

LIS 523 - File Naming and Organization

Conciseness

In texts, conciseness is the degree to which a lot of information can be packed into a small number of words or characters.

File names should be concise for a variety of reasons:

- shorter names are easier to type;
- links to files with short names take up less space in HTML code, making the pages somewhat shorter and so somewhat faster to download;
- shorter names tend to be easier to remember;
- shorter names will not appear truncated in directory displays such as that in Windows Explorer, avoiding the need to adjust column widths or examine file properties more closely;
- shorter names are easier to read and easier to distinguish from other, similar names;
- longer file and directory names can combine to form excessively long path names which software cannot handle.

In DOS, files were constrained to a maximum for the name of 8 characters plus an extension of up to 3 characters. Windows now allows much longer file names, but the considerations listed above still apply.

For many of the same reasons, URLs should also be concise. Relative URLs are shorter than full URLs, especially if they refer to files in the same directory as the linking page. This might be a reason to avoid using subdirectories where possible, or to make directory contents fairly encapsulated, with few links between files in different directories.

Collocation

Collocation means bringing similar things together and separating dissimilar things. Using some sort of expressive notation in naming files can help in collocation. For example, I named files relating to Entity Relationship Diagrams on my LIS 558 site so that they all began with *558er*; thus, when I sorted the file list by name, I could see the complete sublist of files on this topic together.

Order

Even once similar files are grouped together, they may need to be reordered so that they follow a useful sequence. For example, if a series of 12 tutorial pages are named *tuto1.htm*, *tuto2.htm*, though *tuto12.htm*, the names will file in the directory in the order *tuto1.htm*, *tuto10.htm*, *tuto11.htm*, through *tuto9.htm*. It would thus be better to adopt the scheme *tuto01.htm*, *tuto02.htm*, which would show the files in their logical order.

Non-concomitance

Another way of saying this is that information given by file names should not be redundant. For example, calling a file *image01.gif* is redundant because the *.gif* extension already identifies it as an image. It would be more helpful if the file name identified what it was an image of or what page it was used by.

Some kinds of redundancy may be allowed as a safety measure or to provide for greater flexibility in reorganizing the files. For example, it may be useful to give the same short prefix to all the HTML pages in a given directory to avoid conflicts if directory contents need to be merged at a later date and to make the identity of individual files clear when they are transmitted out of context.

Context

With the reservation noted above, file names can usually be understood in the context of the directories in which the files occur. For example, many different HTML files may be called *index.htm*, and this will be all right as long as the files stay in the context of their respective directories, where each serves as a home page for a different site.

Constraints

Servers and other aspects of the Internet impose certain restrictions on how files should be named.

Unless you are allowed to change default server settings, your home page will probably have to have one of a very few predetermined names, typically *index.htm*, *index.html*, or *httoc.htm*. Conversely, such a file name will not be available for other purposes (so, you may not be able to call your site index *index.htm*).

Many non-alphanumeric characters should not be used in file names. Sometimes what is allowed varies from server to server or from time to time on the same server. For example, even though the ampersand (&) is a legal file name character according to the relevant HTML standard, the FIMS intranet started a while ago refusing to deliver the file *dispa&a1.gif* and reporting falsely that it could not be found, even though it in fact existed at the address specified. As a result, the file had to be renamed *dispa_a1.gif*, and the link to it had to be changed.

Some servers treat upper and lower case letters as the same and some treat them differently. So, you should not use case variation in naming your files. Instead, you are advised to use all lower case.

Of course, extensions should always match the file type. Putting the wrong extension on a file name leads to inappropriate and inconsistent behavior on the part of browsers.

Ascertainability and Predictability

It will be easier to find your files again and to name new files if you can remember or predict what particular files are called. Using short file names that follow a consistent scheme can help here. The scheme should be easy to remember; so, you probably do not want to employ a general library classification scheme to name your files.

Sometimes, a brief naming scheme will suggest itself fairly readily. For example, if your site includes a newsletter that is published monthly, you might call the files *news* followed by 2 digits for the year and 2 digits for the month (*news0312.htm*, etc.).

Once exceptions start to arise, you will need a standard way of dealing with them. For example, suppose one month you decide to issue two newsletters. If it is important to have all the newsletter files sort in date order, you might decide on a general rule to add the day of the month to the name of a second or third newsletter appearing in any given month (*news031215.htm*, etc.).

Naming a file should not require too much effort. Having content divided among files according to subjects that can be represented by brief, distinctive key words or phrases can be a great help in quick file naming.

Clarity and Accuracy

A file name should, if possible, not suggest that a file is something that it is not. Occasionally, misleading names can be unavoidable, as in the case of *index.htm*. Problems can also arise when the subject matter of a file changes over time.

Persistence

File names should not be changed too often, to avoid having to update links within the site, but, more importantly, to avoid having to deal with outdated links from outside your site over which you may have little or no control.

If you must rename a file, you should provide for redirection: if you have access to server options, you may be able to get the server to redirect users of expired URLs to the correct pages automatically; otherwise, a good idea is to leave stub pages with the old names that redirect to relevant renamed pages. It is good etiquette to advise visitors explicitly when a redirect happens, so that they know to update their bookmarks.



[Home](#)

Last updated April 24, 2007.

This page maintained by Prof. [Tim Craven](#)

E-mail (text/plain only): craven@uwo.ca

[Faculty of Information and Media Studies](#)

[University of Western Ontario](#),

London, Ontario

Canada, N6A 5B7