepts. If the presentation system has facilities for key word and full text search this will enhance the accessibility of information.

World Wide Web is an example of an information centred system. Authors first create text and other media and then structure these using a HTML (Hypertext Markup Language) editor. The links to other relevant documents are embedded into the document during this markup process. This document is then stored in a web server. Now users can view this document using a presentation system such as Mosaic or Netscape.
Multimedia Viewer from Microsoft is another example of an authoring and a presentation system based on information centred approach. As these systems separate the development of content from the screen layout design, the author can quickly develop the content and use a set of standard screen layouts to present this information to the user, through a presentation system.

Productivity Issues

The approach that should be taken for production of large hypermedia systems is very different from the approach taken for small productions. Handcrafting applications will be too time consuming and costly for large systems. There are many lessons one can learn from the software industry. Traditionally software was predominantly handcrafted (much as hypermedia applications are currently handcrafted). As software systems grew in scope and complexity this approach broke down. This led to a phenomenon known as the “software crisis”, with many systems failing to deliver the required performance or being completely unmaintainable. This was (and still is) addressed through the development of appropriate sophisticated software engineering techniques, methodologies and tools which support the development process [Gibbs 94].

Similarly we need a well defined process to develop large hypermedia information systems. The information centred approach is better suited for this as this approach separates the structuring, linking of information and design of screen layouts into different phases. Thus the method to perform each phase can be optimised.

Before we can use this approach for developing commercially viable hypermedia information systems we have to resolve a few issues.

- Structuring information is very time consuming; specially non textual information such as images, video and sound as these are still predominantly structured manually.

- Linking related information is still done based on location. The author should know the location of associated information to create a link. When developing large hypermedia systems the need to remember the locations becomes a major cognitive burden on the author effecting the quality and the productivity.

- We need to develop an efficient way to re-use and manage information once it has been created.

Structuring Information

Structuring involves breaking the information into nodes, identifying key concepts that best describe the information contained in that node and marking anchor points. Information within a node should contain only a single theme. The key concepts identified during structuring process will assist latter when searching for specific information.

Some researchers have looked at automating the structuring process [Furuta 89, Niblett 89, Rearick 91]. A common strategy for automating the identification of nodes within paper based information is to use existing structural information. Keying on markup languages such as SGML or undertaking layout analysis of the text [Furuta 89] are popular approaches. These approaches can be effective if the information is uniform in structure or has been previously marked up, but otherwise these approaches breaks down quickly. Also researchers have used information retrieval algorithms to produce key phrases and/or anchors. Although this is a step in the right direction, this approach introduces the problem of missing relevant anchors and key phrases plus creating key phrases and anchors which are not relevant (recall and precision) [Bernst 90].

These projects have demonstrated that if the information contains non-structured documents or a document set with divergent structures within it, total automation is not possible. Further we have a major problem if we try to extend these methods to non textual media such as images, video and sound as these do not have a well-modelled syntax.

Linking of Information

Probably the most challenging aspects of the linking process is the need to mentally manage all the existing nodes within a set of information. This problem of finding relevant destination nodes for a given concept is graphically shown in figures 2 and 3.

When the number of information nodes in the system is small the user can remember the information associated with a particular concept and create the links. All authoring tools commercially available at present support this type of manual linking.
When the information space is large (typically over 500 nodes) it becomes impossible for the author to remember information contained in all the nodes. If the system is developed by many authors this adds to the problem. Thus with increasing size the cognitive overload becomes unmanageable.

Experimental Systems

To get a deeper understanding of the structuring and linking issues when developing next generation authoring systems we have developed two experimental systems; one for text (Hypermedia Authoring Research Tool) and another for images (HyperImages).

Hypermedia Authoring Research Tool (HART)

The aim of this project was to develop an effective methodology to convert text to hypertext [Robert 94].

The ability to effectively migrate paper-based information into hypertext is crucial to its commercial success. The identification of hyper-components within paper-based information is the most expensive stage of the media-to-hypermedia transformation.

As discussed earlier automatic identification of hyper-components is not very successful. Rather than use this “computer centred” approach HART places the human at the centre of the conversion process, using the computer to augment this human’s expertise. By providing software tools to assist the human in this editorial process, we found that both the overall cost and time involved can be reduce to a commercially acceptable level. Additionally, we believe this approach will result in better hypermedia systems being developed.

HART takes a fundamentally different approach to the role of the human editor, compared to most other authoring systems, in the conversion process. The user is provided with a single GUI-style environment. Within this environment the user interactively tags hyper-components in a text editing window. The system provides both procedural guidance and support to assist the user in this hyper-component identification process (Intelligent Assistance). The human editor has the option of accepting or rejecting any of the suggestions made by the Intelligent Assistance component of HART. Additionally, the user is not constrained to select hyper-components from the set proposed by the computer. Any text within the document can be tagged as a node, key phrase, anchor, or index value. We are unaware of any other systems which provides this range of support in the same form.

The HART system has been used to develop three hypermedia information systems.

<table>
<thead>
<tr>
<th>Table 1 HART Project Details</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
</tr>
<tr>
<td>No. of Documents</td>
</tr>
<tr>
<td>No. of Nodes</td>
</tr>
<tr>
<td>No. of Key phrases</td>
</tr>
<tr>
<td>No. of Links</td>
</tr>
<tr>
<td>Create Nodes (% of total)</td>
</tr>
<tr>
<td>Create key phrases (% of total)</td>
</tr>
<tr>
<td>Create Links (% of total)</td>
</tr>
<tr>
<td>Time Total</td>
</tr>
<tr>
<td>Original Paper Size</td>
</tr>
</tbody>
</table>
A regulatory agency of the New South Wales Government, WorkCover, has used the system to develop a commercially available hypertext information system about safety-in-the-workplace related legislative information. The system contains 14 acts of legislation and 18 WorkCover commentary documents. Each piece of legislation has an associated commentary document.

The second project was a book containing conference proceedings, with 16 reference papers dealing with Tasmanian forest research.

The third project was the programmer’s documentation of the HART system. The documentation was being written in Microsoft’s Word for Windows. It was then processed by HART, converted into WinHelp format to be used by the programming staff.

Each system was developed by a separate group, independent of each other, following a common conversion strategy. All three projects found that marking up nodes and key-phrasing was best done in one pass. The second pass dealt with linking.

**HyperImage System**

The aim of this project was to consider the role of visual information in hypermedia systems. The work focused on the role that visual information can play as an *active* media in hypermedia applications, and how this can be achieved, especially with respect to the authoring process [Lowe 95].

Textual data has been widely used to construct and manipulate hyper-components in hypermedia systems. If we consider the use of textual data within hypermedia we can see that text can be stored, analysed, manipulated, generated synthetically, and extracted for use elsewhere. Essentially a textual node can be treated as consisting of discrete entities (words, sentences, paragraphs etc.) which obey a series of syntactic and semantic rules describing the interrelationships, and can hence be used as anchor points, index values, and key phrases within a hypermedia system.

The evolution of image data in hypermedia is still at a much lower level, and is predominantly treated as a passive media. Conceptually we should be able to treat image data in the same way as text; the image can be considered as a node, which contains discrete entities (objects and other image entities) which obey certain syntactic and semantic rules (which are admittedly much more complex than for text). In other words we can treat an image as a node, with objects and other entities within the image being used as anchors, key phrases, and index values.

The main problem however lies in identifying these image entities and the associated rules, and then interpreting these. When this is achieved, image data can become an active media as powerful (and in many cases, more so) than textual data. Identifying entities and rules within text is relatively straightforward, being simply a matter of identifying words, and then sentences etc. To identify objects in images is significantly more difficult. The two most common approaches to date have both revolved around manually delineating the syntactic elements of the image. The first method involves masking of selected image areas [Edgar 92] and then tagging these masks (so that they can be used as key phrases, index values, and anchors). The second method uses a manually-added textual description of the image [Consta 91] which can then be processed along the same lines as normal textual hyper-components. Both of these methods are tiresome and cumbersome, especially for large image sets, as each image needs to be independently annotated.

If we are to circumvent these problems, then we need to be able to automate the process of identifying the syntactic elements of images. The most obvious method of achieving this is to use object recognition schemes. These schemes are still in their infancy, and at present do not have the robustness, accuracy and speed that would be necessary for general-purpose hypermedia applications. Despite these problems, the authors have developed a hypermedia application called “HyperImage” [Lowe 92, Lowe 90] which incorporates object recognition. This application allowed, in a restricted fashion, the image data to become truly active by integrating the recognition scheme into the application. The user could select objects within images, and these objects became anchor points to associated information elsewhere in the database. The author of the database was not required to explicitly identify the objects within each image. Instead a model of each object to be identified was developed and stored in a ‘dictionary’ (analogous to storing textual words in a dictionary) and these models were used to automatically identify objects in images.

Using the HyperImage system we developed a hypermedia database of grocery items. A sample screen is shown in figure 4. It was found that because of the nature of image data both the ease of authoring and the useability of the final application increased significantly. The *HyperImage* system
resulted in an application which had a significantly improved level of useability over systems without active image data [Lowe 95]. This active image data was made possible by automating the extraction of the syntactic components of the image data. This indicates that if hypermedia is to reach its potential then we need to ensure that the visual information is fully integrated into the hypermedia applications. This includes the ability for the visual information to provide interactivity. We have recognised that for this to occur manual authoring is impractical - we need to consider ways to avoid the need to handcraft these applications.

The HyperImage system illustrates another important issue. By using a model based coding and recognition scheme we are able to easily implement content-based retrieval of documents or nodes. In the same way in which we are able to ask for all nodes containing a particular piece of text, we could request all image nodes containing a specific object. Since the objects are represented as abstract models, we can directly use these models in content-based search and retrieval operations.

Several additional points are worth noting. Although the application which was developed had a high level of success, this was rather artificial. It certainly illustrated the appropriateness of using active visual information. However it should be recognised that the visual information within the application was limited to relatively simple objects. Every object was rectilinear, relatively simple in shape, and had a simple shading, texture, and colour. It was only because the objects were so simple, that the object recognition scheme had such success in correctly identifying objects. For an object recognition scheme to be effective in implementing active visual information in hypermedia it needs to satisfy at least two criteria. Firstly it must be robust, reliable, and consistent for a very wide range of applications and objects. Secondly, it must be very straightforward to expand the object database which it uses to identify objects. The object recognition scheme used for HyperImage satisfies neither of these criteria. Much research is occurring in the field of computer vision, and great success has been achieved in restricted application domains and for restricted image sets. Nevertheless, a general object recognition scheme which could handle the broad range of visual data present in hypermedia applications is likely to be a considerable distance off.

In the long term, object recognition will become increasingly important in hypermedia authoring and hypermedia applications (as has been

Figure 4: HyperImage application. In this particular example, the user has just clicked on a specific item in the image, which has subsequently been identified. This object will be used to locate and load additional relevant information.
foreshadowed by experiments such as HyperImage). In the shorter term however, object recognition would appear to be insufficiently mature, except in perhaps very isolated cases. Having accepted that object recognition is likely to be too impractical for use for general hypermedia applications, we need to consider possible alternatives.

Framework for Next Generation of Hypermedia Authoring Systems

Based on the above projects, we can identify the need for a properly structured methodology in the next generation of authoring systems. HART illustrated the productivity benefits obtainable through using even a simple set of tools which support an appropriate methodology in the authoring of textual data. The HyperImage system highlighted the importance of providing navigation through visual information and the need to provide mechanisms for support this navigation through assisting the author to identify hyper-components.

In both cases, significant benefits were obtained through introducing authoring methodologies which go beyond the traditional handcrafting of applications. The next stage of development is to develop and refine more sophisticated methodologies. These methodologies will provide us with appropriate techniques for ensuring high quality applications.

These methodologies would be facilitated through the development of two core elements: appropriate representations of information and tools to support these representations. This has been encapsulated in a model for interactive hypermedia authoring (Figure 5) - the MATILDA (Multimedia Authoring Through Intelligent Linking and Directed Assistance) model that we have developed. In this model, layers represent specific representations of the information, and arrows show the process of converting from one form to another.

The MATILDA model separates the information structure (Information Domain) from the way in which we use the information (Application Domain). This has several significant benefits. An important one is a given set of information can be used in a large number of different ways and contexts enabling easy re-useability of information.

This model maps directly to the development process, and hence it provides a structured approach to hypermedia authoring. The development process consist of:

Information Capture

The first step of the process in developing hypermedia applications is to capture the information from its original form (raw information: paper-based, raw video stream, etc.) into a digital form which can be stored, utilised and manipulated on computers (canonical information). In some situations the information may be generated directly in a digital form (e.g. computer graphics and text).

Information Conversion

During the authoring process we will be storing and manipulating the information in various ways. As mentioned above, the basic digital information may not necessarily be in a structure which is most conducive to these functions. We need to determine appropriate information representations (coded information), as well as techniques for
converting the information from its canonical form to its coded form. For example, with video data, the sheer volume of information means that data compression becomes important. We also need to perform various image processing operations to identify various hyper-components within the video stream. We therefore need a representation which supports these disparate requirements (something which is not the case with traditional representation schemes such as MPEG and DVI).

### Information Structuring

Information typically has a certain inherent structure - the structure implicit in the original information (e.g., a section of media may relate to a single concept, which may in turn contribute to a larger theme - for example, paragraphs and chapters in text). Access to, and manipulation of, the information will utilise this structure in various ways. In order to facilitate this process, we wish to make the structure explicit without enforcing a specific (and hence limiting) viewpoint of the information. This process is often referred to as "chunking", or information structuring, and in our model results in the structured information layer.

### Access Structuring

The next stage is to utilise the information in the development of specific applications. Hypermedia systems achieve this through layering additional structure on the information. This structure provides a specific application focus to the information, and supports appropriate interactivity. This process involves two phases. The first is identifying the viewpoint from which we wish to utilise the information - the information viewpoint layer (which will in turn influence decisions on which information is relevant and how it is to be used). The second phase is to identify the inter-relationships within the information which we wish to make explicit - the information linking layer - in order to support user navigation through the information. Often these two phases are combined (i.e., access structuring).

### Presentation Formatting

The presentation layer takes specific information, structured for a specific application, and presents in such a way as to maximise the useability of the application. Issues such as screen layout, general navigation facilities, etc. are addressed at this stage. The information is typically presented in a way which minimises the user disorientation, and improves the clarity of the information and its inter-relationships.

### Implementation of the Model

As described above, the two dominant issues which need to be resolved are representations and processes.

Firstly, we need to consider the data models used to represent the information in each of the layers of our overall model. The most critical element of this will be the structured information layer of the model. This layer provides the interface between the information domain and the application domain. We need to ensure that this interface is developed in such a way as to maximise both flexibility and support for information access. Once this has been defined we can develop suitable representations for the remaining layers which ensure consistency throughout the model.

The second area to be considered is tools which support the processes linking the layers of the model. In particular, we need to consider methods of performing the information structuring and the access structuring. As discussed previously, the quantity of data means that manual structuring techniques are too inefficient and costly. We need to consider methods of automating the process - providing intelligent assistance where possible for the processes. The information structuring process involves identifying the inherent structure within the media and representing this in an appropriate fashion (i.e., the structured information layer). We can use analysis tools suited to the various media to achieve this. For example, with image data we wish to identify the objects within images. We can use tools varying in sophistication from simple graphical interfaces which will enable us to manually mark the object boundaries and tag names to objects to full object recognition schemes.

### Conclusions

We need to ensure that we learn appropriate lessons from the above observations. Although we have been able to successfully develop hypermedia applications to date, these have been rather limited in scope and size. The techniques which we have been using to date will not work for the next generation of applications. Applications are going to grow significantly in scope and complexity. If we attempt to continue to use current tools and techniques (and in particular, handcrafting applications) then we will end up with applications which are poor quality, cost-inefficient, and very difficult to use, reuse, and maintain. In order for the next generation of hypermedia applications to reach their potential we need to develop and adopt the next generation of development tools.
Acknowledgments

The work on HART was done by John Robertson as part of his Ph.D research program. This project was funded by an Australian Postgraduate Award (Industry) together with Allette Systems (Australia) Pty. ltd.

HyperImage system evolved from David Lowe’s Ph.D. work and was funded by Telecom Australia (former OTC) and an Australian Postgraduate Award.
References


