Hypertext in the Web - a History

Robert Cailliau

European Laboratory for Particle Physics   Web: http://www.cern.ch/
Head, World-Wide Web Office
CH - 1211 Geneve 23, Switzerland
Email: robert.cailliau@cern.ch
Web: http://www.cern.ch/CERN/Divisions/ETT/People/WebCommunications/RobertCailliau/

Helen Ashman

University of Nottingham   Web: http://www.nottingham.ac.uk
School of Computer Science and Information Technology   Web: http://www.cs.nott.ac.uk/
Jubilee Campus
Nottingham NG8 1BB, United Kingdom
Email: hla@cs.nott.ac.uk
Web: http://www.nottingham.ac.uk/~hla/

Abstract: In this short paper, we briefly overview the history of hypertext in the World Wide Web. The Web started with hypertext functions that have disappeared from the early popular browsers, and some are still not present in today's dominant browsers. The hypertext community has proposed ways to bring more sophisticated hypertext into the Web, and the new XML proposals are making many of these into mainstream functions.

Categories and Subject Descriptors: I.7.2 **[Text Processing]**: Document Preparation - *hyptertext/hypermedia*; H.5.1 **[Information Interfaces and Presentation]**: Multimedia Information Systems - *Hypertext navigation and maps*

General Terms: Design, Documentation

Additional Key Words and Phrases: Hypertext, hypermedia, World Wide Web, browsers, XML.

In 1989, Berners-Lee proposed a "distributed hypertext system" for the "management of general information ... at CERN" [Berners-Lee 1989]. The exchange of data in this proposed system was based on an interchange format very similar to an SGML [ISO 1986] application. This interchange format had an explicit representations of hypertext links and was to be used in conjunction with a protocol for addressing and requesting documents over a large network.

This SGML-like format was embodied in the HyperText Markup Language (HTML) which later was changed into a conforming SGML Document Type Definition (DTD). The supporting communications protocol likewise reiterated the importance of hypertext in this system, its name being the "hypertext transfer protocol" (http).

HTML did not in itself add hypertext to SGML. Instead, it provided a means of capturing hypertext links and their behaviour. For example, the HREF attribute of the A tag explicitly instructs the browser to use http (or other protocol) to retrieve a named document. The remaining functionality is within the browser and the server, but not within SGML. While SGML does have some universal rules for interpreting documents, such as how a document type definition is specified, it has no inbuilt rules for how to interpret, act on or display hypertext links - that is up to the individual applications and their underlying markup language to specify.

That original 1989 proposal noted that "generality and portability ... [should be] more important than ... complex extra facilities" [Berners-Lee 1989]. Generality and portability were unquestionably the basis of the early success of the Web. And, in some cases, it dispensed with the complex extra facilities, including much hypertext functionality, although one hypertext feature, typed links, was included in the proposal but did not subsequently survive into the Web today.

The original early browser (NeXTStep version) had a simple but effective hypertext functionality, which included the authoring of links, as distinct from the authoring of documents. The owner of a document can easily create links pointing from their document, but it requires a different technology to let an author create links between arbitrary documents which they do not own [Davis 1995].

In 1993, the authoring of links was largely lost, having not been included in the first release of NCSA's XMosaic. This, and the fact that XMosaic used only a single window, gave the impression that hypertexts on the Web were difficult to author and that such authoring was to be done using separate tools. While there are tools for authoring HTML documents and links from those documents, there is (still) no mainstream browser that supports the authoring of links between arbitrary documents, not just those that the link author owns, although the technology to do so has been demonstrated, with various third-party link servers [Carr 1995], [NCSA 1995], [Röscheisen 1995]. This arbitrary link authoring has been considered a key feature of hypertext systems, ever since their first inception, when Bush described the creation of associative links to record one's path of reasoning through a library of information [Bush 1945].

Of course, not everyone believes that it is desirable to have third-party links (and annotations) available, as the controversy surrounding Third Voice(TM) shows [Krantz 1999]. Third-party linking epitomises the contest between free speech and reputation, or perhaps between graffiti and propaganda. However, there is clearly much benefit in the technology. The Web has no "quality control" office, so anyone can (and frequently will) put up materials of questionable integrity and authenticity. The very vastness of the Web now means that it may be impossible to find evidence to support or deny questionable claims, so properly-used, a third-party linking technology can effect a highly democratic form of quality control for the Web.

There still are problems with this concept. For one thing, a collection of links will gradually "rot" because the web changes and links break [Ashman 2000] [Davis 2000]. Secondly, the Web's size means that any link collection published by an individual, however useful, may remain largely unseen and difficult to find by those who wish to obtain a judgement from a third party. Even inside a very local community (for example the Canton of Geneva, which holds about 300,000 people) there would be far too many opinions posted and links made for even a single page about a single topic: it would be impossible for anyone to digest.

The early Web had an extremely simple approach to hypertext: you could only create links from one phrase (or image) to another document or a place inside another document. Browsers, including the NeXTStep version of 1990, implemented one navigational pair of operations: the "Back" and "Forward" operations, which were defined at the currently active point. A path history was kept so one could go back and forth along the path already trodden.

The NeXTStep version had in addition one other navigation pair that was lost in all subsequent browsers: the "Next" and "Previous" operations. The semantics of "Next" was: "go back to the link in the page you came from, and follow the next link on that page". This enabled the building of a "contents" page, containing a set of consecutive links (perhaps with annotative text) which constituted a path through a collection of pages. A reader would click on the

first link in such a contents page, and then use the browser's "next" button to visit all the other pages in turn, thus following a path laid out by the author. This was enabled by keeping the ordered list of links representing the path within the document chosen as the path document.

It was therefore easy to build such paths and make them available to readers without having to physically modify the set of target pages by placing arrows within these pages. A page could be in several "paths", hence an author could make "tours" through a site for various levels of readers. This was a very useful feature for making presentations and preserving them. Today, readers hold two very different models in their brains: one is the back-forward one, for which they happily use the browser's dedicated buttons, and one is the next-previous one, for which they expect arrow-shaped buttons within the pages. The very fact that no-one has suggested to make next-previous a browser function shows that readers don't always know what they want or might need: it needs familiarity with a function before it is appreciated as a feature.

Finally, the early browsers incorporated a useful aspect that also has gone missing from later mainstream browsers. An author could create a named selection in a document of the form

```
<A NAME="ThisParagraphHere">
Some text here...
</A>
```

Then, if the document's URL is

```
http://www.xyz/SomeDoc.html
```

then the reference

```
http://www.xyz/SomeDoc.html#ThisParagraphHere
```

would show that document with the anchored selection highlighted. Today, at least in Netscape and Internet Explorer, there is no such highlighting. The page is scrolled to position the named anchor at the top of the screen, but because there is no highlighting, the result can be confusing to the reader. It is especially confusing if the named anchor is very close to the end of the page and the anchor cannot be positioned at the top of the screen. Also, since there is no distinction between the target and source anchors, there is a problem with creating arbitrarily overlapping selections.

There was no other semantics to the link and no other semantics to the Web.

Ever since those early prototypes, the hypertext community has pointed out shortcomings in the Web's hypertext capabilities [Bieber 1997a] [Lowe 1999a]. There has even been some dispute over whether the Web is a hypertext system at all [Nürnberg 1999]. This has resulted in the development of more hypertext-enabled Web browsers [Maurer 1996] and encapsulating technologies that can passively interface a standard Web browser with a powerful, autonomous hypertext engine [Anderson 2000], [Davis 1994], [Verbyla 1994]. There is even a progressive analysis of the possible forms of giving applications like Web browsers a more powerful hypertext functionality [Davis 1994].

What this meant was that an entire technology of "hypertext for the Web" had sprung up. However, these were all application-dependent hypertext functions, whose use depended on people actively seeking out and using them. But ignorance, and perhaps a lack of understanding of the benefits, limited the adoption of these hypertext tools and contributed to the ongoing lack of understanding of the benefits, a classic case of absence of the "network effect" [1].

On the other hand, the new XML standard changes Web hypertext functionality at the most fundamental level - in the Web's document specification standards, making the changes universal. To date, XML represents the most radical change in hypertext functionality yet seen in the Web [DeRose 1999]. Its widespread adoption should be assured, not necessarily because it gives "better hypertext" but because XML is seen as the new standard for the Web, allowing vastly different document formats to be defined and exchanged, all within the Web infrastructure. This includes more sophisticated linking - adding "one-to-many" links (links that point to more than one destination), adding "behaviours" to links so that they can respond to the click of a link by executing code, and adding pointers to positions within documents so that URLs can refer to not just NAME tags, but to any tag, and to any position either offset or computed from any tag [DeRose 1999].

However, the semantics of such additions to linking do require that either a set of standards is developed so that browsers can implement them, or that code is always shipped with the documents. The former seems rather improbable (especially with the tendency for developers to create proprietary versions of HTML), and the latter defeats the generally held belief that XML should remain largely declarative instead of procedural.

However, XML's acceptance does not rest on its improved hypertext functionality, but largely on its tailorability to specific applications. The business community, for example, is less impressed by XML's linking than they are by the potential for powerful, data-specific Web search engines, based on the idea that arbitrary data types can be represented and queried by using XML [IBSI 1999]. So, it could be that this most important development in Web hypertext functionality is itself a by-product of a larger issue.

Some might argue that powerful hypertext functionality is not really an issue with Web users in general. Perhaps the A-tag links, bookmarks and the "back" button are the only navigation functions that users need most of the time. However, this is more likely to be a case of making the best of the available technology - if better hypertext functions, such as typed links, third-party links and computed links, were readily available, would they be widely accepted and used? Does the technology restrict our browsing habits, or have we already built the browsers that support our needs?

The changes in browser capabilities already implemented since the early Web suggest otherwise: forms and Javascript are now so widely used that there is no question that there is a need for a much richer set of functions. The real question seems to be: can we find a small, elegant, orthogonal set of functions that will make a large improvement and that may be implemented in an agreed standard way? XML as it stands is but a syntax, and the DTD and the code for "executing" the behaviours associated with tags (e.g. the A tag) must be defined elsewhere.

The new hypertext functions of XML are substantial, genuine improvements that will be widely used. Coming "packaged" with XML means that they are available to a larger audience who might not have looked for them specifically, but are happy to use whatever tools are included. But it is immaterial if "better hypertext" is not chosen for its own sake, as long as it is used by those who adopt XML for other reasons. Hypertext is already very much a part of our everyday working habits, because the point-and-click interface is the almost universal means of navigating around files and applications.

XML departs from the SGML rule of not interpreting hypertext structures, because the new "rules" are in one sense now inbuilt in XML. So XML is no longer purely a subset of SGML, stripped down to make it more efficient in an Internet application. The hypertext functions are now being made universal in the XML standard itself, meaning that hypertext is no longer marginalised to specific applications, but has become a core function which must be universally supported. Of course, it is still necessary for browsers to interpret, act on and display hypertext links, just as it is still necessary for SGML browsers to interpret document type definitions. However, the point is that hypertext is no longer seen as an application-specific feature, but is at last considered to be a fundamental characteristic of electronic documentation.

Footnotes

[1]

The network effect is a form of positive-feedback that demonstrates how success leads to more success, especially in software use. Whitehead describes how the success of the Web arises from this network effect [Whitehead 1999].

References

[Anderson 1999] Kenneth M. Anderson. "Supporting Software Engineering with Open Hypermedia Systems" in ACM Computing Surveys, Symposium on Hypertext and Hypermedia, 1999.

[Ashman 2000] Helen Ashman. Electronic Document Addressing - Dealing with Change, to appear, ACM Computing Surveys, 2000.

[Berners-Lee 1989] Tim J. Berners-Lee. Information Management: A Proposal, in-house technical document, CERN, 1989 (revised 1990 with Robert Cailliau), http://www.w3.org/History/1989/proposal.html<

[Bieber 1997a] Michael Bieber, Fabio Vitali, Helen Ashman, V. Balasubramanian and Harri Oinas-Kukkonen. "Fourth Generation Hypermedia: Some Missing Links for the World Wide Web" in International Journal of Human Computer Studies, 47 (1), 31-65 [Online: http://www.hbuk.co.uk/ap/ijhcs/webusability/], July 1997.

[Bush 1945] Vannevar Bush. "As We May Think" in The Atlantic Monthly, 176(1),101-108, [Online: http://www.isg.sfu.ca/~duchier/misc/vbush/], July 1945.

[Carr 1995] Leslie A. Carr, David C. DeRoure, Wendy Hall, and Gary J. Hill. "The Distributed Link Service: A Tool for Publishers, Authors, and Readers" in Proceedings of the Fourth International World Wide Web Conference, Boston, MA, 647-656, [Online: http://www.staff.ecs.soton.ac.uk/lac/dls/link_service.html], December 1995.

[Davis 1994] Hugh C. Davis, Simon Knight, and Wendy Hall. "Light Hypermedia Link Services: A Study of Third-Party Application Integration" in Proceedings of ACM European Conference on Hypermedia Technologies (ECHT)'94, Edinburgh, Scotland, 41-50, September 1994.

[Davis 1995] Hugh C. Davis. "To Embed or Not to Embed..." in Communications of the ACM (CACM), 38(8), 108-109, August 1995.

[Davis 1999] Hugh C. Davis. "Hypertext Link Integrity" in ACM Computing Surveys, Symposium on Hypertext and Hypermedia, 1999.

[DeRose 1999] Steven J. DeRose. "XML Linking" in ACM Computing Surveys, Symposium on Hypertext and Hypermedia, 1999.

[IBSI 1999] IBSI Final Report. Improving Business Search on the Internet (subproject of the European Union KITE programme), [Online: http://www.decade.be/ibsi/report.htm], July 1999.

[ISO 1986] International Organization for Standardization. 1986. Information Processing-Text and Office Information Systems-Standard Generalized Markup Language. ISO 8879: 1986(E).

[Krantz 1999] Michael Krantz. "Spraypainting the Web" [Online: http://www.infowarfare.com/class_1/99/class1_063099c_j.shtml], 1999.

[Lowe 1999a] David Lowe and Wendy Hall. Hypermedia and the Web: An Engineering Approach. Wiley & Sons, 1999.

[Maurer 1996] Hermann A. Maurer. Hyper-G now Hyperwave : The Next Generation Web Solution, Addison Wesley Longman, ISBN 0≠201≠40346≠3, 1996.

[Nürnberg 1999] Peter J. Nürnberg and Helen Ashman. "What was the Question? Reconciling Open Hypermedia and World Wide Web Research" in Proceedings of ACM Hypertext '99, Darmstadt, Germany, 83-90, February 1999.

[NCSA 1995] National Center for Supercomputing Applications (University of Illinois), The NCSA Mosaic Homepage, http://www.ncsa.uiuc.edu/SDG/Software/Mosaic/<

[Röscheisen 1995] M. Röscheisen, C. Mogensen, and Terry Winograd. "Beyond Browsing: Shared Comments, SOAPS, Trails and On-line Communities" in Proceedings of the Third International World Wide Web Conference, Darmstadt, Germany, [Online: http://www.igd.fhg.de/www/www95/proceedings/papers/88/TR/WWW95.html], 1995.

[Verbyla 1994] Janet Verbyla and Helen Ashman. "A User-Configurable Hypermedia-based Interface via the Functional Model of the Link" in Hypermedia 6(3), 193-208, 1994.

[Whitehead 1999] E. James Whitehead. "Control Choices and Network Effects in Hypertext Systems" in Proceedings of ACM Hypertext '99, Darmstadt, Germany, 75-82, February 1999.