Examples of Hypermedia Apps

[http://www.cs.vu.nl/~eliens/media/]

- macro-literary systems -- publishing, reading, criticism
- problem exploration tools -- authoring, outlining, programming
- browsing systems -- teaching, references, information
- general hypermedia technology -- authoring, browsing, collaboration
- embedded hypermedia -- CASE, decision support, catalogs

Main Questions

- What is hypermedia? [cf “traditional media”]
- What is a hypermedia application?
- What is a hypermedia model?

Hypermedia applications

- The goal of a hypermedia application: to support, using associative relationships, the carrying out of actions which result in the identification, effective utilisation and control of appropriate information [p.79]
- NB:
  - object is information, not data
  - information utilisation, not procurement [p.37,41]
Functional Characteristics of Hypermedia Apps (3.4)

- Concerns what the application does (cf. nonfunctional characteristics)
  - Presentation
  - Navigation
  - Querying
  - Document management
  - Customizability
  - Contextualization
  - Security

Functional Characteristics of Hypermedia Apps (3.4)

- Presentation
  - make use of available output devices: CRT, LCD, speakers, …
  - temporal media need synchronization
  - need to adapt to bandwidth constraints
    - progressive downloading of images, streaming of audio/video

Functional Characteristics of Hypermedia Apps (3.4)

- Navigation
  - provide a mechanism whereby available links are visible to user, whereby they can be followed
    - Web browsers: click on link anchor
    - Microcosm: upon selection of any combination of media elements, a set of links is presented
  - big picture visualization (information maps and overviews)
  - information trials (where the user has been and may want to return)

Functional Characteristics of Hypermedia Apps (3.4)

- Querying
  - augments browsing on the basis of associative links
  - user should be able to perform:
    - searches that make use of context
    - queries over non-textual media
Functional Characteristics of Hypermedia Apps (3.4)

- Document management
  - needed for application maintenance

- Customizability
  - flexibility to tailor the application to different users
  - by users or by designers (e.g., creation of novice and advanced profiles)
  - not only presentation, but links and link behaviour (and other aspects, too).

Functional Characteristics of Hypermedia Apps (3.4)

- Contextualization
  - during information-seeking task, context of information provides clues
    - e.g., is this piece of information appropriate for my task?

- Security
  - not everyone should have access to sensitive information
  - need to implement a mechanism to identify who should and should not have access; also to ensure only authorized users do

Nonfunctional Characteristics of Hypermedia Apps (3.4)

- Link completeness
  - underlinked: the links between related concepts are missing
  - overlinked: links become meaningless, navigation becomes totally unmanageable
  - dangling links: destination node is unavailable
  - No objective criteria for completeness
    - even experts disagree on appropriate links
Nonfunctional Characteristics of Hypermedia Apps (3.4)

• Content Validity
  – is material being presented actually correct?
  – definition of “correct” depends on context
    • opinion of factual articles
  – user needs to be able to recognize information for what it is

Nonfunctional Characteristics of Hypermedia Apps (3.4)

• Concept organization
  – appropriateness of node granularity (paragraphs? sentences? document sections?)
    • granularity of atomic components
    • cohesiveness of composite components

Nonfunctional Characteristics of Hypermedia Apps (3.4)

• Consistency
  – of presentation
  – of content
  – of interface
  – inconsistency creates cognitive burden
  – inconsistency can provide information to user
    • e.g., boundary between ACM digital library and outside articles (e.g., Blandford article)

Nonfunctional Characteristics of Hypermedia Apps (3.4)

• General software engineering principles
  – efficiency
  – maintainability
  – reuseability
  – reliability
  – robustness
  – testability, validation, verification
  – portability
  – social aspects (legal constraints, social conventions, ethics)
  – cost effectiveness
Data of Information (2.3)

• Data:
  – the artifacts being used to encode information [p.67]
  – artifacts which exist as a vehicle for conveying information [p.46]

• Information:
  – the interpretation of data within a context set by a priori knowledge and the current environment

Knowledge (2.3)

• Knowledge:
  – the base of personal information which is integrated in a fashion which allows it to be used in further interpretation and analysis of data

  – “What we are essentially doing [for interpretation of meaning] is identifying mappings from concepts [from our knowledge base] to the information that is captured by the data” p.48

Knowledge Media Design (2.3)

• Knowledge Media Design
  – design of media in order to support acquisition of knowledge
  – knowledge acquisition is not passive
    • relies on individual’s extant knowledge, context of data
  – additional challenge for hypermedia design: users can arrive at a point within the information space in many different ways

Hypermedia Goals (2.3)

• “In developing hypermedia applications, we need to be aware of the extent to which we are aiming to facilitate changes in the user’s knowledge base.” [p.50]
Historical Perspectives (2.2)

- Bush:
  - we need a method of managing our expanding stores of information and knowledge in a way that facilitates rapid and natural access
  - information is passive (to be retrieved)
- Nelson
  - coined term hypertext
  - information is active (to be constructed, utilized)

Historical Perspectives (2.2)

- Englebart
  - oN Line System (NLS)
    - “an instrument for helping humans operate within the domain of complex information structures.”
  - The computer:
    - a tool for navigating through complex information structures and examining them in ways that would be too complex otherwise.

Historical Perspectives (2.2)

- Englebart
  - NLS “an instrument for helping humans operate within the domain of complex information structures.”
    - “operate” = compose, study and modify.
    - “complex information structures” = content represents concepts
      - there is a relation between the content of concepts, their structure, and the structure of other domains of human thought that is too complex to investigate in linear text.

The Unfortunate Trend [p.40]

- Focus on:
  - hypermedia as a data structuring tool
  - presentation mechanisms
  - specific information representations
  - tools to manage information representations
  - focus has NOT be on supporting the effective use of information
1945 Vannevar Bush proposes Memex in his article “As We May Think”.
1965 Ted Nelson introduces Xanadu and coins the term hypertext.
1967 Andries van Dam develops the Hypertext Editing System at Brown University, followed by the introduction of FRESS in 1968.
1968 Doug Engelbart gives a demo of NLS, a part of the Augment project, started in 1962.
1975 A team at CMU, headed by Robertson, develops the ZOG system, which later becomes KMS.
1978 A team at MIT, headed by Andrew Lippman, develops the Aspen Movie Map, the first true example of a multimedia application including videodisk.
1985 Janet Walker develops the Symbolics Document Examiner, the first hypertext system used by “real” customers.
1985 Several other hypertext systems are announced, including NoteCards from Xerox, and Intermedia from Brown University.
1986 OWL introduces Guide for the Macintosh, the first widely available hypertext system, based on the Unix Guide system, developed by Peter Brown at the University of Kent.
1987 Apple delivers HyperCard free with every Macintosh.
1987 The ACM organizes the first Conference on Hypertext.
1989 Berners-Lee proposes WWW [deBra, 2004]

What is Hypermedia?

• §2.1 repeats what we already know
  – reading is an active process
    • break information into small chunks, rearrange on basis of one’s information requirements, extant knowledge base
  – writing is a destructive process
    • a web of associative links “weeded” into a linear form

Data of Information

• Data and information should be viewed as separate
  – not done on Web
    • anchors for links are information components
    • the raw data is the HTML file
    • information components embedded in data
    • Microcosm avoids this
Hypermedia/Presentation Systems

- Role of constraints:
  - using the Web as the hypermedia and presentation system constrains the type of applications that can be developed and presented
  - e.g., restrictions wrt: generic links, multiple destination links, overlapping link anchors
  - this is also true of any development and presentation system [what are the constraints of the others?]

Definitions of Media

- **media** [mass comm.]: the main means of mass communication, *esp.* newspapers, radio, and television, regarded collectively
  - newspaper (or radio, television) is a channel or a **medium** of mass communication
- **media** [computing]: a physical object (as a disk, tape cartridge, etc.) used for the storage of data.

Definitions of Multimedia

- **multimedia**: designating or relating to applications which incorporate a number of media, such as text, audio, video, and animation, *esp.* interactively.
  - Each media$_1$ can be created in various media$_2$ (and each media$_2$ has its own affordances)
    - **text**: ink on paper; UNICODE representation (file)
    - **audio**: grooves on vinyl records; magnetic tape; MPEG representation (file)
    - **animation**: a paper booklet of images; celluloid; MJPG representation (file)
Recap

• Basically, hypermedia systems must be able to deal with:
  – components -- text, graphics, audio, video
  – links -- relations between components
  – presentation -- structured display

Data Representation (3.1)

• The main message:
  – Each type of digital representation format has its own affordances
  – These affordances determine the degree to which that particular media can be integrated in hypermedia applications
    • affordance aka “operation”, “functionality”

Types of Operations

• Presentation
• Manipulation
• Analysis

Presentation Operations

• Presentation
  – can a condensed or otherwise quick-and-easy version be generated for the media?
    • thumbnails (text, images, graphics)
    • double-speed clips or other samples (animation, video, audio)
  • Needed for presentation of indexes (for browsing), for presentation of query results
  – images: progressive transmission
  – alignment of dynamic media
    • synchronization of audio, video, animation
Manipulation and Analysis Operations

• Modification
  – Editing the media

• Analysis
  – e.g., automatically identify all instances of a concept/term/object
    • realized in text (easy)
    • realized in images (difficult)

Table 3-1 [p.56, reorganized]

<table>
<thead>
<tr>
<th></th>
<th>Text</th>
<th>Audio</th>
<th>Graphics</th>
<th>Images</th>
<th>Animation</th>
<th>Video</th>
</tr>
</thead>
<tbody>
<tr>
<td>Modification</td>
<td>character-wise, string-wise (RegExp), other editing</td>
<td>of sample of waveform, of primitives of structure, of geometry of pixels, other editing</td>
<td>synchronization</td>
<td>of primitives of structure of lighting, of shading, of compositing</td>
<td>synchronization</td>
<td>of primitives of pixels of frames of effects, other</td>
</tr>
<tr>
<td>Analysis</td>
<td>searching, indexing</td>
<td>searching, indexing</td>
<td>searching, indexing</td>
<td>searching, indexing</td>
<td>searching, indexing</td>
<td>searching, indexing</td>
</tr>
</tbody>
</table>

Thought Questions

• Are media other than text actually integrated or do they merely serve as annotations? p.58

• Distinguish between passive and active media p.59

• The role of “componentialization” [made-up term] in associative linking
  – “componentialization” in text/audio/graphics/images/animation/video

Information Structures (3.2)

1. Information is broken into atomic blocks (nodes)
  – e.g., one screenful, an audio track, a video shot, a block of text
  – size is small to the point that further decomposition “causes information to lose all usefulness”
Information Structures (3.2)

2. Nodes are organized into a structure
   – to support navigation and browsing
   – can make use an intermediate structure such as a document
     • in this case, documents and nodes can be interlinked

Example from R&M (2002)

• The “help phone” hypermedia application

Examples

• Training material
  – nodes are training modules
    • additional nodes can be created for question/answer tests and remedial information
  – want nodes to be presented in sequence
    ∴ use linear structure

• Educational material
  – nodes are {sub}*sections (e.g., Safari)
  – want to support pedagogical reading
    ∴ use hierarchy as base structure
  (and supplement with additional links to glossary, references)
Examples

• Encyclopedia
  – nodes are encyclopedia entries
  – want nodes to be accessible from different contexts
  \[\therefore\] use network structure

Remember…

• “Any given navigation path through a hypermedia application will be linear — a hard constraint set by the unalterable single dimensional nature of time” [p. 29]
• Non-linearity: explicit support for a network of possible paths through the information

Types of Information Structures

• Hierarchical
  – a graph with no cycles
  – mimics printed media
    • e.g., books have chapters, chapters have sections, sections have subsections…
  – table of contents easily generated
  – “the root node is usually the starting point for readers…”

  – meaning of links: “HAS-A” relationship

Types of Information Structures

• Linear structures
  – [a special case of hierarchical structures]
  – enforces a particular sequence of access of nodes
  – a guided tour or hypertrail: a sequence of nodes bound together

  – meaning of links: “IS-FOLLOWED-BY” relationship
Types of Information Structures

- Network/Graph
  - nodes can have arbitrary in- and out-degrees
  - any node can be linked to any other node
  - meaning of links: “IS-RELATED-TO” relationship
  - such “resulting network[s] … are one of the main advantages of a hypermedia applications”

Example of Information Structure

Hypermedia Information Models

Starting with what we already know...

• The Web makes use of a client-server paradigm
  - the client is the browser, which makes requests of the server
  - the server attempts to respond in an appropriate fashion

• is a “computational model that supports hypermedia functionality”
  - not a proper hypermedia model, per se

Lowe & Ginige, 1996

Hypermedia Information Models

- The “hypermedia information model” of the Web:
  - simple “node-link” model
    - information presented in nodes, nodes interconnected with simple, point-to-point, unidirectional, non-contextual links
    - links do not have typing
  - Does not address hypermedia issues
    - does not separation of data from information, …
The “Hypermedia Information Model” of the Web

- Nodes:
  - in general case: multimedia compositions
    - frames, applets, three-dimensional VRML
      models, (other) integrated media displayers
      (e.g., QuickTime)
    - text, graphical images
    - supported to varying degrees by various
      browsers (browsers might make use of plug-ins
      for improvement of functionality)
  - in simplest case: consists of only text
    - browsers can deal with this; file doesn’t need to
      make use of html

The Web

- Re: the Web:
  the hypermedia functionality is
  embedded in the links, not the nodes

  “It is really the links which gives the
  hypermedia functionality (as opposed
  to the multimedia functionality
  embedded in the nodes) ” “the nodes are
  still the basic information units” [p.72,
  wrt WWW]

Other Hypermedia Information Models

- Client-Server Models
  - the simple node-link model used in the
    Web
  - Hyperwave/Hyper-G
- The Dexter Hypertext Reference Model
- CMIF; the Amsterdam Hypermedia
  Model (AHM)
- The Hypertext Abstract Machine
  (HAM)

The Dexter Model

Figure 3: A depiction of the three layers of the Dexter model as
embedded in an actual hypertext system.
The Dexter Model

- Bottom two layers describe a *passive data structure*
  - middle layer: a database of components (atomic and composites) and links between them
  - bottom layer: the representation of the data types for the components themselves
    - Text, graphics, animations, simulations, images, …
    - “the Dexter model treats within-component structure as being outside of the hypertext model per se.”

The Dexter Model

- The top layer
  - a *session* entity keeps track of the mappings between components and their instantiations
  - specifies the access and manipulation of the *passive data structure*
The Amsterdam Hypermedia Model

• criticism of the Dexter model
  – poor handling of temporal media
    • specification of time (e.g., duration, delays, etc)
      must be in within-components layer
    • no expression of timing relationships between components
  – poor handling of context of anchors, context of presentation information
• AHM merges ideas from Dexter and CMIF (xxx will present this)

Hypertext Abstract Machine

Levels
• presentation
• Hypertext Abstract Machine
• database

The Gist of Ch. 4

• RMM and OOHDM are:
  – are examples of “hypermedia development process models” (p.108)
  – also referred to as “hypermedia design methods” (p.111)
What is a *hypermedia development process model*?

- tells you how to conduct/accomplish
  - project management
  - project investigation and specification
  - design of application
  - design of application components
  - putting application into production
  - maintenance of application
  - evaluation of application
  - adequate documentation for project

In Sum…

- work is just beginning on the development of *hypermedia development process models*
- the process models that currently exist are not adequate
  - the ones specific to hypermedia development are incomplete
    - e.g., they do not yet effectively cover project management and application maintenance
  - the ones general to software engineering are inappropriate

Issues…

- “we still do not have effective or accurate metrics for the desired functional and nonfunctional characteristics” [p.91]
  - then how do we expect a process model to specify activities for evaluation

Issues…

- the pursuit of desirable characteristics can conflict, and we need to find “the best balance” p.89
  - but can’t say how to find that balance; only that it depends on the particular application
Issues…

- we can’t tell in advance whether effort applied to a specific activities will actually result in a payoff wrt a beneficial characteristic
  - “enough experience does not yet exist to accurately define specific relationships [between effort and outcome wrt specific characteristics]” p.89

Ch. 4: A final word

- Hypermedia engineering: 
  *The employment of a systematic, disciplined, quantifiable approach to the development, operation and maintenance of hypermedia applications*

- BUT: a systematic, quantifiable approach requires a certain base of knowledge

Chapter 5: Obtaining a Quality Product

The gist of Ch. 5

- Product Goals
  - Relevance
  - Completeness, Correctness
  - Usability
  - Support *use* of information (rather than *procurement of information*)
  - Reduce (or eliminate) cognitive burden
To achieve relevance 5.2

- understand goals of application

To achieve completeness and correctness 5.3

- Identify bounds of information space
  - provide cues to user about those bounds
- Ensure content is not missing
- Choose appropriate node granularity
  - not too fine, not too coarse
- Ensure links are not missing
- “Allow users to recognize information for what it is” (e.g., factual, opinion, etc)

To achieve usability 5.4

- Develop a suitable information structure
- Provide overview of information space
- Make use of local context
  - model user’s short term-history
- Afford appropriate navigation and browsing
- Resist the temptation for interface fanciness/complexity
  - in an attempt to appear “sophisticated”

To achieve usability 5.4

- Provide a search facility
  - ensure indexing is high-quality
  - don’t circumvent deliberate constraints on navigation paths
    - i.e., those that are put into effect by virtue of information structure
  - provide localization information
- Don’t focus solely on content, also create a usable interface
To promote *use* (rather than *procurement*) of information...

- Afford users with ability to make use of information
  - presumes an underlying task
- Provide facilities for:
  - link filtering (see all links by certain author, creation date)
  - information summarization

To reduce cognitive burden

- *increase* ease with which user can localize him or herself in information space
- *decrease* number of concepts user is required to manage
- *increase* ease with which navigation and search can be performed (“functionality”)

Chapter 6: Obtaining a Quality Process
Goals of Hypermedia Dev’t

• to develop high-quality hypermedia applications which have the optimal balance of desired characteristics
• to carry out said dev’t in the most effective and efficient manner possible

Most Time-Consuming Activities

– project management
– project investigation and specification
– design of application, application components
  • media capture, manipulation, structuring
  • UI dev’t
– putting application into production
– maintenance of application*
– evaluation of application
– adequate documentation for project

Cognitive Management

• Much more burdensome for developer than user
  – user doesn’t need to understand the entire information space
    • to structure content (add links) developer needs understanding of entire information space
  – developers, unlike users, need to conduct development process
  – techniques for reduction of burden:
    • procedural guidance, directed assistance

Reuse

• Improved efficiency
  – less time, effort
• Improved effectiveness
  – as related to application goals
    • reliability, maintainability, consistency, robustness
• Types of reuse:
  – media, information, component, process
**Media vs. Information reuse**

- boils down to difference between *media* and *information*
- information representation

**Maintenance**

- Types:
  - corrective, adaptive, perfective
- Spaghetti links
  - are difficult to modify
  - any new links need to be cross-checked with many other links

**Process Management**

- Which activities should be undertaken?
- How should they be integrated into a cohesive, co-ordinated process?
  - Approach #1: figure it out yourself
  - Approach #2: choose an existing methodology, extend and augment it as needed
  - Figure out which skills will be needed

**Product Measurement**

- Navigatibility
  - use link structural analysis:
    - centrality, compactness, stratum
  - use semantic distance
    - very difficult to determine
- Validity
  - automated analyses (anchor/destination overlap)
  - Expert (manual) evaluation
Product Measurement

- Organization
  - partitioning metrics
    - for each node: content “amount”, balance, type and number of links
  - screen analyses
  - intra- and inter-node semantic distance

Process Measurement

- Main metric:
  - Amount of development time (per activity)
- Future work - develop metrics for:
  - quality of dev’t docs
  - extent of reuse
  - effectiveness of project management

Ch. 6: A final word

- Scalability
  - increase in application size translates to large increase in dev’t time and effort

- Ad hoc process management
- Very little work on development of process measures