UNIVERSITY of WISCONSIN-MADISON
Computer Sciences Department

CS 202
Introduction to Computation

Professor Andrea Arpaci-Dusseau
Fall 2010

Lecture 38: How does a computer… find web pages?

"I'M STUCK. CHECK IT OUT ON GOOGLE."

How does Web work?

User wants to look at page specified by URL

- Uniform Resource Locator – http://Server/filename

Implement with two user-level apps

- Client (web browser):
  - Use TCP/IP to find SERVER and ensure requests arrive
  - HTTP protocol: "GET filename" (static content)
- Server: Replies with requested file
  - Reads file from file system; sends over network with TCP/IP
    - Doesn't know anything about contents of file
  - Easy to make your own web server!
    - Implementation Issue: Speed
- Client: Does work to interpret .html file, display nicely in browser
  - Html: HyperText Markup Language (links, headings, bold)

How do Search Engines work?

Web: Billions of pages each identified by URL

How does search for "CS 202 UW" find web page?

- http://pages.cs.wisc.edu/~cs202-1
- And in less than ½ second!

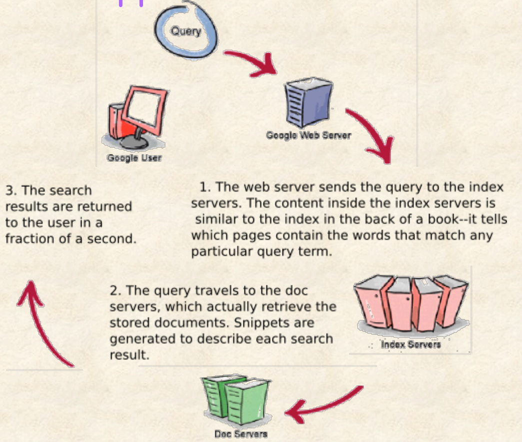
Search for CS?

Search for 202?

Search for UW?

Search for CS 202? CS 202 UW?

What Happens on Web Search?

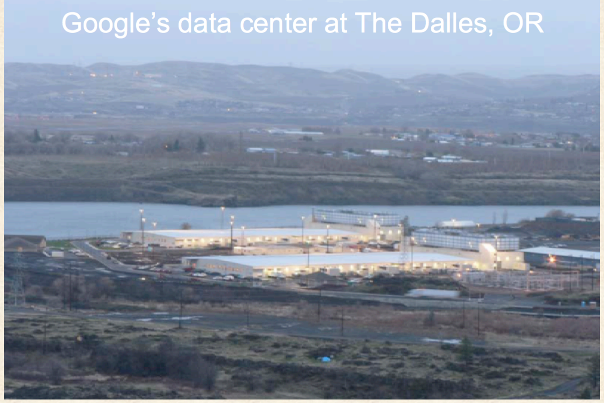


1. The web server sends the query to the index servers. The content inside the index servers is similar to the index in the back of a book--it tells which pages contain the words that match any particular query term.

2. The query travels to the doc servers, which actually retrieve the stored documents. Snippets are generated to describe each search result.

3. The search results are returned to the user in a fraction of a second.

Key #1 to Fast Search: Lots of Machines!


Google's data center at The Dalles, OR

Cluster Environment

Why do they need so many machines?

- Performance
  - Individual searches: < 0.5 second
  - Millions of simultaneous searches
- Service availability: 24x7
  - If one machine fails, have another do work for you
  - If lose power to one floor, have another floor do work
  - If one site fails, have another site do work!
- Data availability: Don't ever lose data
  - Replicate data on 3 machines with local disks

Commodity components (CPUs, disks, OS)

- Cost/performance is important

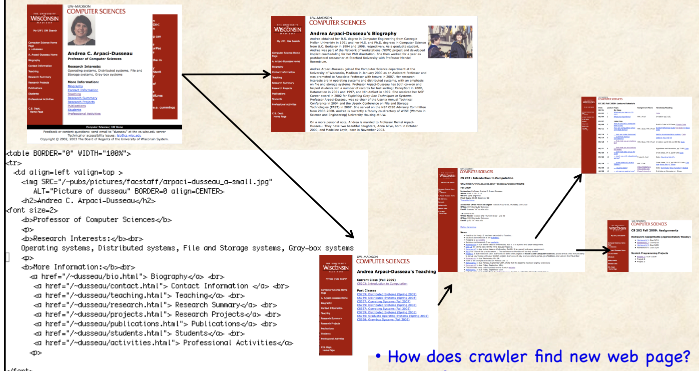
Key #2 to Fast Search: Preparation!

Before user accesses search engine:

1. Crawl the web
   - Gather billions of web pages
2. Index all web pages
   - Look at content of web pages
   - Store keywords associated with each
3. Rank reputation of each page
   - Is this page reputable or not?

Step 1: Crawling the Web

Web can be viewed as a tree: Start at some root



```
<table BORDER="0" WIDTH="100%">
<tr>
  <td align=left valign=top >
    <img SRC="/~pubs/pictures/facstaff/arpaci-dusseau_a-small.jpg"
       ALT="Picture of dusseau" BORDER=0 align=CENTER>
    <h2>Andrea C. Arpaci-Dusseau</h2>
<font size=2>
    <b>Professor of Computer Sciences</b>
    <p>
    <b>Research Interests:</b><br>
    Operating systems, Distributed systems, File and Storage systems, Gray-box systems
    <p>
    <b>More Information:</b><br>
      <a href="/~dusseau/bio.html"> Biography</a> <br>
       <a href="/~dusseau/contact.html"> Contact Information </a> <br>
       <a href="/~dusseau/teaching.html"> Teaching</a> <br>
       <a href="/~dusseau/research.html"> Research Summary</a> <br>
       <a href="/~dusseau/projects.html"> Research Projects</a> <br>
       <a href="/~dusseau/publications.html"> Publications</a> <br>
       <a href="/~dusseau/students.html"> Students</a> <br>
       <a href="/~dusseau/activities.html"> Professional Activities</a>
      <p>

</font>
</td>
```

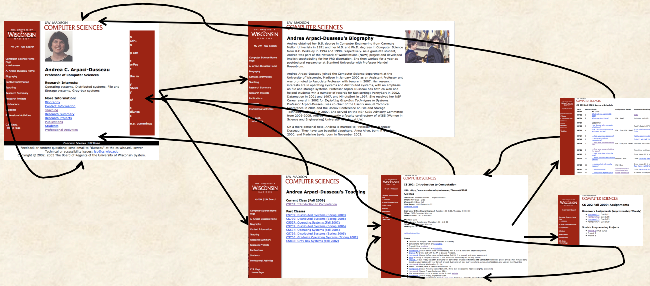
- How does crawler find new web page?
- What if not connected to anyone else?

What is Stored for each Page?

1. Links: What other pages should be crawled
   - Make shared list of URLs to crawl
   - Worker machines grab URL, crawl, add new URLs
     - All in parallel!
2. Cached copy (doc server):
   Copy of content (words, pictures)
   - Why useful?
     - Useful if server holding page is down
     - Useful for highlighting search terms
3. Index terms: What keywords refer to here?
4. Rank or reputation…

Crawling the Web: Some Complexities

Is the Web really a tree?



No, Web is a graph with cycles

Why do cyclic graphs complicate crawling?

Don't want crawler to be trapped in infinite loop

Issues with Crawling

How to ensure don't visit same page again?

- Check to see if URL is already in crawl list
  - If it is, don't add
- Check to see if URL has been visited "recently"
  - If it has, don't add

How do you define "recently"?

- Different for different pages
  - How recently was cnn.com crawled?
  - UW CS dept? My web page? CS 202 web page?

Crawl Frequency

Check date of cached copy

- CNN?
- Wisconsin CS homepage?
- Andrea's homepage?
- CS 202 homepage?
- CS 202 assignment 1 page?

Some pages crawled every day, others 1/month

What should crawl frequency be a function of?

- How frequently page is modified
- How popular the page is

Key #2 to Fast Search: Preparation!

Before user accesses search engine:

1. ~~Crawl the web~~
   - ~~Gather billions of web pages~~
2. Index all web pages
   - Look at content of web pages
   - Store keywords associated with each
3. Rank reputation of each page
   - Is this page reputable or not?

Step 2: Indexing Page Content

How to find page with specified words?

- Can't quickly search thru billions of page content for words
- Must do more work ahead of time

Needed information:

Given keyword, on what pages does it appear?

- Similar to index of terms in a technical book

Update index when crawling

- Give each page a unique ID
- Omit short, common words from index
- Organize index so can update quickly!

Example Index

Page Content (ID: 64)

University of Wisconsin, Madison

CS 202 : Introduction to Computation

Overview

The purpose of computing is insight, not numbers. -- Richard Wesley Hamming

Designed for a diverse audience, this course examines some of the fundamental ideas behind the science of computing. This course, like the field of Computer Science in general, is more than just the study of how computers work, of how to program computers, or of how to use computers. At the highest level, this course focuses on studying algorithms which are step-by-step methods for accomplishing a complex task.

Index (alphabetize!)

University: 1 20 … 64 … 85
Wisconsin: 8 35 42 48 64 …
Madison: … 64 …
CS: … 64 …
202: … 64 …
Introduction: … 64 …
Computation: … 64 …
Overview: … 64 …
Purpose: … 64 …
Computing: … 64 …
Insight: … 64 …
Not: … 64 …
Numbers: … 64 …
…

Would you keep any other info in index?

How to Combine Multiple Search Terms?

Example: Search for "CS 202"

Index of 2 Search Terms:

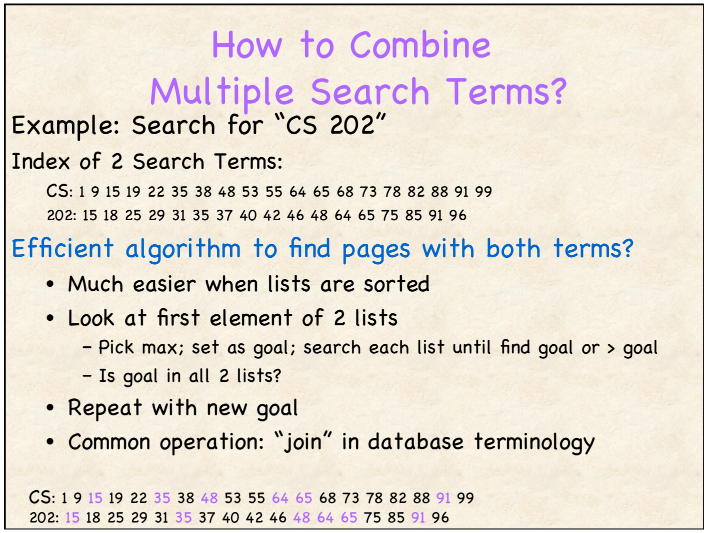
CS: 22 88 91 65 68 48 53 55 1 9 15 19 78 82 64 99 35 38 73

202: 37 85 91 96 40 48 18 25 15 35 29 31 42 46 64 65 75

Algorithm to find pages with both terms?

- First attempt: Lists are not sorted!
- Look at first element of first list
  - Does it appear in second list? Must search through entire list!
  - How many comparisons for this first element?
    - N comparisons
- Repeat for every item in first list (N items!)
  - Total number of comparisons?
  - $O(N^2)$

CS and 202: 22 88 91 65 68 48 53 55 1 9 15 19 78 82 64 99 35 38 73



How to Combine Multiple Search Terms?

Example: Search for "CS 202"

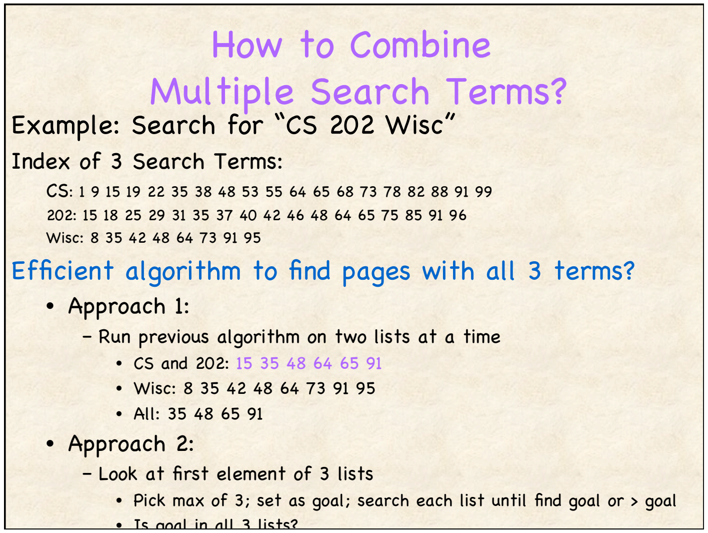
Index of 2 Search Terms:

CS: 1 9 15 19 22 35 38 48 53 55 64 65 68 73 78 82 88 91 99
202: 15 18 25 29 31 35 37 40 42 46 48 64 65 75 85 91 96

Efficient algorithm to find pages with both terms?

- Much easier when lists are sorted
- Look at first element of 2 lists
  - Pick max; set as goal; search each list until find goal or > goal
  - Is goal in all 2 lists?
- Repeat with new goal
- Common operation: "join" in database terminology

CS: 1 9 15 19 22 35 38 48 53 55 64 65 68 73 78 82 88 91 99
202: 15 18 25 29 31 35 37 40 42 46 48 64 65 75 85 91 96



How to Combine Multiple Search Terms?

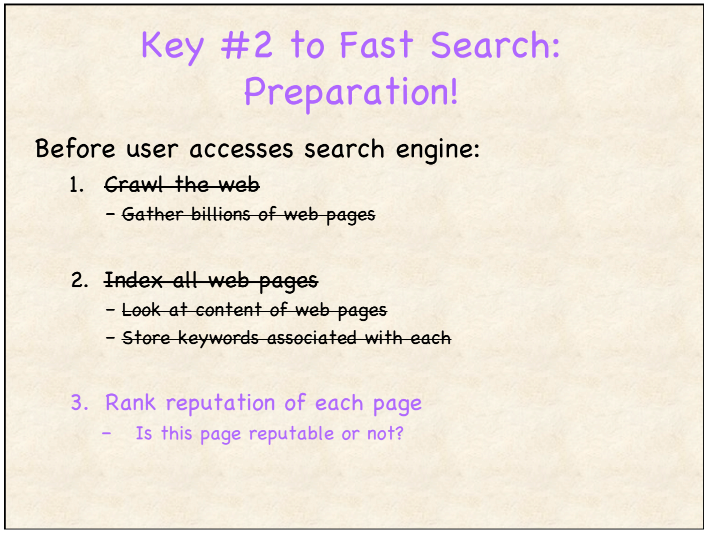
Example: Search for "CS 202 Wisc"

Index of 3 Search Terms:

CS: 1 9 15 19 22 35 38 48 53 55 64 65 68 73 78 82 88 91 99
202: 15 18 25 29 31 35 37 40 42 46 48 64 65 75 85 91 96
Wisc: 8 35 42 48 64 73 91 95

Efficient algorithm to find pages with all 3 terms?

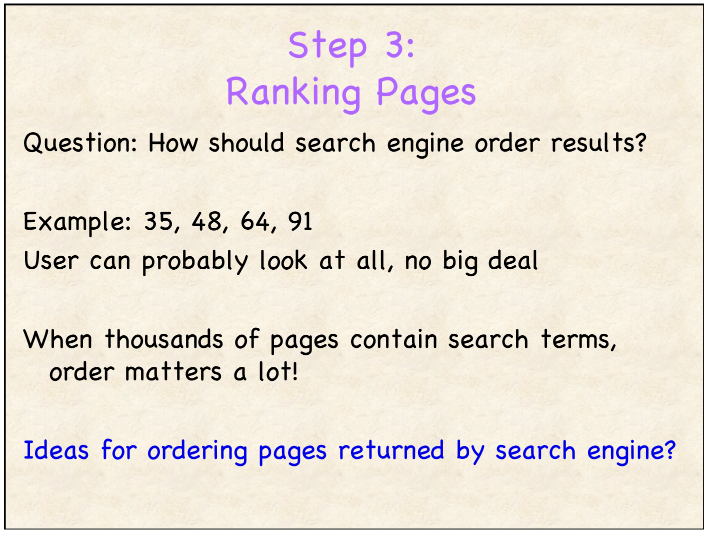
- Approach 1:
  - Run previous algorithm on two lists at a time
    - CS and 202: 15 35 48 64 65 91
    - Wisc: 8 35 42 48 64 73 91 95
    - All: 35 48 65 91
- Approach 2:
  - Look at first element of 3 lists
    - Pick max of 3; set as goal; search each list until find goal or > goal
    - Is goal in all 3 lists?



Key #2 to Fast Search: Preparation!

Before user accesses search engine:

1. ~~Crawl the web~~
   - ~~Gather billions of web pages~~
2. ~~Index all web pages~~
   - ~~Look at content of web pages~~
   - ~~Store keywords associated with each~~
3. Rank reputation of each page
   - Is this page reputable or not?



Step 3: Ranking Pages

Question: How should search engine order results?

Example: 35, 48, 64, 91

User can probably look at all, no big deal

When thousands of pages contain search terms, order matters a lot!

Ideas for ordering pages returned by search engine?

Ranking Attempt #1: Order by Count

Intuition: More often page uses word, more relevant page is to search term

Problem?

- Must view as adversarial process
- Goals of search engine and web page creators not perfectly aligned

Goals of each?

- Search engine: Put page searching user wants at top
- Web page creator: Put my page first regardless!
  - Will do any tricks necessary

Trick #1: Put Bogus Content

```
<html>
money money money money money money money
   money money money money money money money
   money money  money money money money money
   money money money money money money money
   money money money money money money money
   money money money money money money money
   money money money money money money money
   money money money money money money money
   money money money money money money money
   money
</html>
```

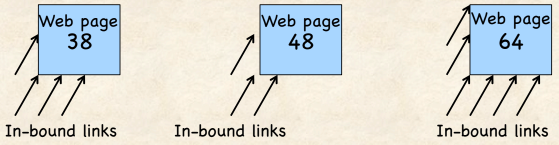
Ranking Attempt #2

How can you tell that page has good information?

- Other pages link to it!

Idea behind Google's PageRank algorithm:

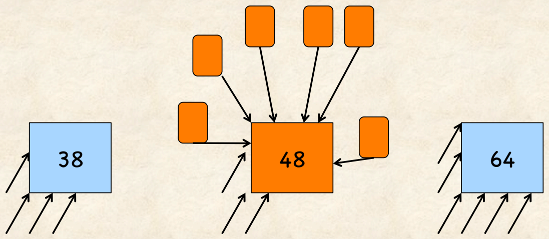
- More importance to pages w/ many in-bound links
- Intuition: Pages are "voting" for you



Problem with Attempt #2

How could adversary trick this algorithm?

- Create bogus pages that point to target page!



Ranking Attempt #3

How can we fix problem of bogus voters?

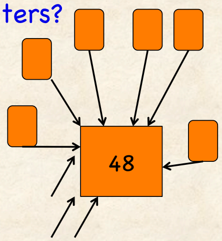
- Also consider reputation of voters
- Scale vote by reputation of voter
  (and how many votes they cast)

Do we know reputation of voters?

- Page Rank

Why is Page Rank of bogus voters low?

- No one points to them (or only other bogus pages do!)



Example Page Rank of 48?

- 3 * (some reasonable page rank/votes cast) + 6 * (0)

Many Variables for Ranking

Exact algorithm for determining ranking is secret!

Why?

Today's Summary

Web search: Resources + Preparation!

- Crawl content, create indices, rank by reputation

Reading: 7.4-7.6

Announcements

- HW 8: Due today (paper by noon, 5pm Learn@UW)
- HW 9: Essay on using Google Trends (Wed 12/8)
- HW 10: Upload draft of Project 2 (Thu 12/9)
  - Comment on 5 others by Friday
- Project 2: Due Monday 12/13
- Final Exam: 12/22
  - Will arrange Review Session after classes…