

Abstract on AUGMENT

Written by : **Jamil Itmazi**, Supervised by : **Dr. Lina Cabrera**, Hypermedia Systems, Granada Univ.

This Abstract is based on :

"Collaboration Support Provisions in AUGMENT", Douglas C. Engelbart, Tymshare, Inc., www.bootstrap.org/augment/OAD/2221.html

"Authorship Provisions in AUGMENT", Douglas C. Engelbart Tymshare, Inc., www.bootstrap.org/augment/OAD/2250.html

"Workstation History & the Augmented Knowledge Workshop", D.C. Engelbart McDonnell Douglas Co., bootstrap.org/augment/AUGMENT/101931.html

This paper summarizes some of the principles, concepts & special provisions embodied in AUGMENT that most directly provide this collaborative support :

AUGMENT is an integrated (text processing) system of knowledge-worker tools originated at SRI International. (name change into "NLS" & the commercial rights were transferred to Tymshare in 1978), The system evolved on time-shared, mainframe computers, & in a packet-switched network environment, Special attributes of the timesharing & the network environments contribute uniquely to the support of collaboration & Authorship has received a great deal of attention in it , *its* architecture, design principles, & specific features, is directly aimed toward bringing high performance to the authorship activities of knowledge workers.



Doug Engelbart

- **The sub-goal in AUGMENT's** development was to "augment" the development, production & control of complex technical documentation - gathering information, planning, creating, reviewing, editing, controlling versions, designing layout, & producing the final documents.
- AUGMENT's most unique architectural feature is its User Interface System (AUGMENT's frontend), a special s/w module, which handles the human/comp. interfaces to all interactive programs. It takes care of all command-language dialog & connection protocols, & provides a framework for building a coherent & integrated user environment while supporting flexible evolution on both user's side & technology side.
- AUGMENT's architecture provides for open-ended expansion & flexible evolution of system functionality & worker command languages using mouse & one-handed chord keyset
- Many options exist for controlling the "view" of a file's text in a window, e.g.: level clipping, paragraph truncation, & content filtering. Structural study & modification of on-line documents are especially facilitated.
- **File Characteristics** : AUGMENT employs explicitly structured files, with hierarchically organized nodes; each can contain up to 2,000 characters of text, a graphic structure. The worker has a definite model in mind for the structuring of any file that he works with; in composing & modifying it he can organize & modify structure using the same verbs as for working with text strings (e.g. Insert, Replace, Copy, Delete), with appropriate structural-entity nouns (e.g., Statement, Branch, Group). For any existing hierarchical structure, he has many flexible alternatives for addressing its entities, modifying its organization, jumping around within it, & viewing it in a most beneficial manner.
- **Augment Televiewing** : where skilled users can collaborate with lesser-skilled people.
- **File Addressing & Embedded Links** : Any given text entity in any AUGMENT document can be addressed, There is a set of addressing features that a worker may use in any command to designate a particular structural node or some element of text or graphics attached to that node.
- **Shared Files** : users able to share the use of files in the process of collaboration, AUGMENT automatically maintains an authorship-change record for each statement in each file, indicating the date, time, & author of the statement's creation or last change.
- **Augment Mail** : The system provides speed & flexibility for all message-processing tasks, including composing, addressing, acknowledging, answering, forwarding, studying, & filing them. This allows our users to collaborate with high effectiveness & the content of a mail item is actually an AUGMENT document & derived from use of AUGMENT links.
- **Augment Journal** : supports a recorded form of dialog having attributes similar to professional journals & the libraries that store, catalog & provide access to them, Having created an on-line record of thoughts & ideas, the author may direct the system to distribute it to a larger group. Such a contribution may from a one sentence message to a 200 pages doc. the author specifies a title, & comments & a distribution list , Other fields will be added automatically upon submission, such as date & time. the item is automatically given a number & is stored permanently in a central location. A full bibliographical citation will be generated & installed in library-like, computer-held catalogs, No one is able to modify it.

Jamil Itmazi, 2/2/2003

Abstract on Intermedia

Concept and the Construction of a Seamless Information Environment

Written by : *Jamil Itmazi*, Supervised by : *Dr. Lina Cabrera*, Hypermedia Systems, Granada Univ.

This Abstract is based on :

- N. Yankelovich et al., "Intermedia: The Concept and the Construction of a Seamless Information Environment," Computer, vol. 21 Iss. 1, pp. 81-96, Jan. 1988. (16 pages)

Intermedia is a tools designed to support both teaching and research in a University environment, contains multiple applications and mechanisms to link the content of documents created with those applications.

The systems was developed at Brown Univ.'s institute for research in information & scholarship (IRIS).

Nelson coined the term hypermedia in early 1960s to describe the idea of "non-sequential writing. Hypermedia is simply an extension of hypertext that incorporates other media in addition to text include static graphics, animated graphics, video and sound.

Intermedia is both an author's tools & a reader's tools.

Abstract:

the systems (which runs on a network of UNIX-based workstation) is a collection of five base layer applications and a hypermedia application. Users select regions of data in the data rendered in base layer applications and link them. The links and their properties are stored in a separate web, not in base documents. The hypermedia application displays the links in all documents opened in the base layer applications. The base layer applications are a part of the system. That is, the base layer applications and the hypermedia application are designed to interoperate among each other from the ground up. The system does not provide linking to documents in applications other than these five applications, the system supports relational and associative links only; it does not allow an notational links. The two ends of the links must be regions in documents (same/different).

The application are:

- 1- The InterText (a text editor/word processing using style shit).
- 2- The InterDraw (a graphics editor , users can create two-dimensional illustration by selecting tools from a palette attached to it's window) .
- 3- The interpix (a scanned image viewer , which can be cropped, copied, and pasted into Interdraw doc.'s).
- 4- The Interspect (a 3-dim. Object viewer-representation).
- 5- The InterVal (a timeline editor).

Although Intermedia appears to have very limited features, it seems to be a good application for its time. It is conceivable that developing compatible base layer applications consumed significant resources. This is evident from information in Section "Intermedia: the construction" and onwards.. Fortunately, most current and applications provide some form of application programming interface to alleviate the need to develop compatible base layer applications.

Jamil Itmazi, 4/2/2003

Abstract on A Partitioning Concept for Hypertext

Written by : **Jamil Itmazi**, Supervised by : **Dr. Lina Cabrera**, Hypermedia Systems, Granada Univ.

This Abstract is based on :

Norman M. Delisle, Mayer D. Schwartz: Contexts - A Partitioning Concept for Hypertext. TOIS 5(2): 168-186(1987)

Hypertext systems provide good information management support for a wide variety of documentation efforts. These efforts range from developing software to writing a book. However, existing hypertext systems provide poor support for collaboration among teams of authors.

This paper starts by briefly describing properties of several existing hypertext systems. Then several models for forming partitions in a hypertext database are examined and contexts, a partitioning scheme that supports multiperson cooperative efforts, are introduced. The semantic issues involved in defining contexts are explored in detail.

The essential idea of *hypertext* is that of *nonlinear or non-sequential* text, In hypertext system, documents consist of a collection of nodes connected by directed links.

A node itself not only text but also graphical images, digitally voice or even animation, & the Links can be made between a node in one document & another node in the same or different document.

Hyperdocument is the complete collection of documents.

Properties of Hypertext Systems : this section outlines of the important properties –

- 1- Editing Hyperdocuments : the ability to create/delete nodes & links, modify both the information in nodes & structure of the hyperdocuments.
- 2- Traversing the Hyperdocuments : the directed-graph structure supported by hypertext systems can be used by authors as the means for structuring documents.
- 3- Multimedia Content : the contents of a node can be arbitrary digital data whose interpretation may include graphics, animations, or digitized speech.
- 4- Multiperson Distributed Access : Several persons can access a hyperdocument simultaneously and the hyperdocument itself can be distributed over multiple networked machines.
- 5- Interactive user interface : A hypertext document is meant to be viewed interactively. As the reader views a node, visible links may be followed or not at the discretion of the reader.

Limitations of existing hypertext systems : All users make change within a single shared storage system. This situation results in numerous conflicts when several authors are collaborating on document.

The areas in which existing hypertext system provide inadequate support for collaboration in a design-support environment :

- 1- A design-support should provide a means for organizing related sets of hypertext nodes & links.
- 2- There should be a way for teams of authors to work together in independent hypertext partitions without the risk of interfering with each other & to allow those independent partitions to be joined at carefully controlled intervals.
- 3- The desire to allow version trees to be built & subsequently, specific branches of the tree to be installed as the primary version.
- 4- If large team of programmers are collaborating, the hypertext database should be distributed across several computers retaining rapid access.
- 5- There should be easy means of allowing configurations to be built and used.

Jamil Itmazi, 6/2/2003

Introduction to Trellis :

a Formally-defined Hypertextual Basis for Integrating Task and Information

Written by : *Jamil Itmazi*, Supervised by : *Dr. Lina Cabrera*, Hypermedia Systems, Granada Univ.

This Abstract is based on : R. Furuta and P. D. Stotts, "Trellis: A Formally-defined Hypertextual Basis for Integrating Task and Information," in *Coordination Theory and Collaboration Technology*, G. M. Olson, T. M. Malone, J. B. Smith (eds.), Lawrence Erlbaum Assoc., 2001, pp. 341-367.

The Trellis project *is* investigating the structure and semantics of hypertextually-described interaction, the study of hyperdocuments broadened to encompass *hyperprograms*. A hyperprogram associates user-manipulatable information (the hypertext) with user-directed execution behavior (the process).

A hyperprogram can be said to integrate task with information, The Trellis model is formally defined using timed, colored Petri nets. Hypertexts in Trellis describe the traditional hypermedia collection of data objects and relationships among the objects, and also define the process by which the objects are used.

Trellis' notion of "hypertext" is very broad, static content (text and graphics), and active content, (video and audio), and procedural content (computations and embedded hypertexts).

Trellis implementations are based around a distributed client-server architecture, and Trellis document defines an environment in which the constituent objects may be dynamic and in which the state of the document also is changing dynamically.

This paper then give a little more detail about Trellis & its prototype implementations & examine a series of example applications of Trellis illustrating its application in hyperprogram specification & then reviews the efforts along those lines then concludes the presentation.

In (Hyperprograms) section they examine 4 applications of Trellis in varying domains. The first three are presented using an older prototype, *_Trellis*, & show Trellis' application in specifying an image browsing index, in representing the message flow in a parallel programming language, & in simulating solutions to classical problems in process synchronization. The presentations will introduce additional characteristics of Trellis implementations, & The fourth example, cast now in the newer prototype, *_Trellis*, illustrates the implementation of a protocol for managing a collaborative meeting.

Trellis & its implementations provide a formal structure for hyperprograms, & net analysis techniques have been developed for exploiting this formalism.

The emergence of hypertext as an authoring-based prototyping mechanism suggests a modification of the view of the role of hypertext, In this work, the hypertext system retains its traditional job of information structuring but also gains the job of process structuring.

The Trellis net-based representation provides a means for specifying a protocol that can be used directly for verification, training, & simulation.

the Trellis implementation decision to interpret the underlying net suggests further experimentation with an incremental methodology in protocol development & in incremental protocol verification. As prototyping proceeds, more sophisticated behaviors are added, refined, & verified. There is a nice symmetry between this view of protocol development & the world-view of a hypertext as a dynamic web, growing & developing as new paths are encountered & envisioned.

Jamil Itmazi, 2/2/2003

Introduction to : (The Flag Taxonomy of Open Hypermedia Systems)

Written by : *Jamil Itmazi*, Supervised by : *Dr. Lina Cabrera*, Hypermedia Systems, Granada Univ.

This Abstract is based on : Østerbye, K. & Wiil, U. K. The Flag Taxonomy of Open Hypermedia Systems. In Proceedings of the Seventh ACM Conference on Hypertext (Hypertext '96), Washington, DC, March 1996, pp. 129-139. ACM Press.

This paper is consist of :-

- **Abstract & Introduction** : This paper presents a taxonomy for open hypermedia systems. The Flag taxonomy provides a framework which allow us to:
 - Classify existing hypermedia systems.
 - Characterize what an open hypermedia system is.
 - 3-examine (describe & compare) OHSs independent of the particularities of specific sys.
 - To provide an overview of the design space of open hypermedia systems/ differentiate between important features such as openness & tailorability.

The Flag taxonomy builds on the achievements of the Dexter model. It extends the terminology of the Dexter model to adequately cover issues that relate to open hypermedia systems such as integration & use of third-party applications to edit & display hypermedia components, the current trend in hypermedia systems design is towards open, extensible & distributed multi-user systems.

Within the framework of the Flag, He presented a number of issues & possible solutions, in particular for the session manager & viewer modules, which can serve as the basis for a more detailed description of existing OHSs & as a starting point for design of new OHSs.

- **In Section 2**, he shows how the taxonomy can be used to classify existing hypermedia systems into broad categories (monolithic/ Hyperbase/ Embedded link/ Link server/ Open hyperbase) & to distinguish OHSs from other hypermedia systems, broad classification will often ignore interesting details, For example, the WWW was characterized as a closed hypermedia system, because it does not have a linking protocol. It consists of 2 main functional modules, a server & a browser.
- **In Section 3**, He focus on runtime aspects :
 - The session manager module.
 - The viewer module.
 - The linking protocol & its relation to the viewer & the session manager.

To validate the usefulness of the taxonomy, he describes & contrasts 2 prominent OHSs

- **In Section 4**, He presented a description & comparison of 2 existing OHSs, **DHM & Microcosm**. The descriptions build strongly on the vocabulary established in the taxonomy. None of these descriptions are very long, the taxonomy provides a framework to understand & evaluate the general characteristics of OHSs with minimal effort. There are of course more subtle aspects of these systems which can not be captured by describing the systems according to the taxonomy. It serves as a check list to ensure that one has indeed covered the necessary aspects of an OHS.
- **Section 5**, He concludes the paper.

Jamil Itmazi, 22/2/2003

Abstract on

(Hypertext Paths & the WWW: Experiences with Walden's Paths)

Written by : *Jamil Itmazi*, Supervised by : *Dr. Lina Cabrera*, Hypermedia Systems, Granada Univ.

This Abstract is based on :

- Hypertext Paths & the World-Wide Web: Experiences with Walden's Paths/ Richard Furuta, Frank M. Shipman III, Catherine C. Marshall, Donald Brenner, & Hao-wei Hsieh

Walden's Paths applies the concept of hypertextual paths to the WWW. Walden's Paths is being developed for use in the K–12 school environment. The heterogeneity of the Web coupled with the desirability of supporting the teacher student relationship make this an interesting & challenging project. he describe the Walden's Paths implementation, discuss the elements that affected its design & architecture, & report on our experiences with the system in use.

Hypertext paths (tours) provide a meta-structuring mechanism that allows the affiliation of elements from many different hypertexts.

Paths appear in implemented systems beginning around 1988 in Trigg's Guided Tours.

The WWW provides an entirely new context for hypertext application. Although the technology of the Web is largely derivative from earlier applications, the environment that it has created is novel. Unlike its predecessors, the Web is highly heterogeneous, both in readers but also in information provided. Readers range from highly-educated academicians to elementary school students. Ages range from retirees to pre-schoolers. People make material available on the Web for all conceivable reasons—to communicate, educate, persuade, promote, defraud, & delude, to name just a few of the possibilities. Path-like mechanisms that support personal-space organization are commonplace in Web browsers.

In this paper he discuss Walden's Paths, a Web-based path implementation. The basic metaphor is that of a network-based meta-document: a document whose elements are themselves documents. In general the author of the path is *not* the author of the supporting documents, a characteristic that raises interesting rhetorical & societal issues.

2 STUDENTS & THE WEB > This section describes the characteristics of the target student population He discuss about How do students experience the WWW as it is today, using Netscape's browser to explore the Web, each student who participated in a session sat at his or her own PC, so each was free to explore , he noticed out their work, their dialogue & their interactive.

3 MECHANISM > While the Internet provides access to a wealth of new multimedia materials, especially through the extensive, encyclopaedic materials available on the World-Wide Web, the students need to be provided a focus based on the curricular objectives. Walden's Paths is designed to enable teachers to make use of materials available via the Internet by creating directed paths over the WWW. Besides providing an ordering of pages, Walden's Paths allows teachers to provide additional context for the page through annotation. By providing text or other annotations in addition to the content of the page, the teacher may provide a rhetorical structure to the path as a whole, create transitions to fill in informational gaps between pages, & create emphasis to particular aspects of the materials.

4 ARCHITECTURE > Walden's Paths consists of 2 components: the Path Authoring Tool & the Path Server. The Path Authoring Tool enables the teacher to create, modify, validate, & reuse paths. When a path is saved by the teacher it is stored with the Path Server. The Path Server is the implementation means through which students navigate the paths. In order to avoid requiring students or teachers to use specific software, the Path Server works as an intermediary between the students' Web

5 EXPERIENCE WITH USE OF WALDEN'S PATHS > he talk about teachers experience with use of walden's paths. most of the teachers focused on the content contained within the path. once again, the material of importance was the contents rather than the container. he find the message of these effects to be both reassuring but also cautionary. The effects are reassuring because they are precisely what he wish to attain—the provision of new layers of meta-structure that enhance rather than diminish access to the content that they place into new contexts. But these reactions also remind us of the changing nature of ownership & attribution for networked information; effects that have implications for the models he implement. he return to this point subsequently.

6 META-DOCUMENTS, DOCUMENTS, & WALDEN'S PATHS > Paths are an interesting form of document: one in which the constituent parts are themselves documents, not the more primitive elements that make up most documents. The rhetorical purpose of a path is likely to be distinct from that of its components. Components frequently are part of a tightly interlinked network devoted to elucidation of a particular point of view about a selected set of topics. On the other hand the path is a unifying structure, directed towards contextualizing multiple points of view about a topic into an organizing framework. Because of Walden's Paths' use in conveying curriculum from teacher to student, there is a strong need to be able to make explicit the path's the rhetorical structure

7 EXTENSION & CURRENT DEVELOPMENTS> This section describes some of the mechanisms that are currently being prototyped, relating them to the issues discussed in the previous section.

7.1 Attribution & separation of content from annotation : The architecture's advantage of using the client-supplied user interface also requires that the annotation share display space with original content. While he can use graphical conventions to reduce confusion (and he continue to refine our designs to accomplish this), the possibility remains that *some* referenced page could be designed in a way to cause confusion.

7.2 Encouraging student engagement : Teachers have told us that they want active ways of generating student engagement with Web-based materials—they wish to avoid passive browsing behaviours, but instead want to make sure that the students are reading & comprehending the material presented. Based on teacher suggestions, he have implemented several mechanisms & are in the process of evaluating them

7.3 Richer path structures : Adopting the KISS approach in designing our path model has been a good choice in most cases. The simple form seemed particularly effective for paths that were based on the extended bookmark metaphor, as each stop in these paths tended to point to a large number of off-the-path resources.

8 DISCUSSION & FUTURE DIRECTIONS > Our initial focus in the Walden's Paths project has been supporting teacher to student discourse. As the project matures, he are also interested in investigating the mechanism's potential for supporting communication in the other direction—from student to teacher.

in this project he have found the WWW provides a provocative environment in which to conduct research. A strong opportunity exists in the Web for transfer of the results of earlier hypertext research. However the distributed, unregulated, heterogeneous nature of the Web raises issues that often did not appear in the original environments, which were more homogeneous both in the sense of users but also in the sense of software architecture. It is interesting to note that even with its heterogeneity, the Web remains surprisingly dependent on the good will of its users in support of its continuing technological survival. Much of the infrastructure of the Web appears to depend on adherence to "gentlemen's agreements," primarily relying on peer pressure for enforcement. Discovering, respecting, & negotiating such agreements adds an extra element of novelty to research in the Web environment.

Jamil Itmazi, 18/3/2003