Web Spider

Implementation Issues

Link Extraction

- Must find all links in a page and extract URLs.
  - <a href="http://www.site.uottawa.ca/~diana/csi4107">
  - <frame src="site-index.html">
- Must complete relative URL's using current page URL:
  - <a href="A1.htm"> to http://www.site.uottawa.ca/~diana/csi4107/A1.htm
  - <a href="../csi4107/paper-presentations.html"> to http://www.site.uottawa.ca/~diana/csi4107/paper-presentations.html

URL Syntax

- A URL has the following syntax:
  - <scheme>://<authority><path>?<query>#<fragment>
- A *query* passes variable values from an HTML form and has the syntax:
  - <variable>=<value>&<variable>=<value>…
- A *fragment* is also called a *reference* or a *ref* and is a pointer within the document to a point specified by an anchor tag of the form:
  - <A NAME="<fragment>">

Link Canonicalization

- Equivalent variations of ending directory normalized by removing ending slash.
  - http://www.site.uottawa.ca/~diana/
  - http://www.site.uottawa.ca/~diana
- Internal page fragments (ref's) removed:
  - http://www.site.uottawa.ca/~diana/csi1102/index.html#Eval
  - http://www.site.uottawa.ca/~diana/csi1102/index.html

Link Extraction in Java

- Java Swing contains an HTML parser.
- Parser uses "call-back" methods.
- Pass parser an object that has these methods:
  - HandleText(char[] text, int position)
  - HandleStartTag(HTML.Tag tag, MutableAttributeSet attributes, int position)
  - HandleEndTag(HTML.Tag tag, int position)
  - HandleSimpleTag (HTML.Tag tag, MutableAttributeSet attributes, int position)
- When parser encounters a tag or intervening text, it calls the appropriate method of this object.

Link Extraction in Java (cont.)

- In HandleStartTag, if it is an "A" tag, take the HREF attribute value as an initial URL.
- Complete the URL using the base URL:
  - new URL(URL baseURL, String relativeURL)
  - Fails if baseURL ends in a directory name but this is not indicated by a final "/"
  - Append a "/" to baseURL if it does not end in a file name with an extension (and therefore presumably is a directory).

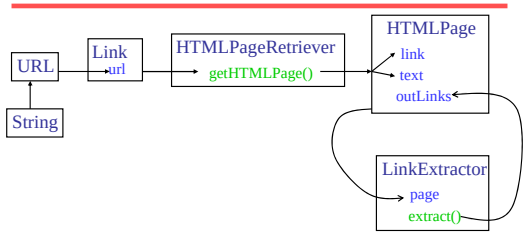
Java Spider

- Spidering code in ir.webutils package.
- Generic spider in Spider class.
- Does breadth-first crawl from a start URL and saves copy of each page in a local directory.
- This directory can then be indexed and searched using VSR InvertedIndex.
- Main method parameters:
  - -u <start-URL>
  - -d <save-directory>
  - -c <page-count-limit>

Java Spider (cont.)

- Robot Exclusion can be invoked to prevent crawling restricted sites/pages.
  - -safe
- Specialized classes also restrict search:
  - SiteSpider: Restrict to initial URL host.
  - DirectorySpider: Restrict to below initial URL directory.

Spider Java Classes



Cached File with Base URL

- Store copy of page in a local directory for eventual indexing for retrieval.
- BASE tag in the header section of an HTML file changes the base URL for all relative pointers:
  - <BASE HREF="<base-URL>">
- This is specifically included in HTML for use in documents that were moved from their original location.

Link-Directed Spidering

- Monitor links and keep track of in-degree and out-degree of each page encountered.
- Sort queue to prefer popular pages with many in-coming links (*authorities*).
- Sort queue to prefer summary pages with many out-going links (*hubs*).

Keeping Spidered Pages Up to Date

- Web is very dynamic: many new pages, updated pages, deleted pages, etc.
- Periodically check spidered pages for updates and deletions:
  - Just look at header info (e.g. META tags on last update) to determine if page has changed, only reload entire page if needed.
- Track how often each page is updated and preferentially return to pages which are historically more dynamic.
- Preferentially update pages that are accessed more often to optimize freshness of more popular pages.