Multivalent Documents: A Platform for New Ideas

Thomas A. Phelps
Robert Wilensky
The UCB Digital Library Project Team

http://elib.cs.berkeley.edu/
Multivalent Documents: Motivation

- Document manipulation is ubiquitous
  - web/collection browsing, word processing, e-mail, net news, help systems, program editors, ...

- Existing system are
  - limited in format
  - not integrative (each app does own editing, etc.)
  - not very extensible:
    » Word, Framemaker: API for given functions
    » OpenDoc, HTML+Applet: juxtaposition, not integration
    » Netscape: Open source, hard to integrate and distribute diverse changes

- Limits use, especially experimentation with new functionality, modes of interaction
Multivalent Documents: Goal

- Provide inventors (you!) with a system
  - of sufficient power
  - and fine-grain control
- that they find it an inviting platform on which to try out their new ideas.
- Enable the widespread distribution of new functionality
  - by not requiring changes to the core, and
  - coordinating potentially conflicting intentions from diverse parties.
Goal: An Anytime, Anywhere, Any Type, Every Way User-Improvable Digital Document System

- Anytime - Add content (annotations or core) and functionality on demand
- Anywhere - over network, read-only media, mobile devices
- Any Type - scanned page images, HTML,... Implement functionality once, works on any type
- Every Way - content, functionality, operation
- User - End-user dynamically loads easily, hacker gets deep access and easy distribution
- Improvable - Seamless integration of improvements for inexpressive (i.e., all current) formats
- Digital Document System - Conform to modern practices: multimedia, structure-based, style sheets, XML, WYSIWYG, GUI, networked, incremental algorithms, ...
Multivalent Documents: The Architecture

- An extension mechanism: behaviors
  - modular, small or large, reusable, composable program units.

- A document tree core data structure
  - perhaps composed from multiple, distributed layers of related information
  - flexible enough to handle diverse formats

- Low-level communication protocols
  - to allow behaviors access to arbitrary document activities, but still compose without conflict

- Higher-level semantic events
  - to allow behaviors to readily participate in logical document activities.

- A behavior management scheme: hubs
  - to give every document a custom browser
Multivalent Documents: The Implementation

The Multivalent Browser, now available
- An open source Java (1.3-4) application, available at http://http.cs.berkeley.edu/~phelps/Multivalent/

Simple, yet:
- Extensions already give it greater format functionality than any other browser
- It has proven capable of supporting unanticipated, novel, even surprising, functionality without any special accommodation.
- Some of these can support new forms of interaction, such as distributed, shared annotations, of unrestricted types.
Multivalent Browser Status

🔗 Multivalent Browser, DR2, available; beta ASN
  - An open source Java (1.3-4) application, available at http://http.cs.berkeley.edu/~phelps/Multivalent/

🔗 Implemented behaviors:
  - Media Adaptors:
    » HTML 3.2 + CSS
    » LaTeX/DVI
    » PDF (prototype)
    » “enlivened scanned images” (XDOC and PDA)
    » ASCII, man pages, Perl POD, Zip, local directories, Java .class,…
    » “multi-page”
  - Standard browser features (cache, composable, native GUI widgets, bookmarks, …),
  - Robust hyperlink and robust location support
  - Annotations: hyperlink, highlight, executable copyeditor marks, style, redaction, notes, …
  - Tools: scrollbar search vis., Notemarks/executive summary
<table>
<thead>
<tr>
<th>BASIN NAME</th>
<th>STATION NAME</th>
<th>AGENCY</th>
<th>ELEV FEET</th>
<th>APR 1 AVG</th>
<th>TODAY</th>
<th>PERCENT OF APR 1</th>
<th>24 HRS AGO</th>
<th>1 WEEK AGO</th>
</tr>
</thead>
<tbody>
<tr>
<td>TRINITY RIVER</td>
<td>MUMBO BASIN</td>
<td>USBR</td>
<td>5700</td>
<td>25.8</td>
<td>1.84</td>
<td>7%</td>
<td>2.88</td>
<td>---6</td>
</tr>
<tr>
<td></td>
<td>SCOTT'S MOUNTAIN</td>
<td>USBR</td>
<td>5900</td>
<td>27.0</td>
<td>1.98</td>
<td>7%</td>
<td>2.60</td>
<td>----</td>
</tr>
<tr>
<td></td>
<td>HIGHLAND LAKES</td>
<td>USBR</td>
<td>6030</td>
<td>34.0</td>
<td>2.5</td>
<td>7%</td>
<td>3.1</td>
<td>12.8</td>
</tr>
<tr>
<td></td>
<td>MIDDLE BOULDER #3</td>
<td>USBR</td>
<td>6200</td>
<td>27.1</td>
<td>6.06</td>
<td>22%</td>
<td>7.02</td>
<td>----</td>
</tr>
<tr>
<td></td>
<td>BONANZA KINGER #3</td>
<td>USBR</td>
<td>6450</td>
<td>40.5</td>
<td>7.5</td>
<td>19%</td>
<td>8.5</td>
<td>12.8</td>
</tr>
<tr>
<td></td>
<td>BIG FLAT D LAKES</td>
<td>USBR</td>
<td>5100</td>
<td>20.0</td>
<td>8.6</td>
<td>43%</td>
<td>9.0</td>
<td>10.6</td>
</tr>
<tr>
<td></td>
<td>PETERSON FLATAIN</td>
<td>USBR</td>
<td>6700</td>
<td>33.0</td>
<td>10.4</td>
<td>32%</td>
<td>10.8</td>
<td>11.6</td>
</tr>
<tr>
<td></td>
<td>SHIMMY LAKE</td>
<td>USBR</td>
<td>6200</td>
<td>49.9</td>
<td>11.6</td>
<td>23%</td>
<td>12.2</td>
<td>----</td>
</tr>
<tr>
<td></td>
<td>RED ROCK MOUNTAIN</td>
<td>USBR</td>
<td>6700</td>
<td>44.0</td>
<td>13.8</td>
<td>31%</td>
<td>14.0</td>
<td>10.6</td>
</tr>
<tr>
<td>SACRAMENTO RIVER</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>CEDAR PASS</td>
<td>SCS</td>
<td>7100</td>
<td>18.1</td>
<td>10.0</td>
<td>55%</td>
<td>10.3</td>
<td>10.5</td>
</tr>
<tr>
<td></td>
<td>BLACKS MOUNTAIN</td>
<td>DWR</td>
<td>7286</td>
<td>8.6</td>
<td>3.4</td>
<td>39%</td>
<td>3.8</td>
<td>----</td>
</tr>
<tr>
<td></td>
<td>SAND FLAT</td>
<td>USBR</td>
<td>6750</td>
<td>42.4</td>
<td>14.3</td>
<td>34%</td>
<td>14.6</td>
<td>----</td>
</tr>
<tr>
<td></td>
<td>MEDICINE LAKE</td>
<td>USBR</td>
<td>6700</td>
<td>32.7</td>
<td>16.1</td>
<td>49%</td>
<td>16.3</td>
<td>----</td>
</tr>
<tr>
<td></td>
<td>ADIN MOUNTAIN</td>
<td>SCS</td>
<td>6350</td>
<td>13.6</td>
<td>7.6</td>
<td>56%</td>
<td>8.0</td>
<td>----</td>
</tr>
<tr>
<td></td>
<td>SNOW MOUNTAIN</td>
<td>USBR</td>
<td>5950</td>
<td>27.0</td>
<td>6.7</td>
<td>25%</td>
<td>7.3</td>
<td>9.5</td>
</tr>
<tr>
<td></td>
<td>SLATE CREEK</td>
<td>USBR</td>
<td>5600</td>
<td>30.0</td>
<td>7.7</td>
<td>26%</td>
<td>9.1</td>
<td>----</td>
</tr>
<tr>
<td></td>
<td>STOUTS MEADOW</td>
<td>USBR</td>
<td>5400</td>
<td>42.5</td>
<td>5.9</td>
<td>14%</td>
<td>6.2</td>
<td>9.1</td>
</tr>
<tr>
<td>FEATHER RIVER</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>KETTLEROCK</td>
<td>DWR</td>
<td>7300</td>
<td>25.5</td>
<td>11.0</td>
<td>43%</td>
<td>11.6</td>
<td>14.2</td>
</tr>
<tr>
<td></td>
<td>GRIZZLY</td>
<td>DWR</td>
<td>6900</td>
<td>29.7</td>
<td>12.0</td>
<td>40%</td>
<td>12.2</td>
<td>13.7</td>
</tr>
<tr>
<td></td>
<td>PILOT PEAK</td>
<td>DWR</td>
<td>6800</td>
<td>52.6</td>
<td>8.8</td>
<td>17%</td>
<td>9.7</td>
<td>12.7</td>
</tr>
<tr>
<td></td>
<td>GOLD LAKE</td>
<td>DWR</td>
<td>6750</td>
<td>36.5</td>
<td>25.0</td>
<td>68%</td>
<td>25.1</td>
<td>25.6</td>
</tr>
<tr>
<td></td>
<td>HUMBUG</td>
<td>DWR</td>
<td>6300</td>
<td>28.0</td>
<td>16.2</td>
<td>58%</td>
<td>16.8</td>
<td>18.7</td>
</tr>
<tr>
<td></td>
<td>RATTLESNAKE</td>
<td>DWR</td>
<td>6100</td>
<td>14.0</td>
<td>3.7</td>
<td>27%</td>
<td>4.2</td>
<td>7.4</td>
</tr>
<tr>
<td></td>
<td>BUCKS LAKE</td>
<td>DWR</td>
<td>5750</td>
<td>44.7</td>
<td>25.6</td>
<td>57%</td>
<td>26.3</td>
<td>30.5</td>
</tr>
<tr>
<td></td>
<td>FOUR TREES</td>
<td>DWR</td>
<td>5150</td>
<td>20.0</td>
<td>10.9</td>
<td>55%</td>
<td>11.2</td>
<td>15.8</td>
</tr>
</tbody>
</table>

YUBA & AMERICAN RIV
equations

\[ \pi_i \prod_{i \neq j} (1 - F_j(p)) \]

\[ = -\pi_i \prod_{i \neq j} (1 - F_j(p)) \]

Dividing one equation into the other, we have

\[ \frac{(1 - F_k(p))^{n-1}}{(1 - F_j(p))^{n-1}} = \frac{1 - (1 - F_k(p))^{n-1}}{1 - (1 - F_j(p))^{n-1}} \]

which implies \( F_j(p) = F_k(p) \).

REFERENCES


This is saved as http://...arpa-anno.mvd
Behaviors with temporal extent

Undersea Garden

Even experienced an Undersea Garden? The colourful reef fishes look like flowers aloft in mid-air. In celebration, the vividly coloured corals sway to the tune of the current.

Note:
The global time source can control (activate or deactivate) behaviors. You can specify an interval relative to the global time source indicating when a selected behavior should be active. Use "Add Time " in the Anno column to associate behaviors with time sources.

Do you know that corals are closely related to jellyfish? Notice the different shades of green, yellow or brown on the corals. These caused by marine algae which photosynthesize and make food to feed the corals.

Clown Fish & Sea Anemone

What are those vertical strands? Watch out, they are razor fish! They swim with their face downwards and look like long razor blades. The surprises at Undersea Garden!
The Multivalent Protocol Suite: Reify
Fundamental Document Lifecycle

Protocols execute methods according to their behaviors’ priority.
- Some have a second ("after") phase in which additional methods are executed in "low-to-high" order
- Some protocols traverse trees

Hub document is the persistent Multivalent object.
Behaviors

- All user-level functionality implemented as behaviors, i.e., system extensions.
- Can be very simple or do a lot.
  - A useful simple behavior might be only a few lines of code, typically leveraging other behaviors.
  - Interestingly, HTML, by far the largest, is only 4000 lines of code (about 5% the size of Mozilla).
- Behaviors invoked by framework according to protocols (= function signatures)
  - I.e., no behavior calls another behavior’s method directly.
Behaviors (con’t)

Some types of behaviors:

- Media Adaptors: OCR (XDOC, PDA), HTML, PDF, LaTeX/DVI, ASCII
- Structural: alt. select-and-paste, tablesort, scrollbar search, vis., Notemarks
- Span: hyperlink, highlight, copy editor mark
- Lens: OCR, magnify, notes, Pilot notes
- Managers: lens coordination, user interface
Multivalent Protocols: Restore Protocol and Hub Document

- External to internal:
  - Instantiates relevant behaviors
    » behaviors initialize
    » some load corresponding layer(s)

- Behaviors to include in a document are listed in a *hub*, an XML document.
  - Content to include is generally an argument to a behavior in a hub.
  - Specifications can be nested; interpretation up to outermost behavior element.
  - Each document gets its own browser!
More About the Hub

- Hubs are like style sheets for functionality.
- In fact,
  - Most documents share lots of functionality,
  - and lots of documents don’t come with an associated hub,
  - so, hubs of are cascaded to determine behaviors.
- The hub cascade:
  - System hub: behaviors to be used by every document (e.g., key bindings, search)
  - Genre hubs: by document type (e.g., scanned image docs get “show ocr” lens)
  - Document specific hub: generally just a few (e.g., annotations)
<MULTIVALENT
  URL="file:H:/wilensky/mvd122/demo/xdoc/620/OCR-XDOC/00000001.xdc"
  PAGES="9"  ORGANIZATION=""  NOTES=""  TYPE="Varian"  SEARCHNB="ON"
  ABSTRACT=""  AUTHOR="Hal Varian"  GENRE="Xdoc"
  BIB-VERSION="CS-TR-v2.0"  TITLE="A Model of Sales"  ID="ELIB//620"
  ENTRY="February 8, 1996"  DATE="February 1996">
  <Layer  NAME="Personal"  BEHAVIOR="multivalent.Layer"  URL="inline">
    <Span  BEHAVIOR="HighlightSpan"  CREATEDAT="941142025281"
      NB="ANNONB"  COLOR="YELLOW"  LENGTH="16">
      <Start  BEHAVIOR="multivalent.Location"
        TREE="0  49/Stiglitz 1/PARA 2/REGION 0/OCR 0/IROOT"
        CONTEXT="Stiglitz and %281977%29.">
      </Start>
      <End  BEHAVIOR="multivalent.Location"
        TREE="7  50/%281977%29. 1/PARA 2/REGION 0/OCR 0/IROOT"
        CONTEXT="%281977%29. Stiglitz They">
      </End>
    </Span>
  </Layer>
</MULTIVALENT>
“Logical-to-union”: Iterates over behaviors, constructing tree for document content.
  – GUI widgets are specialized tree nodes

Runtime data structure: logical/structural tree
  – All media types/genres expose structure for manipulation by other behaviors
  – E.g., augmenting scanned image with table, biblio

Media adaptor behavior bridges between concrete and abstract through leaves, throughout lifecycle
  – E.g., scanned page images = parse XDOC + load image + draw image/OCR

Behaviors request UI categories and elements; system groups and instantiates all requests
The Tree: The Central Data Structure

- Quite ordinary, surprisingly powerful
- All nodes have name, parents, bounding boxes, alignment info, “observers”,…
  - allowing behaviors to move between structural and physical representations.
- Internal nodes have children, capture document structure
- Leaf nodes encapsulate medium-specific qualities.
  - Might paint a word, an image, a clipped image, a video, …
- Division allows other behaviors to be written once against an idealized document tree, and work for any concrete doc format.
  - E.g., annos that put text between lines use same code for HTML, man pages, DVI, scanned paper, PDF, with no hacking
More About The Tree

*Everything* displayed is part of the document tree, *including the UI!*

- Widgets, menu bars, scrollbars are simply other kinds of nodes
  - Perhaps layered on top of other content
- Buttons and textual hyperlinks are just nodes with interactive behaviors attached.
- Everything is laid out by the same process

Behaviors can introduce their own nodes and widgets.
Nested documents are easily implemented as subtrees

- E.g., HTML FRAMEs.
- Notes are subdocuments; since we can annotate documents, we can annotate Notes.
- Widget context is a subtree, so a button can show mix text and images, blink, or be a scanned paper document.
“Logical-to-physical”: Traverse document tree placing elements at geometric locations

Media-specific leaves report dimensions, internal structural nodes position children, i.e., implement layout policies (line breaking, table cells, frames)

Three coordinate systems: screen, absolute document, and for efficiency, parent-child relative

Current display properties in graphics context: font, colors, line spacing, underline, “signals”, ...

Graphic context changed structurally (style sheets), linear range (spans), geometrically (lenses)
More About Formatting

Almost entirely done by format-specific nodes.

However, all nodes must respect a set of core properties during formatting and painting:

- style, alignment, margin features, etc.
- So, e.g., a scanned image node must know that if the background color is changed, it should make background pixels transparent.
Paint Protocol

“System to user”: Paint representation of content on screen within viewport

Printing = reformat + repaint on different canvas

Incremental for good performance with lenses, editing

– In fact, Paint invokes Format of dirty nodes on demand
Paint Example: Move-to annotation

- Determines lowest node common to both arrow end points.
  - Registers interest in that node,
  - Computing coordinates of start and end points relative to common node.

- When there is activity at this node, move-to behavior receives notification.
  - In after phase, behavior draws the arrow between pre-computed coordinates.

- During resetting, behavior unregisters, computes new node, registers again.
Events Protocol and Grabs

• “User-to-system”: For low-level events, e.g., user mouse clicks and keypresses, passed as events to system

• Events distributed to behaviors according to declared interest within tree region
  – e.g., table sorting

• Defaults for usual editing commands (as replaceable behavior)

• Grabs - behavior gets future events directly
  – e.g., hyperlinks
Semantic Events: For Communicating High-level Activities

- Relative infrequent and not performance-critical.
- Sent to all behaviors
- Comprise
  - Message (e.g., openDocument)
  - 3 fields: argument (commonly needed data), in (event sender), out (to collect results from participating behaviors) fields
- Most often acted upon by behaviors to implement a requested action, modify the event, update state in response to an announcement.
DVI is a format that was frozen, but in which new functionality (e.g., hyperlinks) is added by “specials”

Implementation is up to each viewer; involves hacking source code (if available!)

Multivalent DVI media adaptor doesn’t change when “specials” functionality is added.

- Instead, parser announces semantic events
  » passing special string, geometric and logical document positions
- Behaviors implementing specials listen for relevant message, so the required work.
Save Protocol and Robust Locations

“Internal-to-external”: Save reconstitutable description to hub (tag+attributes)

Robust Locations

- Documents change, but rely on registration of parts, especially for annotations
- Redundant descriptions
  - ID - guaranteed correct when available
  - Tree Path - robust to insertion, deletion, change in hierarchy
  - Context - most flexible, but less reliable
Clipboard Protocol and Behavior Interaction

• “System-to-other-system”: Iterate over corresponding media adapters in the selection, each contributing medium-specific representation
• Chunky spans - step by largest wholly-contained subtrees in span
• Before/After/Short-circuit (available on all protocols)
  – e.g., alternative select and paste for biblio
Means of Composition

- Overall coordination by core framework protocols and behavior adherence
- Side-effect on document tree (e.g., table sorting)
- Before/After/Short-circuit (e.g., alt s+p)
- Global and graphics context attributes (e.g., current page number, view as image/OCR)
  - Namespaces
    » of variables with ESIS values.
- Manager behaviors (e.g., lens, UI)
Packaged support for

- Robust locations references
- Spans (across structural boundaries)
- Tree manipulations, traversal
- Templates for lens
- Style sheet-based and fixed-format layout
Multivalent Third Party Work

- PDF developed separately (by Shengdong Zhao)
- Printing, support for other OCR formats, by HP
- Palm Pilots notes, ink (Francis Li)
- Temporal-extent behaviors (Wojciech Matusik)
- Japanese support by NEC; application to office document management
- Chinese character and multilingual lens by UCB Instructional Support staff (Owen McGrath)
Supporting Services

- **Annotation Server (ByungHoon Kang)**
  - Service allows annotation search and storage on DBMS.
  - MVD interface via a behavior.

- **Emailer (ByungHoon Kang)**

- Actually, we’re in the process of rethinking all this...😊
GIS Viewer

- Multivalent applied to geo-referenced data
- GIS data comprise
  - geo-rectified images (raster data)
  - vectors (points, lines, polygons, etc.)
  - geo-positioned data items
- Fits naturally into a layers and behaviors framework
GIS Viewer (cont’)

- Separate Java code-line (by Loretta Willis, Jeff Anderson-Lee)
- GIS Viewer 3.0 released
  - Suitable for general, networked GIS (and other image) apps.
- GIS Viewer 4.0 in beta
  - Supports db querying
- GISlite available
  - Slightly less functionality, much smaller and more “portable”
  - Soon to be interface for Microsoft’s Terraserver.
GIS Viewer 3.0

Supported layer types
- Raster: geo-rectified GIF, JPEG
- Vectors: internal, DLG, ArcInfo Shape files
- Grouping format/protocol: tilePix (data pyramid for raster and vector data)

Behaviors are
- pan
- zoom (with automatic projection transformation)
- change projection
- display context
- display semi-transparently
- “spatial hyperlinks”
- user authoring for annotation
- issue query
Figure 1. Map showing major rivers and other key geographic locations discussed in this status review.
GIS Viewer Example

http://elib.cs.berkeley.edu/annotations/gis/buildings.html
Multivalent: Related Work

- Integration vs Juxtaposition: OpenDoc, OLE, Quill, HTML with Java applets and plug-ins
- Composition vs Shared Library: GNU Emacs
- Deep Extension vs Scripting: Dynamic HTML, Microsoft Word, FrameMaker
- Union vs Confederation: Microcosm, UNIX Guide, Firefly
- Fundamental API vs Open Source: Netscape 5, Tk text widget
- Cross-format vs Single Format: IDVI
Behaviors can define new node types, crucial for
  - non XML-based document types
  - Introducing new widgets

Multivalent covers
  - build, format, paint semantic events
  - In addition to tree, views, style, events, load, save

Java vs. Javascript
  - Hard to compose scripts from different sites.
The “W3C standards set” continues to grow and become more powerful.

- Some (but not all) Multivalent-only functionality could be done by browsers implementing CSS2, ECMAScript, XLink, XPointer, etc.
  » probably also requiring limited extensions (applets or plug-ins)

The Multivalent approach requires providing all this functionality within our framework.

- Even providing good layout, robust HTML parsing is a lot of work.
  » Fortunately, most of the hard part has been done.
- We (at least somewhat) ride the wave of Java improvements.
- We will ultimately depend on a Linux-like network of developments to have a competitive open-source platform.
Multivalent Ongoing Developments

- Support for more genres
  - Perfect fixed image media adaptors: PDF, DVI, other wordbox
  - HTML 3.2 (just about) supported, on to 4….
  - XML parser exists; add XSL, etc.
  - Composite document support
  - Javascript (via Rhino JavaScript-in-Java)
  - Applets, video, etc. via JMF 2.0
Multivalent Developments (con’t)

- Completing CSS support
- Perfect annotative behaviors
- Work out multi-user annotation discipline
- Annotation service issues
  - security, groups, self-administering documents
- Harden, tune user interface, programmers’ guide
  - See on-line behavior writer’s guide.
- Release to community, get feedback, iterate, use in DLIB project.
Layers of Content