Chapter 8

A Framework for Associative Writing in the Web

Associative Writing is not well supported in the current Web environment, requiring many different challenges to be addressed. The preceding chapters have investigated how contributions could be made towards addressing five core challenges, and as it will be subsequently shown, informed the design and implementation of a framework for supporting the Associative Writing process in the Web docuverse. This chapter begins by summarising core challenges identified, and then describes the proposed approach embodied by the Associative Writing Framework using a simple scenario to drive a tour of the framework features. A more detailed description of the implementation is provided before discussing how the framework responds to each of the challenges.

8.1 Recap of Challenges

The investigative work reported over the course of the preceding chapters has highlighted a number of important challenges in supporting Associative Writing in the World-Wide Web, which are summarised here:

8.1.1 The "Lost in Hyperspace" Problem (Chapter 2)

The "lost in hyperspace" problem refers to "the tendency to lose one's sense of location and direction in a nonlinear document" (Conklin, 1987). This 'problem' has attracted much controversy, with many researchers calling for a restriction of the role of links in documents. This work, however, has taken Bernstein's view that "the rich environment and sense of freedom" (Bernstein, 1991) is what has made hypertext (and Associative Writing) desirable in the first place. The challenge therefore, is to avoid disorientation arising from “muddled writing” (Bernstein, 1991) which may itself stem from ‘bad’ system design (Thimbleby et al., 1997).

8.1.2 Deep Linking (Chapter 2)

Recent court cases ruled in favour of organisations which claim that ‘deep’ (associative) links violate the copyright over their material, and allow users to bypass the advertisements on their home pages. Microsoft also withdrew “Smart Tags” (similar to Microcosm’s generic links) from the Internet Explorer browser after legal and moral issues surrounding the modification of existing Web content were voiced (Hughes and Carr, 2002). If such decisions reflect a growing trend towards restricting Web linking, this obviously has significant implications for this work. The challenge therefore, is to demonstrate the advantages of “deep linking” in the context of the Associative Writing process.

8.1.3 Limited Web Hypertext Model (Chapter 4)

The Web’s model of the basic building blocks for integrated hypertexts is restrictive; only uni-directional, binary links can be created, which must be embedded in the documents ‘owned’ by the writer. Work on integrating the Web with open hypermedia systems (Anderson, 1997) has shown that open hypertext technology can be used to enrich the Web’s hypertext model. Alternatively, the XML Linking standards bring advanced hypertext capabilities to the core of the Web, and must be universally supported (whereas open the success of hypertext systems for the Web depended on users actively seeking out and using them). The challenge is therefore to enrich the Web’s hypertext model using open hypertext and/or XLink technology to better support the publication of integrated hypertexts.

8.1.4 Link Integrity (Chapters 4 and 7)

The chaotic nature of the Web (documents are edited, moved, or deleted in an adhoc manner) frequently causes links to ‘break’. A number of Web link management strategies (Ashman, 2000) have been examined, including those embodied by numerous hypertext and annotation systems. The challenge therefore is to leverage these strategies in preventing integrated hypertexts from becoming permanently disconnected from the existing context on which they build.

8.1.5 Supporting the Writing Process (Chapters 6 and 7)

Popular and widespread tools for writing new Web documents (for example, Microsoft's *Frontpage*), typically take the form of 'Web-enabled' word processors. Through the correlation of a number of established cognitive models of writing, Chapter 6 concluded that the activities involved in the Associative Writing process are not adequately supported by such tools. The challenge therefore is to help the user carry out the various activities involved in Associative Writing. A subsequent review of hypertext tools developed specifically to support writing tasks concluded that the challenge should focus on the context building and integrated writing activities of Associative Writing. This focus was explored through an investigation of hypertext, Web, and Semantic Web annotation technologies as approaches for integrating the writer within the docuverse during the context building activity.

8.2 Proposed Approach: Features of an Associative Writing Framework

The purpose of this section is to discuss the proposed responses to the five core challenges listed above that should be embodied by the implementation of a bespoke Associative Writing Framework.

• In response to the first challenge, the *'lost in hyperspace' problem*, the Associative Writing Framework should not restrict the writer's linking capabilities, as suggested by those who believe the problem to be a significant one (De Young, 1990; Lynch and Horton, 1997; Khan and Locatis, 1998), in order to provide "the rich environment and sense of freedom which made hypertext desirable in the first place" (Bernstein, 1991). Therefore, in order to address the potential disorientation arising from unrestricted linking, the framework should include orientation aids, such as overview maps *at the system level*, in line with Thimbleby's "engineering approach" (Thimbleby et al., 1997).

• In response to the *deep linking* challenge, the Associative Writing Framework should not restrict writers to creating only 'shallow links' to the front pages of Web sites; instead the Associative Writing Framework should actively demonstrate, through scenarios and user evaluation, the advantages of the integrated writing vision to writers, readers, and content providers.

• In response to the *limited Web hypertext model* challenge, the Associative Writing Framework should lift the limitations imposed by the Web hypertext model, and effectively increase the range of what can 'be said' by the writer. More specifically, support for *hyperstructures* should be addressed by storing links separately from documents, thus lifting the notion of 'ownership' and allowing any writer to create links across existing material (including anchors in material which the Web does not currently regard as a link source) and to integrate their new contributions with it; support for *navigation* should be addressed by allowing links to be bivisible and bifollowable (in order that newer contributions can be reached from the older material on which it builds), and n-ary and typed, in order to increase the expressive capacity of the writer. In order to facilitate these requirements, and in the light of the limited browser and software support for XLink at the this time, the Associative Writing Framework should use an open hypertext approach, with a hypermedia structure server managing and storing links. Looking ahead to a convergence between Associative Writing technologies and Semantic Web agents (Section 5.3, Section 11.2.1), the hypermedia structure server should be capable of serving/converting its stored structures using a Semantic Web-compatible representation, for example RDF and RDFS, or (eventually) XLink.

• In response to the challenge of *link integrity in the Web*, the Associative Writing Framework should adopt strategies for dealing with the Web's constant and chaotic content change. More specifically, a strategy for addressing the "editing problem" (Davis, 1998) should be adopted since a writer's new contributions will typically undergo many changes before being published in the form of an integrated hypertext, and these changes could compromise internal references. Furthermore, a strategy for addressing the problem of "dangling links" should also be adopted in order to prevent the writer's published contributions subsequently becoming disconnected from the existing material on which they build. Although a complex *notification* strategy, as demonstrated by Hyper-G (Andrews et al., 1995a), may be beyond the scope of this work, *detect and correct* strategies should certainly be incorporated within the environment. The Robust Hyperlink (Phelps and Wilensky, 2000a) and Robust Location (Phelps and Wilensky, 2000b) approaches seem particularly suitable for adoption in this respect, particularly in conjunction with a Semantic Web representation, such as XLink.

• In response to the challenge of *supporting the Associative Writing process*, the Associative Writing Framework should focus on the context building and integrated writing activities, since these activities were identified as least well supported in the review of existing hypertext writing and annotation approaches. Chapter 6 categorised hypertext writing approaches as either *document-based* or *map-based*. In document-based tools, such as WebVise (Grønbæk et al., 1999), and Annotea (Kahan et al., 2001), interaction takes place within one document at a time. Chapter 6 identified that document-based approaches to supporting hypertext writing were best suited to supporting the context building activity of Associative Writing. Chapter 7 demonstrated how document-based annotation tools integrate the writer in the docuverse, allowing existing content to be highlighted/marked and/or annotated with responses. Document-based approaches such as XLibris and TRELLIS also allow writers to capture relationships between annotations, as they are uncovered through careful and creative analysis of existing texts.

Referring back to the 'lost in hyperspace' challenge, it seems that such 'document-at-a-time' interactions may cause "muddled writing" (Bernstein, 1991); early map-based approaches such as NoteCards (Halasz et al., 1987) were developed in response to problems of user disorientation. Map-based systems typically present a visual network containing links and nodes which can be edited and manipulated directly. Whereas related content that has been connected using a document-based tool may be visible in separate browser windows on the writer's desktop, but remains 'disconnected' at the physical boundaries of each window, in map-based approaches related information is always visibly and tangibly connected in the workspace. whether explicitly as in Storyspace (Bernstein, 2002), or implicitly through spatial proximity as in Visual Knowledge Builder (Shipman III et al., 2001). From Thimbleby's engineering perspective (Thimbleby et al., 1997), a map-based approach would be more suited to supporting context building since writer orientation is better supported and hence the writer may be less likely to produce "unwieldy and inexpressive" (Bernstein, 1991) hypertexts.

However, the map-based approaches explored in Chapter 6 were shown to provide limited support for the context building activity, since they 'disconnected' the writer from existing Web content. To express relationships between existing material (and/or annotate existing material with new contributions) the content had to be copied and pasted from source documents into nodes in the workspace and then arranged, with no 'link' retained between the node and the original location of the content in the source document (in contrast, the pre-Web Synthesis Writing Toolkit made these links available to the writer). This causes the writer to constantly shift their "forced divided attention" (schraefel et al., 2002) from reading in the docuverse to representation in the workspace.

Therefore, in order to contribute to work which attempts to address this challenge, the Associative Writing Framework should demonstrate a new approach which combines the advantages of both document- and map-based approaches in a single environment by facilitating map-based interactions (nodes visibly connected, overview facilities) in a document-based context (nodes are directly annotated Web documents, viewed in a browser). The Associative Writing Framework should not attempt to *replace* either approach (indeed, the document- and map-based approaches reviewed in Chapters 6 and 7 offer many features above and beyond the scope of this work) — the purpose of the Associative Writing Framework will be to explore the possibilities of 'blurring' the boundaries between document- and map-based systems in order to evaluate the suitability of this approach in supporting Associative Writing. The Associative Writing Framework should be implemented in such a way as to facilitate the integration of other approaches in the future (for example, an integrated Visual Knowledge Builder component to add support for the structure building activity to the framework).

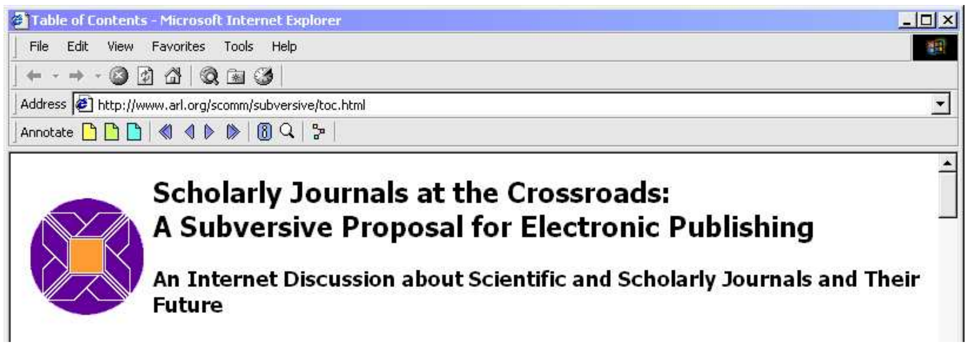


Table of Contents - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Address http://www.arl.org/scomm/subversive/toc.html
Annotate

Scholarly Journals at the Crossroads:
A Subversive Proposal for Electronic Publishing

An Internet Discussion about Scientific and Scholarly Journals and Their Future

FIGURE 8.1: Online version of *Scholarly Journals at the Crossroads: A Subversive Proposal for Electronic Publishing*

8.3 Tour of the Associative Writing Framework Implementation

Following the consideration of the desired features in the previous section, an Associative Writing Framework has been implemented as a toolkit of open hypertext components which integrate with the popular Internet Explorer browser and Microsoft Frontpage editor. Interaction with AWF takes place through three main tools designed to support the context building activity of Associative Writing: *Annotate* allows writers to highlight content in existing Web documents relevant to the writing task (optionally adding comments, interpretations, abstractions *etc.* and semantics to the marked content); *Relate* supports writers in capturing (and optionally adding semantics to) connections between marked content; and the *AWF Server*, as well as performing open hypertext structure storage and management tasks transparent to the user, provides customisable overview maps of the writer's emerging hypertext structures.

Integrated writing is supported through integration with the Microsoft Frontpage editor, enabling writers to build new contributions on existing content and relationships (uncovered and recorded using the *Annotate* and *Relate* tools). Behind the scenes, the framework creates associative links which integrate the new contributions with the existing content: when the new hypertext is subsequently published in the Web, not only will the new material be immediately and visibly connected to the wider context on which it builds, but readers of the older material will be able to follow links to its reinterpretation in the new context.

In order to better demonstrate the features and use of the AWF prototype, the following sections are supplemented by a simple, illustrated scenario[1], based around the book *Scholarly Journals at the Crossroads: A Subversive Proposal for Electronic Publishing* (Okerson and O'Donnel, 1995). This book brings together in print emails which formed the basis of a 1995 electronic discussion about scientific and scholarly journals and their future, initiated by a proposal that scholars should self-archive their work on

[1]The scenario is differentiated from other text by using the `arl.org` logo.

public FTP servers to allow fellow researchers to access their work for free rather than having to pay for subscriptions to paper-based journals. The online version of the book[2] (Figure 8.1) makes an ideal resource for demonstrating the Associative Writing process within AWF, since the site is structured in the same way as the book (each chapter of the printed version reproduced as an unlinked Web page). This organisation hardly gives justice to the many tangled webs of debate underlying the linear pages, which become apparent to the reader with careful analysis of the texts.

The purpose of the scenario is therefore to illustrate descriptions of the AWF implementation and interaction with examples of how a writer might use the framework to capture some of the hypertext structures implicit in the online version of *SJC*, and then use these structures to build a new hypertext essay which is tangibly and visibly linked into the ideas and structures which it describes.

8.3.1 Annotating Web Content

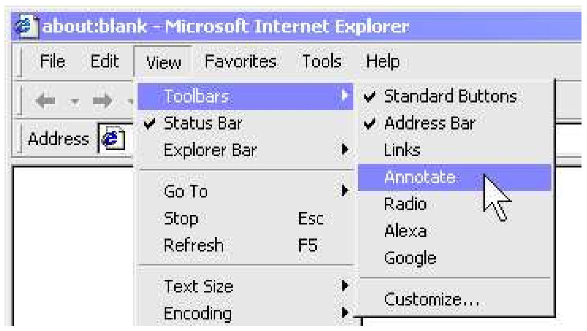
The Annotate tool integrates directly with Internet Explorer adding a dedicated annotation toolbar to the browser interface (Figure 8.2). To annotate document content, the writer selects a text span and assigns it a (user-defined) annotation type by choosing the appropriate type button from the toolbar (the text span is subsequently highlighted with the colour associated with that type). If a document contains several annotations, the 'navigate' buttons of the toolbar allow the writer to quickly move between them.

At the simplest level, annotations in AWF exist as a marker of 'something' significant in existing work (an idea, concept, datum, example, description, experience, claim, theory, suggestion, *etc.*). Writers may also associate some new content with the marked content (a new idea, interpretation, summarisation, abstraction, *etc.*). At a higher level, annotated content becomes a potential link anchor which may later be connected to other annotations.

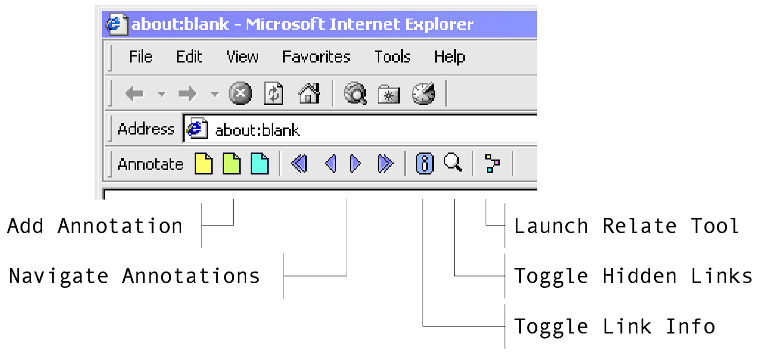
The Annotate tool utilises the RDF templates specified by the Annotea project (Kahan et al., 2001) to describe each annotation (Section 8.4.1 will discuss AWF annotation semantics in more detail), capturing the identity of the creator of the annotation, the date and time that the annotation was created, the type of the annotation, the URL of the Web page which contains the annotation, and an XPointer which describes the exact region of the page that was annotated. This metadata is stored and managed by the *AWF Server*.

SJC Scenario: In Figure 8.3, the writer reads Stevan Harnad's claim that electronic-only journal publishing costs are far less than paper counterparts (Okerson and O'Donnel, 1995, pg 12), and highlights it.

[2] `http://www.arl.org/scomm/subversive/`

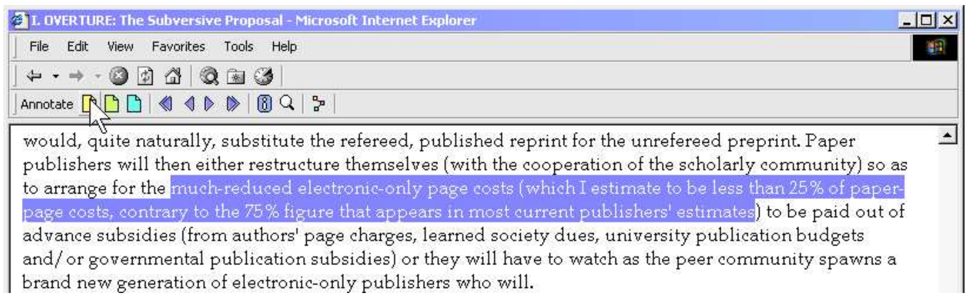


(a)



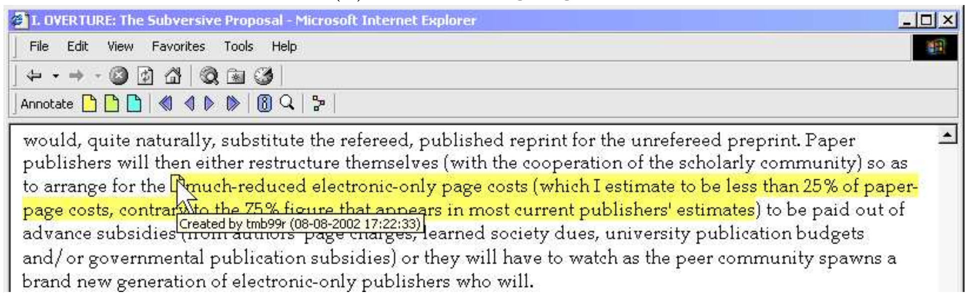
(b)

FIGURE 8.2: Integration of the Annotate component with the Microsoft Internet Explorer browser.


I. OVERTURE: The Subversive Proposal - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Annotate

would, quite naturally, substitute the refereed, published reprint for the unrefereed preprint. Paper publishers will then either restructure themselves (with the cooperation of the scholarly community) so as to arrange for the much-reduced electronic-only page costs (which I estimate to be less than 25% of paper-page costs, contrary to the 75% figure that appears in most current publishers' estimates) to be paid out of advance subsidies (from authors' page charges, learned society dues, university publication budgets and/or governmental publication subsidies) or they will have to watch as the peer community spawns a brand new generation of electronic-only publishers who will.

(a) Content highlighted.


I. OVERTURE: The Subversive Proposal - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Annotate

would, quite naturally, substitute the refereed, published reprint for the unrefereed preprint. Paper publishers will then either restructure themselves (with the cooperation of the scholarly community) so as to arrange for the much-reduced electronic-only page costs (which I estimate to be less than 25% of paper-page costs, contrary to the 75% figure that appears in most current publishers' estimates) to be paid out of advance subsidies (from authors' page charges, learned society dues, university publication budgets and/or governmental publication subsidies) or they will have to watch as the peer community spawns a brand new generation of electronic-only publishers who will.

Created by tmb99r (08-08-2002 17:22:33)

(b) Annotation created, metadata shown in tooltip.

FIGURE 8.3: Annotating a Web page with the *Annotate* tool.

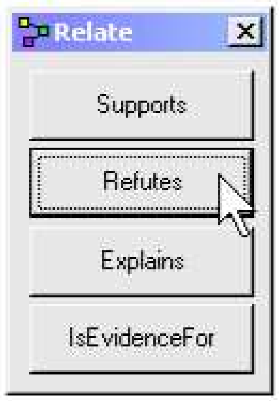


Figure 8.4: The *Relate* tool.

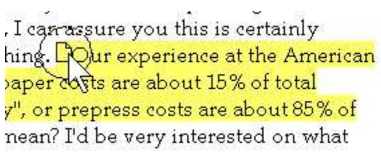
8.3.2 Linking Web Content

In an attempt to bring map-based interactions to a document-based context, the Relate dialog (Figure 8.4) — launched from the Annotate toolbar (Figure 8.2b) — allows writers to create links between annotations *in situ*, without having to switch to a separate workspace, by 'drawing' connections directly onto the desktop (Figure 8.5). Once recorded, these connections are displayed whenever the two endpoints are visible on the desktop, following the movement of windows and scrollbars, and thus keeping related information visibly and tangibly *linked* across the physical boundaries of the document windows.

To create links, the writer first chooses a link type from the Relate dialog (semantics are user-defined — discussed in Section 8.4.1). Each visible annotation on the desktop then becomes a 'hot spot' (Figure 8.5a) — activated when the user mouses over it. The writer then simply 'draws' links between visible annotations by dragging the mouse from one hot spot to another. Link types may have a semantic direction (e.g. A *Refutes* B does not imply that B *Refutes* A), in which case the writer indicates directionality by the direction of drawing (note arrowhead in Figure 8.5b). Bi-directional semantics have no arrowheads.

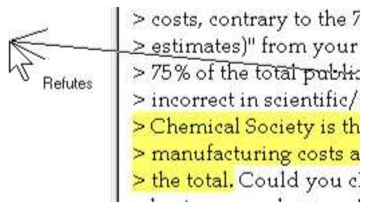
In Figure 8.5, the writer opens a new page of the online SJC *and reads about Lorrin Garson's experiences at the American Chemical Society Journal, including the claim that electronic publishing costs would be much higher than the figure put forward earlier by Stevan Harnad. Realising this is significant, the writer quickly highlights Garson's claim and then connects it to the annotation made earlier (Figure 8.3) using the Relate tool. The writer uses the 'refutes' link type to show the contradictory relationship between the two claims.*

The Relate dialog may also be used to connect Web pages as a whole (rather than parts highlighted using the Annotate tool) — like AWF annotations, Web pages on the desktop also have a 'hot spot' (Figure 8.6).



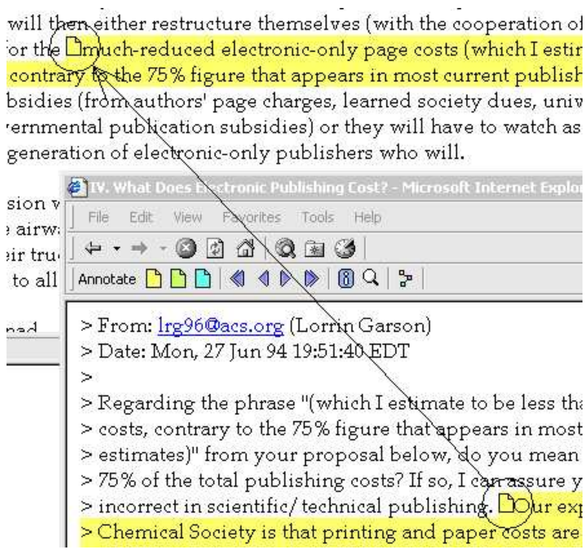
, I can assure you this is certainly
hing. Our experience at the American
paper costs are about 15% of total
y", or prepress costs are about 85% of
nean? I'd be very interested on what

(a) Annotation 'hot spot'.



> costs, contrary to the 7
> estimates)" from your
> 75% of the total publi
> incorrect in scientific/
> Chemical Society is th
> manufacturing costs a
> the total. Could you c

(b) Annotations connected by drawing links.


IV. What Does Electronic Publishing Cost? - Microsoft Internet Explor
File Edit View Favorites Tools Help
Annotate

will then either restructure themselves (with the cooperation of
or the much-reduced electronic-only page costs (which I estir
contrary to the 75% figure that appears in most current publish
bsidies (from authors' page charges, learned society dues, univ
ernmental publication subsidies) or they will have to watch as
generation of electronic-only publishers who will.

sion v
airw
ir tru
to all

> From: lrg96@acs.org (Lorrin Garson)
> Date: Mon, 27 Jun 94 19:51:40 EDT
>
> Regarding the phrase "(which I estimate to be less tha
> costs, contrary to the 75% figure that appears in most
> estimates)" from your proposal below, do you mean
> 75% of the total publishing costs? If so, I can assure y
> incorrect in scientific/ technical publishing. Our exp
> Chemical Society is that printing and paper costs are

(c) A completed link.

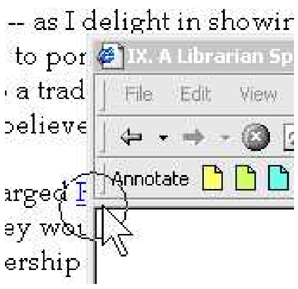
FIGURE 8.5: Capturing connections between annotations with the *Relate* tool.

In Figure 8.6, after having read and annotated Stevan Harnad's further elaboration of his experiences in publishing electronic only journals (Okerson and O'Donnel, 1995, pg 24), the writer realises that Richard Entlich's contribution to the discussion (Okerson and O'Donnel, 1995, pg 74) is supportive of Harnad's experiences (Okerson and O'Donnel, 1995, pg 24), and uses the Relate dialog to capture this connection.

Although the writer 'draws' point-to-point links one at a time, multi-headed links can be expressed by drawing more than one link from the same annotation.

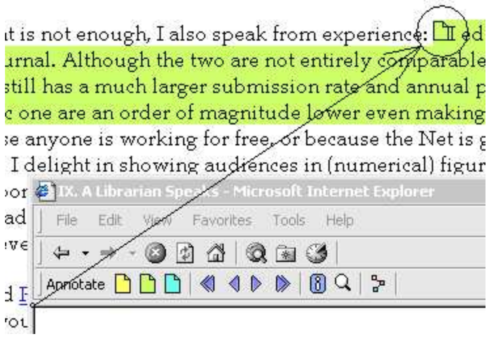
In Figure 8.7, the writer has connected Stevan Harnad's original claim about reduced electronic publishing costs (Okerson and O'Donnel, 1995, pg 12) to the experience Harnad relates to justify this claim, and to Lorrin Garson's further evidence to the contrary (Okerson and O'Donnel, 1995, pg 50).

AWF provides a number of options for displaying links, accessed through the Annotate toolbar (Figure 8.2b). Link metadata, in the form of link type, creator, and creation date, can be shown on screen (Figure 8.8) by selecting the *Toggle Link Info* button. Also, when both endpoints of a link are on the desktop, but obscured from view (for example, an when an AWF annotation is hidden by another document window), the *Toggle Hidden Links* button causes these links to be displayed using a 'ghosted' effect (Figure 8.9).



-- as I delight in showir
to por
a trad
believe
arged
ey wo
ership

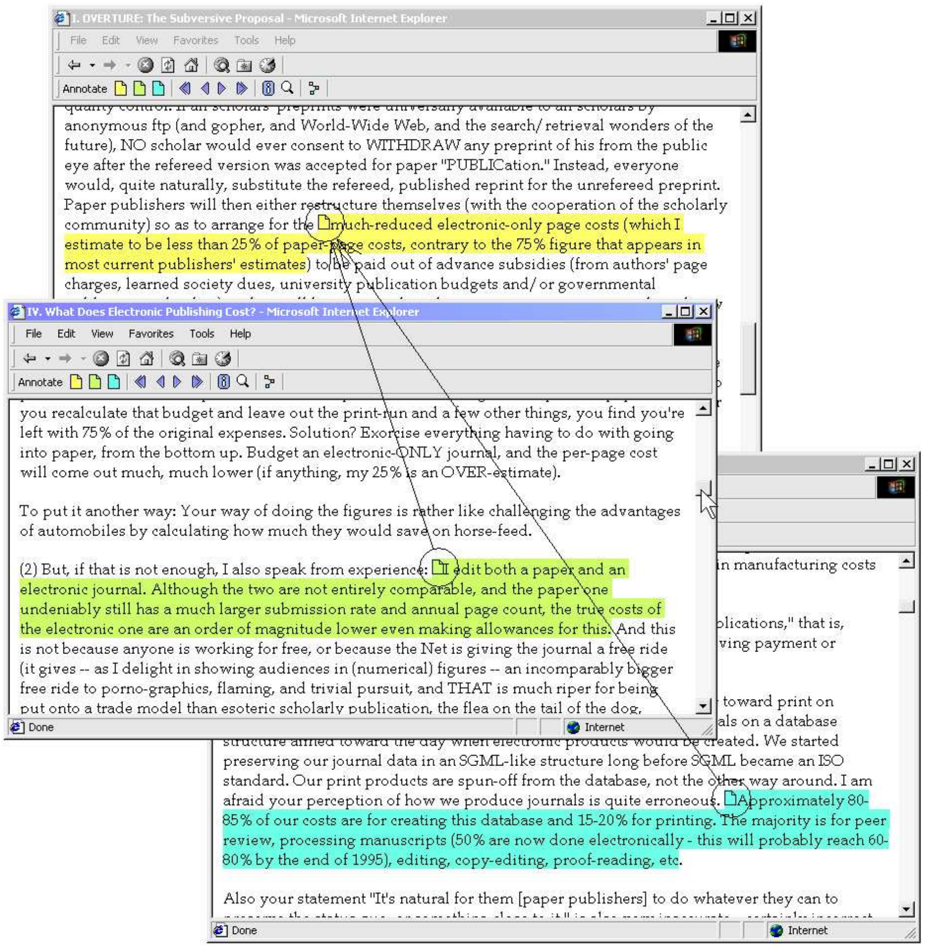
(a) Page 'hot spot'.


IX. A Librarian Speaks - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Annotate

it is not enough, I also speak from experience: I ed
urnal. Although the two are not entirely comparable
still has a much larger submission rate and annual p
c one are an order of magnitude lower even making
se anyone is working for free, or because the Net is g
I delight in showing audiences in (numerical) figur

(b) Page connected to AWF annotation.

FIGURE 8.6: Capturing a connection between a Web page and an AWF annotation.


I. OVERTURE: The Subversive Proposal - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Annotate
IV. What Does Electronic Publishing Cost? - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Annotate
Done
Internet
Done
Internet

anonymous ftp (and gopher, and World-Wide Web, and the search/ retrieval wonders of the future), NO scholar would ever consent to WITHDRAW any preprint of his from the public eye after the refereed version was accepted for paper "PUBLICation." Instead, everyone would, quite naturally, substitute the refereed, published reprint for the unrefereed preprint. Paper publishers will then either restructure themselves (with the cooperation of the scholarly community) so as to arrange for the much-reduced electronic-only page costs (which I estimate to be less than 25% of paper-page costs, contrary to the 75% figure that appears in most current publishers' estimates) to be paid out of advance subsidies (from authors' page charges, learned society dues, university publication budgets and/ or governmental

you recalculate that budget and leave out the print-run and a few other things, you find you're left with 75% of the original expenses. Solution? Exorcise everything having to do with going into paper, from the bottom up. Budget an electronic-ONLY journal, and the per-page cost will come out much, much lower (if anything, my 25% is an OVER-estimate).

To put it another way: Your way of doing the figures is rather like challenging the advantages of automobiles by calculating how much they would save on horse-feed.

(2) But, if that is not enough, I also speak from experience: I edit both a paper and an electronic journal. Although the two are not entirely comparable, and the paper one undeniably still has a much larger submission rate and annual page count, the true costs of the electronic one are an order of magnitude lower even making allowances for this. And this is not because anyone is working for free, or because the Net is giving the journal a free ride (it gives -- as I delight in showing audiences in (numerical) figures -- an incomparably bigger free ride to porno-graphics, flaming, and trivial pursuit, and THAT is much riper for being put onto a trade model than esoteric scholarly publication, the flea on the tail of the dog,

in manufacturing costs

plications," that is,
ving payment or

toward print on
als on a database
structure aimed toward the day when electronic products would be created. We started preserving our journal data in an SGML-like structure long before SGML became an ISO standard. Our print products are spun-off from the database, not the other way around. I am afraid your perception of how we produce journals is quite erroneous. Approximately 80-85% of our costs are for creating this database and 15-20% for printing. The majority is for peer review, processing manuscripts (50% are now done electronically - this will probably reach 60-80% by the end of 1995), editing, copy-editing, proof-reading, etc.

Also your statement "It's natural for them [paper publishers] to do whatever they can to

FIGURE 8.7: Multi-headed links on the desktop.

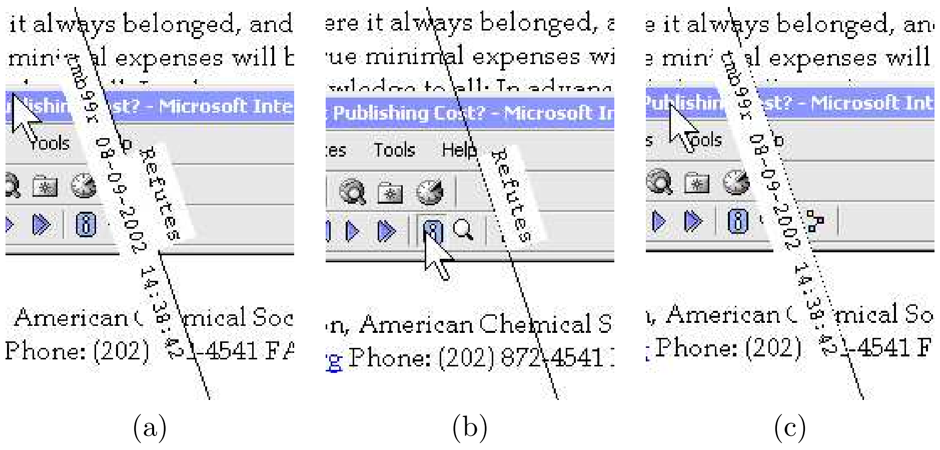


FIGURE 8.8: Toggling metadata on visible links.

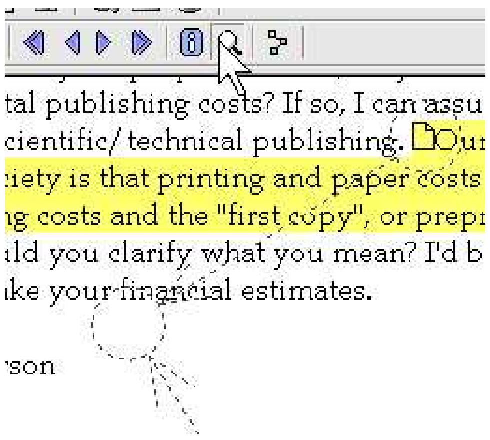
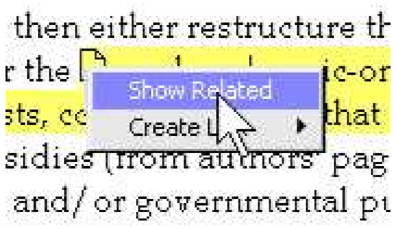


FIGURE 8.9: Displaying obscured links.

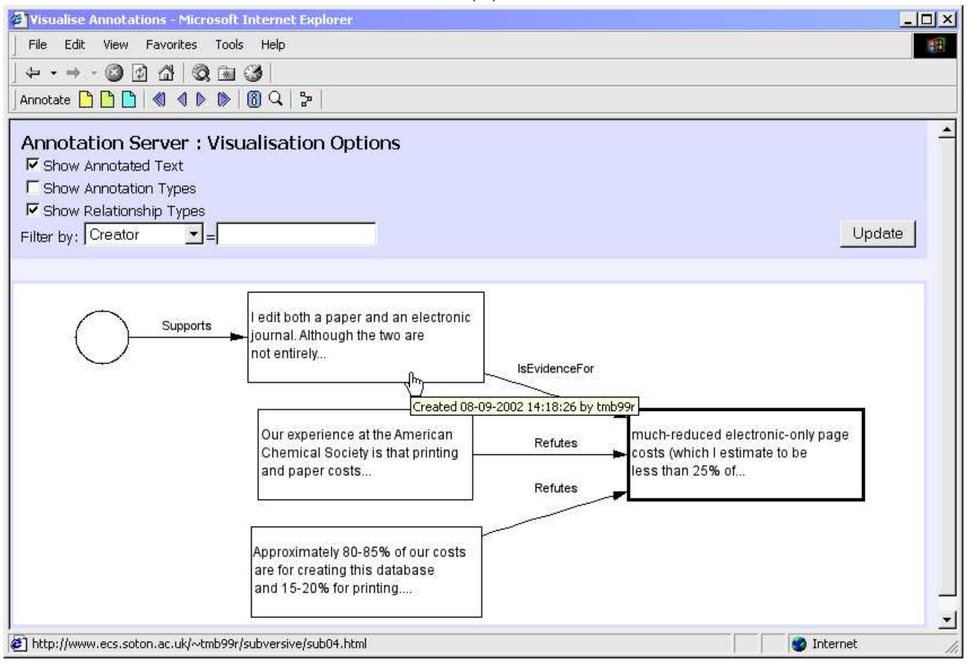
As with AWF annotations, links created through the Relate dialog are described internally using RDF and are managed by the *AWF Server*. Link metadata includes the identity of the link creator, the date and time that the link was created, the link type, and the unique identifiers of the AWF annotations which the link connects. If the link has semantic direction, the metadata also distinguishes the ‘source’ and ‘destination’ annotations.

8.3.3 Visualising Emerging Hypertext

The AWF Server is a distributed, multi-user, open hypermedia structure server, which stores and manages annotations and links created by writers (readers and writers configure their local installation of the AWF framework to store and retrieve structures from at least one AWF server). To demonstrate how the possible advantages of aiding writer orientation provided by map-based approaches could be leveraged in a document-based context, the AWF server also provides customisable overview maps of the writer’s emerging hyperstructures (Figure 8.10). Each annotation has a ‘pop-up’ context menu (Figure 8.10a) — selecting the *Show Related* option from the menu instructs the AWF



(a)


Visualise Annotations - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Annotate
Update
Internet

Annotation Server : Visualisation Options

- [✓] Show Annotated Text
- [ ] Show Annotation Types
- [✓] Show Relationship Types

Filter by: Creator =

Supports

I edit both a paper and an electronic journal. Although the two are not entirely...

IsEvidenceFor

Created 08-09-2002 14:18:26 by tmb99r

Our experience at the American Chemical Society is that printing and paper costs...

Refutes

much-reduced electronic-only page costs (which I estimate to be less than 25% of...

Refutes

Approximately 80-85% of our costs are for creating this database and 15-20% for printing....

http://www.ecs.soton.ac.uk/~tmb99r/subversive/sub04.html

(b)

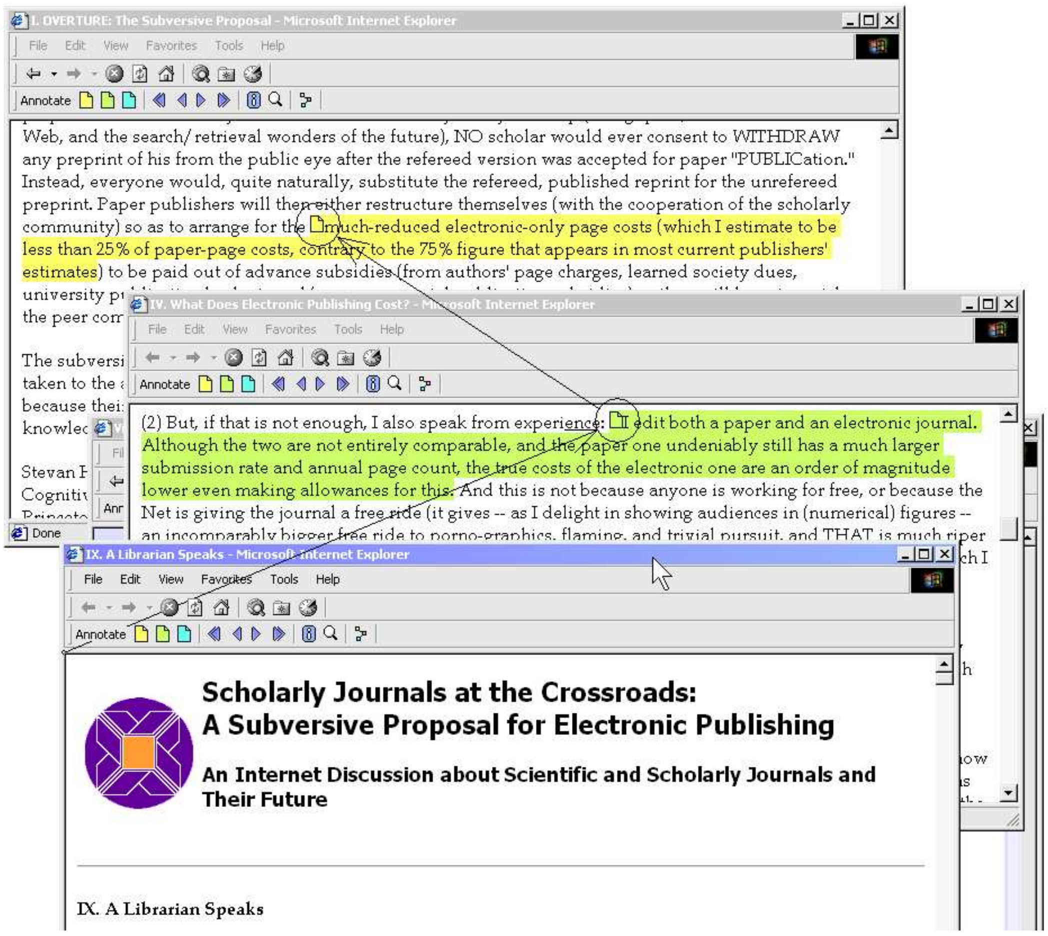
FIGURE 8.10: Overview map provided by AWF Server.

Server to generate an overview map of the hyperstructures surrounding that annotation (Figure 8.10b). The overview map is interactive only in that clicking on a node of the hypertext map leads to the display of the original annotation in the context of the original source text.

The overview map facility is designed to provide the writer with a mechanism for orientation, and for revisiting or re-evaluating the hyperstructures they have created; and for readers to explore structures surrounding the writer's annotations with reference to the original source texts. The map itself [3] can be filtered according to the identities of the writers using the server (for example, render only structures created a specified writer), the creation date of structures (for example, render only annotations created in the last week), and Web URL (for example, render only the hyperstructures surrounding a specified document).

In Figure 8.10, the writer revisits the hyperstructure created around Stevan Harnad's claim (in the map, the current or 'focus' annotation is highlighted by a bold outline). Annotations are rendered as rectangular symbols containing an extract of the

[3]Rendered using the open graph visualisation toolkit *GraphViz* (Gansner and North, 2000)


I. OVERTURE: The Subversive Proposal - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Annotate
Done
IV. What Does Electronic Publishing Cost? - Microsoft Internet Explorer
IX. A Librarian Speaks - Microsoft Internet Explorer

Web, and the search/ retrieval wonders of the future), NO scholar would ever consent to WITHDRAW any preprint of his from the public eye after the refereed version was accepted for paper "PUBLICation." Instead, everyone would, quite naturally, substitute the refereed, published reprint for the unrefereed preprint. Paper publishers will then either restructure themselves (with the cooperation of the scholarly community) so as to arrange for the much-reduced electronic-only page costs (which I estimate to be less than 25% of paper-page costs, contrary to the 75% figure that appears in most current publishers' estimates) to be paid out of advance subsidies (from authors' page charges, learned society dues, university p

(2) But, if that is not enough, I also speak from experience: I edit both a paper and an electronic journal. Although the two are not entirely comparable, and the paper one undeniably still has a much larger submission rate and annual page count, the true costs of the electronic one are an order of magnitude lower even making allowances for this. And this is not because anyone is working for free, or because the Net is giving the journal a free ride (it gives -- as I delight in showing audiences in (numerical) figures -- an incomparably bigger free ride to porno-graphics, flaming, and trivial pursuit, and THAT is much riper

Scholarly Journals at the Crossroads:
A Subversive Proposal for Electronic Publishing

An Internet Discussion about Scientific and Scholarly Journals and Their Future

IX. A Librarian Speaks

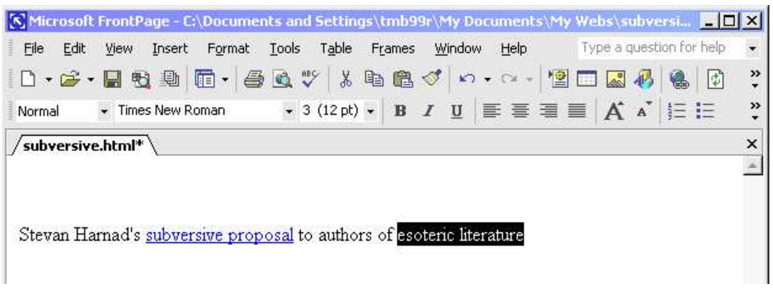
Figure 8.11: Revisiting source documents through AWF Server overview map.

annotated text, with links rendered as (possibly directional) arrows. Circular symbols represent Web pages — in this case, Richard Entlich's contribution to the discussion (Okerson and O'Donnel, 1995, pg 74) (Figure 8.6). In Figure 8.11, the author has clicked on three nodes to reopen the original source documents (pages are automatically scrolled to the annotation position), and the links are automatically redrawn on the desktop.

8.3.4 Integrated Writing in Microsoft Frontpage

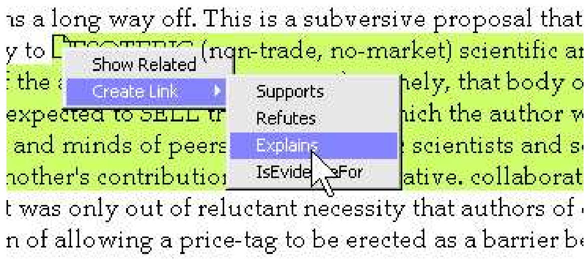
We have seen how the *Annotate* and *Relate* components of the framework support the writer in highlighting significant bits of existing content and capturing links between them. By integrating with the popular Microsoft Frontpage editor, AWF attempts to extend the utility of this tool to support writers in creating a new hypertext which *integrates* their new contributions with the background (or 'global') 'context' that they have identified and connected in the context building activity.

To integrate the new hypertext with existing 'context', the writer first highlights text in the Frontpage editor which will form a link anchor. To link to an AWF annotation, the writer chooses 'Create Link' from the annotation 'pop-up' context menu, and then selects


Microsoft FrontPage - C:\Documents and Settings\tmb99r\My Documents\My Webs\subversi...
File Edit View Insert Format Tools Table Frames Window Help
Type a question for help
Normal
Times New Roman
3 (12 pt)
subversive.html*

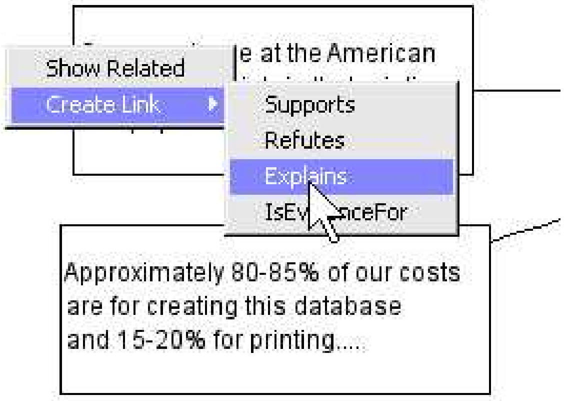
Stevan Harnad's subversive proposal to authors of esoteric literature

(a) Highlight link anchor.



ıs a long way off. This is a subversive proposal that
y to (non-trade, no-market) scientific ar
the ely, that body o
expected to SELL th hich the author w
and minds of peers scientists and s
nother's contributio ative. collaborat
t was only out of reluctant necessity that authors of
n of allowing a price-tag to be erected as a barrier b

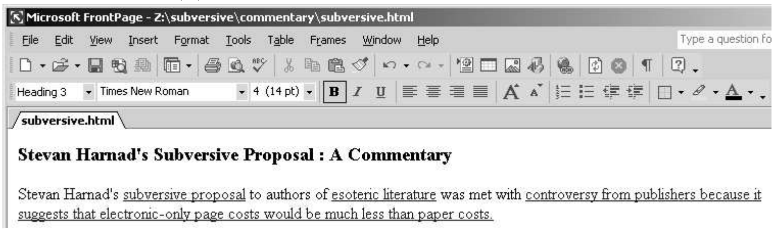
(b) Creating a link to an annotation.



e at the American

Approximately 80-85% of our costs are for creating this database and 15-20% for printing....

(c) Creating a link to a structure.


Microsoft FrontPage - Z:\subversive\commentary\subversive.html
File Edit View Insert Format Tools Table Frames Window Help
Type a question fo
Heading 3
Times New Roman
4 (14 pt)
subversive.html

Stevan Harnad's Subversive Proposal : A Commentary

Stevan Harnad's subversive proposal to authors of esoteric literature was met with controversy from publishers because it suggests that electronic-only page costs would be much less than paper costs.

(d) Associative links created automatically in Frontpage.

FIGURE 8.12: Integrated writing in Microsoft Frontpage.

a link type (Figure 8.12b). To link to a structure, the writer chooses 'Create Link' from the (possibly customised) overview map 'pop-up' context menu (Figure 8.12c). The selected link anchor in Frontpage is then automatically transformed by AWF in two ways:

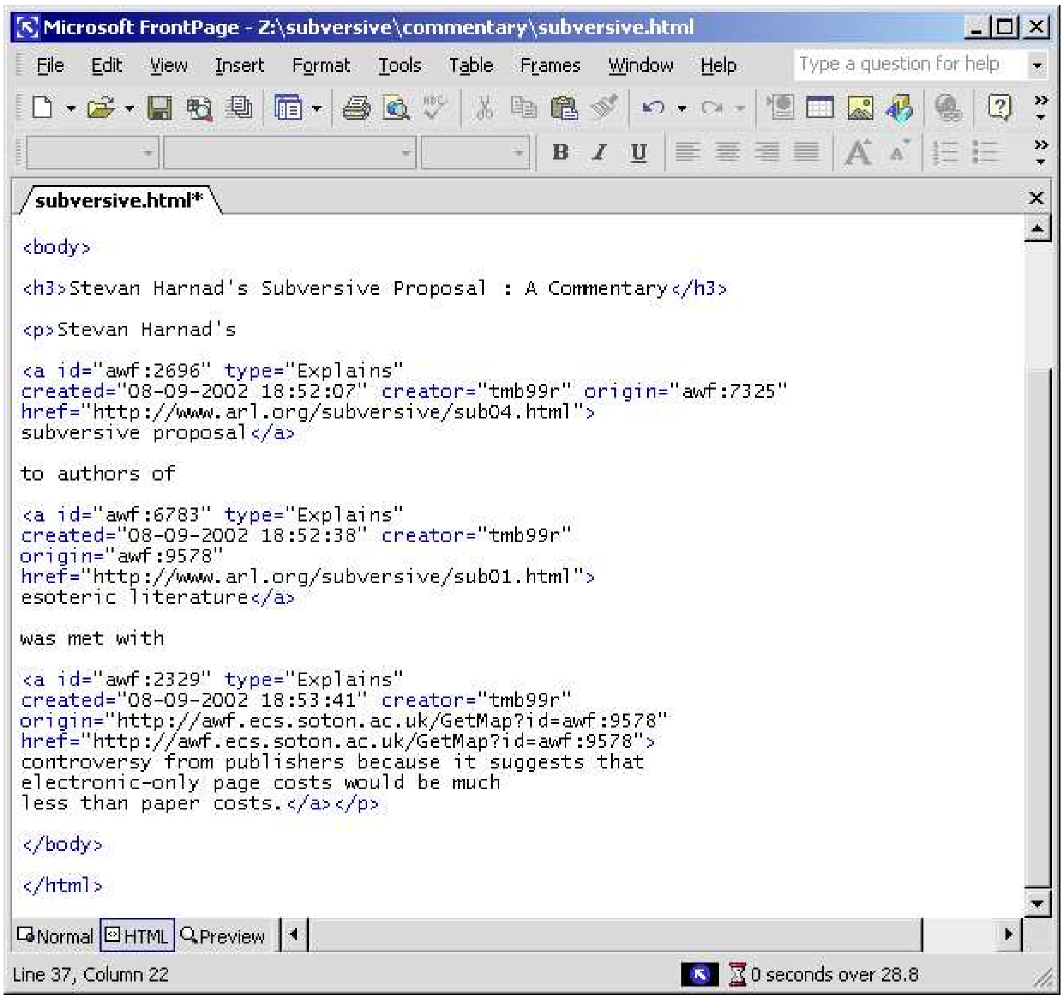
1. The selected text becomes a new AWF annotation, linked to the selected annotation/structure.
2. The selected text becomes an HTML link (Figure 6d), in order to provide some (limited) functionality to users without the AWF system installed (elaborated in more detail in Section 8.3.6).

In Figure 8.12, the writer begins a hypertext essay commenting on the discussions underlying the online SJC. *The writer uses an annotation made earlier on Stevan Harnad's definition of 'esoteric' (Okerson and O'Donnel, 1995, pg 11) to provide an explanation of the term in the context of the essay (Figure 6a,b). The map of the structure surrounding Stevan Harnad's claim that electronic-only journal publishing costs are far less than paper counterparts is used to illustrate the idea that this claim proved controversial amongst other journal publishers.*

Since the writer's new hypertext may not yet be published in the Web (typically being stored locally while still under development), metadata describing the new AWF annotations and links created in this integrated writing process is not immediately registered with the AWF Server — instead, AWF embeds this metadata in the Frontpage document so that it can be later extracted when the new hypertext is published. Figure 8.13 shows a hypertext essay as rendered by the 'HTML view' mode of Frontpage. Note that each HTML link has additional attributes storing the annotation and link information, including the unique id assigned to the annotation, the id of the linked AWF annotation/structure (*origin*), the creator of the link, the creation time and date, and the link type. In the 'Normal' Frontpage view (Figure 8.12d), these details are hidden from the writer.

8.3.5 Publishing Integrated Hypertexts

To publish the new, integrated, hypertext in the Web, the writer uploads the pages to a Web server (this process may be transparently handled by Microsoft Frontpage's 'Publish Web' mechanism), and then manually triggers an automatic AWF 'update' process which crawls the published hypertext and stores extracted annotation and link metadata from each page on the AWF Server. This update process must be performed each time the content (or Web location) of the hypertext is updated so that the information held by the AWF Server is up to date.


Microsoft FrontPage - Z:\subversive\commentary\subversive.html
File Edit View Insert Format Tools Table Frames Window Help
Type a question for help
subversive.html*
Normal HTML Preview
Line 37, Column 22
0 seconds over 28.8

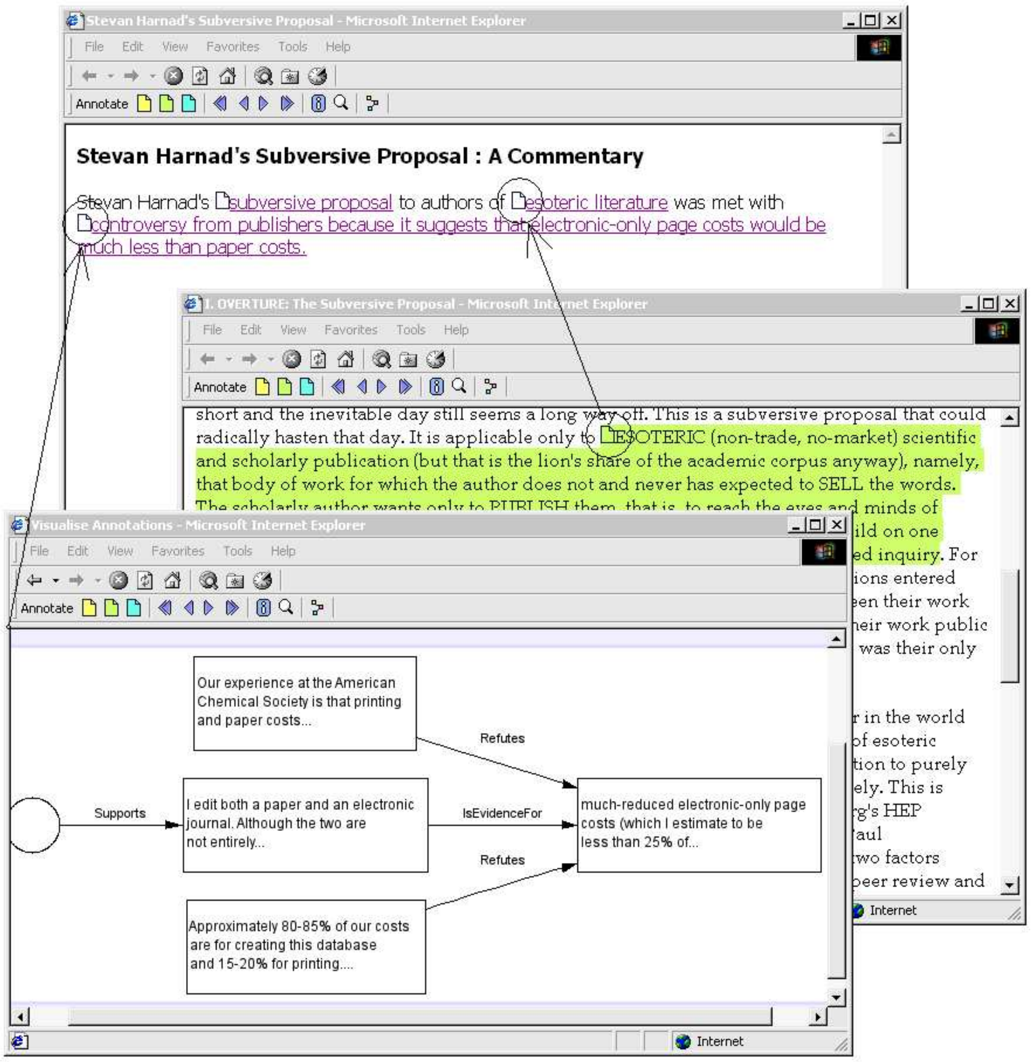
```
<body>

<h3>Stevan Harnad's Subversive Proposal : A Commentary</h3>

<p>Stevan Harnad's

<a id="awf:2696" type="Explains"
created="08-09-2002 18:52:07" creator="tmb99r" origin="awf:7325"
href="http://www.arl.org/subversive/sub04.html">
subversive proposal</a>

to authors of

<a id="awf:6783" type="Explains"
created="08-09-2002 18:52:38" creator="tmb99r"
origin="awf:9578"
href="http://www.arl.org/subversive/sub01.html">
esoteric literature</a>

was met with

<a id="awf:2329" type="Explains"
created="08-09-2002 18:53:41" creator="tmb99r"
origin="http://awf.ecs.soton.ac.uk/GetMap?id=awf:9578"
href="http://awf.ecs.soton.ac.uk/GetMap?id=awf:9578">
controversy from publishers because it suggests that
electronic-only page costs would be much
less than paper costs.</a></p>

</body>

</html>
```

FIGURE 8.13: Essay in 'HTML view' mode showing AWF metadata embedded in HTML links.

In Figure 8.14, a reader views the published hypertext essay, clicking AWF annotations to open both Stevan Harnad's original explanation of the term 'esoteric' and the structure captured by the writer illustrating the controversy surrounding Harnad's claim — the essay visibly and tangibly integrates with the material on which it builds, and may now be annotated and linked by other writers to form part of the 'context' of future contributions.

8.3.6 After Publishing: The Reader Experience

The AWF 'reading' experience currently mirrors that of the 'writing' experience outlined above (in fact, AWF makes no distinction between the roles of 'reader' and 'writer' — indeed, the 'reader' may be a writer engaged in the context building activity — writers see the hyperstructure in the same way that readers do). Firstly, the 'reader' configures their local installation of the AWF framework to retrieve structures from at least one AWF server. As Web pages are subsequently browsed, writer's annotations are retrieved



Stevan Harnad's Subversive Proposal - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Annotate
1. OVERTURE: The Subversive Proposal - Microsoft Internet Explorer
Visualise Annotations - Microsoft Internet Explorer
Refutes
Supports
IsEvidenceFor
Refutes
Internet

Stevan Harnad's Subversive Proposal : A Commentary

Stevan Harnad's subversive proposal to authors of esoteric literature was met with controversy from publishers because it suggests that electronic-only page costs would be much less than paper costs.

short and the inevitable day still seems a long way off. This is a subversive proposal that could radically hasten that day. It is applicable only to ESOTERIC (non-trade, no-market) scientific and scholarly publication (but that is the lion's share of the academic corpus anyway), namely, that body of work for which the author does not and never has expected to SELL the words. The scholarly author wants only to PUBLISH them, that is, to reach the eyes and minds of
ild on one
ed inquiry. For
ions entered
een their work
heir work public
was their only
r in the world
of esoteric
tion to purely
ely. This is
rg's HEP
aul
wo factors
peer review and

Our experience at the American Chemical Society is that printing and paper costs...

I edit both a paper and an electronic journal. Although the two are not entirely...

much-reduced electronic-only page costs (which I estimate to be less than 25% of...

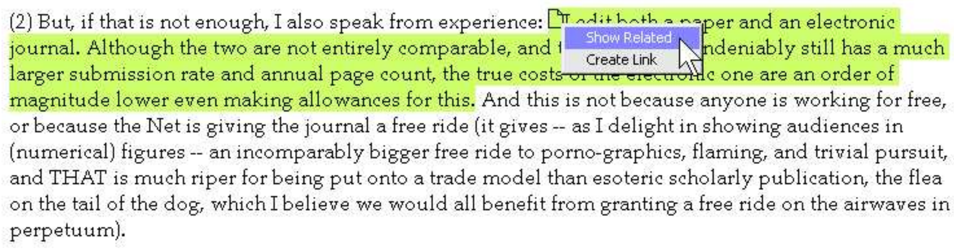
Approximately 80-85% of our costs are for creating this database and 15-20% for printing....

FIGURE 8.14: Exploring the published hypertext essay.

by the Annotate tool and merged into the document. Using the AWF annotation 'pop-up' context menu, the reader can access the structure surrounding the annotation.

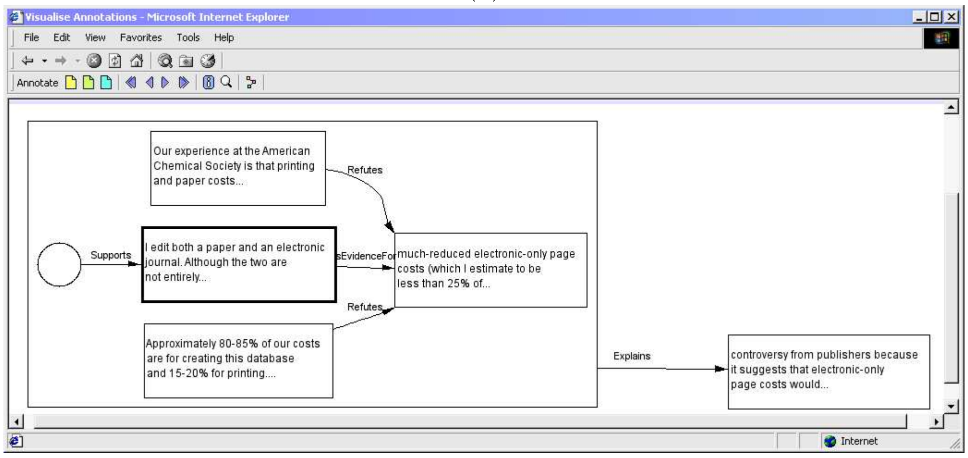
In Figure 8.15, the reader browses the online SJC, *and discovers that a writer has previously annotated Lorrin Garson's claim that electronic publishing costs are high (Okerson and O'Donnel, 1995, pg 23). By selecting the "Show Related" option from the annotation menu (Figure 8.15a), the reader can explore the writer's interpretation of the related ideas surrounding this claim (Figure 8.15b). In the overview map, the reader can see that the writer has published some work which builds on this structure (node on far right of map), and by clicking on this node can reach the writer's own contributions.*

Readers without a local AWF installation can still read and explore hypertexts written within the framework, although with restrictions; the aim here was to ensure that integrated hypertexts created within AWF are not totally exclusive to AWF users. This 'minimum functionality' reading environment is equivalent to the 'normal' Web browsing



(2) But, if that is not enough, I also speak from experience: I edit both a paper and an electronic journal. Although the two are not entirely comparable, and t undeniably still has a much larger submission rate and annual page count, the true costs of the electronic one are an order of magnitude lower even making allowances for this. And this is not because anyone is working for free, or because the Net is giving the journal a free ride (it gives -- as I delight in showing audiences in (numerical) figures -- an incomparably bigger free ride to porno-graphics, flaming, and trivial pursuit, and THAT is much riper for being put onto a trade model than esoteric scholarly publication, the flea on the tail of the dog, which I believe we would all benefit from granting a free ride on the airwaves in perpetuum).

(a)


Visualise Annotations - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Annotate
Supports
Refutes
sEvidenceFor
Refutes
Explains
Internet

Our experience at the American Chemical Society is that printing and paper costs...

I edit both a paper and an electronic journal. Although the two are not entirely...

much-reduced electronic-only page costs (which I estimate to be less than 25% of...

Approximately 80-85% of our costs are for creating this database and 15-20% for printing....

controversy from publishers because it suggests that electronic-only page costs would...

(b)

FIGURE 8.15: Reading in AWF.

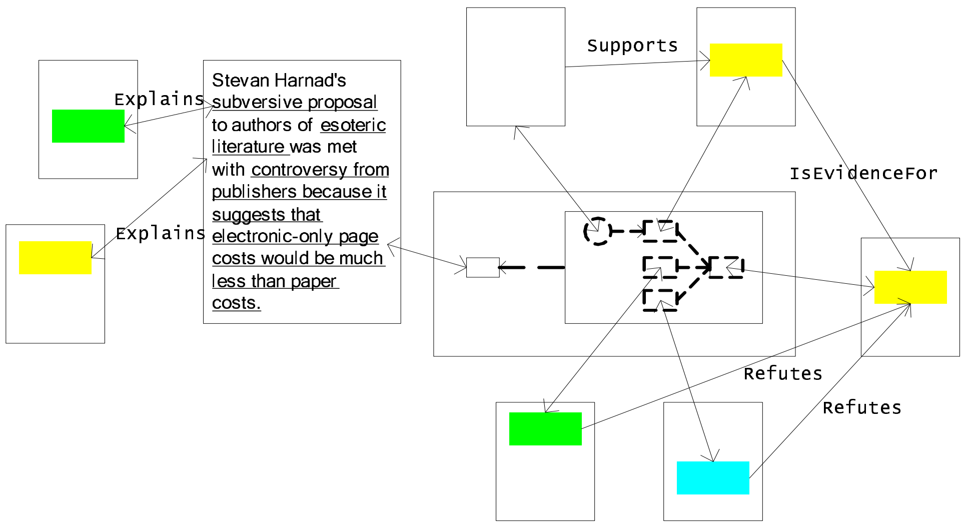
experience and is subsequently subject to the limitations of the Web hypertext model discussed in Chapter 4. Figure 8.16 illustrates the differences in reading experiences with and without the framework installed, using the *SJC* hypertext essay as an example. Note that readers can still take advantage of the services provided by a distributed AWF Server: overview maps with semantic and contextual information may help readers without AWF installed to understand the significance of connected material (since link semantics are not preserved in the Web), and to discover the original context of the annotated content by searching through the source Web page for the partial content shown in the map (since AWF's point-to-point links are replaced by Web point-to-page links).

8.4 Implementation Detail

This section briefly outlines some specific details of the Associative Writing Framework implementation[4], beginning with a brief overview of each tool: *Annotate* is a COM[5] component which implements (amongst others) the `IDeskBand` object interface, and

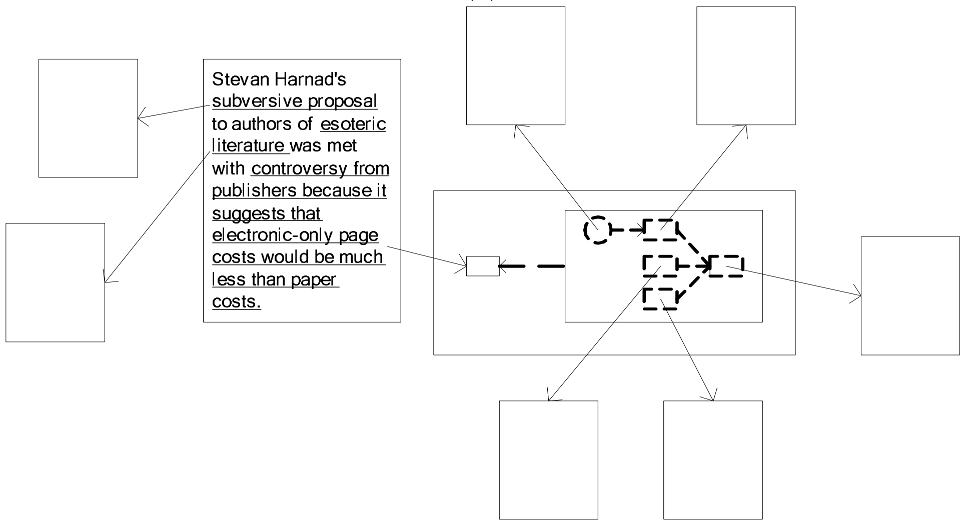
[4]Note that some aspects of the AWF implementation have built on earlier implementation work carried out by the author in the context of the Management Reporting System (Miles-Board et al., 2003a) and Perspectives in Electronic Publishing (Hitchcock, 2002) projects.

[5]Microsoft's Component Object Model — `http://www.microsoft.com/com/`



Stevan Harnad's subversive proposal to authors of esoteric literature was met with controversy from publishers because it suggests that electronic-only page costs would be much less than paper costs.

(a)



Stevan Harnad's subversive proposal to authors of esoteric literature was met with controversy from publishers because it suggests that electronic-only page costs would be much less than paper costs.

(b)

With AWF (a)	Without AWF (b)
New contributions can be reached from existing content.	No links lead to new contributions.
Point-to-point links	Point-to-page links
Link semantics	No semantics

FIGURE 8.16: Reading an integrated hypertext with and without AWF.

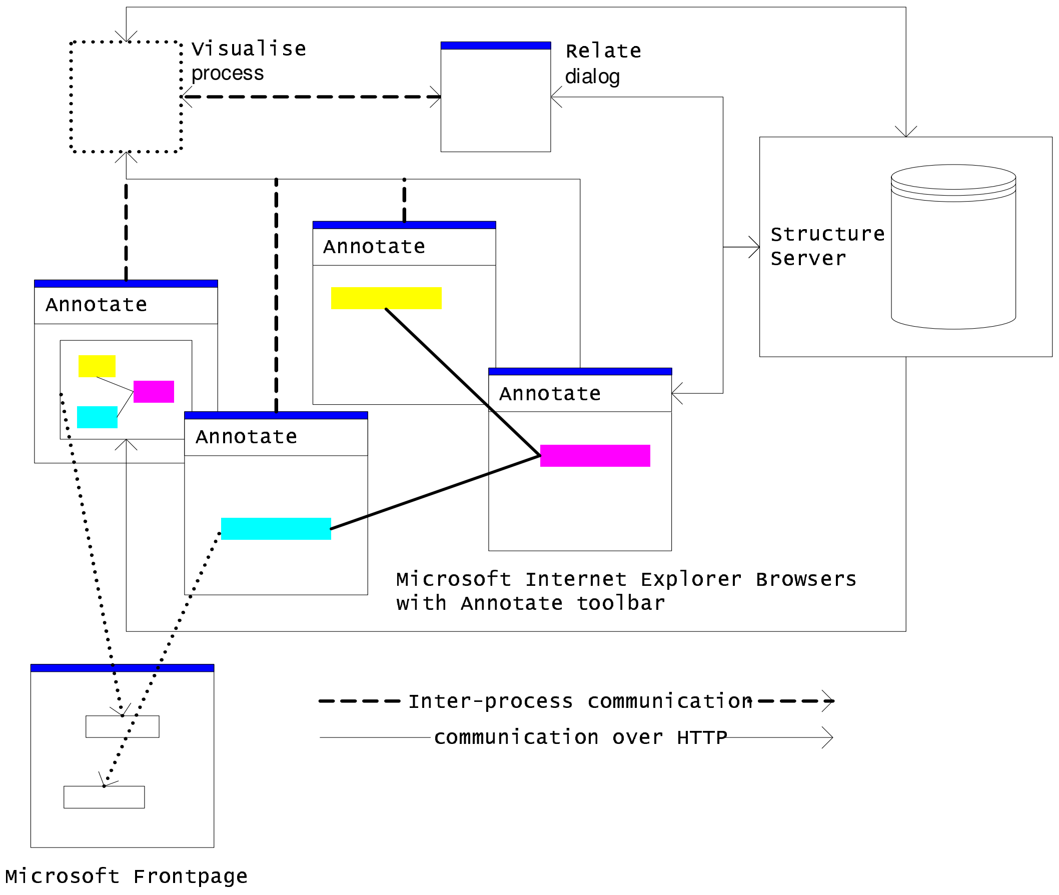


FIGURE 8.17: Architectural overview of the Associative Writing Framework.

therefore can be configured to appear as a toolbar in the Internet Explorer browser; *Relate* is a standard Microsoft Windows application which runs on the user's desktop; the *AWF Server* provides access over HTTP to the information stored in an RDF database through a set of Common Gateway Interface (CGI) scripts. The mechanism for displaying links on the desktop is handled by a further COM process, *Visualise*, which has no user interface. Figure 8.17 attempts to demonstrate the interactions between each of the AWF components.

One Annotate component is instantiated for every Internet Explorer browser window open on the desktop. Each Annotate component can access the Document Object Model (DOM) of the document currently displayed by its host Internet Explorer browser, and can also respond to events fired by that browser. For example, when a `DocumentComplete` event is fired by the browser to signify that a document has been completely loaded and initialised, the Annotate component can request an RDF description of existing annotations for that document from the AWF Server, and then manipulate the DOM to display these annotations (a process which includes extracting information about the annotation from the RDF description, resolving the annotation XPointer to highlight the annotated content, and inserting an annotation icon with metadata tooltip next to the highlighted content).

Each Annotate component also communicates directly with a single instance of the Visualise component. As each annotation is added to the DOM on the fly, the Annotate component calculates its on-screen coordinates, and (provided the annotation is within the visible region of the document) sends this information to the Visualise component — Visualise uses these coordinates to draw links on the desktop. In order to keep the annotation coordinates synchronised with the on-screen display as the user carries out normal interactions with the windowing environment (for example, minimising/maximising Internet Explorer windows, scrolling displayed documents, moving windows, covering/uncovering windows with other application windows) each Annotate component must constantly feed the Visualise component updated coordinate information. By responding to Internet Explorer events such as `OnScroll`, the Annotate component can update on-screen coordinates when the document is scrolled. Responding to other user interactions with the Internet Explorer window is more difficult: Annotate components use a window sub-classing technique in order to detect when the window is moved on screen. By sub-classing the hosting Internet Explorer window, the Annotate component is able to intercept low-level messages from the operating system, such as `WM_WINDOWPOSCHANGED` (which instructs the window to move to a new position) and send updated annotations positions to the Visualise component accordingly.

The Relate component also uses low-level Win32 API programming techniques to allow writers to 'draw' links directly on the desktop. When the writer chooses a link type from the Relate dialog, an up-to-date list of annotation screen positions is immediately requested from the Visualise component, and used to create annotation 'hot spots'. The Relate component then keeps track of the hot spots active when the writer starts and ends the mouse drag which draws the link on screen, in order to record the link endpoints.

When the Visualise component is instantiated, it immediately communicates with an Annotation Server to request an RDF description of all existing relationships. This list is updated as the user uses the Relate component to create new links in a session — metadata describing new links is communicated directly to the Visualise component, and via HTTP (encoded as RDF) to the Annotation Server. By combining the list of screen coordinates for currently visible annotations (indexed according the the unique ID of each annotation and the Internet Explorer window which it is displayed in) with this relationship information, the Visualise component can calculate which links should currently be visible on the desktop and display them directly. As the on-screen annotation coordinates are constantly updated by the user's interactions with the desktop, so to are the visible links constantly redrawn on the screen.

In handling storage and retrieval queries from the Annotate, Relate, and Visualise components, and creating overview maps, the AWF Server is able to respond to a number of HTTP requests:

`StoreAnnotation/StoreRelationship` Stores the RDF encoded annotation or relationship description (posted with the request) in the database.

`GetAnnotations?url=` Returns an RDF document describing each of the annotations which annotate the given document URL.

`GetRelationships` Returns an RDF document describing all links in the database.

`GetMap?id=` Returns a graphical overview map of the structure surrounding the given annotation ID. If no id is specified, returns a series of maps showing all stored structures.

8.4.1 Semantics in AWF

The framework allows semantic information to be attached to both annotations and links, in order to help authors organise information more effectively and to lend context for readers (Bieber et al., 1997): annotation types can be used to convey the significance of the highlighted text; link types convey the relationship between annotations.

Annotations are modelled within the framework as metadata, according to the annotation templates specified by the Annotea project (Kahan et al., 2001). Figure 8.18 lists an RDF description of an AWF annotation. In order to model links in AWF, a further template (which extended the Annotea model) was defined (Figure 8.19).

Annotea provides a number of default annotation types facilitating discussion around documents, including Advice, Change, Comment, Example, Explanation, Question, See Also, and 'none' (i.e.. no type specified). Similarly, in adopting the Annotea representation, AWF provides a set of 'generic' annotation and link types — as opposed to domain or discipline specific types: special-purpose types for a particular genre often do not apply well to other genres (Haas and Grams, 1998b) — more suited to Associative Writing, drawn from argumentation and discourse relations used in ScholOnto (Buckingham-Shum et al., 2000) and TRELLIS (Gil and Ratnakar, 2002a), including Claim, Data, and Theory (annotations), and Supports, Refutes, and IsEvidenceFor (links). The default AWF configuration file points at the RDF resource(s) describing these semantics.

With reference to Trigg's extensive taxonomy of link types (Trigg, 1983), DeRose argued that "no closed taxonomy is likely to be adequate for all future purposes, so coining new types should be possible" (DeRose, 1989). A feature of the Annotea representation is that users can extend or customise the type hierarchy with their own types. In adopting this representation, writers can add new types to AWF by publishing type descriptions and pointing the AWF configuration file at new resource(s). Writers may also use semantics from existing schemas (for example, published by other writers, communities, or organisations), provided they are addressable by a URL. The Annotate toolbar interface and Relate dialog are dynamically adjusted by AWF to reflect the available annotation

```
<r:RDF xmlns:r="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
       xmlns:d="http://purl.org/dc/elements/1.1/"
       xmlns:a="http://www.w3.org/2000/10/annotation-ns#">

<r:Description ID="awf:0001">

  <r:type resource="http://www.w3.org/2000/10/annotation-ns#Annotation"/>
  <r:type resource="http://www.milesboard.org/awf/annotationType#Claim"/>

  <a:annotates r:resource="http://www.arl.org/scomm/subversive/sub01.html"/>
  <a:context>#xpointer(body[0]/p[3])</a:context>
  <d:creator>Tim Miles-Board</d:creator>
  <a:created>2002-10-14T12:10Z</a:created>
  <a:body>Stevan Harnad's claim</a:body>

</r:Description>

</r:RDF>
```

`ID`	unique id of resource described.
`r:type`	type of resource described; in this case an `Annotation` with a user-specified type `Claim` (optional).
`a:annotates`	(Annotea) URL of Web page annotated.
`a:context`	(Annotea) the specific location in the text which the annotation refers to; in this case the 3rd paragraph of the document body.
`d:creator`	(Dublin Core) identity of the writer who created the annotation.
`a:created`	(Annotea) date and time annotation was created.
`a:body`	(Annotea) text entered by writer (optional).

FIGURE 8.18: RDF description of an AWF annotation.

```
<r:RDF xmlns:r="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
       xmlns:d="http://purl.org/dc/elements/1.1/"
       xmlns:a="http://www.w3.org/2000/10/annotation-ns#"
       xmlns:awf="http://www.milesboard.org/awf/awf-ns#">

<r:Description ID="9999">

  <r:type resource="http://www.milesboard.org/awf/awf-ns#Relationship"/>
  <r:type resource="http://www.milesboard.org/awf/awf-ns#Directional"/>
  <r:type resource="http://www.milesboard.org/awf/linkType#Supports"/>

  <d:creator>Tim Miles-Board</d:creator>
  <a:created>2002-10-14T12:10Z</a:created>
  <awf:source r:resource="awf:0001"/>
  <awf:target r:resource="awf:0002"/>

</r:Description>

</r:RDF>
```

`r:type`	type of resource described; in this case a `Relationship` which is `Directional`, with a user-specified type `Supports` (optional).
`awf:source`	(AWF) the ID of the source of the directional link.
`awf:target`	(AWF) the ID of the target of the directional link.

FIGURE 8.19: RDF description of an AWF link.

Challenge	
Lost in hyperspace	●
Deep linking	●
Limited Web hypertext model	●
Link integrity	○
Supporting the writing process	●

●	Addressed
○	Partially addressed

Table 8.1: Response to challenges in current implementation of AWF.

and link types (to distinguish between different types in the AWF interface, writers may assign different colours to each type).

Although the arbitrary creation of nodes and link types gives writers more flexibility to express exact meaning, it also carries the danger of inconsistent type names (Rada, 1990). DeRose advocated allowing users to add to a default link taxonomy, but only by choosing to make each new type a sub-type of one which already exists (DeRose, 1989). This kind of mechanism is possible using the Annotea type hierarchy, but not currently enforced within AWF.

A further consideration is that the rich schemas of node and link types in hypertext systems such as TextNet (the system for which Trigg designed his extensive link taxonomy), NoteCards (Halasz et al., 1987), and Aquanet (Marshall et al., 1991), often overwhelmed users, resulting in types not being assigned at all (Marshall et al., 1994). Spatial hypertext systems, such as VIKI (Marshall et al., 1994) and Visual Knowledge Builder (Shipman III et al., 2001) emphasised an easy, informal interaction in which node and link types did not have to be assigned immediately. Through simple document-based interface design, and adopting the Annotea 'none' type (no semantics assigned), AWF could also facilitate this less formal interaction. Chapter 10 will consider how AWF could be integrated with a spatial hypertext tool, allowing node and link types to be explored and expanded in the map-based workspace after the writer captures content and (initial) links in the document-based AWF.

8.5 Discussion: How AWF Responds to Challenges

The current AWF implementation responds to each of the core challenges formulated by the investigative work carried out over the course of the preceding chapters. Table 8.1 lists the 'status' of each challenge in terms of the implementation work carried out in AWF, and serves to introduce a more detailed description of each response.

AWF enables writers to produce integrated hypertexts without restrictions on linking, in line with Bernstein's view that such restrictions are detrimental to the reader's experience (Bernstein, 1998a). In order to avoid the potential *lost in hypertext* problem arising from "muddled writing" (Bernstein, 1991), AWF takes Thimbleby's "engineering approach" (Thimbleby et al., 1997) by including writer orientation aids in the system design. The document-based interactions of AWF's Annotate and Relate components are supplemented by the map-based overviews produced by the AWF Server, following similar orientation approaches demonstrated by systems such as NoteCards (Halasz et al., 1987). The use of link types and visible hypertext links on the desktop (managed by AWF's Relate and Visualise components) have also been designed to help writers orientate themselves during the context building activity, and subsequently for reader orientation after publication. Guide's replacement links addressed the 'go-to' problem by using "come-tos" instead (Brown, 1992), giving the user the illusion of hierarchy (and thus retaining surrounding context) rather than jumping to a different node in the network. AWF's visible link approach seems to echo this solution — although clicking on an annotation can lead to separate windows being displayed on the desktop, the visible links show exactly how the information in these windows is connected to the original reading context, with annotations clearly highlighting the 'scope' of each link destination.

In response to recent legal issues surrounding *deep linking* in the Web, the scenario presented above has demonstrated some important advantages of the AWF approach (and Associative Writing in general). Key observations include:

- Supporting integrated writing means new hypertexts can link directly to existing context rather than using hundreds or thousands of words to establish that context. In the scenario, the writer links the concepts 'subversive proposal" and "esoteric literature" are directly linked to the exact text in the *SJC* pages describing Stevan Harnad's proposal and explanation. The writer also links to an overview map to illustrate the controversy surrounding the issue of the cost of electronic publishing, rather than having to delineate each opposing viewpoint and evidence.
- Associative Writing demonstrates the reliability of the conceptual foundation being built on, and the innovation and significance of new ideas. In the scenario, the writer links to an overview map to illustrate and elaborate the idea that the Stevan Harnad's proposal proved controversial.
- The bi-directionality of links in AWF's integrated hypertexts means that newer contributions become reachable from older texts. In the scenario, a reader browsing the online *SJC* Web pages would be able to discover subsequent interpretations, analyses, and commentaries about the topic.

AWF successfully addresses the *limited Web hypertext model* challenge by adopting an open hypertext approach. Hyperstructures created in AWF are managed and stored separately to document content, enabling the Web's notion of 'ownership' to be lifted (readers/writers can annotate and link arbitrary documents), and facilitating better support for navigation (links are bi-directional and may be typed) and structure (links can share a source, giving the effect of multi-headed links, although currently not represented internally as such). Although the XML Linking standard was not adopted in this initial implementation (due to limited browser and software support at the time), meaning that users must actively seek out and install AWF, the AWF approach enriches the writer's expressive capabilities and supports the publication of integrated hypertexts in the Web.

The challenge of *link integrity* has been partially addressed by AWF. A *preventative* strategy has been applied to dealing with the "editing problem" as the writer writes and revises the new hypertext using Microsoft Frontpage. Maintaining the location of AWF structures within the hypertext externally would cause problems as the content was continually revised. AWF therefore embeds structure metadata directly in the hypertext. This metadata survives content changes and can be extracted and stored by the AWF Server when the hypertext is subsequently published. Although no *corrective* strategy has been applied to the problem of links to existing content breaking as content beyond the writer's control is edited, moved, or deleted, the open RDF representation used by AWF will facilitate the use of Robust Locations (Phelps and Wilensky, 2000c) and Robust Hyperlinks (Phelps and Wilensky, 2000a) technologies alongside the XPointer (`a:context`) and URL (`a:annotates`) information currently recorded.

The challenge of *supporting the writing process* has been addressed by introducing a novel interface and interaction designed to bridge the advantages of document-based (good for integrated context building) and map-based (good for context building and structure building, but not integrated) approaches. AWF's Annotate component allows writers to mark significant material in existing content. Annotated content then becomes a potential link anchor which may later be connected to other annotations using AWF's Relate component as the writer uncovers relationships and structures.

Related content that has been connected using other document-based tools, such as WebVise (Grønbæk et al., 1999), may be visible in separate browser windows on the writer's desktop, it remains 'disconnected' at the physical boundaries of each window. By contrast, in map-based approaches, such as Storyspace (Bernstein, 2002) and Visual Knowledge Builder (Shipman III et al., 2001), related information is always visibly and tangibly connected in the workspace. To demonstrate how map-based interactions may be applied in a document-based context, AWF's Relate component allows writers to create links between annotations directly, by 'drawing' connections directly onto the desktop. Once captured, these connections are displayed whenever the two endpoints are visible on the desktop, following the movement of windows and scrollbars. This approach also has the advantage that "forced divided attention" (schraefel et al., 2002) between reading in the docuverse and representation in the workspace is reduced. The

AWF Server provides customisable, interactive, overview maps of the emerging hyperstructures created with the Annotate and Relate components; clicking on a node of the map leads to the display of an annotation its source context.

AWF extends the utility of the word-processor based Microsoft Frontpage editor to support the integrated writing activity of Associative Writing. Writers can easily create associative links from new contributions to the annotations and structures captured in the context building activity.

8.6 Summary

The five core challenges surrounding Associative Writing in the Web identified over the course of the preceding chapters have informed the design and implementation of the Associative Writing Framework (AWF), which has been described in detail in this chapter.

The proposed approach of AWF — the bridging of document-based and map-based hypertext environments — has been described, and an example scenario used to illustrate a how a writer could create an integrated hypertext essay using the framework. A more detailed insight into the implementation of the framework has been presented, which provided the necessary background for a discussion of how AWF successfully addresses each of the five challenges.