Building and Applying Digital Libraries

(Part 1)

Edward A. Fox
Professor
Dept. of Computer Science
660 McBryde Hall, Virginia Tech
Blacksburg, VA 24061-0106
(540) 231-5113, FAX 231-6075
fox@vt.edu

Robert M. Akscyn
President
Knowledge Systems
RD2 213A Evans Road
Export, PA 15632
Phone/Fax: (412) 327-8159
rma@ks.com

0. Overview
1. Introduction to Digital Libraries
2. Key Requirements
3. Future Directions
4. Recommendations

To conserve presentation time, please review slides in Overview section before session
Overview

- Instructor background
- General Plan for Course
- Specific aspects

Background

- 1981 --> 1996: Knowledge Systems
- 1988 --> 1992: Dexter Group
- 1990 --> 1993: SIGLINK Chair
- 1990 --> 1996: Editor, ACM Hypertext Compendium
- 1994 --> 1996: PetaPlex Project (US Intelligence Community)
- But most important is experience from using hypertext
ZOG Project at Carnegie Mellon

- Ten year project: 1975-1985
  - Funded by Office of Naval Research
- General-purpose research on human-computer interaction
  - Newell et al
- ZOG: Large-scale, rapid-response menu-selection system
- Major application:

Knowledge Systems

- Spun out of Carnegie-Mellon University in 1981
- Goal: develop commercial follow-on to ZOG
- Purpose: general-purpose enterprise-wide management of knowledge (esp. collaboration)
- Focus: Networked workstations
  - (e.g., Sun, HP, DEC)
- Product: KMS® (Knowledge Management System®)
- Current efforts: Applying hypermedia technology to task of constructing large-scale digital libraries
ACM Hypertext on Hypertext project

- ACM Project to explore Hypertext Publishing
- Task: to represent one issue of CACM (July88)
  - 6 articles from HT87 + keynote + guest editor piece
- Three systems chosen
  - Hypercard
  - HyperTIES
  - KMS
- Lessons learned (Panel HT89)

Dexter Group

- Informal group of researchers from academe and industry
- Started meetings in 1988
- --> Dexter Reference Model
  - The Dexter Hypertext Reference Model Frank Halasz, Xerox PARC Mayer Schwartz, Tektronix Labs
- Good reference on the Dexter Model is the Feb 94 issue of Communications of the ACM.
  - Diagram of Dexter Model
- --> Responsible for organizing
  - Hypertext ’89
  - Hypertext ’91
  - SIGLINK
SIGLINK

• ACM Special Interest Group on Hypertext and Hypermedia
• Founded in 1990
• Approximately 1200 members
• Upcoming events:
  • Hypertext ’97 (Southhampton, UK)
  • Digital Libraries ’97 (where??)
• Please join!

ACM Hypertext Compendium

• Hypertext Database on field of Hypertext
• Published by ACM Press
• First Edition: ~130 articles
  • Bush, V --> ECHT ’90
• Developed by Knowledge Workshop under contract to ACM
  • (Examples from Compendium later in course)
• Second Edition in progress (triple size)
  • HT91
  • ECHT ’92
  • HT93
  • Major works of Engelbart et al
  • Major works of Intermedia Project
PetaPlex Project

• Goal:
  • Develop architecture for massive-scale digital libraries
• Objectives:
  • Capacity: 20 Petabytes on-line
  • Response over Internet: < 1 second for random URN
  • Throughput: 30 million web pages per second

  ◦ Approach
  ◦ Status

Approach

• Massively-Parallel Architecture
  • internetworks 1 million 20-Gb "SmartDisks"
• Content: hypermedia-structured
• "Supercompression" --> high compression level
• "DNE" protocol --> single round trip packet
• "Knowledge File System" --> single disk seek
• Proxy on client-side
  • to allow use of existing browsers
• Current prototype based on KMS
Status

- DNE single round-trip of packets works well
- Average internet response: .5 seconds
- Can construct 9Gb "SmartDisk" under $2000
- Constructing 1 billion web-page database

But most important is experience from using hypermedia

- Active hypermedia user
- Using hypermedia on daily basis for 18 years
  - Do virtually everything within HM environment
- ~200,000 nodes in KSI corporate database
- --> source of most of my knowledge about hypermedia
Hypermedia as foundation for 'corporate memory'

- Articles
- Budgets
- Drawings/Diagrams/Tables
- Email
- Financial statements
- Forms
- Letters
- On-line documents
- Software Programs
- Proposals
- Slides
- Training testing

General Plan for course

- Issue oriented
- Organize issues by useful categories
- Help build a conceptual framework
- Identify positions on issues
- Identify tradeoffs
- Recommendation(s) about each issue
- Generate discussion about issues
Specific aspects

- Organization of course notes
  - Model 1 (printout)
  - Model 2 (printout)

- Start slow, speed up
- Start talking, shift to more discussion
  - Use slides more as scaffolding to talk about what really interests us

- Rhetorical conventions for slides
  - Hollow bullet --> expanded below in slides
  - Solid bullet: --> no further expansion in slides
  - Red solid bullet (on-line): not in slides, but may show
  - No slide number --> a slide not in handout

Model 1

Iceberg Model

Printed Notes

mostly

Outline

More Detail

~600 slides

Examples

~200,000 nodes
Introduction to Digital Libraries

- Concepts and Practices
- Relevant Technologies
- Existing projects
- Existing frameworks
- What is different?
What are digital libraries?
- Any remote, any size "database"?

What's outside the scope?
- Traditional Databases?, WWW?
- Networking? CSCW?
- Where are the boundaries of digital libraries?

Types of Digital Libraries

Central Issue: Digital Library Architectures
- What tradeoffs are appropriate to best meet all expectations for digital libraries?

Types of Digital Libraries

- Private and Public
- Commercial and Non-commercial
- Personal and Corporate
- Intra-Organizational and Inter-organizational
- National (France, Britain, Singapore)

--> Central Issue:
- In what ways are these different that lead to different design decision?
- What factors are different about these types, that suggests need for different approaches?
### Relevant Technologies

- Library and Information Science
- Database systems
- Information retrieval
  - Indexing, ...
- Hypertext/Hypermedia/Multimedia
- Human-Computer Interaction
- Artificial intelligence
  - Expert Systems, Natural Language Processing
- Storage systems
- Networking systems
- Image Processing
- Compression schemes

### Existing projects (on the rise!)

- Mercury
- Dissertations
- Handle System
- JANUS
- Envision
- TULIP
- Gutenberg Project
- QUEST
- RightPages
- CS Tech Reports
- ELVYN
- NSF/ARPA DL Initiative
- CORE
- NASA Mission to Planet Earth
- ARPA BAA
- FreeLore Project
- CoLib Project
- ACM 'Electronic Community'
- IBM Global Digital Library
- Project Athena
- Library of Congress
- Univ Michigan Digital Library
- PetaPlex Project
- Gutenberg Project
- IBM Global Digital Library
- Library of Congress
- Univ Michigan Digital Library
Existing frameworks

• Not much to date!
  ◦ Exception: Gladney et al.
• Part of the purpose of this course is to reflect on what should be in useful frameworks
  ◦ Why are frameworks useful
• Starting points
  ◦ Key requirements
  ◦ Architecture
  ◦ Activity-based
  ◦ Example 'key requirements’ framework

Why are frameworks useful?

• Provide a way to conceptualize field
• Provide a way to organize knowledge
• Provide a common frame of reference that facilitates communication
• Help compare alternate approaches
• Help make design decisions
• Help make make/buy decisions
• Help promote interoperability
Example framework

Utility

Sustainability

Scalability

What is same/different

- From existing libraries
- From other aspects of electronic publishing
- For authors
- For users
  - But should also strive to identify and preserve what is the same as traditional libraries
    - Universal access?
    - Sense of organization
    - Sense of discipline
    - Better resources than individuals can afford
    - Completeness
What is different from existing libraries?

• More users, More contexts
  • Forever open, always ‘close by’

• Economics change significantly
  • Scalability issues very different

• More possibilities for organizing content
  • Multiple indexes and views, multiple search engines

• More possibilities for integrating with task environments
  • Access reference within context of constructing knowledge artifact
  • Link directly to references
  • Link directly to components (e.g., for reuse)
  • Copy components into another context

From other aspects of electronic publishing

• Scale
  • Not just small number of documents

• Interlinking
  • Widespread, inter-document linking
  • Within-DocA --> Within-DocB

• Indexing
  • Indexing across many documents/artifacts

• Range of size of objects
  • Poem --> Essay --> Article --> Book --> Encyclopedia
  • Image --> Video
  • Code: Statement --> Routine --> Module --> Library

• Greater usability problems
  • Some induced by scale
  • Some induced by heterogeneity
For authors

- Scale of what is considered publishable/valuable
  - (Paragraphs and Diagrams) vs (Articles and Books)
- Collaboration with co-authors
  - Co-authors can work on same version
  - Rise of ‘Sou-authors’ (authors who prepare for others)
- Maintenance of documents (versus frozen)
  - Authors more like editors of series
- Rhetorical freedom/limitations for Diagrams
  - - Less resolution for diagrams
  - + Color more available
  - + Use of animation
  - + Use of current data
- ’Dual document’ considerations
  - What to do if publishing paper (or other forms) as well

For users

- Access material previously could not afford to access
- Ability to dive directly into depths of documents
- Annotation/collaboration
- Note taking (with links to references)
- Searching
- Re-use (under fair use)
  - Clipping of portion of another artifact
  - Linked quotes
  - Reference by linking
  - Overlays
Key Requirements

- Perspectives
  - 1. Utility
  - 2. Scalability
  - 3. Sustainability

Utility = Usefulness + Useability

- Centrality of task environment
- Usefulness
  - Framework-2: User-Task-Info-Cap
  - Nature of the Content
    - (Media, Data models)

Useability
- Capabilities
- User interface design
- User studies
Centrality of Task Environment

• More than any other factor, the nature of the task is key to good design
  • "Know the user" is too indirect
    ° "Know the task" more appropriate
• Best way to understand task is to personally do task (repeatedly)
  • People you interact with may themselves not be true end-users
  • Incremental feedback better than quantum leaps
  • Important to understand differences among tasks and how needs differ
• Particularly difficult is relationship to outer task environment
  • E.g., Computing environment often most binding constraint

"Know the task" more appropriate

• If you look at most physical tools (e.g., hand tools)
  • their design is mostly a function of the task
• There is surprising little about their design that is human factors inspired!
• The task environment is a given and is not as flexible at higher levels
• The tool is often very specific to the task
  • for simplicity, ruggedness, effectiveness, etc
  • thus usually not flexible
  • exception: swiss army knife
• But humans EXTRAORDINARILY flexible and adaptable
• Tool designers should exploit this flexibility
Usefulness

- What forms/organizations of information do people want/need?
  - Depends on task environment!
  - Some answers are known
  - but many issues still are empirical questions yet to be answered

- Ways to add value

- A Fundamental Design Tradeoff
  - All things to all people
  - Some things to some people

- Recommendation:
  - Work bottom-up -- first be some things to some users
  - E.g., programmers (as a particular class)

Principal Task Environments: "3 E’s"

- Entertainment
- Education
- Enablement
  - Tools that enable users to perform tasks
  - Faster, better, cheaper,...
(some) Ways to add value

- Accessibility/Scale/Authentication
- Completeness
- Indexing/Cataloging
- Chunking/Organizing/Structuring
- Interlinking/Cross-referencing/Naming
- Review/Critique
- Abstraction/Overviews/Condensations/Surveys
- Import/Exporting/Conversion
- Discovery/Search/Awareness services
- Integration with other tools
- Integration across libraries
- --> Improve productivity of user

Framework: User-Task-Info-Capability

- Purpose of framework
  - flush out needed information and capabilities
    - U-T-I-C Diagram
    - Types of USERS
    - Types of TASKS
    - Types of INFORMATION (content)
    - Types of DL capabilities to process information
    - Diagram: subtyping of each level of framework
    - Impact of Organization
Types of Users (roles)

- Administrators
- Engineers
- Investors
- Managers
- Programmers
- Scholars
- Scientists
- Secretaries
- Students
- Teachers

But note: people often perform multiple roles
same/different times
same/different tasks

Also as members of teams

Types of tasks

- Communicating
- Deciding
- Designing
- Evaluating (Grade, Critique, ...)
- Fact finding
- Learning
- Modelling
- Planning
- Presenting
- Researching
- Writing/Publishing
Types of information

- Editorials
- Diagrams
- Calendars
- Data
- Lists
- Memos
- Images
- News
- Regulations
- Predictions
- Invoices
- Policies
- Speech
- Letters
- Photos
- Schedules
- Formulae
- Essays
- Trends
- Procedures
- Events
- Maps
- Code
- Scripts
- Instructions
- Clips
- Orders
- Laws
- Hints
- Patents
- Forms
- Drawings
- Directories
- Symbols
- Tables
- Plans
- Stories

Types of capabilities to process information

- Authoring
- Citing
- Clipping
- Converting
- Displaying
- Exporting
- Formatting
- Linking following/Link creation
- Searching
- Sending
- Transforming
- Notetaking/Notemaking
Nature of the Content

- Media
  - Text
  - Graphics
  - Images
  - Sound
  - Video

Data Model is more important than the media!

- Data models
  - Linear
  - Tabular
  - Hierarchical
  - Network
  - Relational, Object oriented, Hypertextual ...
  - Hybrids

Capabilities

- Access Methods
- Processing content/Executable content
- Integration with others systems/tools
- Support for Specific Applications/User Communities
Access Methods

1. Navigation
   • Browsing
   • Problems
   • Approaches

2. Search and query
   • Keyword
   • Full-text
   • Concept search
   • Structure search

Navigation (’Wayfinding’)

• Browsing

• Problems
  ◦ Disorientation
  ◦ Cognitive Overload

• Approaches
  ◦ How can system address disorientation?
  ◦ How can system address cognitive overload?
Disorientation

• Difficulties
  • Not knowing where one is
  • Not knowing what surrounding terrain is like
  • Not knowing where info sought is located
  • Not knowing how got to current location

• However
  • "Lost in Hyperspace" is often over-dramatized
  • Solutions proposed (e.g., overviews) don’t address multiple facets of problem

Cognitive Overload

° Description
° Examples
Description of Cognitive Overload

• User confused about his/her objectives,...
• Stress of difficulty of using system overloading user and distracting from task
• Whereas disorientation is confusion about the hypertext itself

Examples of Cognitive Overload

• Not sure how 'current node' relates to user’s task
• Not sure what links to follow
  • Relative to current task
• Not sure how to structure latest thought
  • Need to just get it out somehow versus extensive interrupt)
  • Example: navigating to 'safe harbor' for idea, but then forgetting idea 'enroute'
• Logistics of using system saps user’s task orientation
How can system address disorientation?

• What can cause disorientation?
• Problems that may result
• How can it be reduced?
  • Traditional answer: Overviews, but ...
  • Traditional answer: Search and Query
  • Traditional answer: Graphical cues
  • Traditional answer: Bookmarks
  • Another approach: Aliases

How can system address cognitive overload?

• Keep data model very simple
• Make link anchors as informative as practical
  ("rhetoric of departure" --- Landow)
• Provide mechanisms for ’lazy organizing’
• Facilitate ’Switchyards’
Search and query

• Keyword
  • Known to be suboptimal (but scalable)
• Full-text
  • Scalability to DL levels is open question
  • Good opportunity to think ’tradeoffs’
• Concept search
  • Use of terms other than ones provided
• Structure search
  • Part navigation to sub-scope, then search ’within’
  • Surprisingly useful type of search!

Processing/Executable content (Agents)

• Searching, Transforming, Executing, ...
• Document composition
• Document printing
• Authoring assistance
Integration with others systems/tools

• Very important
  • Source of considerable ’task leverage’
  • This is where DL’s might derive most of their value-added

• Types of integration
  • Programs
  • Databases
  • Other DL’s
  • Other types of DL’s (personal, organizational, public)

• Means of integration
  • Import/Exporting (shared formats)
  • Reading shared store
  • Dynamic exchange (real time)

Support for Specific Applications/User Communities ("task environments")

• Key theme: What needs are different enough to have design implications?
  • 1. Research and development
  • 2. Education and Training
  • 3. Business and Government
  • 4. Collaborative Work
Research and development

• Literature search/read/notetaking
• Issue analysis
  • Need for ‘Structured’ Discussion groups
• Modelling
• Computer-supported cooperative work
  • High need to co-author documents
  • Interaction with peer ‘concurrent engineers’
• Publishing

Education and Training

• Testing
  • Course examinations
  • On-line self-assessment
• Distance learning
• Self-paced learning
• Curriculum enrichment
  • Cross-course linking
• Large-screen presentation
• Stepping stone to other communities
  • --> Need to experience on-line tools and techniques of those communities
Business and Government

- Access to conformance documents (regs, policies, standards)
- Communication
  - Internal
  - External (e.g., marketing collateral, Announcements)
- Document development
  - Authoring/Co-authoring
  - Dissemination
  - Use as templates
  - Use for submission
- Evaluation (of proposals)
- Record keeping

User interface design

- Types of interaction
- Amount of use affects importance of efficiency
- User-system cycle time
- Underappreciated role of User/Org Conventions
- Relationship to data model
- Recommendations:
  - Bias system toward rapid system response
  - Bias system to support rapid interpretation
  - Bias system to support rapid user response
Types of interaction

• Command
• Menu
• Direct manipulation
• Hybrids
  • May be more natural even if not consistent

User-system cycle time

• What makes a difference?
  • System response time
  • User interface directness
  • Simplicity of conceptual data model
  • Fewer more homogeneous node layouts
  • Node and object Schemas

• Why important?
  • Makes significant difference in style and productivity
  • Products developed are not the same as otherwise would be (with less-facile systems)
Underappreciated role of Conventions (versus features)

• Norms users decided for themselves
  • Not enforced by system
• Important roles for Conventions
  • Allows more exploration within specific task environments
  • Conventions substitute for ’Designer Knows Best’
  • Good pre-test of future functionality

Relationship to data model

• Data Model is Heart of User Interface Design
• Constructs and their nature determine possibilities
  • Operands * Operators --> sets stage for User Interface
  • Provides many issues concerning defaults
User studies

• Instrumental for improving usability
• What we know about existing libraries?
  • Electronic and non-electronic
• What we know from hypertext/hypermedia studies?
• What do we know about use of information retrieval systems?
• What would we like to know and how to incorporate in design cycle
  • Embedded instrumentation

Scalability

1. Why is scalability important?
2. Architecture
3. 'Stocking the shelves'
4. Price/Performance
5. Security and Integrity
Why is scalability important?

• Small scale libraries less interesting, less complete, less interlinking
• Permits economies of scale
  • Construction
  • Maintenance
• Enables simpler fee mechanisms
  • E.g., subscription fees
• But large scale systems face technological challenges
  • Response time
  • Cost of construction and maintenance

Architecture

• 1. Hardware
• 2. Software
• 3. Networking

--> Architecture can determine overall price/performance of the system

--> Need to avoid drowning in possibilities!
'Stocking the shelves'

1. OCR
2. Document analysis
3. Conversion
   - Especially structuring for on-line access
4. Formats -- SGML, HTML, HyTime, ....

Price/Performance

1. Response time
   - Bandwidth conservation
   - Load-balancing
2. Storage efficiency
   - Hypothesis: $50 per Gb is a key threshold for large-scale digital libraries
   - At this point other cost factors dominate (e.g., systems administration)
Security and Integrity

- More functionality more problems
- 1. Permissions
- 2. Concurrency control
- 3. Annotation

Sustainability

- Costs of construction and operation
- Pricing
- Fee mechanisms
- Who actually pays?
- Incentives to all stakeholders
- Intellectual Property Issues
- Relationships to other communication mechanisms
- Competition
Costs of construction and operation

- Acquiring rights
- Processing raw material
- Editorial effort
  - May be the dominant cost!
- Maintenance of material
- Maintenance of infrastructure
- Archiving

Pricing

- Who pays how much for what?
  - Need for two orders-of-magnitude reduction in price
  - Otherwise we remain ‘Knowledge Rich, Access Poor’
  - Otherwise we remain ‘Access Rich, Knowledge Poor’

- Topic of active discussion on/for Internet
  - Internet-wide
  - Electronic Communities (e.g., ACM Electronic Community)
Fee mechanisms

- Usage-based vs. Flat Fee Subscriptions
- Priority-based scheme could be way to deal with congestion
- How do these impact on the user?
  - Usage-based schemes destructive of productivity
  - Organizations prefer peace of mind from flat-fees
- How do these impact on the library?
  - Assuming expenses < revenues
- Recommendation: Strive at all costs for flat fees!

Who actually pays?

- Information provider?
  - Perhaps for marketing 'collateral'
- User?
- User’s patron?
  - Parent
  - Dean
  - Employer
- --> Perhaps User’s patron will likely be most important decision maker to convince
Incentives to all stakeholders

• How do we motivate all parties to collaborate?
• Authors
• Editors
• Publishers
• Educators
• Librarians
• Subscriber Users/User Patrons
• Technologists
• Vendors
• --> Need 'We-all-need-each-other' type of Win-Win arrangement

Robert M. Akscyn -- Multimedia '96 -- November 1996

Copyright © 1996 Knowledge Systems Incorporated

Intellectual Property Issues

• Rights
  • Electronic version blur boundaries (Is it a book or movie?)
• Fair Use
  • Where are new boundaries?
  • Derivative works, colorization problems, Clipping, Linking to/from
• Enforcement mechanisms
  • Authorization
  • Authentication (e.g., Kerberos)
• Contracts
Relationships to other publishing mechanisms

• How does digital libraries displace/supplement other channels?
  • Paper
  • Email
  • Fax
  • Phone
  • Existing libraries

• Market share of user’s time?

Market share of user’s time?
(and productive use of user’s time)

<table>
<thead>
<tr>
<th>User’s Time</th>
<th>F2F</th>
<th>Phone</th>
<th>Fax</th>
<th>Email</th>
<th>Mail</th>
</tr>
</thead>
<tbody>
<tr>
<td>100%</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Status Quo</th>
<th>Digital Libraries</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Consume</td>
</tr>
<tr>
<td></td>
<td>Produce</td>
</tr>
<tr>
<td></td>
<td>Consume</td>
</tr>
<tr>
<td></td>
<td>Produce</td>
</tr>
<tr>
<td></td>
<td>Consume</td>
</tr>
<tr>
<td></td>
<td>Produce</td>
</tr>
<tr>
<td></td>
<td>Consume</td>
</tr>
<tr>
<td></td>
<td>Produce</td>
</tr>
</tbody>
</table>

Key question is how each enables productivity!
Competition

- Paper
- Culture (inertia, perceived threats)
- Competing designs/paradigms (to traditional commercialization)
  - Free browsers
  - Free content (e.g., promotional materials)
  - Free authoring (vanity publishing)
- Cost (has to be significant reduction)
  - To be interesting
  - To overcome inertia
  - To make a significant difference

Future Directions

- Need to develop practical theoretical framework
  - Otherwise we get 'synergy-free' trajectory
  - Information 'SuperHypeway'
  - Problems Digital libraries may face
  - Potential Pitfalls
  - Conferences
  - Specific Initiatives
Information 'SuperHypeway'  

- Some consider Internet de facto "information superhighway"
  - 30+ million users and growing rapidly
  - Dramatic interest in WWW/Mosaic/Netscape technology
  - But individual users can cause significant problems

- Others: digital video on demand, ...

- Issues proliferating faster than consensus
  - E.g., Need for (or not) "Universal Access"

- HTML Extensions now out of control
  - Multiple groups vying for supremacy

Problems Digital libraries may face  

- Scale problems
  - Bandwidth congestion
  - Server congestion

- Sustainability
  - Not clear how best to recoup costs

- Location dependence for documents/nodes/links

- Cross-platform support

- Cross-library support

- Fear from existing publishers
Potential Pitfalls (for industry)

- Bias toward entertainment rather than economic value
- Everyone is an author and no one is an editor
  - Low quality diminishes demand
  - But (nightmare) ease of access could breed complacency with substandard info
- Trying to support too many variations
  - Formats, protocols, views, platforms, rights, fees)
- Usage-based pricing squelching true demand
- Fragmentation into many small libraries with no meaningful integration
- Not making optimal tradeoff for in/out bandwidth
  - Asymmetric bandwidth is acceptable tradeoff

Conferences

- Feb94: Massive Digital Data Systems Workshop
- Mar94: Digital Library Workshop in San Antonio
- May94 Rutger’s Digital Libraries Workshop
- Jun94: On-line Publishing ’94
- Jun94: Digital Libraries ’94
- Sep94: ECHT ’94
- Oct94: Multimedia ’94
- Jun95 Digital Libraries ’95
- Nov95: Multimedia ’94
- 1996: HT96, DL96, and a hundred Internet/WWW conferences
Specific Initiatives (in U.S.)

- 1. NSF Digital Library Initiative
  - NSF Announcement
- 2. ARPA
  - ARPA BAA94-28
- 3. US Intelligence Community
- 4. Universities
- 5. Publishers

Key Recommendations

- Simplicity is key -- but is incredibly difficult
  - You must be immovable object
  - successful against irresistible forces
- Experimentation and incremental development
  - ’Perpetual Prototyping’
- Focus first on data model tradeoffs
  - then user interface
  - then script language
- Strive for rapid system response
  - Probably the single most important usability factor
- Avoid ’premature automation’
  - Get hands dirty doing tasks manually