Web Spiders

To avoid overloading our servers, we need to manage the way web crawlers/spiders find our pages. Googlebot is of primary concern, because it uses the most resources. However, we don't want to design solely for Google at the expense of other search engines.

Problems we need to fix

- P1. Make sure all of our (public) content can be indexed in some way.
- P2. We want our servers to behave gracefully when under heavy loads. If everything goes through Tomcat, Tomcat sessions must be managed in an organized manner. If users are passed between several servers, those servers must interact in a way that doesn't cause resources to be wasted.
- P3. It would be nice if search engines displayed our PURLs in their result listings instead of more specific URLs, but this isn't an absolute must, because all entries will be updated eventually.
- P4. We don't want spiders to accidentally get in infinite loops.

Decisions to make

- What is the best architecture of servers and applications?
- What is the best way to structure URLs that access our applications?

Research to be done

- What do other users of the PURL server do about spiders?
- How do other digital libraries manage this problem?
- How do other large sites (e.g., Amazon, Walmart, Google) manage this problem?
- How do other DSpace instances manage this problem?
- How do other handle-based servers (ACM Digital Library) manage this problem?
- Can we give bots cookies to manage sessions? Or do we have to include the session ID in the URL? (Sarah investigating)
    - There has been some evidence that major bots use cookies, but this is difficult to confirm in a non-production environment
    - you can either use mod_rewrite in Apache, possibly in conjunction with a few other tricks to strip the URL or you can use something called UrlRewriteFilter that will work with Tomcat to do the same thing.
- When does Googlebot follow redirects, and what does it do with them?
- Do SEOs (Search Engine Optimizers) have any pointers on dealing with spiders?
- Is there some way to always force googlebot sessions to expire quickly, or better yet, force it to always use the same session?
- Are loops ok? Have spiders in the past gotten into loops because they were generating multiple session IDs? Would we get better PageRanks by allowing Google to follow loops?
- Can the Tomcat's PersistentManager session manager solve the problem of too many sessions?

Questions answered:

- *Does Tomcat 5 handle sessions better than Tomcat 4? The Tomcats on thalia never list sessions. Is this a problem, or a setting we can take advantage of? Is there any new Tomcat functionality that we can use to improve session management?* Tomcat 5 does seem to handle heavy use a bit better than Tomcat 4. Tomcat 5 allows us to better see what is happening to an application, and the LambdaProbe app gives even more detail. Tomcat 5 also allows for replication of applications between servers, which can be used for load balancing and to provide a fail-over point.
- *Can we transmit the Tomcat session ID in a way that spiders can understand?* Google prefers not to receive a session ID.

Proposed solutions

Option 1 – Allow controlled spidering and supplement with sitemaps

1. Build applications so it they are possible to navigate without sessions, and robots never see session ID's in URLs. **May not be possible with Struts applications.**
2. Ignore the fact that many pages will be indexed with the non-PURL form of the address. Most search engines update their links often enough that they will have a working address, even if it is not the permanent address.
3. Open robots.txt to allow everything to be spidered.
4. When possible/convenient, create sitemaps to ensure every object in a collection is spidered.
5. Don't allow browse pages to be placed in the index. Add a robots meta tag with "noindex" to these pages. (links to these pages will still be followed)
6. Don't allow links out of detail pages to be followed. Add a robots meta tag with "nofollow" to these pages. This keeps the spider from accidentally falling into a loop (although a loop shouldn't normally happen if we're not displaying session IDs).

Advantages:

- All content gets indexed
- We can control how each collection is used.

Disadvantages:

- May still have problems with overwhelming Tomcat.
- Needs a certain amount of control for each collection.
- We can't fully evaluate this solution until most content is moved to Tomcat 5.

Option 2 – Generate static "landing pages"

1. Pre-generate a static copy of the detail page for each item. Links from the detail page point to the "live" webapp.
2. Place this detail page directly on the PURL server, served by Apache.
3. Allow web spiders to index content on the PURL server only.

Advantages:

- Search engines could index everything without creating Tomcat sessions.
- We could control how/when updates are performed.
- Our PURLs would always point to content that is suitable for viewing outside the context of a session.

Disadvantages:

- Essentially, we would need to write our own spider or hook an export mechanism into the Fedora messaging system. This would take a reasonable amount of development work.
- There may be inconsistencies between the static content and the live content.

Googlebot

Official Googlebot FAQ
Google Webmaster blog

Do a Google Image Search for "Pacific Shore line at Laguna Beach. Sunday" (with quotes) and you can see that the 600 pixel image is indexed by a webapp1 address, while the 1000 pixel image is indexed by a PURL. It appears that the image search takes its URLs from the URL that resulted in page that contains the image rather than the image src attribute (in this case, both images are referenced by PURLs, but the detail page is only referenced by a webapp1 URL).

According to Google, Googlebot is relatively conservative in its handling of robots.txt and meta-tag robot instructions.

Google sitemaps

URLs provided in sitemaps must point to the same server, and redirects aren't allowed. This means that sitemaps won't solve the problem with getting Google to recognize PURLs. The main advantage to sitemaps is ensuring that Google doesn't miss any content in the collection.

Articles

- 302 redirects (what our PURL resolver uses):
- Does Googlebot use cookies? This indicates that it does
- Using a tool called URLRewriteFilter, we can strip jsessionids just for googlebot - or we can turn off URL rewriting altogether
- Some sample robots.txt files