UNIVERSITY OF SOUTHAMPTON
Faculty of Engineering and Applied Science
Department of Electronics and Computer Science

A mini-thesis submitted for transfer from MPhil to PhD

Supervisor: Leslie Carr
Examiner: Nigel Shadbolt

An Annotative Approach to Better Hyperauthoring and Associative Linking

by Timothy J. Miles-Board

May 2001

UNIVERSITY OF SOUTHAMPTON

ABSTRACT

FACULTY OF ENGINEERING AND APPLIED SCIENCE
DEPARTMENT OF ELECTRONICS AND COMPUTER SCIENCE

A mini-thesis submitted for transfer from MPhil to PhD

by Timothy J. Miles-Board

Early hypertext visionaries proposed entire online archives of the world's literature, with everything associatively linked to everything else. Today, the most widespread hypertext system is the World-Wide Web (WWW), a publicly accessible and globally distributed medium. However, the WWW is not living up to the promise of hypertext associativity - the majority of hypertext linking on the WWW is estimated to be intended for navigational purposes only.

WWW authors typically have new ideas to contribute, and assert particular relationships between these and existing ideas already published in order to demonstrate both the reliability of the conceptual foundation being built on, and the innovation and significance of the new ideas. However, these associations are rarely rendered as associative links which seamlessly link the new material into the global context. This research investigates the possibility of capturing these implicit inter-document associations through annotation, and then using these annotations to assist the hyperauthoring process.

The hypothesis of this work is therefore that by capturing inter-document associations through annotation, a better hyperauthoring process will result, both in terms of the quality and coverage of the new writing, and in terms of the seamless (associative) integration with the global context, helping the WWW evolve to achieve all of its potential hypertextual richness.

The **A**nnotation **LI**nking **EN**vironment (ALIEN) has been implemented to demonstrate techniques for capturing inter-document associations made by an author whilst reading, using free form annotations.

Further work proposed includes the re-purposing of these captured associations to assist the authoring and linking processes through dynamic visualisation of the association structures "as-you-type", and automatic associative linking.

Contents

List of Figures

Chapter 1

Introduction

Early hypertext visionaries (such as Bush Bush (1945) and Nelson Nelson (1980)) proposed entire online archives of the world's literature, with everything associatively linked to everything else. Hyperauthors would have no need to reiterate material that had already been written - readers follow associative links from concepts used in newer writings to explanations that have been written before, to a depth that satisfies their own personal requirement for understanding.

Today, the most widespread hypertext system is the World-Wide Web (WWW), a publicly accessible and globally distributed medium. However, the WWW is not living up to the promise of hypertext associativity - the majority (as much as 99%) of hypertext linking on the WWW has been estimated to be intended for navigational purposes only Carr et al. (2000). Factors contributing to this lack of associative linking activity on the WWW have been identified as the fact that authoring associative links incurs a significant development effort Mendes and Hall (2000), and recommendations that associative linking be minimised to avoid well documented problems such as readers becoming "lost in hyperspace" and suffering cognitive overload as a result (for example Lynch and Horton (1997)).

WWW authors typically have new ideas to contribute, and assert particular relationships between these and existing ideas already published in order to demonstrate both the reliability of the conceptual foundation being built on, and the innovation and significance of the new ideas. However, these associations are rarely rendered as associative links which seamlessly link the new material into the global context. This research investigates the possibility of capturing these implicit inter-document associations through annotation, and then using these annotations to assist the hyperauthoring process.

The hypothesis of this work is therefore that by capturing inter-document associations through annotation, a better hyperauthoring process will result, both in terms of the quality and coverage of the new writing, and in terms of the seamless (associative) integration with the global context, helping the WWW evolve to achieve all of its potential hypertextual richness. Annotation is an integral and fluent part of reading, so it is posited that the effort required to author inter-document associations in this manner will be reduced. Furthermore, by providing annotation tools that allow the reader to capture associations in a meaningful way, problems arising from the net-like characteristics of the resulting hypertext by can potentially be alleviated by disciplining the hypertext network with semantic information.

1.1 Roadmap

Chapter 2 provides a more detailed discussion of the theoretical background of this hypothesis, followed by reviews of previous approaches in hyperauthoring systems research (Chapter 3) and annotation systems research (Chapter 4) which are particularly relevant to this work.

Chapter 5 introduces the implementation work carried out, primarily focusing on the **A**nnotation **LI**nking **EN**vironment (ALIEN) system, which demonstrates techniques for capturing inter-document associations made by an author whilst reading, using free form annotations.

Chapter 6 outlines a number of proposals for future research directions, including the implementation of a mechanism for re-purposing captured associations and using them to assist the authoring and linking processes through dynamic visualisation of the association structures "as-you-type", the transclusion of annotated material, and automatic associative linking.

Chapter 7 concludes this report.

Chapter 2

Background

2.1 The promise of hypertext

Hypertext in its purest form is an associative network of nodes (semantically discrete concepts or ideas) which are connected by links, mimicking the brain's ability to access information quickly and intuitively by reference Fiderio (1988). Indeed, Engelbart saw hypertext as a means to augment human intellect Engelbart (1963).

Conklin Conklin (1987) proposed that true hypertext systems should provide computer supported navigation through hypertextual space, and Bush's conceptual Memex system Bush (1945) - a huge database of the world's literature that everyone can access - first envisioned this. Through associative links, Memex offers parenthetical information, footnotes, digressions, or parallel themes that enrich the main content, providing a computer-based medium for thinking and communicating.

Nelson, the originator of the word "hypertext" itself also envisioned an online network of the world's literature using his Xanadu system Nelson (1980).

The promise of hypertext is that it allows everything to be associatively linked to everything else, with no need for authors to reiterate material that has already been written - readers can follow associative links from concepts used in newer writings to explanations that have been written before, to a depth that satisfies their own personal requirement for understanding, and become fully immersed in the hypertext network.

2.2 Hypertext on the World-Wide Web

The World-Wide Web (WWW) Berners-Lee (1989) is a publicly accessible, globally distributed, hypertext platform. The promise of hypertext means that every document published on the WWW can potentially be associatively linked to every other related document, effecting the introduction of a new non-sequential (hyper) writing paradigm. However, the WWW is not living up to this promise.

In an investigation of linking practices on the WWW, Carr et al Carr et al. (2000) concluded that, largely by inspection, the majority (as much as 99%) of hypertext linking on the WWW is for navigational purposes (either in structured site navigation menus, or ad-hoc, non-hierarchical "related information" links), after finding few examples of associative linking from within the actual content of hypertext documents.

The WWW therefore exemplifies Moulthrop's concept of *local coherence* Moulthrop (1992), where nodes exist as stable units which stand in isolation connected purely by transitional links, rather than *global coherence*, whereby nodes serve to associatively augment the global context.

Current research suggests three contributing factors to the current state of the WWW:

- Widespread concerns that hypertext users face problems of disorientation and cognitive overload have led to recommendations that linking on the WWW be restricted to serve navigational purposes only Lynch and Horton (1997).
- Similarly, concerns that the net-like character of true associative hypertext makes it hard for users to grasp the overall structure of a document and understand the meaning of its links have led to calls for such associative linking to be avoided unless structured or otherwise meaningful links can be constructed De Young (1990).
- The hypertext community has always maintained that the authoring of associative links is *hard*; hypertext development metrics have been used to demonstrate that the number of links desired in a hypertext application does increase the development effort significantly Mendes and Hall (2000).

These factors are discussed in more detail in the following sections.

2.2.1 The navigation problem

Hypertext readers as well as authors have to struggle with a variety of problems arising from the net-like character of hypertext, typically having to face two major difficulties. The first is known as the *navigation problem*; Conklin describes this as the "lost in hyperspace" problem Conklin (1987). Problems of cognitive overload arise from the additional work the user has to do to keep track of their location in hyperspace.

Much research has attempted to address these potential problems, and proposed solutions include the use of navigational metaphors (for example the book- and newspaper-like metaphors presented to the user by the OpenBook Ichimura and Matsushita (1993) and VOIR Golovchinsky and Chignell (1997) systems), sophisticated overview maps (Gloor (1991); Kreutz et al. (1999); Nanard and Nanard (1991); Bieber and Wan (1994); Robert and Lecolinet (1998)), link reduction algorithms (Walden's Paths Furuta et al. (1997)), and recommender systems (MEMOIR Pikrakis et al. (1998)).

Lynch and Horton Lynch and Horton (1997), the authors of the influential *Yale C/AIM Web Style Guide*, express strong concern about associative linking on the WWW, claiming that overuse or poor placement of associative links can cause disorientation. Associative linking is claimed to disrupt the narrative flow by inviting readers to go elsewhere, and instead of enhancing the reader's understanding of a subject, an associative link may "send them to a foreign land without a guide" in the attempt to discern the link's significance.

Bernstein validates widespread concerns of confusing the reader through associative linking, but disagrees with the strategy of restricting the role of links in WWW documents Bernstein (1991). Bernstein's view is that disorientation arises not from the fact that information is associatively linked, but the fact that it is linked badly. Bernstein further claims that no convincing evidence exists that associatively linked information necessarily prevents readers from getting "lost", a claim supported by Davies et al Davies et al. (1998).

2.2.2 The comprehension problem

Research which focuses on the navigation problem often overlooks the second difficulty arising from associative hypertexts: *comprehension*. Readers often fail to grasp the overall structure of a hyperdocument or to understand the semantics of its links Streitz et al. (1992), leading to calls for "structured" or otherwise disciplined hypertext.

Glushko, in a commentary on the issues involved in creating multi-document hypertexts Glushko (1989), rejects the Xanadu hypertext model in which multiple documents are integrated to the extent that their separate structures disappear Nelson (1980), favouring instead a solution which limits associative linking in order to preserve structural and contextual clues.

The use of link taxonomies to disipline the hypertext network has been described by researchers such as DeRose DeRose (1989), Trigg Trigg (1983) and Baron Baron et al. (1996).

De Young De Young (1990) proposed that hypertext networks should be disciplined still further, and advocates the provision of general structures on to hypertext links which reflect the structure of the task being performed, claiming that this would mean that users no longer have to think in terms of links or navigation, but simply access the information they need.

2.2.3 The linking effort problem

Mendes et al Mendes and Hall (1999, 2000) have investigated the use of software engineering metrics in measuring quality characteristics of hypertext applications and development processes, in an attempt to provide metrics for estimating the effort required to develop new hypertext applications. They report that hypertext applications which have a higher number of links will contribute significantly to a larger developmental effort. Mendes et al suggest that by adding type and context information to links at creation time, this information may be used in order to facilitate link reuse in later developments.

Some research has proposed automatic link suggestion mechanisms for link authors (for example Bernstein (1990)), in an attempt to reduce the authoring effort. However, observers such as Glushko Glushko (1989) posit that associative links are better identified by the hypertext author rather than extracted automatically (for example, with natural language processing mechanisms), since the author will create links based on their understanding and context of the document, in terms of their writing goals - considerations that are difficult, if not impossible, to reproduce mechanically.

2.3 Linking with annotations

2.3.1 Annotations and reading

In a diary study describing how people read in the course of their daily working lives, Adler et al Adler et al. (1998) report that *cross-referencing* was the most frequently observed reading mode. This provides an indication of the scale on which associative linking between documents could take place - cross-referencing between multiple documents is a common activity. The study also highlighted the fact that reading frequently co-occurred with some form of writing, with a significant amount of time spent *annotating* documents whilst reading them.

Marshall Marshall (1998) takes up the question of annotation as a fundamental activity for hypertext readers, arguing that past research has been geared more towards the initial design of a hypertext, rather than the annotation of existing materials, and therefore emphasises the role of the *reader-as-navigator* rather than the role of the *reader-as-annotator*.

From a hypertext perspective, annotation facilitates promote growth and change by way of addition, with readers augmenting an existing body of material through commentary, the creation of new connections and pathways, and the promotion of structural coherence between gathered materials Marshall (1998). In doing so, they crucially augment an existing body of interrelated materials, increasing the value of the hypertext for future readers. Indeed, such an annotative perspective has contributed to foundational

- **Form**
  - Formal (structured metadata) ⟷ Informal (readers marginalia)
  - Explicit (intended for others to read) ⟷ Tacit (personal)
- **Function**
  - Writing (polyvocal, participatory medium) ⟷ Reading
  - Hyperextensive ⟷ Extensive ⟷ Intensive (see Levy (1997))
  - Permanant (always available) ⟷ Transient (last only for duration of reading)
- **Role**
  - Published (visible to all) ⟷ Private (visible only to annotator)
  - Global ⟷ Institutional ⟷ Workgroup ⟷ Personal

FIGURE 2.1: Marshall's dimensions of annotation Marshall (1998)

work in the hypertext field. The Memex machine envisioned by Bush Bush (1945) focuses on annotation through trail blazing, Nelson's Xanadu Nelson (1980) introduced a transclusive approach whereby new hypertext seamlessly assimilates portions of older writings, and Engelbart's Augment system Engelbart (1963) emphasised a capacity for journal system commentary.

Marshall provides a useful overview of the dimensions of annotation Marshall (1998) (Figure 2.1). Annotation has variously been interpreted as link making, path building, commentary, marking in and around existing text, decentering of authority, as a record of reading and interpretation, or as community memory.

After carrying out an investigation of the annotations made in paper-based textbooks, Marshall posited that annotations on paper are in fact *hypertextual* in nature, existing in non-linear relationships to the printed linear text, interrupting and orthogonal to linear reading, connecting disparate passages, and generally functioning as hypertext is intended to.

Readers create hypertext links on paper by associating annotations with the printed document elements. Marshall observed four kinds of associations that readers make:

Composite associations Annotations that refer to many subparts of a single document, for example a note that refers to an entire chapter.

Node-to-annotation associations Produced when annotations are localised to a particular document part, but don't visibly refer to any particular document element, for example a note written orthogonally to the printing on a page.

"Standard" hypertext associations Formed when an anchored portion of text is associated directly with an annotation using a connecting arrow, bracket, brace, or other spanning mark, or by simply using proximity to imply a relation.

Word-to-word associations Particularly likely to be constructed in foreign language texts where a word is directly associated with its translation.

Readers may also use annotations to create new nodes from document segments, often because the author's structure does not suit the readers purposes. Marshall observed techniques such as the hierarchical numeration of passages, and the switching of colours between segments in order to achieve this.

2.3.2 A simple model of writing

Carr Carr (1995) describes a simple model of writing in which an author first adopts the role of *reader*, searching for relevant ideas and information in the literature, then adopts the role of *censor*, choosing the information appropriate to the task. Finally they adopt the role of *organiser* to arrange the selected information within an informal structure. In the role of organiser, the author may partition and sort the gathered information into themes. Since each theme may be divided into subthemes requiring further partitioning and sorting, this process is inherently iterative. Using this structure as a guide, the author then adopts the role of *writer* and actually produces new material, revisiting earlier roles as new ideas initiate further searches, censorship, and organisation.

2.3.3 Making associations with annotations

We have seen that annotating frequently occurs with reading, and that reading is both a precursor and ongoing part of the writing process. The hypertext author in the role of reader is actually producing annotative constructs which represent an early form of the final hypertext.

Marshall's work shows how annotative hypertexual links can be produced by the reader within a single document, but Carr's model of the authoring process implies that authors typically read many more than one document in order to obtain a clearer picture of how the new writing fits into the context of what has been written before. Indeed, Adler et al Adler et al. (1998) frequently observed the practice of reading from multiple sources, and furthermore found that readers often read from multiple sources in parallel.

Experience tells us that as a number of documents are read, implicit associations are discovered between them (for example, different viewpoints, supporting ideas). Usually the author has some new ideas to contribute, and also asserts particular relationships between these and existing ideas already published in order to demonstrate both the reliability of the conceptual foundation being built on, and the innovation and significance of the new ideas, linking the new material into the global context. It is these implicit relationships that are closest to the vision of hypertext: associative links that are not explicit between related documents but that can be extracted by careful and creative analysis of the texts and the relationships between them. However, these implicit associations exist only in the authors short-term memory, and are in danger of being forgotten, unless they can be made explicit through annotation of the documents.

Nelson visualised two documents positioned side by side on a computer screen, with interconnecting lines drawn between related ideas Nelson (1980). In a paper medium, it would be difficult to capture associative links between documents by drawing lines between ideas.

Alternatively, readers may use symbols to link physically separate document sections, for example an asterisk in the margin next to several paragraphs selected from a number of documents indicates that the ideas presented in these paragraphs are related. However, it then becomes difficult for the reader to remember which symbols have been used to mark up which type of relation, especially as the number of unique symbols used increases. It also becomes difficult to gather together all related ideas once writing begins, since, in the worst case, the reader must search through all the documents to find them.

Price et al Price et al. (1998a) describe how the XLibris personal annotation system could be extended to provide support for linking using symbols in the computer medium. Digital ink marks which are circled become "ink anchors": passages are linked together by annotating each passage with a similar ink anchor. Activating an ink anchor would create a composite view of all the linked passages in which all the passages marked with the same ink anchor are displayed end to end in a linear fashion. A list of all the ink anchors created by the reader can also be viewed as a reminder of the anchors that have been used before.

2.3.4 Summary and definition of approach

We have seen the promise of hypertext, but have been disappointed to discover that, although a publicly available and global medium, the WWW largely fails to live up to this promise. We have posited that the reason for the lack of associative linking on the WWW derives from at least three contributing factors - the net-like properties of hypertext which cause *navigation* problems; the lack of structural cues in a hypertext which lead to *incomprehension* problems; and the significant *effort* required to author associative links which restricts their abundant creation.

The approach of this research is to use *annotations* to capture implicit knowledge about associations between documents as the author actively reads a document set. Since annotations are a ubiquitous part of reading (particularly when the reading activity is part of a larger authoring effort), we posit that using the reader's annotations to capture these potentially volatile associations between ideas and concepts presented in different documents will not accrue a significant extra effort. By capturing information about the context and type of these relationships, it is hypothesised that navigation and incomprehension issues may be kept to a minimum. This information can then be used to support the author during the writing process, by providing a mechanism for automatically constructing associative links from the new writing to the global context, and also describing and making available the associations between documents in the global context, in the context of the new writing. As Bush Bush (1945) reminds us :

One cannot hope thus to equal the speed and flexibility with which the mind follows an associative trail, but it should be possible to beat the mind decisively in regard to the permanence and clarity of the items resurrected from storage.

The next chapter briefly overviews traditional hypertext authoring approaches, with particular emphasis on facilities for describing and authoring associations between concepts and ideas.

Chapter 3

Hyperauthoring Systems Review

The emphasis of this review is on mechanisms for describing and authoring associations between information nodes, since these mechanisms are most applicable to the approach of this work. Other hyperauthoring facilities and concerns such as collaboration (Haake and Wilson (1992)), author ability (Joyce (1991)), and thinking tools (Thimbleby (1994)) are also important considerations for hyperauthoring research but will not be discussed here.

3.1 SEPIA

The SEPIA (Structured Elicitation and Processing of Ideas for Authoring) environment Streitz et al. (1989, 1992); Haake and Wilson (1992) provides hyperwriting tools for both the individual and collaborative writer, based on the principle of cognitive compatibility. There is no difference between single and cooperative authoring in SEPIA in terms of the way users manipulate and navigate hyperdocuments.

Authors create hyperdocuments by interacting with four activity space browsers dedicated to the tasks of content generation and structuring, planning, arguing and writing the final hyperdocuments under a rhetorical perspective (content space, planning space, argumentation space, and rhetorical space respectively). SEPIA's basic hypertext objects are atomic nodes, composite nodes, and labelled links. The contents of each composite node created in an activity space is displayed in a new activity space browser when the composite node is opened.

The planning space serves as a meta space for coordinating the activities in the other three spaces and for controlling the progress of the hypertext design process. The planning space contains the overall goal structure and plans for writing, and acts as a switchboard between the other activity spaces - for example, nodes created in the planning space can be migrated as topics forming the basis of argumentation networks in the argumentation space.

The content space utilises the structuring facility of hypertext to support idea dumping, the grouping of related ideas into topic related clusters by composite nodes, and the connection of ideas via part-whole relationships.

The argumentation space supports the development of argumentative structure, through the ability to relate information nodes with links typed according to an argumentation schema. Writers are able to use this argumentation space to elaborate arguments by generating support or objections on different levels, and by formulating contradictions and argumentative chains.

The node types and links, and the operations which can be carried out on them in the argumentation space are summarised below :

- **Nodes**
  - statement *(position, claim, datum)*
- **Links**
  - so (may exist with *warrant* and/or *backing*)
  - contradicts
  - contributes_to
- **Operations**
  - support
  - object_to
  - justify
  - negate
  - generalise
  - specialise

The nodes of the network created in the argumentation space represent statements generated by the author during the development of an argumentation. Nodes can be typed as either positions, datum, or claims. By applying the *support* operation, a new link is created, and visualised as a *so* relation leading from the supporting to the supported statement, the supporting node becoming a *datum.* The *negate* operation can be used to indicate that a statement is a direct contradiction of another statement, and is visualised as a *contradicts* relation from contradicting to contradicted node.

To attack a claim, the author applies the *object_to* operation to an existing claim. This operation creates two new nodes (a claim which opposes the existing argument and a datum that supports the new claim) linked together with a supporting *so* relation, and links the new claim to the claim under attack via a *contradicts* relation. This structure is necessarily complex in order to make the implications of the attack explicit to the author.

The *justify* operation can be used to create supporting relations (through the *so* link) with warrant and backing to cover the relation between datum and claim. The warrant and backing are essentially properties of the supporting relation.

Authors often start with a global argument, and then gradually refine it by creating other arguments which are more specific instances of the global argument. Alternatively, authors may start with several arguments and draw a more general position by summarising their claims. Hence, in order to be able to model arguments at multiple levels of abstraction, generalisation and specialisation operations are provided. The *generalise* operation allows a more general statement to be related to one or more statements at a lower level of abstraction via the *contributes_to* relation. Conversely, the *specialise* operation enables more specific statements to be linked to the appropriate general statements.

The rhetorical space is where the author creates the reader-oriented, final document (which can be either a linear text or hyperdocument) by imposing a document structure on the ideas and arguments explicated and elaborated in the content and argumentation space. The rhetorical space provides three working modes : *outline mode* (global outline of the document), a*rgumentation strategy mode* (rhetorical reorganisation of positions and arguments for each subissue), and *text edit mode* (sentence-level editing).

SEPIA emphasises smooth ad-hoc transition between activity spaces, in recognition that the production and transformation of knowledge during writing is not done in a linear sequence of stages from idea generation to text generation.

3.2 Writing Environment

The Writing Environment (WE) system Smith et al. (1987), specifically designed as an experimental platform to study what tools and facilities are useful in a writer's environment, is based on a cognitive model of the communication process. WE regards hypertext in its fundamental form - a directed graph of information components - as being consistent with one particular mode of thinking - exploration. Exploratory thinking usually occurs early in the development of a set of ideas, and therefore hypertext is defined as an (early) stage in the development of a document rather than its final form. WE thus provides tools for transforming loose associative networks of ideas into a

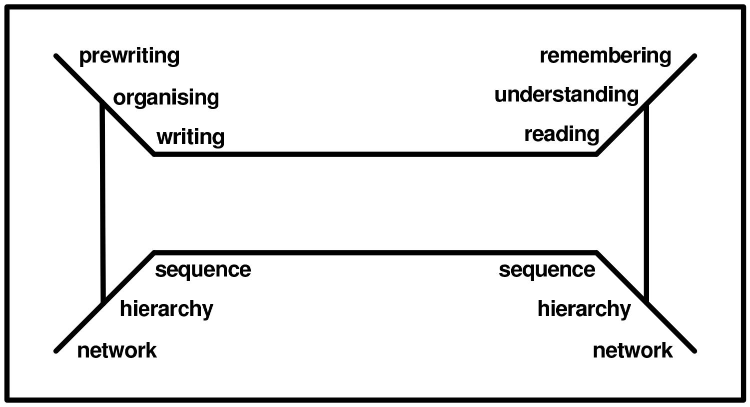


FIGURE 3.1: Cognitive Framework for Written Communication (after Smith et al. (1987)).

hierarchical structure, and then writing a document in accord with that structure. The developers of WE posit that hypertext is therefore a supporting utility over which more constrained applications will be developed, rather than the primary application system itself.

The cognitive model on which the WE is based models reading as the process of taking the linear stream of text, comprehending it by structuring the concepts hierarchically, and absorbing it into long-term memory as a network. Writing is seen as the reverse of this process - the organisation of a loosely structured network of internal ideas and external sources into a hierarchy or outline which is then used as the foundation of a linear stream of words and sentences (Figure 3.1).

The WE contains two major view windows, a graphical window and a hierarchical window. A writer begins by creating nodes in the graph view, but imposes little or no structure on the conceptual material. Nodes may be placed in "piles" if they seem related, and simple subordinate/superordinate relations imposed. As the conceptual structure emerges, nodes can be copied into the hierarchy window and arranged according to this structure.

In a follow-up study Lansman et al. (1993), WE was used to investigate how writers developed the structure of technical documents. The ability to view the organisational structures of the document as the text was being written proved popular among participants.

3.3 StorySpace

The Storyspace writing tool Bernstein et al. (1991); Inc. provides a graphical interface within which graphic elements or *writing spaces* are represented as rectangular icons. Each writing space has two roles: it contains text (including pictures) and it occupies a place in the evolving structure of elements. The visual representation of a writing space therefore contains two components: a textual component (in the form of a title bar which leads to viewing and editing of the text inside the writing space), and a structural component which can hold additional writing spaces (in fact, the writer may locate an indefinite number of nested writing spaces inside one superordinate space). Writers manipulate structure directly by arranging, opening, and closing spaces. The inherent hierarchical nature of writing spaces can also be used to provide a tree-like view of the graphical layout. Bernstein et al argue that these writing spaces are easier to manipulate than chunks of continuous writing because the writer identifies with each space as a discrete symbolic entity Bernstein et al. (1991).

Storyspace allows labelled links to be used to graphically model associations between related spans of text, regions of graphics, or entire writing spaces. If the source and destination of a link are not on the same level of the hierarchy, only an abbreviated incoming or outgoing arrow is displayed. Storyspace allows the construction of both *univalent* links (links with a single beginning and a single end) and *multivalent* links (links with multiple sources and/or destinations), which can be labelled arbitrarily by the writer.

3.4 Aquanet and the Visual Knowledge Builder

Aquanet Marshall et al. (1991) is a tool designed to support knowledge structuring tasks. Users are able to create and browse knowledge structures (interconnected networks of information bearing nodes). Node types and links can be created, according to a user definable schema, for example the Toulmin argumentation structure Toulmin (1959) (Figure 3.2).

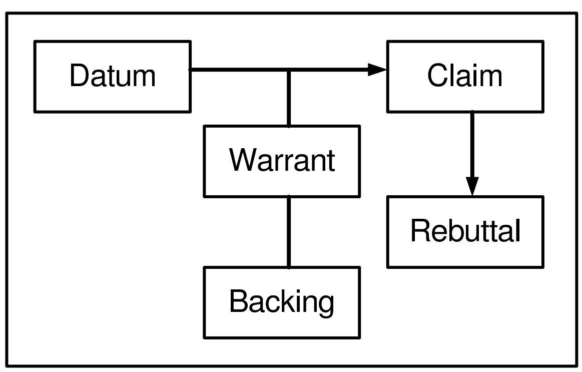


FIGURE 3.2: Abstract Toulmin micro-argument structure Toulmin (1959)

Experiences from the use of Aquanet record that authors did not rely on a predefined library of network structures: instead they tried to define their own schemas for hypertext structures, and were consequently frustrated by a system unable to support flexible schema modification. Aquanet users circumvented this problem by using the main display space as a drawing board, expressing developing relationships between the object icons spatially, for example placing similar objects in a pile. Marshall and Rogers describe various manipulations of representational structures as crucial to the author's interpretive process, and this work has led to the development of tools which support these manipulations in all their richness.

The Visual Knowledge Builder (VKB) Shipman et al. (1999) (based on the earlier VIKI system Marshall and Shipman, III (1997)) is based on a spatial notion of hypertext, providing users with spatial and visual affordances for organising information. Marshall and Shipman describe this prewriting/analytic process of sorting through materials and organising them to meet the needs of the task at hand as "information triage" Marshall and Shipman, III (1997).

The VKB system provides a rich feature set in terms of the visual manipulation of information nodes, or objects, represented as symbols in the spatial area. The position, size, border width, font style, transparency, background color, font, and border colours of visual symbols and collections are able to be modified by direct manipulation. Information objects can be also represented by more than one visual symbol, allowing multiple interpretations of a single piece of content.

Information objects can be assigned a set of attributes and values - when a common set of attributes are shared by a number of objects, an object type can be defined. Types combine semantic information about the objects (a set of expected attributes and their default values), and visual information about the symbols (as default visual characteristics, including color, shape, and details of which attribute values are shown in the visual presentation).

The VKB system is also able to integrate with the user's working enviroment - objects can use an attribute value to point to external information, for example Microsoft Word documents, Adobe Acrobat documents, documents on the WWW, or any other data type with a registered viewer in the user's working environment.

In order to express relationships or categories in VKB, symbols are simply placed near one another, or placed in a collection. Hierarchies of collections can be navigated by double clicking on the borders of each collection object; navigating into a collection causes that collection to fill the entire workspace window. Relationships are constructed using spatial structures, such as piles, lists, and repeating patterns of visual symbols.

An integrated spatial parser recognises three main types of structure : stacks, lists and composites. Stacks are overlapping symbols of the same type. Lists are vertically or horizontally aligned symbols of the same type. Composites are repeating visual patterns of symbols of specific types, such as a "list label" above a number of "list elements". Types for untyped objects can be inferred implicitly based on their visual similarity to other *typed* objects (largely based on colour similarities, since colour tends to dominate the visual categorisation of symbols). The spatial parser allows the VKB workspace to be exported to formats such as XML and HTML.

VKB also includes an embedded history mechanism. As information spaces are rarely completely consistent, it is important for users to be able to rewind, replay, or step through the interpretative process leading to the current workspace when they are unable to remember initial interpretations.

3.5 Describing associations

Mechanisms for describing associations have also been suggested by other research not directly related to hyperauthoring systems.

Rittel introduced the idea of Issue Based Information Systems (IBIS) Kunz and Rittel (1970) (see also Conklin and Begeman (1988)). IBIS systems provide teleconferencing components for collaborative understanding of the major issues and implications surrounding what Rittel describes as "wicked problems" (problems that lack a definitive formulation), and hypertext components for creating structured arguments surrounding the issues. IBIS allow three types of node to be created - issues, positions, and arguments - and used eight different types of relationship to link these node (responds_to, questions, supports, objects_to, specialises, generalises, refers_to and replaces).

The SYNVIEW system Lowe (1986) takes the IBIS approach in a different direction by allowing the nodes to be collaboratively rated in terms of validity and relevance - new readers are therefore able to focus on those argument trails having the highest voted validity.

Trigg Trigg (1983) describes a taxonomy of over 70 link types, classified into two major categories. *Normal links* serve to "connect nodes making up a scientific work", whilst *commentary links* "connect statements about a node to the node in question".

In an attempt to structure hypertext for Classics and Religious studies, DeRose DeRose (1989) presents a taxonomy of links categorised at the highest level as either *extensional* or intensional. Extensional links are futher split into *relational* and *inclusive* categories. Relational links consist of *associative* links (connecting arbitrary pieces of text) and *annotational links* (referential links connecting portions of the text to information about the text). Inclusive links consist of *sequential* and *taxonomic* links, which associate lists of properties with particular document elements. The second highest-level category, intensional links, describes other link types that follow strictly from the structure and content of the document (for example, the linking of references of the form "see Section *n*" to the appropriate section).

Baron et al Baron et al. (1996) used three classes of link to demonstrate that augmenting hyperlinks with meaningful labels enabled more effective searching across a hypertext document. The three classes of link used were: *semantic* links (similar, contrast, part of), *rhetorical* links (definition, explanation, continuation, illustration, summary), and *pragmatic* links (warning, prerequisite, usage, example).

3.6 Discussion

This chapter has examined the mechanisms which have been applied in previous hyperauthoring-related research for describing and authoring associations between information nodes. Other link taxonomies that describe associations between documents have also been examined.

Although the authoring systems share many common features in terms of the manner in which information nodes can be visual manipulated on-screen, the mechanisms each system offers to the hyperauthor for making associations are quite different.

Both the SEPIA and WE systems base their interaction modes on established cognitive models, supporting the author through various processes of refinement from forming a unstructured network of ideas to the final structured hypertext. The WE allows the user to express simple ordinate/subordinate relationships between ideas with the intention of capturing the intended hierarchical structure of the document. The WE is therefore oriented toward authoring a linear document. In contrast, the SEPIA system (and also the Storyspace system) allow the user to create true hypertext links between ideas and produce a rich hypertext. The VKB system allows authors to spatially organise and classify information nodes into "piles", but does not allow authors to explicitly express meaningful associations between interrelated nodes. Aquanet and SEPIA in contrast allow authors to apply argumentation-based structures to ensure that associations can be meaningfully expressed.

The link types with which each system allows authors to express associations between information nodes varies diversely (Trigg suggests more than 70 link types, the SEPIA system provides only 3). Aquanet allows the author to define a strict taxonomy of link types it offers, and Storyspace goes a step further to allow the author to unrestrictedly describe an association in any manner that the author feels is appropriate. It is clear that care needs to be taken to achieve the right balance - too few link types may limit the expressiveness of the author and the degree to which associations between ideas are able to be captured; too many links may overwhelm the author and be left largely unused; unrestricted typing of links may lead to inconsistencies.

None of the systems discussed in this chapter make any attempt to tightly integrate the hyperauthoring process with the foundation of published literature upon which new writings inevitably build. As authors add ideas gleaned from this literature and interweave them with their own, the system does not record where these ideas originated (the author must "cut and paste" ideas into the system, rather than seamlessly "transclude" them). Therefore the author makes associations between the information nodes in the system, but not between the literature itself - the system has no sense of the underlying literature foundation from which the ideas are derived. The system lacks the global perspective which this work requires, unless this is explicitly (and inefficiently) orchestrated by the user (e.g. in VKB by assigning properties to each information node which are placeholders for metadata about the document from which an information node transcludes).

The SEPIA system highlights the importance of allowing the author to switch between different writing processes quickly, whilst maintaining the current context. However, neither the SEPIA system nor any of the other systems discussed attempt to integrate the reading processes that invariably occur during writing. Indeed, this work proposes that the annotations the author makes during reading, in terms of both the highlighting or otherwise emphasising of ideas and the meaningful association of contextually interrelated ideas would provide an ideal source for the information nodes and associations that form the early stages of hyperwriting.

Chapter 4

Annotation Systems Review

Whittington Whittington (1996), and Cousins et al Cousins et al. (2000) identify common characteristics of annotation systems. Whittington identifies 12 characteristics which define and differentiate annotation systems. Cousins et al describe a framework for annotation systems, based on the dimensions of:

1. The platform used to create the annotations
2. The platform used to read the annotations
3. The annotations themselves
4. The target object being annotated

In this review, investigation of the following five characteristics, derived from both Whittington's and Cousins et al's models, have been identified as being most relevant to this work :

- **Architecture** - what are the main architectural features of the annotation platform?
- **Authoring** - how are annotations created?
- **Granularity** - at what granularity can annotations be associated with documents? For example, the coarsest granularity annotations are those at the document level.
- **Storage** - how are the annotations represented and stored?
- **Typing** - what facilities exist for attaching semantics to the annotations?

An additional consideration in exploring annotation systems, not included in Whittington's or Cousins et al's taxonomies, is the facilities that existing annotation systems offer for **associative linking** - the expression of relationships between different annotations.

4.1 Annotea

Koivunen et al Koivunen et al. (2000); Kahan et al. (2001) describe the Annotea annotation system, developed by the World Wide Web Consortium (W3C), in order to support annotative collaboration through an open metadata infrastructure.

Architecturally, Annotea consists of two main components: an annotation server that stores and serves the annotations in RDF form, and a WWW browser/editor that provides users with an interface for creating and reading the annotations. The primary WWW browser/editor for the system is Amaya, an Open Source system also developed by the W3C. When annotations are in use, the Amaya browser consults the annotation server every time the user loads a new document and the annotations related to that document are retrieved and presented. Annotation Servers can be hosted by any individual, and users wishing to view annotations stored on a particular server can configure the Amaya browser to request annotations from one or more different servers.

Annotations in the Annotea system are different statements about a document, such as comments, typographical corrections, hypothesis or ratings. Annotations are posted to an annotation server in RDF format, and stored in an RDF database as general metadata triples. A single triple is a (*subject*, *property*, *value*) statement and states that for a WWW object whose Web address is *subject*, the given *property* has the specified *value*. This provides a flexible framework for describing objects on the WWW, permitting separate communities to develop independent metadata vocabularies and then freely mix statements using these vocabularies in a single database of triples. By the nature of the RDF standard, the property names themselves have Web addresses, so other statements about the meaning and use of each property and its relationships to other properties can be inferred. The RDF schema definitions for the basic annotation properties are shown in Figure 4.1.

Currently Amaya supports two kinds of annotation: annotations that apply to the whole document, and annotations which apply to a specific point or span in the document. Document level annotations are created by selecting the “Annotate Document” command from Amaya’s menu. Similarly, finer grained annotations can be created by moving to a point or selecting a span and choosing the “Annotate Selection” command. When an annotation has been created, a small icon appears either at the top of the document, at the specified point in the document, or at the beginning of the selected span. Clicking on the icon opens a new window which displays the content of the annotation.

With the aim of supporting the evolving needs of collaborative groups, Annotea provides expandable annotation types through the creation of different annotation classes which subclass an Annotation superclass. Sample subclasses such as Comment, Deletion, Addition, Query, and Warning have been defined. Annotation sets can also potentially be created. Other metadata can be added to annotation subclasses when the need arises, for example a property that describes the rating of an annotation according to a scale created by the collaborative group.

rdf:type The type of the annotation, which should indicate the author's intention when making an annotation. The value should be of **rdf:type Annotation**, or any of its subclasses.

annotates The resource to which the annotation applies.

body The content of the annotation.

context The context defines where exactly inside the document the annotation is attached to. The XPointer standard is used to define this location.

dc:creator The creator of the annotation (from Dublin Core element set).

created The date and time on which the annotation was created.

dc:date The date and time on which the annotation was last modified (from Dublin Core element set).

related Related resources that augment the body of the annotation, for example, backing evidence for an argument, related issues, or discussion threads.

FIGURE 4.1: Annotea RDF schema definition for basic annotation properties (after [Koivunen])

4.2 InterNote

The InterNote annotation system Catlin et al. (1989) is an extension to the InterMedia system Meyrowitz (1986), implemented in an attempt to better support collaborative document review and revision.

Annotations of InterMedia documents are made using *Notes*, which are created by making a selection in the text of the document and picking the "Create Annotation" menu command. A Note consists of two frames: the *Incorporation Frame* and the *Commentary Frame*. Annotators enter specific suggestions to replace the selected text in the main document into the Incorporation Frame (this frame is automatically filled with the contents of the selection in the main document). The editing tools available to annotators in the Incorporation Frame correspond to the editing tools available in the document being annotated - for example, if a section of an image is annotated, the Incorporation Frame will have the full compliment of graphics editing tools available.

The Commentary Frame is used to provide a textual justification or commentary of the suggested changes entered into the Incorporation Frame.

A link anchor then appears at the beginning of the selection in the annotated document. Selecting the annotation marker causes the Note to appear in a separate window to the main document, which contains a link back to the document. The InterNote system introduces the concept of warm linking, linking that involves not only navigation, but also allows data transfer across the link. The author of the document can therefore view the annotations made by reviewers and decide whether to incorporate them into the next revision. Suggestions made in the Incorporation Frame of annotations can be incorporated by executing the *Pull* command on the link anchor in the main document, replacing the content which the Note annotates with the content in the Incorporation Frame. Alternatively, the *Push* command can be executed on the link anchor in the Note window to achieve the same effect.

4.3 Multivalent Annotations

Phelps and Wilensky have developed Multivalent Annotations Phelps and Wilensky (1997), based on the Multivalent Document system Phelps and Wilensky (2000b). In the Multivalent Document system, a hub document is an XML description of the layers and behaviours that comprise a single conceptual document. For example, a hub document may consist of a scanned document image, and its corresponding OCR-inferred text. Other layers and behaviours may provide arbitrary functionality, including various forms of annotation.

Three broad types of multivalent annotation have been developed: annotations that make use of point-to-point spans of media elements, annotations that make use of geometric regions, and annotations that make use of structures within the document tree. Span annotations include highlighting of text (created by selecting a region of text, and then choosing the "highlight" behaviour), hyperlinks (created by selecting a region of text, choosing the "hyperlink" behaviour, and then entering the URL), and copy editor markup (created by selecting a region of text and then associating copy editor markup with the span by drawing freehand marks on the page, typing in text, or choosing from a palette of common copy editor markup symbols). Another example of a span annotation is that used to visualise version differences, with deleted text overstruck, inserted text in italics and changed text in bold.

Annotations associated with a geometric region are lenses. Lenses can arbitrarily transform the appearance of their contents, with full access to the semantic representation of that content. For example, lenses can be used to magnify underlying text or images, opaque lenses can act as "post-it note" type annotations, and lenses can even be used as language translators - an experimental "language translation lens" displays French or German translations of underlying English sentences.

Structural annotations hook into the hierarchical document tree. For example, by annotating an entry in a bibliography with structural information (e.g. author, title), the bibliographic entry could automatically be transformed into BiBTeX or Refer format when the entry was pasted into another application.

Multivalent annotations may also be combined - Phelps and Wilensky introduce NoteMarks, a combination of structural and span annotations that allow the overall structure of a document to be sensed by collapsing sections so that only headings are visible, except for section content that contains higher priority annotations. For example, in reading a manual page, the page could be annotated such that only the section headings are visible except for content within the section which contained search hits for a particular topic that the user is interested in (presented as span annotations).

As the concept of "hub documents" in the Multivalent Document model implies, annotation behaviours can be stored apart from the other layers and behaviours that make up the document. A hub document can be also constructed that incorporates annotations created by one or more persons.

4.4 XLibris

Price et al Golovchinsky et al. (1999); Schilit et al. (1998); Price et al. (1998a,b) describe the XLibris system, an annotation system geared towards the support of the active reading process - the combination of reading with critical thinking and learning which involves not just reading, but also underlining, highlighting, scribbling, and commenting.

XLibris uses the paper document metaphor by providing a display that emulates the appearance of a sheet of paper, an interface for viewing pages in the linear order of the document, the ability to mark in any place on the page, and cues about the size of a document and the reader's location within it. The physical experience of reading a paper document is also emulated - readers hold a lightweight pen tablet that displays one page of a document at a time, and can mark anywhere on the page with coloured pens, highlighters, and an eraser, taking the granularity of annotations to the pixel level. Different types of annotation can be effected by using different ink colours.

XLibris enriches these paper-like annotations by adding the possibility of *active annotations*. Documents are scanned or "printed" into XLibris, and stored in an "image + text" format. The image of each displayed page is tied to the text of the page (possibly inferred from OCR) by a bounding box for each word. Underlined or circled words and passages are translated into queries which search for related information in the context of the users annotations. Links to related passages are then unobtrusively presented in the margin of the page next to the annotation, allowing the user to peruse the related information at their leisure. This process of initiating searches related information in the context of the users reading is known as *linking by inking*.

The XLibris system also provides a Reader's Notebook, and a "workspace view". The Reader's Notebook is a notebook of clippings derived from user's annotations, providing review and retrieval facilities. Clippings are derived from the bounding boxes of the user's annotations, and are presented end-to-end on a page, labelled with the document title and page number, and acting as a link to the underlying page. Clippings from one or more documents may be viewed simultaneously, and may be sorted by creation time or page number, or filtered according to the ink color used. The "workspace view" displays a thumbnail of each document, with the size of the documents shadow used to indicate the size of the document.

Price et al Price et al. (1998a) describe the possibility of linking documents together using digital ink, proposing a design whereby circling any annotation converts that annotation into an anchor. The system would then maintain multi-way links between similar ink anchors - tapping on a particular instance of an ink anchor should produce a list of clippings that contain matching target ink anchors (similar to the Reader's Notebook). Readers can use the flexibility of free form ink to make the ink anchors meaningful; this could potentially include assigning types to the links.

Since the XLibris system is primarily a personal reading device, annotations are expected to be stored locally and not be accessible to anyone other than the reader herself. However, Marshall et al Marshall et al. (1999) have demonstrated how the XLibris system can be used in a collaborative reading environment. Suggestions arising from this study include:

- The overlaying of several sets of annotations from different readers, in order that the group can see multiple annotations at once.

- Allowing readers to access the annotations and underlying text apart from the document to create a compact representation of the passages that other readers have annotated.

- The processing of different readers' annotations in order to produce a document which has passages emphasised where one or more readers annotated them, but preserves anonymity by not showing the original annotations. By looking at *where* people annotate, rather than *how* people annotate, areas of concensus can be identified - the annotations may create paths through a document that others may benefit from.

4.5 CritSuite

The CritSuite Yee (1998) system provides a referential annotation system for the WWW, supporting the practice of critical discussion.

The CritSuite system uses a proxy architecture (CritSuite's creator, Yee, calls this proxy service a *Mediator*), which augments requested web documents with the "CritLink" interface, and any annotations that have been made on the document. The mediator also modifies the links in the documents it processes so that the client software will continue to invoke the mediator on each request.

CritSuite supports two types of annotation granularity, annotations that apply to the entire document, and annotations that apply to a specific phrase (arbitrary span of text) in the document. In order to create an annotation of either type, the user selects the "comment" icon from the CritLink interface bar inserted at the top of each page requested through the mediator, which in turn opens a new window displaying the "CritWriter" interface - an HTML form containing fields for the annotation content and author. In order to apply the annotation to the document as a whole, the "link to entire document" option is selected. Linking to a specific phrase is more cumbersome and requires the annotator to copy and paste the phrase and its surrounding context into appropriate fields of the CritWriter interface. The annotation can then be stored on the server, or alternatively, downloaded and made available on the WWW by the annotator. CritSuite supports four types of annotation; support (agree), issue(disagree), query, and comment.

When documents are subsequently viewed through the mediator, annotations specific to spans are presented as small arrows at the start and end of the span, which can be clicked on to open a new window displaying the annotation. The mediator also inserts a summary of links to all annotations of the document at the bottom of the document, including span specific annotations (with the annotated phrase reproduced verbatim), orphan annotations, and annotations that apply to the document as a whole.

4.6 ScholOnto

Buckingham Shum et al Shum et al. (1999); Buckingham-Shum et al. (2000) describe a formal ontological approach to annotating WWW resources, in an attempt to model the claims that scholars make about their research.

The ScholOnto system is a digital library server that overlays on conventional documents and metadata a semantic web of scholarly claims, discourse and perspectives, enabling researchers to describe and debate via the semantic network the contributions a document makes, and its relationships to the literature. These human-encoded document

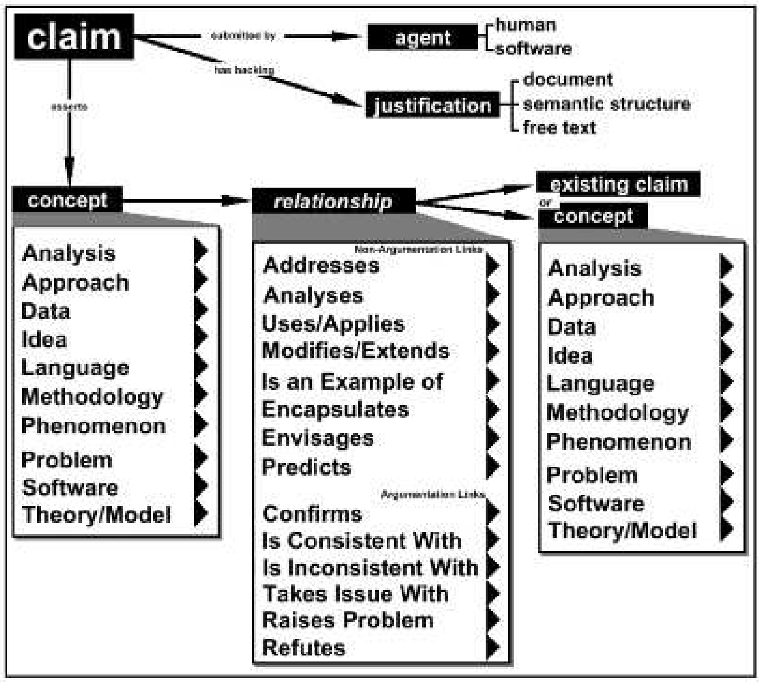


FIGURE 4.2: The structure of a scholarly claim in the ScholOnto ontology Buckingham-Shum et al. (2000). All claims are owned by an agent, and have some form of justification. Claims assert new relationships with other claims, or between concepts.

descriptions have the advantage that authors promoting their work take ownership of, and have control over, the way in which their work is presented.

The ScholOnto system can then be used by researchers to answer queries such as *Are there any arguments against the framework on which this paper builds?*, and *Has anyone extended language L?*.

Authors can use a direct annotation interface to describe their contributions to a research community. This interface provides the opportunity to annotate specific points of the document using the available ontological relationships (Figure 4.2) - for example the author can specify exactly which part of the document refutes a theory proposed in another document. Using a 1-to-N link, a number of points in the document can be associated with an ontological concept. The annotations are stored on the ScholOnto server, separate from the documents being annotated.

4.7 Discussion

This chapter has examined previous annotation systems research, emphasising architectural features, storage models, mechanisms for creating annotations, possible levels of granularity in attaching annotations to documents, mechanisms for describing annotations, and mechanisms for expressing associations between annotations.

In terms of architecture, the Annotea framework adopts the client-side model, in which a WWW browser is augmented with the capability to request and display annotations from a dedicated annotation server as the WWW is browsed. This architecture allows users to view annotations from multiple sources by configuring the browser to request annotation from multiple servers. In contrast, the CritSuite system is based on a proxy architecture, in which requests to view WWW documents are routed through the CritSuite "mediator", which decorates each document with annotations before returning it to the browser. The InterNote, Multivalent Annotations, XLibris, and ScholOnto systems are dedicated annotation enviroments in which the user must work in order to take advantage of annotative capabilities. The impetus of this work is to allow the WWW to evolve into all its hypertext richness - the working environment of the author is the WWW, and thus tools to support the capture of implicit relationships between documents need to be integrated into the WWW browsing environment in much the same way as the Annotea framework extends the Amaya browser to include annotation facilities.

The Annotea framework uses an RDF syntax to describe annotations of WWW resources. These descriptions are then stored in a (freely distributable) database held on a dedicated server which can be queried extensively by any application able to use the same communication protocol. Annotations created in the other systems are not accessible outside that systems storage environment. In the context of this work, it would be sensible to allows "open" access to stored annotations, particularly in a situation where collaborating authors share each others annotations.

The ScholOnto and Multivalent Annotation systems differ from the other systems described in terms of the concept of what exactly constitutes an "annotation". The Multivalent Annotation system regards mechanisms which alter the presentation of the text as a form of annotation. ScholOnto regards annotations as formal statements that are made about a resource in order to allow computers to perform intelligent reasoning and inferencing over a set of resources, as opposed to personal responses to a text that are intended to be human readable. The annotations authors make when reading literature clearly fall into this category.

In terms of granularity, the Annotea, CritSuite, ScholOnto, Multivalent Annotations and Internote systems allow annotations to be made at the character level by selecting text in the document to which an annotation will be anchored. These systems also allow annotations to be attached to a document as a whole. InterNote also allows true multi-media annotations, for example images can be annotated with image-editing tools.

Taking granularity further than character- and document-level annotations, the XLibris and Multivalent Annotations systems allow annotations at the pixel level - freehand marks can be drawn on electronic documents in much the same way as annotations are made on paper with a pen. The underlying systems are able to determine which text is being annotated and perform actions according to these annotations (e.g. link suggestions in XLibris when passages are underlined).

In terms of this work, character level granularity seems best for highlighting important parts of documents, and pixel-level granularity seems best for expressing associations.between these highlighted parts.

Chapter 3 concluded that link taxonomies suggested by the literature vary diversely, and should be chosen carefully. The same is true of annotation taxonomies advocated by the annotation systems reviewed in this chapter, and the same cautionary note should be heeded. The Annotea system allows the available annotation types to be extended. The Multivalent Annotations and XLibris systems offer unconstrained annotation types - freeform marks or notes can be used to describe the purpose of the annotation.

Further to the link taxomony discussion in Chapter 3, the ScholOnto system provides a number of link types which are particularly appropriate to this work.

The link types with which each system allows authors to express associations between information nodes varies diversely (Trigg suggests more than 70 link types, the SEPIA system provides only 3). Aquanet allows the author to define a strict taxonomy of link types it offers, and Storyspace goes a step further to allow the author to unrestrictedly describe an association in any manner that the author feels is appropriate. It is clear that care needs to be taken to achieve the right balance - too few link types may limit the expressiveness of the author and the degree to which associations between ideas are able to be captured; too many links may overwhelm the author and be left largely unused; unrestricted typing of links may lead to inconsistencies.

Of all the systems described in this chapter, only the XLibris and Annotea systems provide mechanisms for associating interrelated annotations. XLibris proposes a "ink anchor" technique for associating related annotations. All passages marked with the same freeform symbol are automatically associated and can be viewed together, separately from their source documents. Meaningful associations can be made using freeform marks near the ink anchors (for example, noting the type or reason for the association) - these marks are also presented when all the associated passages are viewed together. This mechanism does not provide for the creation of more complex association structures however, and it is not clear how the associations created can be integrated directly into the writing space.

Annotea contains "related" field in its RDF annotation syntax but this is intended to be used in the context of providing additional information to that contained in the annotation (examples given by Koivunen et al include backing evidence for an argument, related issues, and discussion threads). The Annotea framework does not therefore provide a principled approach for authors to express relationships between resources as annotations. However, it is possible that the framework could be extend to include information about inter-annotation relationships by subclassing and extending the RDF Annotation class.

The “warm linking” mechanism demonstrated in the InterNote system may be useful for importing or transcluding annotation content into a visual manipulation environment (see authoring discussion) where associations between annotations can be expressed.

Chapter 5

Implementation Work

5.1 Annotation Linking ENvironment (ALIEN)

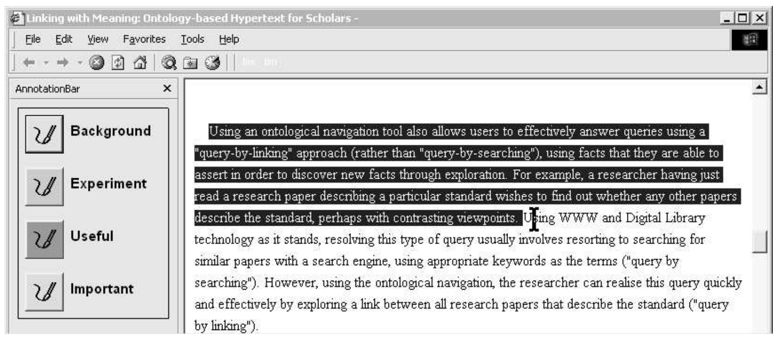
The **A**nnotation **LI**nking **EN**vironment (ALIEN) was implemented to demonstrate the principles of "linking by annotating"; how annotations could be used to capture associations between documents as they are read.

ALIEN consists of 4 major components, the Annotator, Annotation Linker, Annotation Desk, and Annotation Server. WWW Documents can be annotated using the Annotator, which is integrated into the Internet Explorer WWW browsing environment. The Annotation Linker is then used to express the relationships between annotated parts of WWW documents using typed links. More complex annotations can be constructed by dragging document annotations created with the Annotator onto the Annotation Desk and using the advanced annotation facilities. The Annotation Desk also provides mechanisms for associating meaningful labels and statements with the various annotations and relationships. The Annotator, Annotation Linker, and Annotation Desk communicate with the Annotation Server, which maintains a central annotation repository.

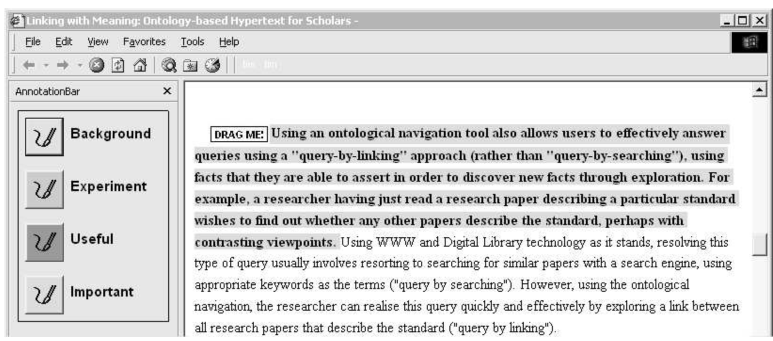
5.1.1 Annotator

Architecturally, the Annotator component has many similarities to the Annotea annotation system Koivunen et al. (2000); Kahan et al. (2001). The Annotator is a tool for annotating web documents that is integrated into the Internet Explorer WWW browsing environment as a user interface "plug-in". Annotations are stored separately from the annotated document - on the addition of a new annotation, the Annotation Server is informed that a new annotation has been added to this particular WWW document. Subsequently, when the browser returns to the WWW document, the Annotator queries the Annotation Server for annotations of the document, and displays the return annotations.

Currently, the Annotator facilitates annotations of the type that may be achieved in a paper environment with a highlighter pen, that is, *span* annotations. Web documents are annotated by selecting a text span from the currently displayed document, and then choosing an annotation type. The selected area is then highlighted according to the colour associated with that annotation type. The granularity of the annotations is therefore at the character level. Figure 5.1 demonstrates how the Annotator facilitates the in-place annotation of a WWW document.



(a)



(b)

FIGURE 5.1: Creating an annotation with the Annotator tool. After selecting the span of text to be annotated (a), the author chooses an annotation type from the palette, and the selected text becomes highlighted in the corresponding colour (b).

The Annotator can be configured to present a number of different annotation types to the user, each with a corresponding (user defined) colour. For example, the Annotator could be configured to provide annotation types based on argumentation models (for example Toulmin Toulmin (1959), SEPIA Streitz et al. (1992), IBIS Kunz and Rittel (1970)), or those annotation types provided by existing annotation systems such as Annotea Koivunen et al. (2000), CritSuite Yee (1998), or ScholOnto Buckingham-Shum et al. (2000). Alternatively, the annotation types provided by the Annotator could be configured to tie annotations directly to their proposed roles in the writing being produced, for example, "background" for highlighting background material in documents, and "important" for highlighting an important experimental result or conclusion.

In order to record the position and extent of the span annotations created through the Annotator, an XPointer-like construct is used. This approach is comparable to that of Phelps and Wilensky Phelps and Wilensky (2000b) and Annotea Koivunen et al. (2000); Kahan et al. (2001), who use a similar syntax to describe the positions of annotations.

Annotations created by the Annotator are described (and subsequently stored by the Annotation Server) as XML fragments (Figure 5.2). This XML format is regarded as a stepping stone to RDF representation, and compliance with the W3C standard. An RDF representation would also mean that the Annotea RDF annotation server could be deployed in place of the current Annotation Server component.

```
<annotation>
    <location>
        <url>http://www.ecs.soton.ac.uk/~tmb99r/lwm.html</url>
        <doc_location type='html'>BODY(0)/P(19)
            </doc_location>
        <media_location type='text'>Using(0),0,447
            </media_location>
    </location>
    <properties>
        <type>Important</type>
        <title>Linking with Meaning: Ontology-based
            Hypertext for Scholars</title>
        <content>Using an ontological navigation..</content>
    </properties>
</annotation>
```

Figure 5.2: ALIEN internal XML representation of a span annotation

The `location` construct of the XML representation shown Figure 5.2 describes the location of the annotation within the document in two separate parts - the *document* location and the *media* location. Together, these two parts form the "location descriptor" for the annotation, which acts as an indice into the document content.

The structure level component (`doc_location`) describes the location of the annotation in terms of the structural elements of the document, using a syntax which may vary according to the type of the document. The modular implementation of the ALIEN system means that different types of document handler (which are able to create and interpret the structural component) can be loaded dynamically according to the document type specified in the `type` attribute. In Figure 5.2, the document type is HTML, and an XPointer-like construct can be used to represent the structural position of the annotation in the document, in this example the 19th paragraph (`P`) of the body of the HTML document.

Although the XPointer standard is intended to be used to reference nodes of XML documents, and is often unsuitable for HTML documents that are not well formed, it was found that provided the same technique was used for creating and resolving the structure level locator for HTML documents (more specifically, provided that HTML nodes that are not well formed are treated consistently), these locators can be used.

The media level component (`media_location`) describes the location of the annotation in terms of the lexia found at the structural element referenced by the structure level component. In Figure 5.2, the structural locator contains `text`, and the media level locator identifies the span annotation as beginning at the first occurence of the word "Using", and extending for 447 characters from the beginning of this word (position 0). Again, the modular implementation of the ALIEN system allows different media handlers to be initiated depending on the media type in order to display the annotation appropriately.

Currently document handlers have been implemented for HTML documents, and media handlers for text-based media. These handlers are delegated responsibility for creating and resolving structure and media level location descriptors, and the media handler must also display the annotation (for example, using the `type` and `colour` information of the `properties` construct in Figure 5.2).

In order to integrate the system fully with the document types and embedded media available on the WWW, document handlers for PDF, PS, plain text, and Microsoft Word documents need to be implemented. Since these types of documents do not have the structural markup of an HTML document, a different structural model, requiring a different syntax, will be required. Media handlers supporting, for example, image and video types, would allow diagrams and presentations to be annotated.

5.1.1.1 Referential integrity of annotations

A significant problem with this implementation of the ALIEN system is that although the referential integrity of the annotations is obviously important, the location descriptors are not robust to (often inevitable) changes in the document content. If a WWW document that has previously been annotated is modified and subsequently viewed, the annotations may be displayed incorrectly. The ALIEN system is robust to changes in the content of the document (but not the content of an annotation), since changing the content does not affect the document structure. Any changes to the structure of the document increase the likelihood of the location descriptor becoming unresolvable, and the subsequent loss of the annotation.

Lessons can be learnt from Phelps and Wilensky Phelps and Wilensky (2000a), in their description of a scheme for intra-document locations - an integral part of the Multivalent Annotations system Phelps and Wilensky (1997). Phelps and Wilensky demonstrate how these intra-document locations can provide robustness, that is, one should be able to make an intra-document reference to a location within an arbitrary resource, save this description, and then re-establish the location in the future, after the document has undergone some class of mutations.

In order to achieve robustness, Phelps and Wilensky posit that two elements are needed: a location descriptor (Figure 5.3), which describes a location, and a reattachment algorithm, which attempts to reposition the location descriptor within a target resource that has undergone changes.

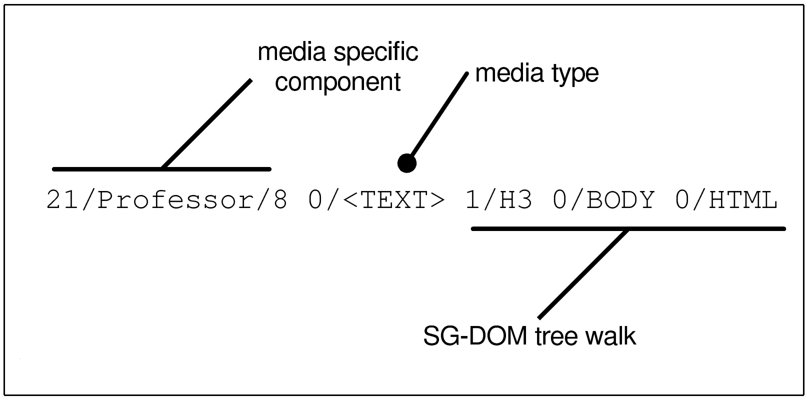


FIGURE 5.3: Components of the Multivalent Annotation location descriptor (after Phelps and Wilensky (1997)). The location descriptor is read from right to left, and interpreted as follows: Starting at the root of the SG-DOM, move to the 0th (first) child element which should be labelled `HTML`, then to the first child which should be labelled `BODY`, then to the second child which should be labelled `H3`, and finally move to the first child which should be a media leaf element labelled with its type (<`TEXT`>). The media-specific component specifies in this case that the 22nd word should be "Professor", and resolves the locator to an offset 8 characters into this word, to the point between the 'o' and 'r'.

In the Multivalent Annotation system, location descriptors are constructed and resolved using a special Document Object Model (DOM) which is much simpler and more general than the XML or HTML DOM, and accommodates common aspects of a wide variety of possible document types. This SG-DOM ("Simple, General Document Object Model") describes a document as a tree of typed internal nodes, with terminal nodes being 'media' elements. In mapping an HTML document to its equivalent SG-DOM, HTML tags that do not structure content (such as `B`, `A`, and `SPAN`) are spliced out, so that the SG-DOM represents only the structural content of the HTML document. Location descriptors consist of a core set of records which are easy to harvest from the SG-DOM, that as a whole provide considerable resilience against the independent and chaotic changing of resources and locations on the WWW:

Unique Identifier Name unique to a location within the document. These survive the most violent document changes, except its own deletion, but are available only through the foresight of the document author, and only then at locations corresponding to document elements, but not sub-element or non-element locations.

Tree Walk Describes path from the root of the SG-DOM, through internal structural nodes, to a point within media content at a leaf. These are robust to deletions of content that defeat unique id and context descriptors.

Context Small amount of previous and following information from SG-DOM leaf node.

If the location descriptor fails (that is, if any of the conditions outlined in Figure 5.3 are not met), a reattachment algorithm is initiated. Firstly, if the location descriptor has a unique ID, the document is checked for the existence of this unique ID. If the unique ID is located in the document, then the location is resolved - unique ids survive normal editing operations, unless the element itself is deleted.

If the unique ID relocation method fails, a number of operations are carried out using the tree walk record. If the structural tree walk itself fails, the siblings surrounding the element at the furthest extent to which the tree walk can be resolved are tested against the remainder of the tree walk, on the hypothesis that an element has been added or removed. Failing this, the surrounding levels of hierarchy may also be examined, on the hypothesis that one or more levels of hierarchy have been added or removed. If it is the media-specific descriptor that fails, the surrounding words are examined on the assumption that one or more words have been added or removed before the original point. These examinations of the surrounding structural or lexical context contribute appropriately to an overall uncertainty factor, for example resolving a location descriptor by moving one sibling in the tree walk adds a small amount of uncertainty; skipping or assuming a level of hierarchy in order to resolve the location descriptor adds a large amount of uncertainty.

Upon failure of the tree walk operations, the context record of the location descriptor is used to search forward and backwards from the furthest extent to which the tree walk was able to be resolved. As searching continues unsuccessfully, parts of the context phrase may be shed, and the search expanded to higher subtrees until the root is reached, adding appropriately to the uncertainty of any matches found.

If the location descriptor(s) for an annotation cannot be resolved, but are then successfully reattached, the annotation is displayed in its new position, flagged with its uncertainty factor. For annotations which cannot be attached, users are given the opportunity to reattach them manually to a new position.

In the Multivalent Annotation system, two location descriptors are required for an span annotation, representing the start and end points of the span. Spans of text are therefore robust to changes to the text within the span, including the addition of new elements, since the start and end points of the span are resolved (and reattached) independently.

5.1.2 Annotation Linker

The Annotation Linker component is used to express relationships between annotations of WWW documents, effectively annotating the annotations created with the Annotator component with associations. The Annotation Linker provides a mechanism for recording the implicit associations made by the reader as documents are read.

Similar to the configurable link types of the Annotator, the Annotation Linker can be configured to present a number of different link types to the user, each with a corresponding colour. For example, the Annotation Linker could be configured to provide link types based on the taxonomies of IBIS Kunz and Rittel (1970), SEPIA Streitz et al. (1992), DeRose DeRose (1989) or ScholOnto Buckingham-Shum et al. (2000).

Associations between annotations can be formed by simply selecting a link type from the Annotation Linker interface and then drawing a freehand line on the screen between the annotations. The Annotation Linker therefore allows annotations to be created at the pixel level of granularity (see also the Multivalent Annotation system Phelps and Wilensky (2000b), and XLibris Price et al. (1998b)), although these freeform marks are currently not preserved in the subsequent presentation of the annotation (Figure 5.4).

After drawing associations, the Annotation Linker calculates which annotations have been linked, and updates the Annotation Server with the new association information. The WWW browsers currently displaying the annotations involved in associations are instructed to refresh in order to the force the annotation information for the displayed document to be re-requested from the Annotation Server. The Annotator then presents the updated annotation information, using a semi-transparent popup window which is activated by clicking on the "links" icon of the newly associated annotations (Figure 5.5). This approach to linking annotated passages of multiple documents is different to that of XLibris (see Section 4.4), in that WWW documents can be annotated as they are read in parallel.

5.1.3 Annotation Desk

When a more complex relationship between annotations needs to be formulated by the author, annotations created with the Annotator can be dragged and dropped onto the Annotation Desk. The Annotation Desk facilitates the creation of composite nodes (the grouping of related annotations together in hierarchies), labelling of links and annotations, and the capacity for the author to meaningfully associate their own statements with their annotations.

These more complex associations are formulated through direct manipulation of a visual representation of the annotations, in a similar style to those manipulations observed in

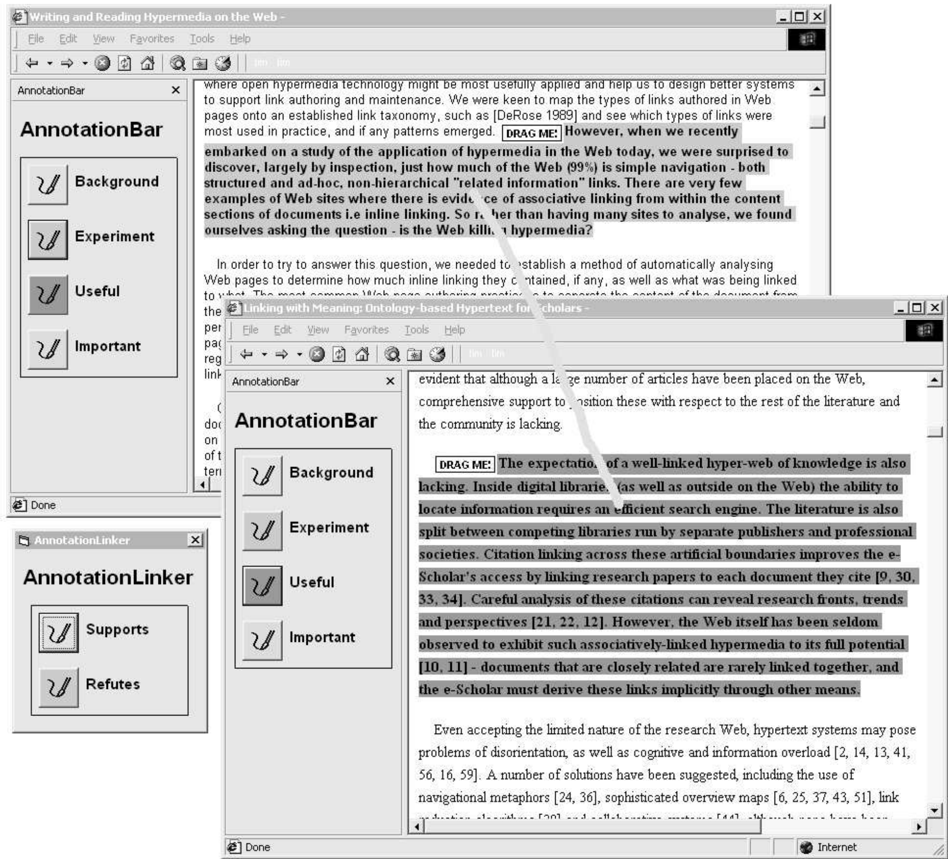


FIGURE 5.4: Creating an association with the Annotation Linker tool

the graphical mode of the Writing Environment, and the VIKI and VKB systems. However, the facility to make meaningful associations between annotations, and augment them with additional information surpasses the simple ordinate/subordinate relationships enforced by the Writing Environment, and the information "piles" classification metaphor adopted by the VIKI and VKB spatial hypertext systems. The Annotation Desk has more in common with the activity spaces of SEPIA Streitz et al. (1992), Storyspace Bernstein (1991), and Aquanet Marshall et al. (1991) systems, where more meaningful association structures are possible. The Annotation Desk extends these approaches by providing a higher level integration with both the author's operating environment (in that information nodes can be dragged and dropped on to the Annotation Desk), and the document corpus from which the annotations are derived (the visual objects represent annotations which are tied to their parent documents, rather than text nodes that have been pasted in, offering opportunities for citation and transclusion).

Associations between objects on the Annotation Desk are made using freehand marks in a similar fashion to the Annotation Linker. A palette of the available link drawing tools is derived from a configurable taxonomy of link types. The Annotation Desk is

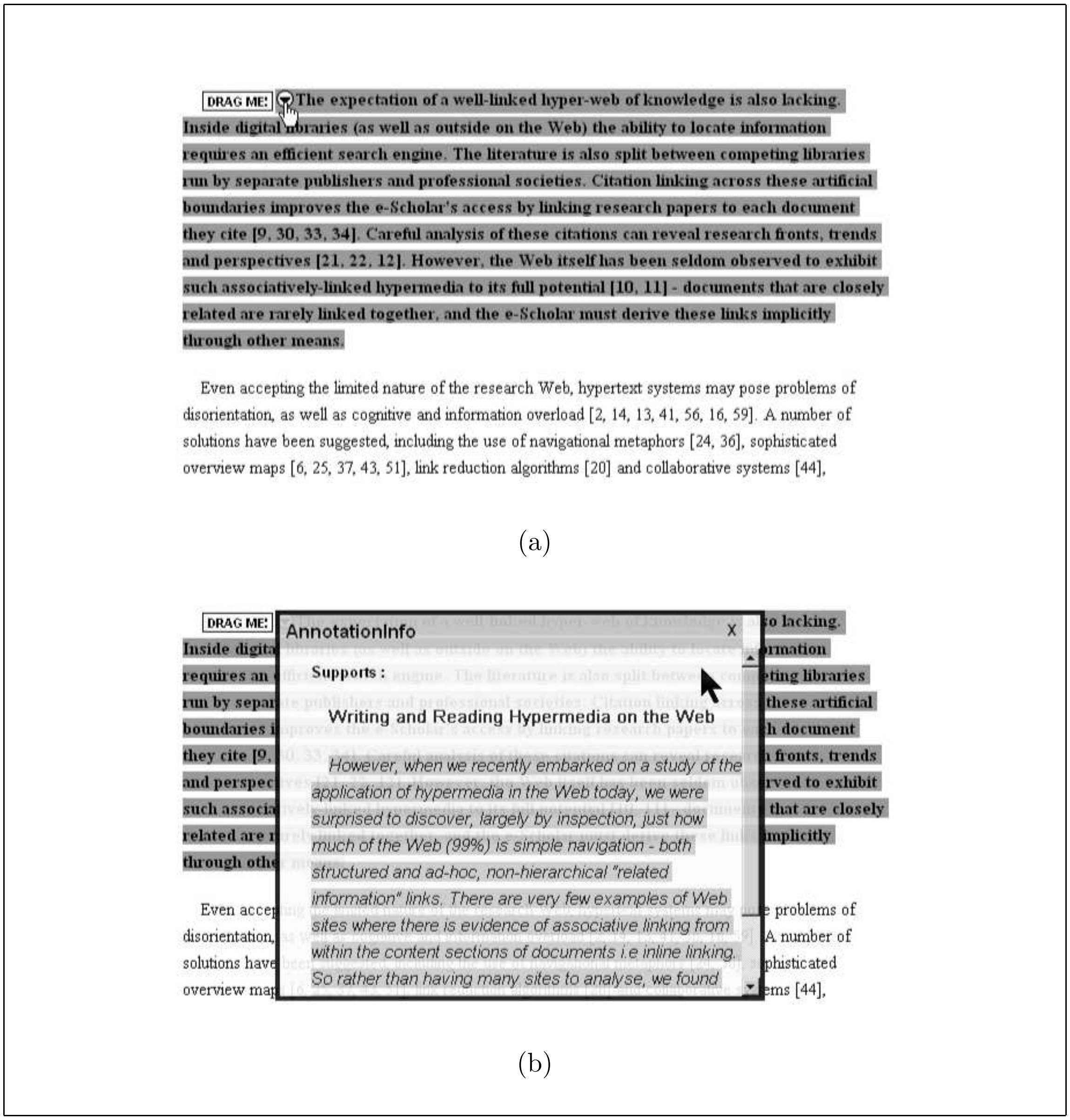


FIGURE 5.5: The result of creating an association with the Annotation Linker tool

currently able to recognise marks corresponding to the grouping of objects, the linking of two objects, and the linking of multiple objects to a single object (many to one relationship); see Figure 5.8.

The URL of a WWW document can be dragged and dropped on to the Annotation Desk, creating a visual document icon which can be meaningfully associated with other structures and objects. This may be useful when a document needs to be referenced as part of an association, but not discussed in detail.

Figure 5.6 shows how a network of annotations, labels, statements and documents can be created using the Annotation Desk. The resulting XML representation that the Annotation Server subsequently stores is shown in Figure 5.7.

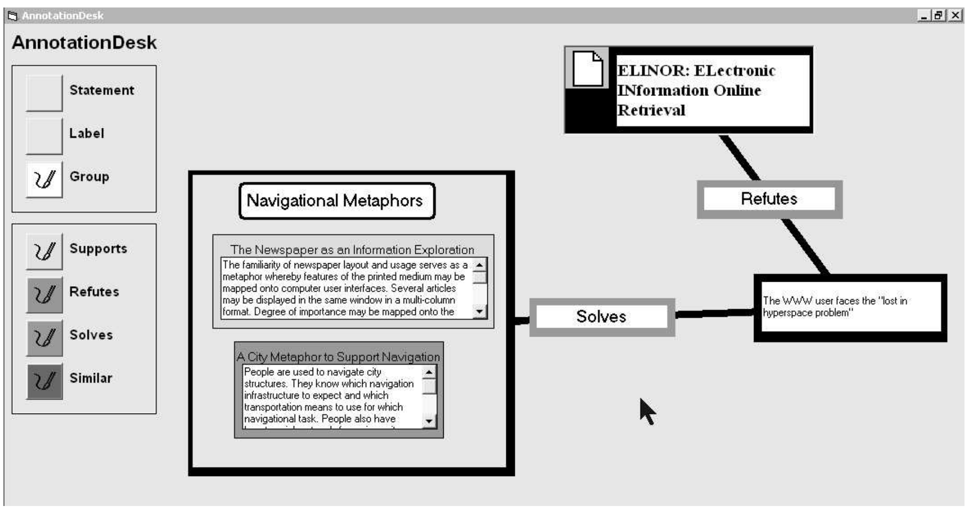


FIGURE 5.6: An association structure takes shape on the Annotation Desk.

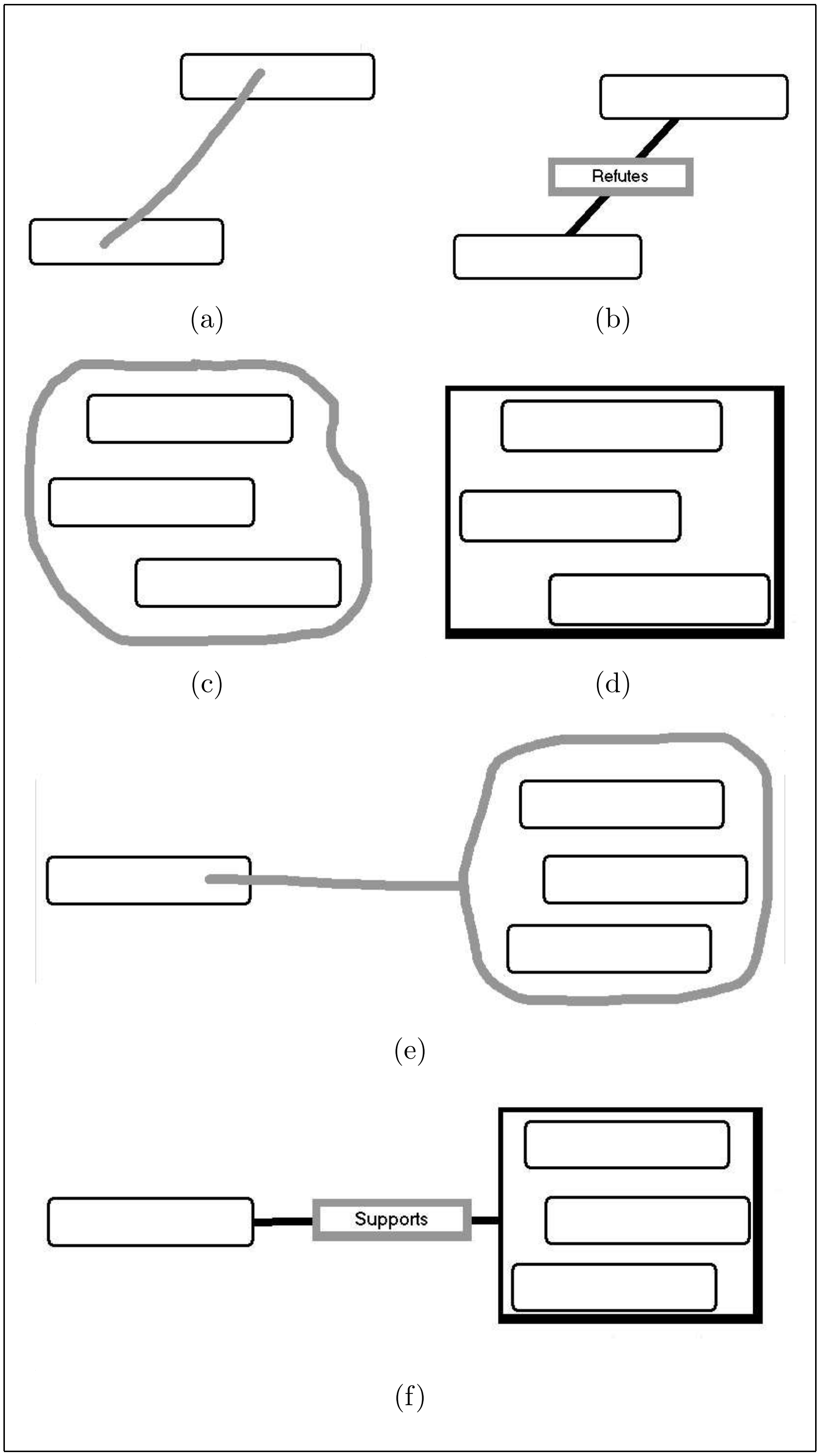
```
<item type="Annotation">
    <id>1</id>
    <location>..</location>
    <properties>
        <title>The Newspaper as an Information
            Exploration Metaphor</title>
        ...
    </properties>
</item>

<item type="Annotation">
    <id>2</id>
    <location>..</location>
    <properties>
        <title>A City metaphor to Support Navigation</title>
        ...
    </properties>
</item>

<item type="Label">
    <id>3</id>
    <content>Navigational Metaphors</content>
    ...
</item>

<item type="Group">
    <id>4</id>
    <endpoints>
        <endpoint>
            <id>1</id>
        </endpoint>
        <endpoint>
            <id>2</id>
        </endpoint>
        <endpoint>
            <id>3</id>
        </endpoint>
    </endpoints>
</item>
```

Figure 5.7: XML representation of association structure shown in Figure 5.6



(a) (b) (c) (d) (e) (f)

FIGURE 5.8: Creating associations using freeform marks on the Annotation Desk. A freeform mark joining two visual objects (a) creates a labelled association between the objects (b). A freeform mark encompassing multiple visual objects (c) creates a group or composite object (d). A freeform mark which encompasses multiple visual objects and then joins a further object to this group (e) creates a labelled one-to-many link between the new group and the individual (f).

5.1.4 Annotation Server

The Annotation Server is responsible for storing the annotations and associations created using the Annotator, Annotation Linker, and Annotation Desktop. The Annotation Server takes the form of a web server which the other components of the ALIEN system are able to communicate with using the `HTTP` protocol. Requests made to the Annotation Server invoke a script which decodes the request parameters through the CGI interface. The script is able to consult a database to store and retrieve annotation descriptions, and returns results in XML. The Annotation Server may be accessed by several applications simultaneously.

The interface offered by the Annotation Server to ALIEN components, and other applications able to communicate using the `HTTP` protocol is described in Figure 5.9.

addItem Adds an annotation, document, statement, label, or association to the database. **Parameters:** `type` the type of item (e.g. "Annotation", "Statement") `url` the URL of the resource to which this item is associated (valid only for annotations) `description` XML description of item (for example, see Figure 5.2) `endpoints` list of items involved in relationship (valid only for adding associations).
Returns: id assigned to new item.

getItemsForUrl Retrieves the annotations applicable to the resource at the given URL.
Parameters: `url` the URL for which to retrieve annotations.
Returns: XML description of annotations relevant to the given URL.

getLinksForItemId Retrieves the associations in which a given item is implicated.
Parameters: `id` id of item for which to retrieve associations.
Returns: XML description of associations in which given item is implicated.

getItem Retrieves the item with the given id.
Parameters: `id` id of item to retrieve.
Returns: XML description of item with given id.

FIGURE 5.9: The interface offered by the Annotation Server to ALIEN components

5.2 Other Work

This section describes other implementation work carried out that while not directly relevant to this work, provided practical foundations for the Annotation Linking Environment.

5.2.1 The Client DLS system and its applications

This section briefly describes the Client DLS system, a java-based link service which runs on the client machine. The client-side "proxyless" architecture developed in the implementation of the Client DLS, and its application to both the Perspectives in Electronic Publishing (PeP) online journal Hitchcock and the Conceptual Open Hypermedia System (COHSE) Bechhofer et al. (2001a,b) has provided a solid foundation for the development of the Annotation Linking Environment. HTML DOM access and manipulation technologies developed for the Client DLS have also proved useful when applied to this work.

5.2.1.1 Brief tour of link services

A number of hypertext systems have advocated the separation of links from the content of documents. In such systems, links are not embedded in node contents, but stored separately in a link database, and mediated by a link service. This has the advantage of better site manageability. Examples of hypertext link services include :

- **Intermedia**
  The Intermedia hypertext system used a link service which provided hyperlinks to other components in the system through socket connections Meyrowitz (1986); Yankelovich et al. (1988).

- **Sun's Link Service**
  Sun's Link Service provided a protocol and development tools to add hypertext functionality to existing Sun applications Pearl (1991).

- **Microcosm**
  Microcosm was developed to provide open hypertext linking functionality across a variety of different sources of information, such as video, images, sound, and text Fountain et al. (1990). The open architecture of the system meant that third party applications could be integrated, allowing its documents to be augmented with hypertext links from the link service Davis et al. (1994). When a document is opened in an application which is integrated with the Microcosm system, the application requests links appropriate to that document. A number of specialised

link services are chained sequentially - suggested links for the document pass along the chain, with each link service adding or removing links as appropriate. The final set of suggested links for the document are returned. Link services may provide three different link types: specific links can only be applied between specific points in specific nodes; local links can be applied from any point in a specific document that matches the link anchor; generic links can be applied from any point in any document that matches the link anchor.

- **Hyperwave** The Hyperwave link service was originally part of the Hyper-G system Andrews et al. (1995b), which was adapted to provide a link service to the WWW Andrews et al. (1995a) before becoming a commercial product called Hyperwave Maurer (1995).

- **The Distributed Link Service** The Distributed Link Service (DLS) takes the Microcosm philosophy a stage further by providing a link service for the WWW Carr et al. (1994), implemented as a proxy server Carr et al. (1995).

5.2.1.2 Moving a WWW link service to the client side

The implementation of a WWW link service as a proxy ensures platform and browser independence, and integrates seamlessly with the user's browsing experience. However, the use of a proxy raises other issues. The user has to manually configure their browser to ensure that requests for WWW pages are sent through the proxy, and if the user is already using a proxy for some other service it is difficult to chain the proxies together. The user may also have to subscribe to the link service.

The Client DLS was implemented to address these issues, and does so by providing a java applet which is downloaded over the WWW and runs on the client machine. The applet is able to detect when a new WWW document is loaded in the Internet Explorer web browser, and traverse the DOM of the WWW document, augmenting it with appropriate links derived from a linkbase.

The Client DLS has been deployed to provide an added value service to the Perspectives in Electronic Publishing (PeP) online journal Hitchcock. The PeP journal currently stores information about approximately 450 papers that are related to electronic publishing, with the articles themselves distributed across the WWW. The Client DLS uses meta-information about the articles in the PeP journal to construct a linkbase of concepts, which include author's names, article titles, keywords, publishing organisations, high level topics, and journal titles. These links are used to augment the PeP website itself (for example, each occurence of an authors name links to all the articles that that author has produced, each occurence of an article title links to the article itself), and also the articles, which are distributed across the WWW (for example, citations of articles published by PeP are linked to the meta-information for the cited article held at the PeP site). The Client DLS therefore allows the services provided by the PeP online site to go beyond the boundaries of the site itself, without the casual visitor having to subscribe or configure a proxy link service.

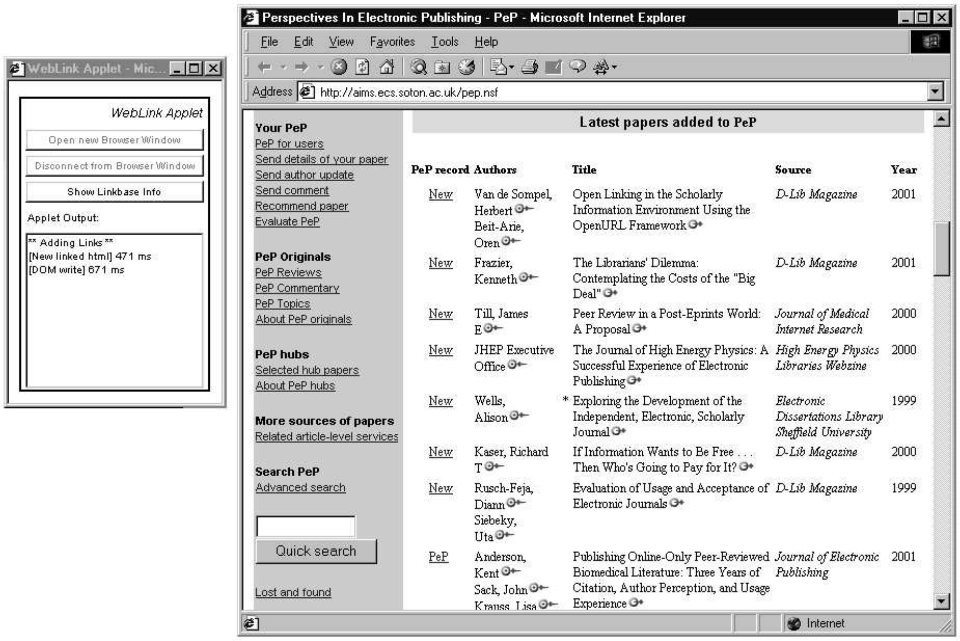


FIGURE 5.10: The Client DLS in action, augmenting the PeP website with links. The Client DLS applet runs in a separate window (a), and reacts to changes in the main browser window (b). When the user visits a new page, the Client DLS consults the PeP linkbase, and inserts links appropriate to the viewed page into the main browser window. These links appear as small icons - an "onsite" icon is used to represent links which point to further information on the PeP website itself, and an "offsite" icon represents links which point to resources on the WWW, not hosted by PeP. In this example, the author names are linked to further information about the author (including a list of publications), and the article titles are linked to the actual articles on the WWW.

The Client DLS principles have also been applied to the COHSE project Bechhofer et al. (2001a,b), where a Client DLS-based component communicates with a thesauri-based Ontology Service (able to provide similar, broader, and narrower alternatives for a given term in a document) and a Resource Service (able to provide URLs to resources related to a given term in the thesaurus) in order to determine the links most suitable for augmenting the document currently viewed in the browser.

5.2.2 OntoPortal - an ontological hypermedia portal

This section briefly describes the OntoPortal system Kampa et al. (2001), an ontology-based portal service which annotates the resources of a research community with meaningful associative links (Figure 5.11). The OntoPortal system uses an ontology to improve the linking of research literature together with the WWW pages of projects, institutions, and individual researchers - providing a principled way of describing both the topic under discussion, and the process by which it was produced in order to allow other researchers to better understand the work and its surrounding context (Figure 5.13).

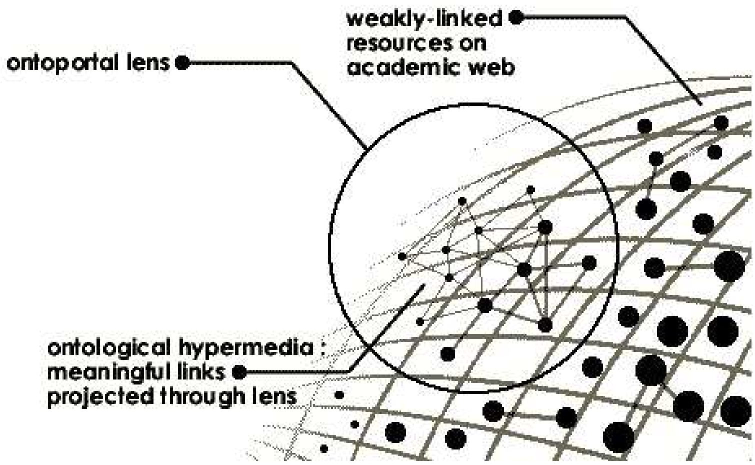


FIGURE 5.11: OntoPortal overlays a meaningful meta-layer of information onto a sparsely linked research community

The OntoPortal ontology (Figure 5.12) makes explicit the different types of resources available to a researcher in any field, and formally defines the relationships between them. The ontology is not domain-oriented (it can reasonably be applied to any research field), nor simply bibliography-oriented (hence transcending the features available through independently published Digital Libraries). Since the ontology is relatively simple and intuitive, it is promoted to the forefront of the user interface to provide an overview and navigation tool. The ontology enforces typed links (derived from the named relationships between concepts in the ontology), and bi-directional, n-ary associations.

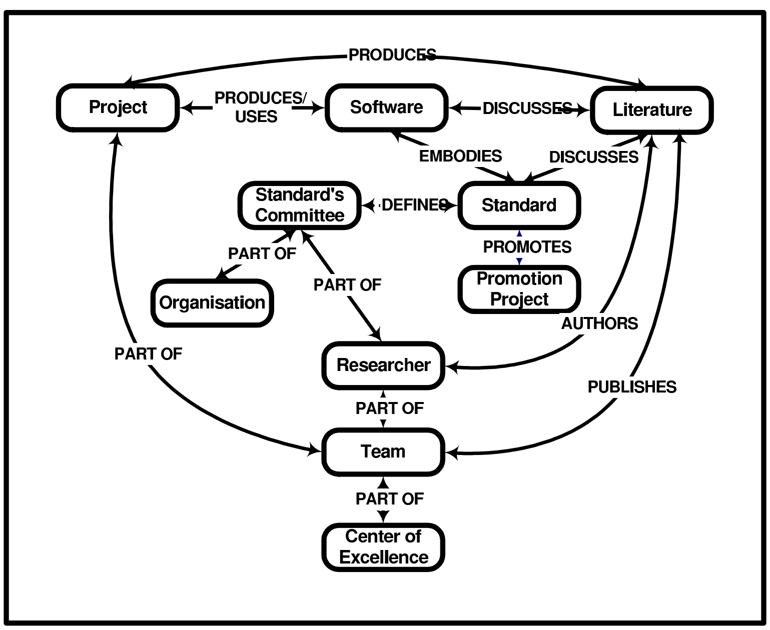


FIGURE 5.12: The OntoPortal Ontology

The annotation of resources in a scholarly community is facilitated through simple

HTML forms, into which the metadata describing the resource, and more importantly, the associations with other resources, is entered by dedicated editors. To integrate the resource with its proper context, all valid ontological relationships are presented to the annotator (based on the resources that have already been annotated), who chooses the most appropriate. Annotations can be made only at a the coarsest level of granularity - the entire document will be annotated with meaningful link information. The annotations are stored by the portal in XML form, and an XSL transformation process is used to render the annotations into a presentation format.

Researchers themselves can also annotate any one of the community resources which OntoPortal encapsulates, using a threaded discussion facility.

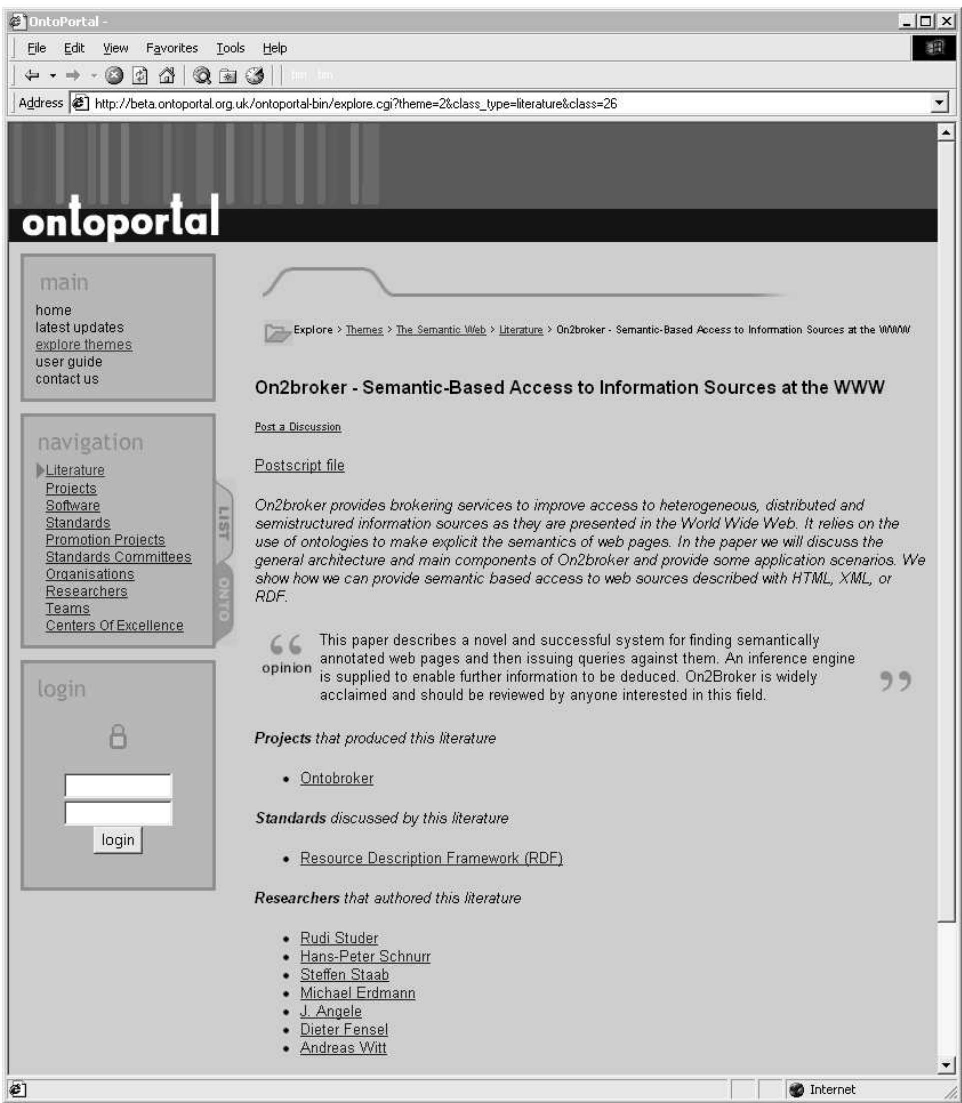


FIGURE 5.13: The application of the OntoPortal server to the metadata research community. In this example, metadata about a research paper is displayed, along with ontology-driven links to related projects, standards, and researchers.

5.2.2.1 Related Work

SHOE Heflin et al. (1998) allows researchers to annotate their WWW resources with metadata, in order to build an ontology-driven distributed knowledge base. Inferences can then be made across the knowledge base, producing supplementary knowledge.

The KA^2 initiative Benjamins et al. (1998) applies the OntoBroker framework Fensel et al. (1998) to provide a coherent set of tools with which to design ontology-based portals, placing particular emphasis on the ontological engineering and inferencing processes.

ESKIMO Kampa et al. (2001) uses a more scholarly support-oriented ontology to model a scholarly community, focussing in particular on the useful inferences that can be formulated, and the augmentation of traditional bibliographic analysis methods with ontological knowledge.

The ScholOnto system Buckingham-Shum et al. (2000) is a digital library server that overlays on conventional documents and metadata a semantic web of scholarly claims, discourse and perspectives.

Chapter 6

Further Work

The Annotation Linking Environment demonstrates the principles by which the capturing of associations between documents as they are read may be facilitated. The next logical step is to demonstrate how the association structures created in the ALIEN system can be most usefully applied to aiding the authoring process and the creation of associative links between the author's new ideas and their theoretical foundations.

This chapter discusses extensions to the current ALIEN system, highlights interesting findings from the reading and writing diary recorded during the authoring of this report, and introduces possible approaches to the re-purposing of associations captured in the ALIEN system to the writing and associative linking processes.

Finally, directions in which this work can be further extended are considered, including suppporting collaborative writing, enriching existing WWW content with authors' associations, and using authors' associations as a recommender system.

6.1 Extensions to the Annotation Linking Environment

A number of useful extensions which could be made to the Annotation Linking Environment are listed below.

Standard Compliance For compliance and interoperability, the ALIEN system should be extended to support the W3C annotation standard. This would involve subclassing the W3C RDF annotation description Koivunen et al. (2000), in order to meaningfully describe the associations with other annotations. The Annotea RDF annotation server could be deployed in place of the ALIEN Annotation Server, to store and serve these annotations.

Robust Annotations Phelps and Wilensky Phelps and Wilensky (1997) have demonstrated that robust annotations can be achieved by means of a location descriptor and reattachment algorithm. The ALIEN system already uses a similar location descriptor to describe the positions of annotations, but has no implementation of a reattachment algorithm. At the very least, the ALIEN system should be adapted to use two location descriptors to describe the position of a span annotation, to make the system robust to changes within the span.

Zooming/Panning Annotation Desk As association structures constructed on the annotation desk become more complex, facilities for zooming and panning across the structure may be required (see for example Bederson et al. (2000)).

Comprehensive WWW Document Support In order to support the full range of document types on the WWW, the ALIEN system needs to be augmented with document handlers for Adobe PDF, plain text, Microsoft Word, and Postscript documents, in order that these documents can be annotated. Annotation facilities for images (such as those provided by InterNote Catlin et al. (1989)) could also be considered.

6.2 Considerations arising from writing diary observations

During the writing of this report, a diary was kept for the purpose of recording elements of the reading and writing process that were applicable to the use of the ALIEN environment.

The ALIEN system was used to record associations made in reading a small subset of the literature referenced by this report, with traditional annotative techniques deployed on the remainder. The ALIEN system proved promising in improving the permanence and clarity of the recall of the associations made and recorded using annotative associations. Reconstructing associations made using pen marks on paper was less effective.

Where these observations are relevant to the future development of the Annotation Linking Environment, they are reproduced below.

No clear separation between reading and writing process Associations are still being made during the writing process (for example when the context of an annotation is being re-read, and a new association comes to mind, and needs to be captured). Writing is therefore continuously interrupted in order to capture new associations. This confirms the writing models of Carr (1995); Smith et al. (1987), and emphasises the importance of seamless switching between writing and capturing associations, for example, if the current focus of writing is describing a particular association, the writer should be able to be easily switch to tools for augmenting of otherwise editing this association as new associations come to mind (compare with activity space switching in SEPIA Streitz et al. (1989)).

Richer link taxomony required During the capture of associations, it was observed that the simple set of available relationship types (*Supports*, *Refutes*, *Solves*, and *Similar*) was insufficient to capture all the associations made, even across the relatively small subset of the literature annotated in the ALIEN system. An *Analyses/Discusses* relation was required, as well as a mechanism for capturing the realisation that certain research projects formed a basis for other projects (*Precursor/Successor* relationships). A richer taxonomy of associations (for example Buckingham-Shum et al. (2000)) would form a more solid foundation for association capture.

Adding context to captured associations It would be useful to be able to give some indication as to where associations are intended to be discussed/described in the new writing. This could take the form of capturing the context under which the association is intended to appear (for example, "introductory material", "authoring systems review"). The WE system Smith et al. (1987) allows authors to transform graphical networks to hierarchical structures, which are then populated with the new writing. Similarly, associations created in the ALIEN system could be filtered according to their context, to produce a hierarchical basis for writing. This would be useful in the case that a documents contains many relevant ideas, but these ideas are not associated with any other documents.

Composite annotations In using the ALIEN system, it was observed that creating associations between documents was not as simple as creating one annotation on each document and associating these annotations using the appropriate link type. The annotations are intended to highlight the parts of the annotated documents that are relevant to the association, but often these parts are spread throughout the document. It would be more useful to be able to combine these annotations into a composite annotation which represents a documents contribution to an association. This composite annotation could then be manipulated in the same way that a single annotation is currently manipulated. This would enable an association-based view of documents, where the content of a document is viewed in the context of a particular association in which it is involved, rather like the Reader's Notebook tool in the XLibris system Price et al. (1998b), where readers can view all their annotations end to end, with the surrounding content removed. A similar concept could be applied to representing the contributions of multiple papers (for example, papers written by the same group of authors, but discussing different areas of a system) as a single composite annotation.

Transclusions The ability to "pull" the content of an annotation into a document, preferably as a "warm link" (see for example Nelson (1980); Catlin et al. (1989)), or even as a copy of the text highlighted by the annotation could be useful, particularly when text needs to be transcluded or copied verbatim. Associations could also be "pulled" into the new writing, resulting in the appropriate associative links and semantics being inserted.

6.3 Applying captured associations to hyperwriting

We have seen how authors typically have new ideas to contribute, and asserts implicit relationships between these new ideas and existing ideas already published in order to demonstrate both the reliability of the conceptual foundation being built on, and the innovation and significance of the new ideas. It is these implicit relationships that are closest to the vision of hypertext - associative links that are not explicit between related documents but that can be extracted by author in their reading of the literature.

The ALIEN system demonstrates how these associations can be captured using an annotation-based interface and interaction paradigm, but the important work is demonstrating the benefit of capturing this information, and discovering whether a better hyperwriting process can result, in terms of both the content and coverage of the new writing itself, and the meaningful associative links that it provides, leading the reader into the global context.

The next logical step in the implementation is therefore to design and build a test platform which forms a basis for the investigation of methods for re-purposing these associations in order to provide support for writing and linking, by allowing the writer to review and navigate the annotative structures that were created during the reading process.

Possible approaches include :

Visual navigation of associations in parallel with writing Associations should be traversable in both directions, and annotations at endpoints of associations should be linked back to their surrounding context in the document they annotate.

Synchronisation of visualisation with writing process As the writer enters text, the current context is detected (for example, the idea which is currently being discussed), and the visualisation automatically changes focus as appropriate (for example, focusing centrally on the idea being discussed, with surrounding visualisation of associated ideas).

Links created using a "link-as-you-type" paradigm Automatic insertion of meaningful associative links appropriate to the current topic under discussion. Associative links could be stored separately from the hypertext, and inserted as and when the hypertext is viewed (c.f. Client DLS system, section 5.2.1), perhaps applying particular filtering mechanisms.

6.4 Collaborative writing

Writing very often takes place not by an author in isolation, but by many authors in collaboration, and tools for collaborative authoring must provide support for many complex communication and cooperation processes including synchronisation, videoconferencing and versioning (see for example Streitz et al. (1992)). The abundance of collaborative writing projects, especially in research, suggests that providing support for collaborative writing in the context of this work may be important. For example, collaborating authors would need to be able to view and navigate associations created by their co-authors, and augment these associations with further relationships and viewpoints.

6.5 Enriching the WWW

The product of using the ALIEN system is a repository of meaningful knowledge about relationships between possibly unlinked resources on the WWW. We have already seen how this repository can be re-purposed to provide support for hyperauthoring and associatively linking the new writing into the global context, but these inter-document associations could potentially be used to enrich the existing WWW document base. After all, it is unlikely that authors of these already (electronically) published writings would return to their earlier work and make explicit the associations between the ideas it describes.

The associations created by authors as they describe and build upon the existing WWW idea base could be presented to readers of these older documents, making clear the context of (and of course, linking to) the later work in which associations between these documents are discussed. Adaptive hypermedia systems may even present only particular associations, dependant on user profiles or previous browsing history.

6.6 Annotation-based recommender system

We have seen how the combining of annotations made by several users can be used as a census to decide which parts of an annotated document are most important 4.4. The repository of meaningful knowledge about relationships between WWW resources produced by the ALIEN system could be repurposed to provide a census-based association network between WWW documents. Those associations being made most frequently by authors as they read through the document corpus could be presented as the strongest associations tying two ideas. New readers could therefore familiarise themselves with the community of literature by traversing the strongest associations first before attempting the weaker ones (compare with history-based recommendation techniques presented in Pikrakis et al. (1998)).

6.7 Work schedule

- **June 2001 - September 2001**
  - Evaluate and implement proposed extensions to Annotation Linking Environment.
  - Review knowledge management and lifecycle literature for theoretical grounding in knowledge reuse principles.
  - Review information visualisation and navigation literature to provide theoretical grounding for implementation of test platform for visualisation and navigation of associations.
  - Implement test platform for evaluation of the visualisation and navigation of associations, in conjunction with writing process.
- **October 2001 - December 2001**
  - Review evaluation literature for theoretical grounding in model-based evaluation.
  - Finalise implementation in preparation for model-based evaluation.
  - Carry out formal model-based evaluation of annotation linking environment and association visualisation and navigation tools.
- **January 2001 - March 2002**
  - Evaluation and pursuit of suggested further research directions (collaborative authoring considerations, the application of captured associations to the WWW document corpus, and the application of captured associations to an annotation-based recommender system).
- **April 2002 - September 2002**
  - Write thesis.

Chapter 7

Conclusion

This report has focussed on how an annotation-based approach could be used to capture inter-document relationships in order to allow the WWW to evolve into the rich associative hypertext network envisioned by early hypertext pioneers. The **A**nnotation **LI**nking **EN**vironment (ALIEN) has been implemented to demonstrate how the implicit associations made between documents by authors as they read the literature can be captured using an annotative interface.

As a foundation for this research, the early hypertext pioneer's visions of an online database of the world's literature, where everything can be associatively linked to everything else, were explored. The globally available distributed hypertext platform known as the World-Wide Web (WWW) was identified as the closest hypertext system to this vision (in terms of widespread public access), but it was found that the WWW is not living up to the promise of hypertext - that is, associative linking was found to be in little evidence on the WWW. It was posited that the current state of the WWW in terms of associative linking practices is influenced by recommendations that associative linking be minimised to avoid potential problems of disorientation, comprehension, and cognitive overload. Metrics which suggest that the authoring of hypertext links incurs a significant effort to the hypertext development process may also be a contributing factor.

In response to these concerns, research into the annotation process was initiated. As authors read, they make annotations, and also form implicit relationships between ideas presented in documents. It was hypothesised that if these associations could be captured through annotation (difficult using a traditional paper medium) in a fluent and effortless manner at the time that they are formed by the author, the effort required to make associative links in the new writing would be reduced. Furthermore, by attaching semantics to these associations (for example, the type of association, and the context in which it was made), it was also hypothesised that *meaningful* associative links could be produced, reducing the risk of disorientation, miscomprehension, and cognitive overload through discipline of the hypertext network.

A survey of hyperauthoring and annotation systems research was carried out, in order to provide a theoretical foundation to the implementation of a system able to provide tools for annotating and associating ideas in documents as they are read, which led to the implementation of the ALIEN system. The implementation of the Client DLS and OntoPortal systems were also introduced, and their practical contributions to the ALIEN system discussed.

Even before the proposed changes listed in 6 have been made to the ALIEN system, experiences with its use in capturing associations as part of the authoring process of this report have certainly demonstrated the advantages of this approach compared to capturing and applying these associations in a traditional paper-based medium.

Further implementation is scheduled to produce a tool that repurposes the associations captured at the reading stage of authoring to assist the hyperwriting and linking processes. Proposed approaches include interactive visualisation and navigation of the captured associations, automatic refocussing of the visualisation as the context of the writing changes (for example, as the author leaves one idea to begin discussion of another, the display should change to reflect the associations surrounding the new idea), and automatic insertion of associative links into the hypertext.

Further research directions were also suggested, including the applications of such a system for the capturing and repurposing of associations to a collaborative authoring environment, and the repurposing of captured associations to enrich the existing WWW document corpus in order to allow the WWW to involve in all its richness.

Bibliography

Annette Adler, Anuj Gujar, Beverly L. Harrison, Kenton O'Hara, and Abigail Sellen. A diary study of work-related reading: Design implications for digital reading devices. In *Proceedings of CHI '98 Human Factors in Computing Systems, Los Angeles, California, USA*, volume 1 of *Reading and Writing*, pages 241–248, 1998.

K. Andrews, F. Kappe, and H. Maurer. Serving Information to the Web with Hyper-G. *Computer Networks and ISDN Systems*, 27(6):919–926, 1995a.

Keith Andrews, Frank Kappe, and Hermann A. Maurer. The Hyper-G network information system. *The Journal of Universal Computer Science*, 1(4):206–220, April 1995b.

Lisa Baron, Jean Tague-Sutcliffe, and Mark T. Kinnucan. Labelled, typed links as cues when reading hypertext documents. *Journal of the American Society for Information Science*, 47(12):896–908, 1996.

Sean Bechhofer, Les Carr, Carole Goble, and Wendy Hall. Conceptual open hypermedia = the semantic web? In *Proceedings of The 2001 International Workshop on the Semantic Web*, May 2001a.

Sean Bechhofer, Les Carr, Carole Goble, Wendy Hall, and Dave De Roure. Conceptual linking: Ontology-based open hypermedia. In *Proceedings of the Tenth International World Wide Web Conference, Hong Kong*, May 2001b.

B Bederson, J Meyer, and L Good. Jazz: An extensible zoomable user interface graphics toolkit in java. In *Proceedings of the 2000 ACM Conference on User Interface Software and Technology*, pages 171–180, 2000.

R. V. Benjamins, D. Fensel, and A. G. Perez. Knowledge management through ontologies. In *Proceedings of the 1998 Conference on Practical Aspects of Knowledge Management*, 1998.

Tim Berners-Lee. *Information Management: A Proposal*. CERN DD/OC, March 1989.

Mark Bernstein. An apprentice that discovers hypertext links. In *Hypertext: Concepts, Systems and Applications, Proceedings of the Hypertext '90 Conference, INRIA, France*, Turning Text into Hypertext, pages 212–223, 1990.

Mark Bernstein. Deeply intertwingled hypertext: The navigation problem reconsidered. *Technical Communication*, pages 41–47, 1991.

Mark Bernstein, Jay David Bolter, Michael Joyce, and Elli Mylonas. Architectures for volatile hypertext. In *Proceedings of the ACM Hypertext '91 Conference, San Antonio, Texas, USA*, Discovering Structure II, pages 243–260, 1991.

Michael Bieber and Jiangling Wan. Backtracking in a multiple-window hypertext environment. In *Proceedings of the ACM Hypertext '92 Conference, Milano, Italy*, Papers, pages 158–166, 1994.

Simon Buckingham-Shum, Enrico Motta, and John Domingue. Scholonto: An ontology-based digitial library server for research documents and discourse. *International Journal on Digital Libraries*, 3(3), August/September 2000.

Vannevar Bush. As we may think. *Atlantic Monthly*, 176(1):101–108, July 1945.

L. A. Carr, D. DeRoure, W. Hall, and G. J. Hill. The Distributed Link Service: A Tool for Publishers, Authors and Readers. *Proceedings of the Fourth International World Wide Web Conference, Boston, Massachusetts, USA*, 1:647–656, 1995.

L. A. Carr, W. Hall, H.C Davis, D. DeRoure, and R. Hollom. The Microcosm Link Service and its Application to the World Wide Web. In *Proceedings of the First International World Wide Web Conference, Geneva, Switzerland*, May 1994.

Leslie Carr. *Structure and Hypertext*. PhD thesis, Department of Electronics and Computer Science, University Of Southampton, 1995.

Leslie Carr, Wendy Hall, and Timothy Miles-Board. Writing and reading hypermedia on the web. Technical Report ECSTR-IAM00-1, University of Southampton, Southampton, UK, February 2000.

Timothy Catlin, Paulette Bush, and Nicole Yankelovich. Internote: Extending a hypermedia framework to support annotative collaboration. In *Proceedings of the Hypertext '89 Conference on Hypertext, Pittsburgh, Pennsylvania, USA*, Applications in Writing, pages 365–378, 1989.

Jeff Conklin. Hypertext: a survey and introduction. *IEEE Computer*, 20(9):17–41, September 1987.

Jeff Conklin and Michael L. Begeman. gIBIS: A hypertext tool for exploratory policy discussion. In *Proceedings of the 1988 ACM Conference on Computer Supported Cooperative Work*, Structured Communication Technologies, pages 140–152, 1988.

Steve Cousins, Michelle Baldonado, and Andreas Paepcke. A systems view of annotations. Xerox PARC Tech Report P9910022, April 2000.

Clare Davies, Mel Collier, Anil Sharma, and Dian Zhao. *ELINOR Electronic Library Project*. Bowker Saur, 1998.

H. C. Davis, S. J. Knight, and W. Hall. Light hypermedia services: A study of third party application integration. In *Proceedings of the ACM Hypertext '94 Conference, Edinburgh, Scotland*, pages 41–50, September 1994.

Laura De Young. Linking considered harmful. In *Hypertext: Concepts, Systems and Applications, Proceedings of the Hypertext '90 Conference, INRIA, France*, Designing and Reading Hyperdocuments, pages 238–249, 1990.

Steven J. DeRose. Expanding the notion of links. In *Proceedings of the Hypertext '89 Conference on Hypertext, Pittsburgh, Pennsylvania, USA*, Usability, Links, and Fiction, pages 249–257, 1989.

Douglas C. Engelbart. A conceptual framework for the augmentation of man's intellect. In P. D. Howerton and D. C. Weeks, editors, *Vistas of Information Handling*, volume 1, pages 1–29. Spartan Books, Washington, D.C., 1963.

D. Fensel, S. Decker, M. Erdmann, and R. Studer. Ontobroker in a nutshell. *Lecture Notes in Computer Science*, 1513, 1998. ISSN 0302-9743.

J. Fiderio. A grand vision. *Byte*, 13(10):237–244, October 1988.

A. Fountain, W. Hall, I. Heath, and H. C. Davis. Microcosm: An Open Model for Hypermedia With Dynamic Linking. In A. Rizk, N. Streitz, and J. André, editors, *Hypertext: Concepts, Systems and Applications, Proceedings of the Hypertext '90 Conference, INRIA, France*, pages 298–311, November 1990.

Richard Furuta, Frank M. Shipman, III, Catherine C. Marshall, Donald Brenner, and Hao wei Hsieh. Hypertext paths and the world-wide web: Experiences with walden's paths. In *Proceedings of the ACM Hypertext '97 Conference, Southampton, UK*, WEB Integration and Application, pages 167–176, 1997.

P. A. Gloor. CYBERMAP: Yet another way of navigating in hyperspace. In *Proceedings of the ACM Hypertext '91 Conference, San Antonio, Texas, USA*, Presentation Issues, page 107, 1991.

Robert Glushko. Design issues for multi-document hypertexts. In *Proceedings of the Hypertext '89 Conference on Hypertext, Pittsburgh, Pennsylvania, USA*, page 51, 1989.

Gene Golovchinsky and Mark Chignell. The newspaper as an information exploration metaphor. *Information Processing and Management*, 33(5):663–683, September 1997.

Gene Golovchinsky, Morgan N. Price, and Bill N. Schilit. From reading to retrieval: Freeform ink annotations as queries. In *Proceedings of the 1999 ACM SIGIR conference on Research and development in information retrieval*, pages 19–25, 1999.

J. M. Haake and B. Wilson. Supporting collaborative writing of hyperdocuments in SEPIA. In *Proceedings of the 1992 ACM Conference on Computer Supported Cooperative Work*, Collaborative writing, pages 138–146. ACM Press, 1992.

Jeff Heflin, James Hendler, , and Sean Luke. Reading between the lines: Using shoe to discover implicit knowledge from the web. In *Proceedings of the 1998 Conference on Artificial Intelligence*, 1998.

Steve Hitchcock. Perspectives in electronic publishing. http://aims.ecs.soton.ac.uk/pep.nsf.

Satoshi Ichimura and Yutaka Matsushita. Another dimension to hypermedia access. In *Proceedings of the ACM Hypertext '93 Conference, Seattle, Washington, USA*, Papers, pages 63–72, 1993.

Eastgate Systems Inc. Storyspace: A hypertext tool for writers and readers. http://www.eastgate.com/Storyspace.html.

Michael Joyce. Storyspace as a hypertext system for writers and readers of varying ability. In *Proceedings of the ACM Hypertext '91 Conference, San Antonio, Texas, USA*, Technical Briefings, pages 381–387, 1991.

José Kahan, Marja-Riitta Koivunen, Eric Prud'Hommeaux, and Ralph R. Swick. Annotea: An open RDF infrastructure for shared web annotations. In *Proceedings of the Tenth International World Wide Web Conference, Hong Kong*, 2001.

Simon Kampa, Timothy Miles-Board, Les Carr, and Wendy Hall. Linking with meaning: Ontological hypertext for scholars. Technical Report 0-854327-37-1, University of Southampton, Southampton, UK, 2001.

Marja-Riitta Koivunen, Dan Brickley, José Kahan, Eric Prud'Hommeaux, and Ralph Swick. The W3C collaborative web annotation project ... or how to have fun while building an RDF infrastructure. http://www.w3.org/2000/02/collaboration/annotation/papers/annotationinfrastructure, 2000.

R. Kreutz, B. Euler, and K. Spitzer. No longer lost in WWW-based hyperspaces. In Klaus Tochtermann, Jörg Westbomke, Uffe K. Wiil, and John J. Leggett, editors, *Proceedings of the ACM Hypertext '99 Conference, Darmstadt, Germany*, pages 133–134. ACM Press, February 21–25 1999. ISBN 1-58113-064-3.

W. Kunz and H. Rittel. *Issues as Elements of Information Systems*. Center for Planning and Development Research, University of California at Berkeley, 1970.

Marcy Lansman, John B. Smith, and Irene Weber. Using the writing environment to study writer's strategies. *Computers and Composition*, 10(2):71–92, April 1993.

David M. Levy. I read the news today, oh boy: Reading and attention in digital libraries. In *Proceedings of the 1997 ACM International Conference on Digital Libraries*, User Communities, pages 202–211, 1997.

D. Lowe. SYNVIEW: The design of a system for cooperative structuring of information. In *Proceedings of the 1986 ACM Conference on Computer Supported Cooperative Work*, pages 376–386. ACM Press, 1986.

Patrick Lynch and Sarah Horton. Imprudent linking weaves a tangled web. *IEEE Computer*, 30(7):115–117, July 1997.

Catherine Marshall. Toward an ecology of hypertext annotation. In *Proceedings of the ACM Hypertext '98 Conference, Pittsburgh, USA*, Novel Systems and Interfaces, pages 40–49, 1998.

Catherine C. Marshall, Frank G. Halasz, Russell A. Rogers, and William C. Janssen, Jr. Aquanet: A hypertext tool to hold your knowledge in place. In *Proceedings of the ACM Hypertext '91 Conference, San Antonio, Texas, USA*, Discovering Structure II, pages 261–275, 1991.

Catherine C. Marshall and Frank M. Shipman, III. Spatial hypertext and the practice of information triage. In *Proceedings of the ACM Hypertext '97 Conference, Southampton, UK*, Structure and Spatiality, pages 124–133, 1997.

C.C. Marshall, M.N. Price, G. Golovchinsky, and B.N. Schilit. Collaborating over portable reading appliances. *Personal Technologies*, 3(1), 1999.

H. Maurer. *Hyper-G now Hyperwave: the next generation Web solution.* Addison Wesley, 1995.

Emilia Mendes and Wendy Hall. Applying measurement principles to improve hypermedia authoring. *The New Review of Hypermedia and Multimedia*, 5:105–132, 1999.

Emilia Mendes and Wendy Hall. Towards the prediction of development effort for web applications. In *Proceedings of the ACM Hypertext 2000 Conference, San Antonio, Texas, USA*, pages 242–243. ACM Press, May 2000.

N. Meyrowitz. Intermedia: The Architecture and Construction of an Object-Oriented Hypermedia System and Application Framework. In *Proceedings of OOPSLA '86*, pages 186–201, September 1986.

Stuart Moulthrop. Toward a rhetoric of informating texts. In D. Lucarella, J. Nanard, M. Nanard, and P. Paolini, editors, *Proceedings of the ACM Hypertext '92 Conference, Milano, Italy*, pages 171–180. ACM Press, November 30–December 4 1992. ISBN 0-89791-548-8.

J. Nanard and M. Nanard. Using structured types to incorporate knowledge in hypertext. In *Proceedings of the ACM Hypertext '91 Conference, San Antonio, Texas, USA*, Hypertext – Integrative Issues, page 329, 1991.

Theodor H. Nelson. Replacing the printed word: A complete literary system. In S. H. Lavington, editor, *Information Processing 80*, pages 1013–1023. North-Holland Publishing Co., 1980.

A. Pearl. Sun's Link Service: A Protocol for Open Linking. In *Proceedings of the Hypertext '89 Conference on Hypertext, Pittsburgh, Pennsylvania, USA*, pages 137–146, November 1991.

Thomas Phelps and Robert Wilensky. Robust intra-document locations. In *Proceedings of the Ninth International World Wide Web Conference, Amsterdam, NL*, 2000a.

Thomas A. Phelps and Robert Wilensky. Multivalent annotations. In *Proceedings of the 1997 European Conference on Research and Advanced Technology for Digital Libraries*, 1997.

Thomas A. Phelps and Robert Wilensky. Multivalent documents. *Communications of the ACM*, 43(6):82–90, June 2000b. ISSN 0001-0782.

A. Pikrakis, T. Bitsikas, S. Sfakianakis, D. De Roure, W. Hall, S. Reich, and G. Hill. MEMOIR: Software agents for finding similar users by trails. In Hyacinth S. Nwana and Divine T. Ndumu, editors, *Proceedings of the Third International Conference on the Practical Application of Intelligent Agents and Multi-Agent Systems*, pages 453–466, March 1998.

Morgan N. Price, Gene Golovchinsky, and Bill N. Schilit. Linking by inking: Trailblazing in a paper-like hypertext. In Kaj Grønbæk, Elli Mylonas, and III Frank M. Shipman, editors, *Proceedings of the ACM Hypertext '98 Conference, Pittsburgh, USA*, pages 30–39. ACM Press, June 20–24 1998a. ISBN 0-89791-972-6.

Morgan N. Price, Bill N. Schilit, and Gene Golovchinsky. XLibris: The active reading machine. In *Proceedings of CHI '98 Human Factors in Computing Systems, Los Angeles, California, USA*, volume 2 of *Demonstrations: Dynamic Documents*, pages 22–23, 1998b.

Laurent Robert and Eric Lecolinet. Browsing hyperdocuments with multiple focus+context views. In *Proceedings of the ACM Hypertext '98 Conference, Pittsburgh, USA*, Short Papers: Advanced Browsing Interfaces, pages 293–294, 1998.

Bill N. Schilit, Gene Golovchinsky, and Morgan N. Price. Beyond paper: Supporting active reading with free form digital ink annotations. In *Proceedings of CHI '98 Human Factors in Computing Systems, Los Angeles, California, USA*, pages 249–256. ACM Press, April 18–23 1998. ISBN 0-201-30987-4.

Frank Shipman, Haowei Hsieh, and Robert Airhart. Analytic workspaces: Supporting the emergence of interpretation in the visual knowledge builder. http://www.csdl.tamu.edu/ shipman/vkb/vkb.html, 1999.

Simon Buckingham Shum, Enrico Motta, and John Domingue. Representing scholarly claims in internet digital libraries: A knowledge modelling approach. In S. Abiteboul and A.-M. Vercoustre, editors, *Proceedings of the 1999 European Conference on Research and Advanced Technology for Digital Libraries*, number 1696 in Lecture Notes in Computer Science, LNCS. Springer-Verlag, September 1999.

John B. Smith, Stephen P. Weiss, and Gordon J. Ferguson. A hypertext writing environment and its cognitive basis. In *Proceedings of the Hypertext '87 Workshop, University of North Carolina, USA*, November 13-15 1987.

Norbert Streitz, Jörg Haake, Jörg Hannemann, Andreas Lemke, Wolfgang Schuler, Helge Schütt, and Manfred Thüring. SEPIA: a cooperative hypermedia authoring environment. In D. Lucarella, J. Nanard, M. Nanard, and P. Paolini, editors, *Proceedings of the ACM Hypertext '92 Conference, Milano, Italy*, pages 11–22. ACM Press, November 30–December 4 1992. ISBN 0-89791-548-8.

Norbert A. Streitz, J. Hannemann, and M. Thüring. From ideas and arguments to hyperdocuments: travelling through activity spaces. In *Proceedings of the Hypertext '89 Conference on Hypertext, Pittsburgh, Pennsylvania, USA*, pages 343–364. Association for Computing Machinery, 1989.

Harold Thimbleby. Designing user interfaces for problem solving, with application to hypertext and creative writing. *AI and Society*, 8:29–44, 1994.

S. Toulmin. *The uses of argument*. Cambridge University Press, 1959.

Randall H. Trigg. *A Networked-based Approach to Text Handling for the On-line Scientific Community*. PhD thesis, University of Maryland, 1983.

C. D. Whittington. Annotation systems: A review. ftp://ftp.cs.strath.ac.uk/research-reports/rr-96-202.ps.Z, 1996.

N. Yankelovich, B. J. Haan, N. Meyrowitz, and S. M. Drucker. Intermedia: The Concept and the Construction of a Seamless Information Environment. *IEEE Computer*, 21 (1):81–96, 1988.

Ka-Ping Yee. CritLink: Better hyperlinks for the WWW. http://crit.org/ ping/ht98.html, 1998.