

Some Thoughts on Hyperlinks

Harald Krottmaier

Institute for Information Systems and Computer Media

Inffeldgasse 16c, A-8010 Graz, Austria

hkrott@iicm.edu

Abstract

As already mentioned in many reports, the best hyperlink is the one, you don't have to follow. Therefore authors of hypertext-documents should make it clear to the reader, what is "behind" the link. This is not just information about size of the linked content, but also information about the content itself, such as topic, number of incoming links etc. However, unexperienced authors are not aware of this fact. It would be nice to have tools plugged-in into your favorite web-browsing software, which enhances hyperlinks in some respects.

In this article we are going to explore some of the available tools (such as thumbnail-previewing) and we will discuss in detail a mechanism of enhancing and "ennobling" links. We will discuss different mechanisms on how to attach information.

Keywords: links, annotations to hyperlinks, enhancing hyperlinks

1 Introduction

Even before the term *hypertext* was invented by Ted Nelson in 1965, Vannevar Bush wrote in his famous article "As We May Think" (see [Bush, 1945]) that it is very important to "tie things together". I.e. different users create different relations between documents using some linking technologie. *Memex* was the name of the system based on analog technology.

Although it was foreseen by Bush, that many "trails" will be build by users and that trails will be imported to other systems, it is still not very common, that users add links to information sources. It is even not possible to *export links* from one system and import *just the links* to another system.

Conventional systems support hyperlinks mainly coded in HTML. Therefore links are stored *inside* the document. This may be fine on the first look: they are available to users at their fingertips, no further configuration is needed, the browser just have to parse the HTML-code and render it to the user. However, there are may drawbacks:

- It is not possible to *add links* to a document, where you don't have permissions to modify the document. But this is the usual situation: information-providers allow users to read the document, but not to modify the content of the document. Just take a look at ordinary publishing systems. As a (passive) information consumer your are just allowed to read the content.

Some current encyclopedia-systems, such as Wikipedia, allow users to modify content. However, this is an exception in the publishing world. Ordinary information-systems such as systems for newspapers, or systems serving scientific publications, do not allow users to modify any content.

- Links are unidirectional: In HTML links have a direction. They point from a single source *to* a single destination. It is not easily possible, to get the information on how may documents are pointing to a specific document.

Usage of Google's algorithms ("A document, where many other documents are pointing to it is more important than a document with less links pointing to it...") may make it possible to get

a list of documents which are pointing to a document. However, Google (and all other search engines) have just an index of the “surface Web”, therefore information available in the “deep Web” is not indexed and therefore not available to users of search engines.

Using a feature of the Hypertext Transfer Protocol (HTTP) makes it possible to extract information of the page-referrer using the HTTP-header-information. Unfortunately this powerful feature is not available in most of the systems. This feature must be implemented in **all** the different systems to be useful to users.

- Links are unstable and possibly out of date: as already mentioned, links are unidirectional. If the target of a link disappears, there is no chance to automatically inform all sources of a link (i.e. the authors of the source documents). The situation is similar if the content of the destination changes: There is also no chance to notify the authors of source documents.
- Links are the same to all users: Once links are added by an author of a document, every user will see the same links. To give a short example: imagine two types of readers: a student and a professor. The student should be directed to the basics of a description of a word, while the professor should be guided to the latest research activities on the topic.
- Links have just one destination: In many applications it is necessary to point to more than one destination. Given the example above: A student may also be interested in current research of a specific topic and whats to be supported by the system in this aspect.

It had been shown extensively in the literature that it is much better to *store links outside* of documents in link-databases. Some systems (e.g. Hyperwave, [Hyperwave 2004]) support this paradigm. There are many advantages of this approach. If the reader wants to add some links, it is not necessary to have write access to a document. Since links are stored outside the document in a database, it is easily possible to extract linking information such as what is the target of the link and which other documents are pointing to another document. All disadvantages mentioned above (such as adding links, direction of links, unstable links, uniform links, single destination links) may disappear if links are stored outside the document.

However, there are many different systems around and it is obvious that users will not change their existing systems because of these missing features in HTML. We have to cope with many existing documents which are coded in HTML. In this article we will discuss how to make things better using different types of visualization aids for users.

We are not going to discuss legal aspects of hyperlinking. In this article we just take a look at some technical aspects on how to improve hyperlinking on the web reusing information which is already available to users.

2 Improve Existing Links

A link in HTML has two anchors (a source and a destination anchor) and a direction. The destination may be any resource accessible via the Hypertext Transfer Protocol (HTTP). The destination may also be a part of a resource. This part may be addressed using the fragment-identifier in HTML (`href="destinationResource#theNameOfTheFragment"`). However, the author of the destination must include such a name of a fragment in advance. The following is an example of a destination resource with a named fragment.

```
...some text. Jump directly to <a name="theNameOfTheFragment">the fragment</a>...
```

If headings of a document should be addressed, it is possible to add just an id-attribute to the headings.

```
<h1 id="#theNameOfTheFragment">This is the Title</h1>
```

But there are more attributes available in the specification, unfortunately they are infrequently used. We will now list and shortly describe the attributes for the anchor-tag as listed in the recommendation:

href: as already mentioned this is the target of the link.

title: this attribute is probably the one which is mostly used. A short description of the target of the link is given by authors.

name: this names the current anchor so that it may become the destination of another link. Names must be unique inside a document and they share the same namespace as the id-attribute.

hreflang: this specifies the language of the resource defined in the href-attribute. It may only be used when href is specified.

type, charset: this specifies the content type, and the charset of the target address. Browsers may hide links to specific types if they do not support this type of content.

rel: the type of link; please refer to the following paragraphs where the link-element is explained.

rev: this attribute describes the relation of the target resource to this document.

accesskey: some browsers support this attribute to make keyboard-navigation possible.

shape, coords: used in the context of client-side image maps.

tabindex: position in tabbing order

onfocus, onblur: if the element is in focus or lose the focus.

target: this controls the behavior of the browser where to display the target of the link. Targets may be named. Special names such as "**_blank**" are available.

As you can see, there are a lot of attributes available to specify in detail the target of the link. Unfortunately there are no separate attributes to describe the size of the target, or even more detailed information on the content of the target. See the reference ([W3C:HTML40]) for a more information on the mentioned attributes.

While anchors are valid just in the body of an HTML-document, there is a different type of link using the `link`-element in the header of a document. This element may express another relation of the document to others. It is often used to add additional relations such as: "what document is the next or the previous document:". Some browsers have even implemented a prefetching mechanism for these types of links. E.g. Firefox ([Firefox, 2005]), a browser from the Mozilla group, supports prefetching of link-elements in the header of a HTML-document with appropriate `rel`-attributes (`prefetch` and `next`), the Link-HTTP-header, and the link-attribute in the HTML-Meta-element.

To give a short example of these types of link:

```
<head>
  <title>The Title of this Document</title>
  <link rel="prev" href="previousResource"/>
  <link rel="next" href="nextResource"/>
</head>
...
```

User agents (i.e. browsers) may add additional navigation using these `link`-elements to help users in browsing the document. In the current HTML-recommendation ([W3C:HTML40]) there are several different types specified and user agents should make use of these elements:

Alternate: another substitute version of the current document. In addition to the language-attribute (`lang`) it is therefore possible to create multilingual versions of a document. In addition to the media-attribute it implies a version designed for a different medium.

Stylesheet: this is probably the most common attribute used to link to a external stylesheet for rendering the document.

Start, Next, Previous, and Contents refer to related documents in a collection; there are also other attributes such as `index`, `glossary`, `copyright`, `chapter`, `section`, `subsection`, `appendix`, and `help` which also refer to the related documents in a collection of documents.

Bookmark: A bookmark is defined as a key entry to a document. It is possible to create different bookmarks for different user-groups using different title attributes.

As you can see, several attributes are already foreseen by the committees. Several applications such as e-learning applications need different extended attributes which should be handled by the browser. However, we are not going into detail of this special kind of webapplication. In the next section we will point to some additional attributes which are of interest to users in many web-based-application.

3 Additional Desirable Attributes

There are some additional attributes which should be provided by the hypertext-authoring systems as attributes to the anchor-object. In the following we are going to list some of “nice-to-have” attributes and how they may be realized. We will also show some screenshots of popular applications.

thumbnail-preview: A small thumbnail of the target resource is displayed inside the browser, if the mouse is right over the link. Thumbnails may either be created using available freeware-tools, or they may be retrieved using already established servers. [Alexa, 2005] and [Thumbshots, 2005] are two of such service providers. A screenshot of a plugin using this kind of preview is displayed in Figure 1.

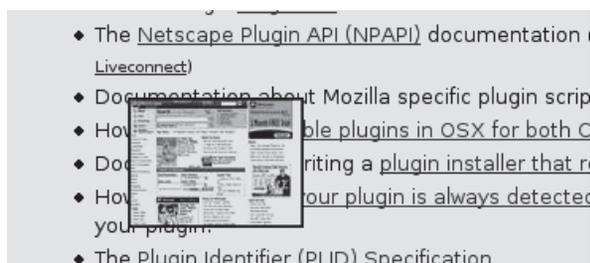


Figure 1: Thumbnail-Plugin in Firefox

users of the target: Information about users (how many users, what resources are of interest to that users, etc.) of the target-resource should also be displayed. An example is displayed in Figure 2.

information of the target: Information such as what is the size of the target, how many words are there, etc. are also important to users. The document-type of the target resource is already supported by the anchor-element. However, additional information must be coded in the title-attribute. Additional information about the target-website may also be included in the preview. A very nice example of additional information is available via [A9, 2005] (see Figure 2).

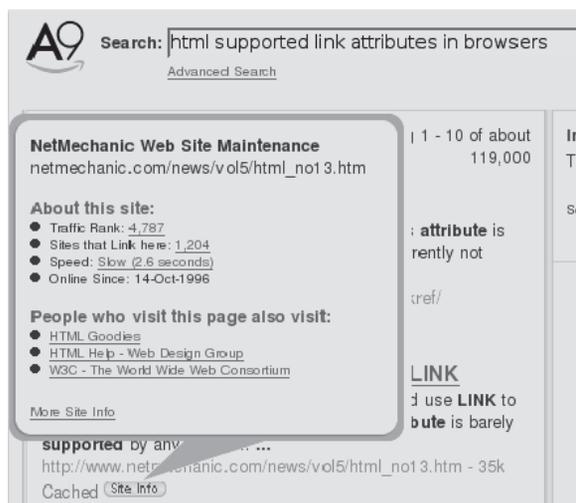


Figure 2: More Information about the Target: Statistical and Related Information about Users of that Site

Additionally information about the categorie of the target, rating of content, an excerpt of the target, number of incoming and outgoing links, etc. should also be available to users.

metadata related to the source-document: If the target was modified *after* the sourcedocument was written, this should be displayed using some colorcode of the hyperlink. One may also think of using user-dependend linking-colors: if the content of the target changed since the last visit of this specific user, this fact should be visible to the user. [DublinCore 2004] may be used to get this information out of the resource, however, if this type of metadata is not supported by the target-system, then attributes provided by the HTTP-header should be used to notify users.

Many of this information may be visualized using the `title`-attribute of the anchor-element. However, there is limited space on the screen and users should not be flooded by the mentioned kind of information. Beside the title attribute, it is possible to use the `onfocus`-attribute

4 Conclusions and Future Work

It had been shown, that there are many additional attributes available in the HTML-recommendation. They are unfortunately not widely used and there is few support in web-browsers for all of the mentioned attributes. However, some attributes, which are of interest to the user, are not available and these attributes should be included in the title-attribute. This attribute is usually displayed by browsers.

The usage of this attribute may be a generic solution in all applications. Several plugins exist for specific browsers (such as the thumbnail-previewing plugin for Firefox/Mozilla) which are able to do sophisticated visualization of additional metadata of the target. However, there is no generic support for different browsers and users should not be forced to use a specific browser.

We therefore suggest to use the attribute of an anchor-element to add additional information of the target, such as who is the author of the target-document, modification date and size of the target document, the number of incoming links, to name just a few. All this information may be extracted out of the search-engines. However, it will also be important to have other, content related information attached to the link, such as an automatically generated abstract of the target.

Acknowledgment

This work is supported by DELOS, a Network of Excellence on Digital Libraries (EU FP6, G038-507618).

References

- [Bush, 1945] Bush, V. (1945). As we may think. *The Atlantic Monthly*, 176(1):101–108. <http://www.theatlantic.com/unbound/flashbks/computer/bushf.htm> (2005/01/15) .
- [Alexa, 2005] Alexa Internet Inc. An Amazon.com Company <http://www.alexa.com> (2005/04/24)
- [A9, 2005] A9.com Inc. An Amazon.com Company <http://www.a9.com> (2005/04/10)
- [Thumbshots, 2005] Thumbshots Web Previewing Technology Smartdevil Inc. <http://www.thumbshots.org> (2005/04/24)
- [Hyperwave 2004] Hyperwave (2004). Hyperwave Information Server. <http://www.hyperwave.com> (2005/04/21)
- [W3C:HTML40] Dave Raggett and Arnaud Le Hors and Ian Jacobs (1999) HTML 4.01 Specification. <http://www.w3.org/TR/html4/> (2005/03/21)
- [Firefox, 2005] Mozilla Firefox. <http://www.mozilla.org/products/firefox> (2005/04/20)
- [DublinCore 2004] Dublin Core Metadata Initiative (2004) <http://dublincore.org> (2004/04/07)
- [Krottmaier 2001] Harald Krottmaier (2001). Improving the Usability of a Digital Library Proceedings of the 5th ICC/IFIP International Conference on Electronic Publishing (ELPUB 2001)
- [Nielsen 1998] Jakob Nielsen (1998). Personalization is Over-Rated Alertbox (1998) <http://www.useit.com/alertbox/981004.html> (2004/03/01)