Relative URL – location (directory path) to a file *relative* to another file

Relative URLs are used for *internal links*, which are hyperlinks between files internally within a Web site.



Such a linking structure does not contain any ambient information about where the files are located. The name of the containing folder can be changed or the containing folder moved elsewhere, and the links will still work. A Web site with relative internal links is *portable*, meaning the whole structure could be moved without breaking the links. The first link in the previous slide could use an Absolute URL.

href="http://www.lfc.edu/~smith/sports/volleyball.html"

But, if you move the Web site to a different Web server, or even a different user account on the same server, the links will no longer work.

A site built with internal links that are absolute like above is not portable since all the links contain ambient information about where the files are located. Such ambient information is not necessary just to link files together relative to each other.

Note: It is convenient to show only the *href* portion of a link in the examples for this lesson href="URL" instead of the entire link <a href="URL">Click Me</a> because only the *href* part really matters in these examples. The notation .../ means up one folder in the file system tree. Said another way, it means go outside the current folder into the folder that contains it.

The site below uses Relative URLs for internal linking. Thus, the site is completely portable and could be moved. If moved, the internal structure would have to be preserved, or else the Relative URLs would no longer be accurate.

href="resume.html"
href="sports/default.html"
href="sports/football/stats.html"
href="../resume.html"
href="../football/stats.html"
href="../default.html"



The previous slide shows a few random hyperlinks linking the site together. Specific links were chosen to illustrate different Relative URLs.

But in practice, a Web site needs to be easily *navigable*. That means it should be linked together in a user friendly way, making it easy for people to navigate among the pages.

The following slides show examples of how (and how not to) link a site together internally. The linking strategy generally depends upon the nature of the site.

The site below has a terrible internal linking structure, where the linking is done in a seemingly random fashion.

Pages A and B form a "loop", with no escape (except using the back button on the browser).

Page C is isolated on the site since nothing links to it. You would need to know the name and location of the file in order to type an Absolute URL into a browser to find it.

home



A site where the internal linking structure is not logically organized. A common linking structure is called *linear* or *sequential*. Linear linking is ideal for a site (or part of a site) in which pages are best read in sequence, like a tutorial or online book, for example.

You move through pages in a linear fashion using <u>next</u> and <u>back</u> links. Some sites force you through the pages in a next/back linear fashion.

Some linear sites have a home page that works like a table of contents with links to each page, which may each have a <u>home</u> link to get back.



Another common linking structure is called *hierarchical*, which is ideal for sites with natural categories, sub-categories, and so forth.

There is usually a main page for each level of the *hierarchy* (each category) that has links to pages within that category. Same for sub-categories, etc.

The up/home links are often organized into a *breadcrumb* path that shows where you currently are in the hierarchy, and allows you to go back up all or part of the way. <u>Animals</u> > <u>Arthropods</u> > <u>Crustaceans</u> > <u>Crabs</u>

