

## CMS/CS/EE 144

### Networks: Structure & Economics

#### Administrivia

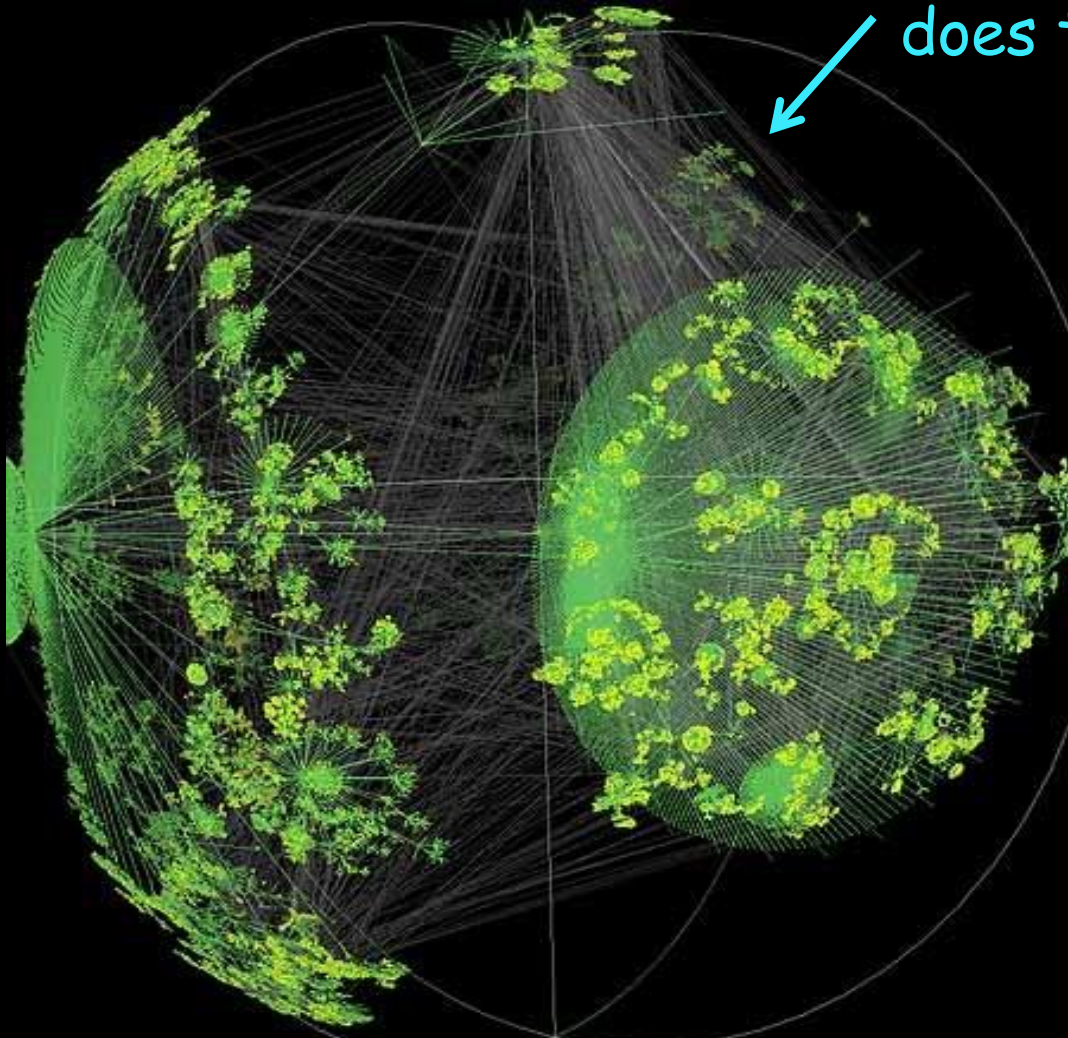
- 1) HW3 is due today  
→ Grab solutions up front
- 2) HW4 is out today
- 3) Don't forget your blog posts!
- 4) Start thinking about projects...
- 5) Rankmaniac is coming up...

## COURSE OUTLINE

- 1) Understanding network structure
- 2) Exploiting network structure
- 3) Network economics

RECAP

What structural properties  
does the web have?



Courtesy of Young Hyun, CAIDA

## RECAP

Four “universal” properties of networks

- 1) A “giant” connected component
- 2) Small diameter
- 3) Heavy-tailed degree distribution
- 4) High clustering coefficient

## Scientific question

What causes the emergence  
of these properties?

## RECAP

### Four “universal” properties of networks

- Indep. random choices –  $G(n,p)$  → 1) A “giant” connected component  
2) Small diameter
- “rich get richer” → 3) Heavy-tailed degree distribution  
4) High clustering coefficient

## RECAP

### Four “universal” properties of networks

1) A “giant” connected component

2) Small diameter

3) Heavy-tailed degree distribution

4) High clustering coefficient

Correlated local  
connections  
combined w/  
distance-dep.  
global  
connections



## RECAP

### Four “universal” properties of networks

1) A “giant” connected component

2) Small diameter

3) Heavy-tailed degree distribution

4) High clustering coefficient

Correlated local  
connections  
combined w/  
distance-dep.  
global  
connections



Note: These are only possible “explanations  
for these properties.

There is much more we could cover...

Many more properties...

- weak ties vs strong ties
- expansion
- densification
- shrinking diameter
- power law eigenvalues
- ...

Many more models...

- copying model
- random web surfer
- kronecker graphs
- geometric random graphs
- dot product graphs
- ...



## COURSE OUTLINE

- 
- 1) Understanding network structure
  - 2) Exploiting network structure
  - 3) Network economics

## Where can network structure be exploited?

TODAY

Search ... *the killer app for network structure*

High clustering both aids and inhibits spread of viruses (or information)

Heavy-tailed degrees make networks “robust yet fragile” to attack

Identifying “fake” accounts and suggesting friends in facebook

Community detection, controlling the spread of fake news, and many more...

TODAY

Search ... *the killer app for network structure*

caltech - Google Search - Mozilla Firefox

File Edit View History Bookmarks Tools Help

http://www.google.com/search?hl=en&q=caltech&aq=f&oeq=&aqi=g10

caltech - Google Search


Web Images Videos Maps News Shopping Gmail more ▼

adamwierman@gmail.com | Web History | Settings ▼ | Sign out

Google caltech Search Advanced Search

View customizations

Web Show options... Results 1 - 10 of about 4,310,000 for caltech. (0.12 seconds)



California Institute of Tech  
www.caltech.edu

201 15 Nail Stop  
Pasadena, CA 91125-0001  
(626) 395-6357  
[Get directions](#) - [Is this accurate?](#)

[Write a review](#)  
[More information >](#)

[California Institute of Technology](#)  
Academic home of NASA's Jet Propulsion Laboratory. Research affiliation: Association of American Universities.  
www.caltech.edu/ - [Cached](#) - [Similar](#) - [Print](#) - [Close](#)

<a href="#">Graduate Studies</a>	<a href="#">Faculty</a>
<a href="#">Caltech Undergraduate Admissions</a>	<a href="#">Campus Maps</a>
<a href="#">Job Opportunities</a>	<a href="#">Caltech Personnel Directory</a>
<a href="#">Division of ENGINEERING &amp; ...</a>	<a href="#">Caltech Catalog</a>

[More results from caltech.edu >](#)

[California Institute of Technology - Wikipedia, the free encyclopedia](#)  
The California Institute of Technology (commonly referred to as **Caltech**) is a .... **Caltech** is on the trimester system. Because of its schedule, **Caltech's** ...  
en.wikipedia.org/wiki/California\_Institute\_of\_Technology - [Cached](#) - [Similar](#) - [Print](#) - [Close](#)

[Caltech Seismological Laboratory \[Earthquakes, Research, Academics\]](#)  
Earthquake information, geodynamics, seminar schedules.  
www.seismolab.caltech.edu/ - [Cached](#) - [Similar](#) - [Print](#) - [Close](#)

[Caltech Astronomy](#)  
The department, observatories, academics, research activities, events, resources, and public outreach.  
www.astro.caltech.edu/ - [Cached](#) - [Similar](#) - [Print](#) - [Close](#)

Physics, Mathematics, and Astronomy at Caltech

Done

How does google choose search results?

## The search landscape

>800 million searches per day

>\$20 billion in paid search revenues

Goal: Make money

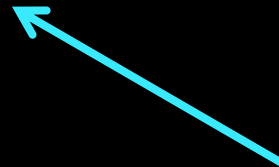
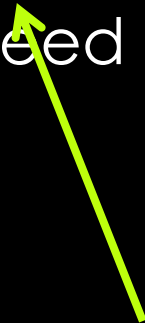
Challenges:

- Need lots of users
- Need to serve ads effectively

Give users what  
they want (speed & quality)

Later in course

Data centers!



Users want: to get results quickly and have the most relevant pages in the first few positions of the search

Challenges:

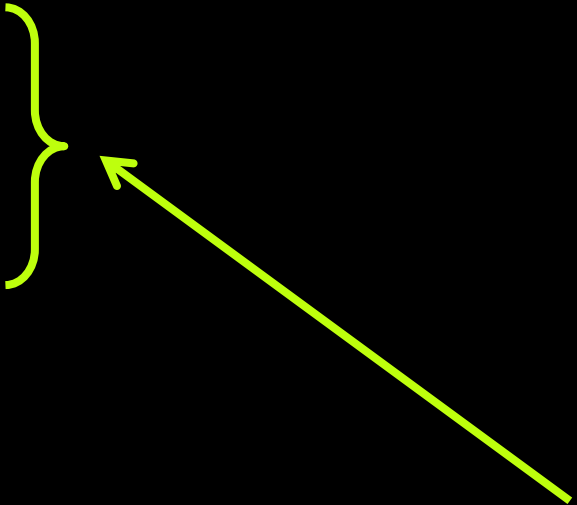
- Scale of the web (~1 trillion pages indexed)
- Needs to be really fast (the speed of a keystroke)

## What goes into making a search engine run?

- 1) Maintain a huge index of web pages and links
  - 2) Organize the info about the web to allow fast access
  - 3) Choose the best pages for a given query
  - 4) Display the results to the user
- 
- ```
graph LR; display[display] --> ranking[ranking]; ranking --> indexing[indexing]; indexing --> crawling[crawling]
```
- The diagram illustrates the search engine process flow using four yellow arrows pointing upwards and to the right. The arrows are labeled 'display', 'ranking', 'indexing', and 'crawling' in yellow text. The 'display' arrow points from the bottom step to the 'ranking' step. The 'ranking' arrow points from the 'ranking' step to the 'indexing' step. The 'indexing' arrow points from the 'indexing' step to the 'crawling' step. The 'crawling' arrow points from the 'crawling' step to the 'indexing' step.



## Four components of a search engine

- 1) Crawling
  - 2) Indexing
  - 3) Ranking
  - 4) Display
- 

Key to search engines → preprocessing  
(speed is more important than space)

- 1) Crawling
- 2) Indexing
- 3) Ranking
- 4) Display

You did this on your homework!

Important issues you didn't consider:

- 1) Avoiding “spam” pages
- 2) Dynamic content – “hidden web”
- 3) Parallelism (maintaining lots of connections)
- 4) Freshness of the crawl  
→ time between changes is heavy-tailed
- 5) Avoid causing DoS!

- 1) Crawling
- 2) Indexing
- 3) Ranking
- 4) Display

Preprocess database to allow efficient access for ranking

This is done by building an “inverted index”

- For each word, maintain a list of every web page it appears on.
- List points to compressed version of each page (~850 TB as of 2006)

### Document 1

The bright blue butterfly hangs on the breeze.

### Document 2

It's best to forget the great sky and to retire from every wind.

### Document 3

Under blue sky, in bright sunlight, one need not search around.

### Stopword list

a  
and  
around  
every  
for  
from  
in  
is  
it  
not  
on  
one  
the  
to  
under  
.  
.  
.

### Inverted index

| ID | Term      | Document |
|----|-----------|----------|
| 1  | best      | 2        |
| 2  | blue      | 1, 3     |
| 3  | bright    | 1, 3     |
| 4  | butterfly | 1        |
| 5  | breeze    | 1        |
| 6  | forget    | 2        |
| 7  | great     | 2        |
| 8  | hangs     | 1        |
| 9  | need      | 3        |
| 10 | retire    | 2        |
| 11 | search    | 3        |
| 12 | sky       | 2, 3     |
| 13 | wind      | 2        |

Image from <http://developer.apple.com>

[/documentation/UserExperience/Conceptual/SearchKitConcepts/searchKit\\_basics/chapter\\_2\\_section\\_2.html](http://documentation/UserExperience/Conceptual/SearchKitConcepts/searchKit_basics/chapter_2_section_2.html)

- 1) Crawling
- 2) Indexing
- 3) Ranking
- 4) Display

### Details:

- 1) Omit common words, called “stop words”
- 2) “stem” the words (cats → cat, etc)

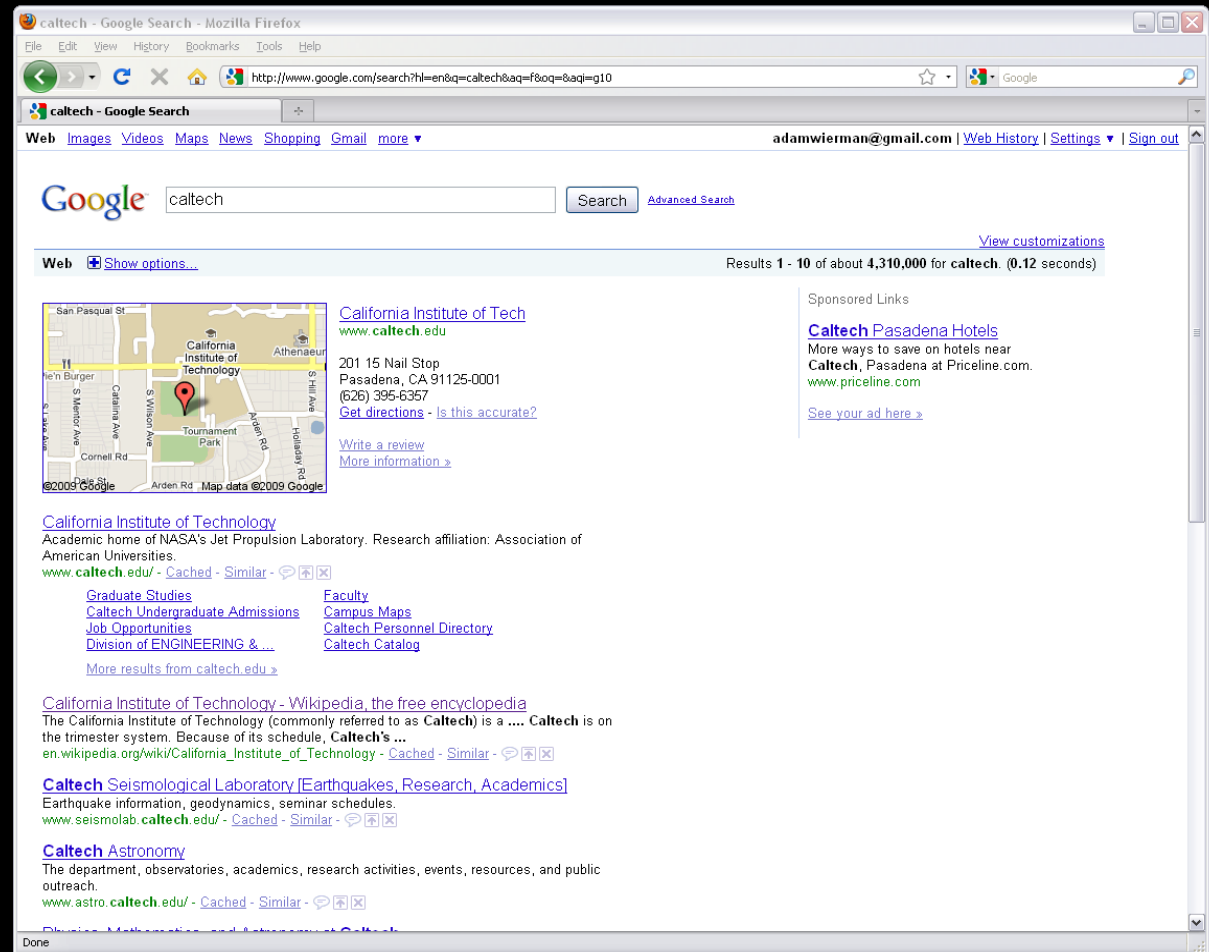
- 1) Crawling
- 2) Indexing
- 3) Ranking
- 4) Display

The heart of the search engine  
...we'll come back to it

- 1) Crawling
- 2) Indexing
- 3) Ranking
- 4) Display

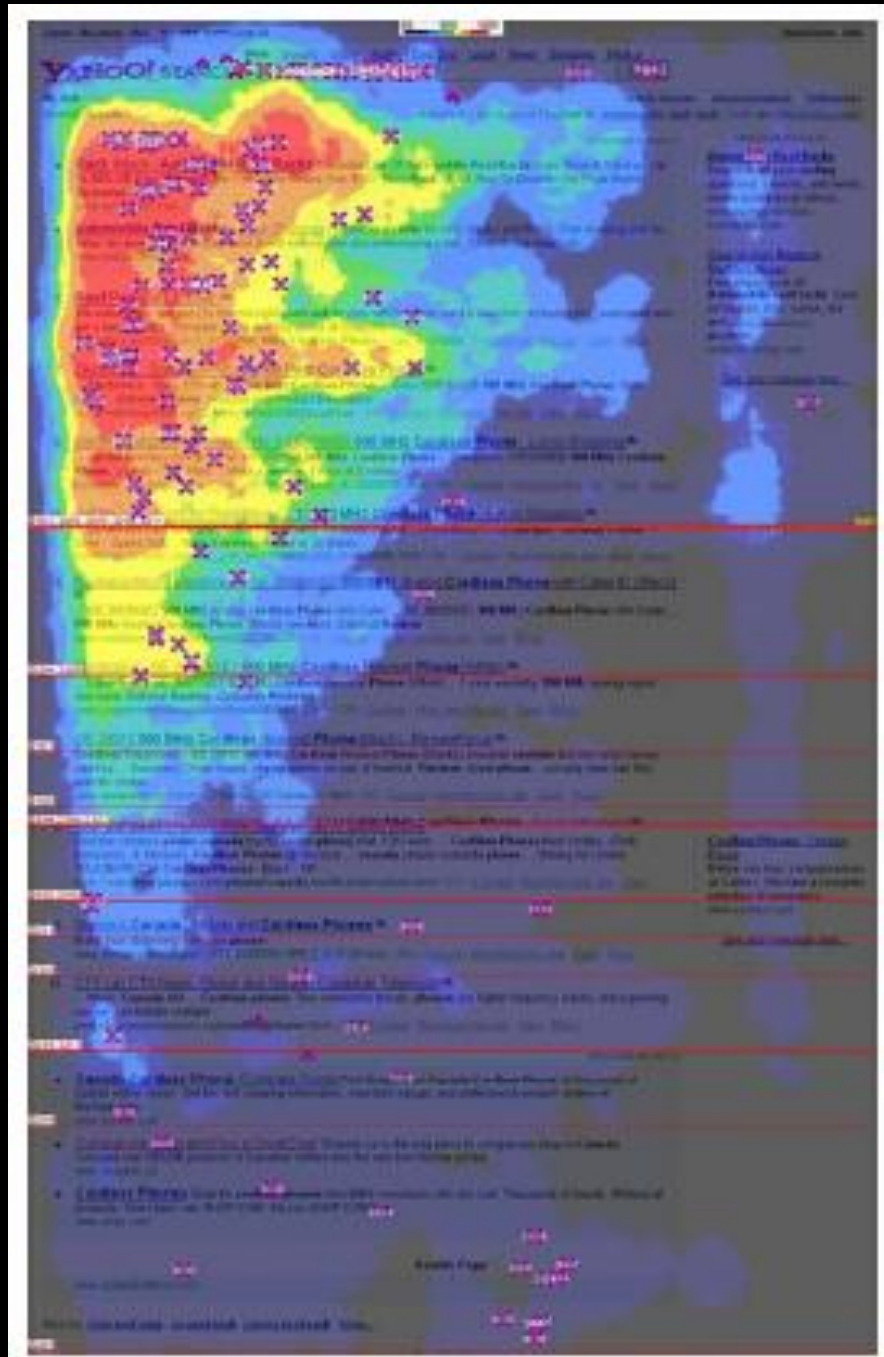
Present results to users

Seems boring and  
“done”, but...



- 1) Crawling
- 2) Indexing
- 3) Ranking
- 4) Display

Seems boring and  
“done”, but...



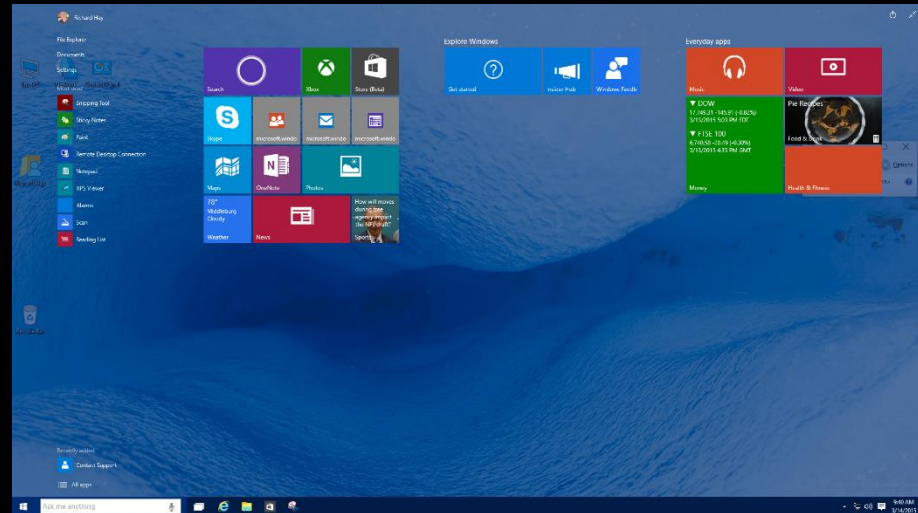


- 1) Crawling
- 2) Indexing
- 3) Ranking
- 4) Display



Seems boring and  
“done”, but...

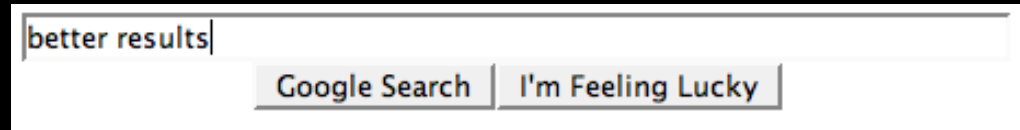
...it's on the verge  
of a big change



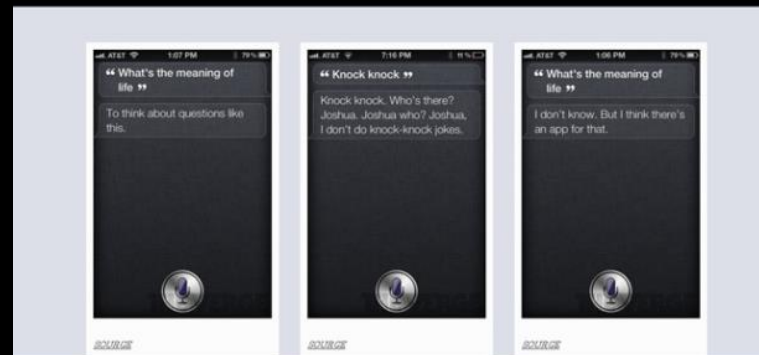
- 1) Crawling
- 2) Indexing
- 3) Ranking
- 4) Display

Seems boring and  
“done”, but...

...it's on the verge  
of a big change

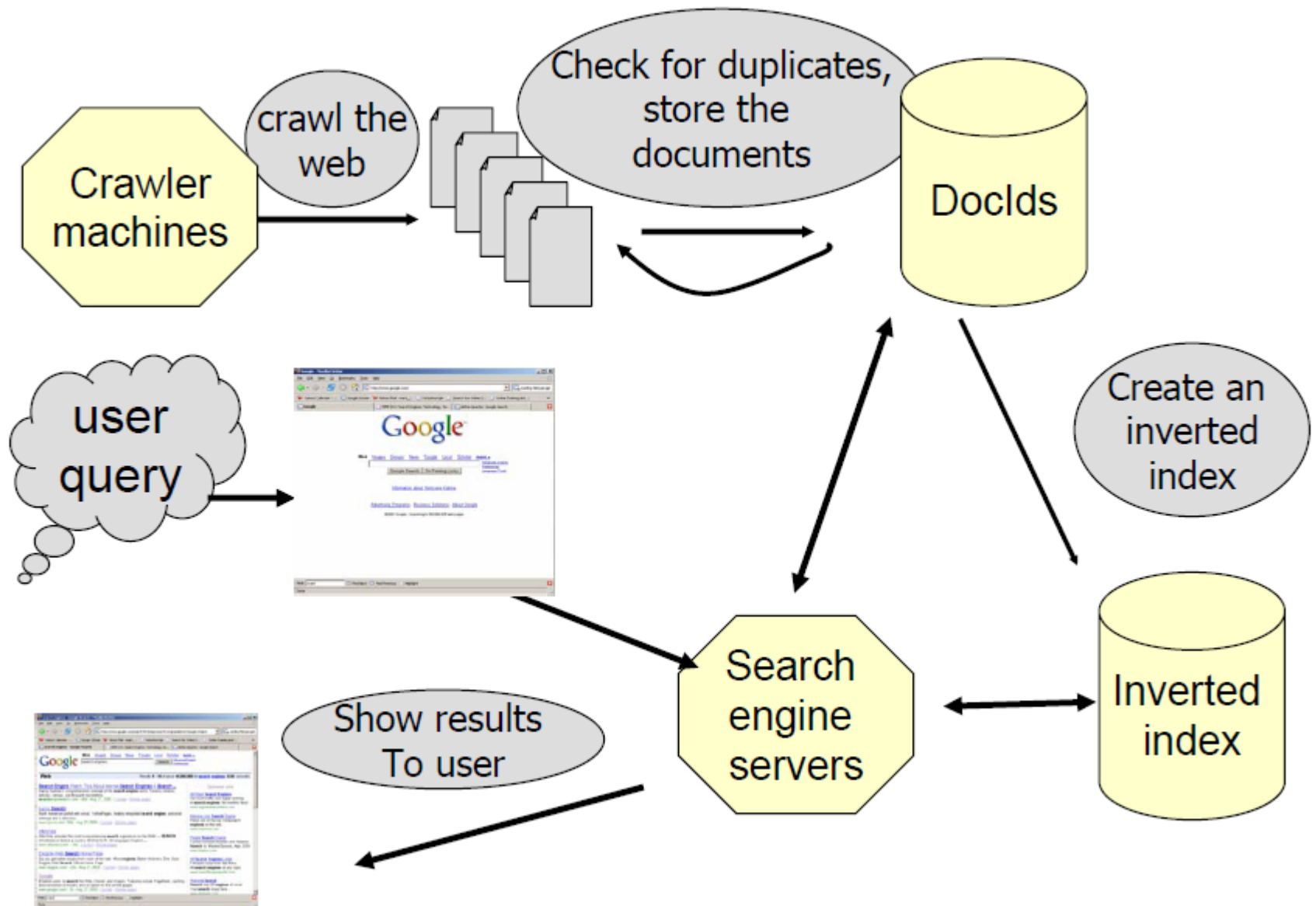


## STUFF THAT SIRI SAYS



- 1) Crawling
- 2) Indexing
- 3) Ranking
- 4) Display

# Typical search engine architecture



(from Marti Hearst)

- 1) Crawling
- 2) Indexing
- 3) Ranking
- 4) Display

The heart of the search engine

Goal: Given a query, place the most relevant pages in the first few positions of the search

Challenges:

- Scale of the web (~1 trillion pages indexed)
- Needs to be fast (<100msecs)

- 1) Crawling
- 2) Indexing
- 3) Ranking
- 4) Display

The heart of the search engine

Question: How do we solve this?



It's a machine learning problem!  
Measure ~20(?) features for each page and  
then use these to predict relevance

Input:

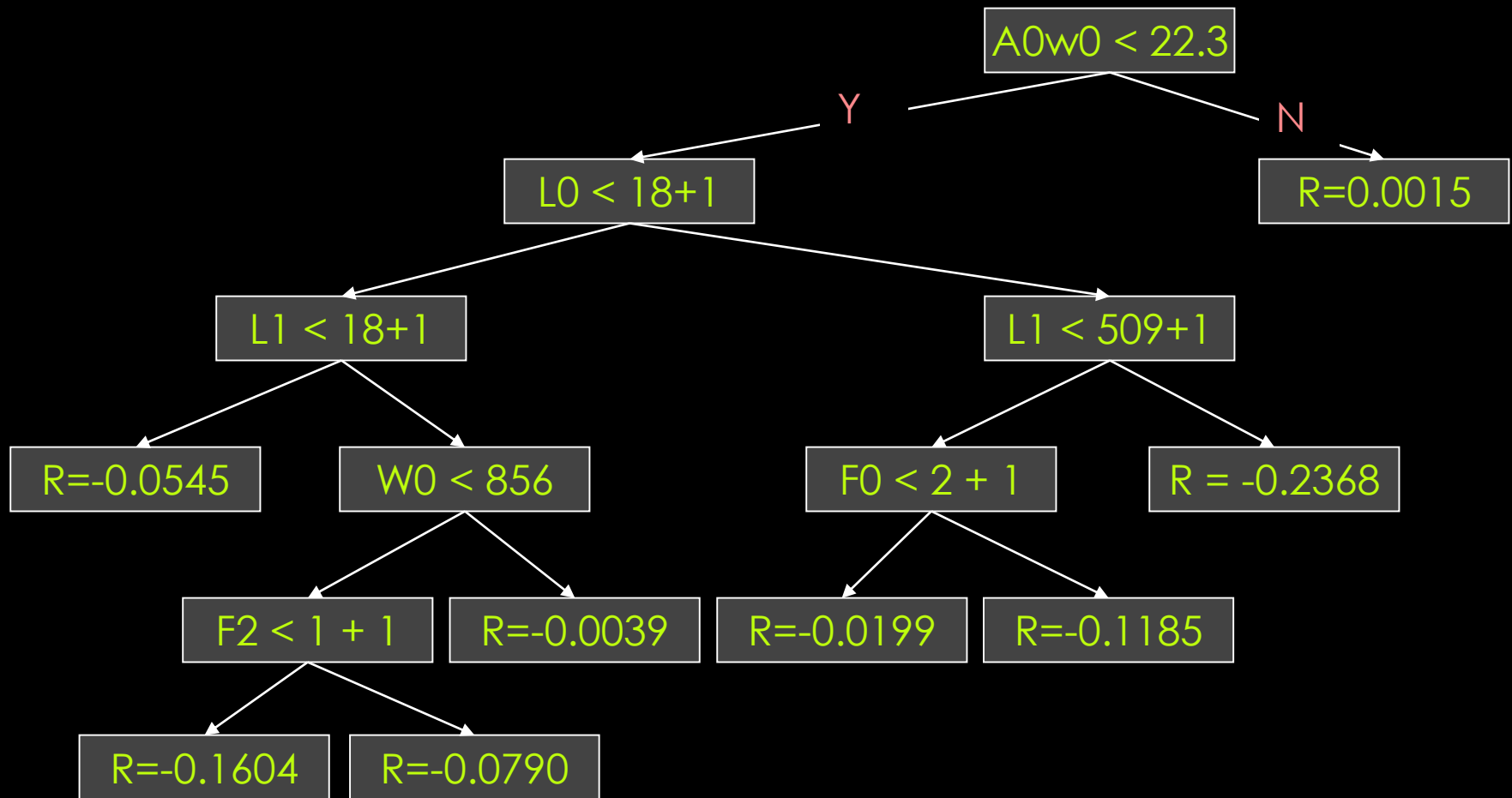
- large training set
- data for each feature

Output

- ranking function / decision tree

# Decision tree example

(From Jan Pederson of Yahoo)





### Input:

- large training set
- data for each feature

### Output:

- ranking function / decision tree

### Key:

this allows rapid, ongoing development

- adjust features
- test new features

## Question: What features do you think they use?

- Frequency of matching query words in the page
- Proximity of matching words to one another
- Location of terms within the page
- Location of terms within tags e.g. <title>, <h1>, link text, Body text
- Anchor text on pages pointing to this one
- Frequency of terms on the page and in general
- Click-through analysis: how often the page is clicked on
- How “fresh” is the page
- Graph structure – how “important” the page is
- Page load time

Question: Which can be precomputed?  
Which are query specific?

## Question: What features do you think they use?

- Frequency of matching query words in the page
- Proximity of matching words to one another
- Location of terms within the page
- Location of terms within tags e.g. <title>, <h1>, link text, Body text
- Anchor text on pages pointing to this one
  - Frequency of terms on the page and in general
  - Click-through analysis: how often the page is clicked on
  - How “fresh” is the page
- Graph structure – how “important” the page is
  - Page load time

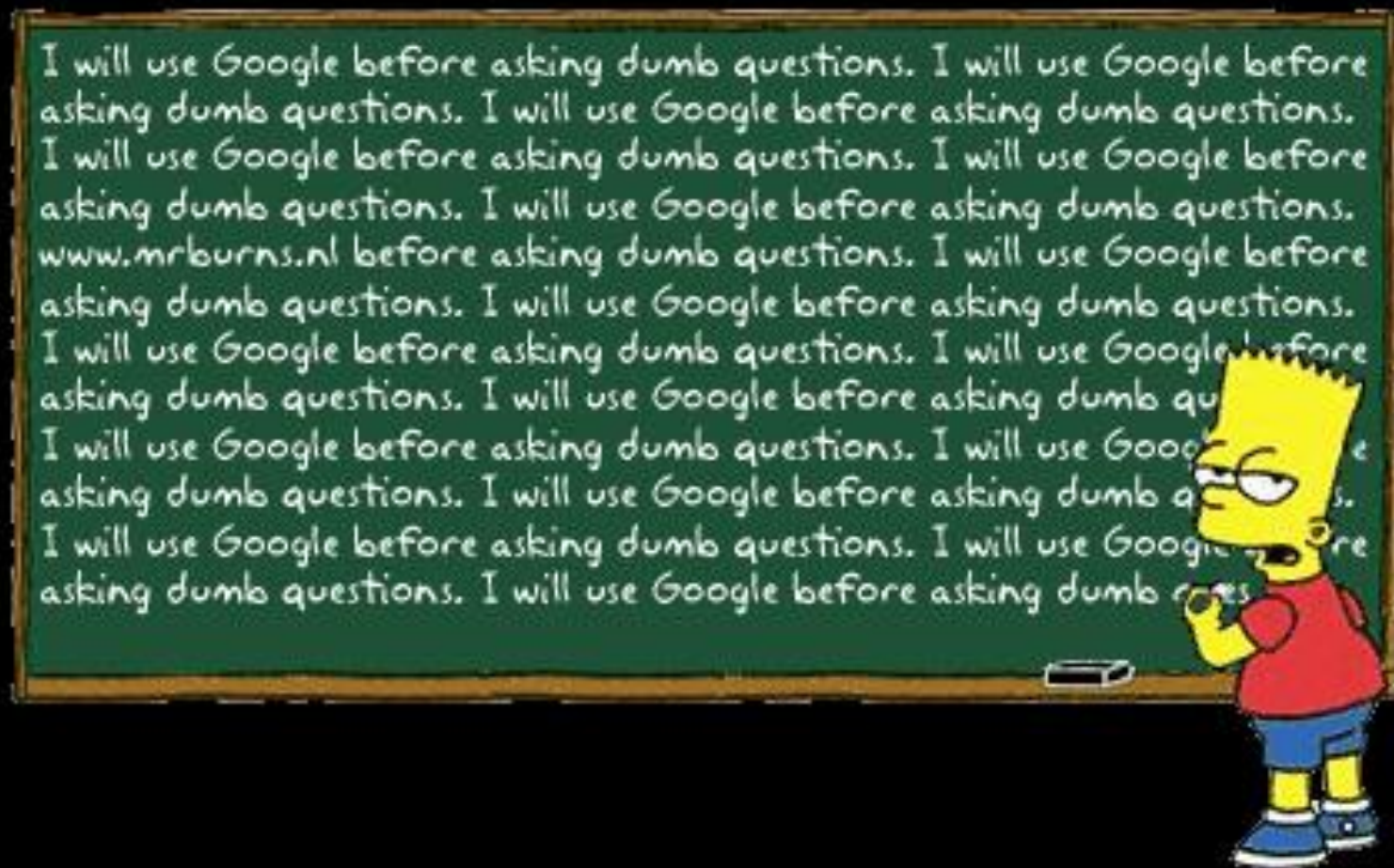
Key realization of google → Graph structure is helpful

# SEARCH BEFORE GOOGLE



Use search engine X to search for X, and only 1 would even return a link to itself on the first page!

# SEARCH AFTER GOOGLE



## The idea:

Use graph structure to tell you which pages are important – “**pagerank**”

→ There were a handful of related measures floating around academia in CS

[Marchiori 97] [Spertus 97] [Kleinberg 98] [Page 98]

In social networks related ideas had been studied since the 50s, but I don't think the CS folks knew...

**Brin & Page built a large scale prototype**





**Sergey Brin** received his B.S. degree in mathematics and computer science from the University of Maryland at College Park in 1993. Currently, he is a Ph.D. candidate in computer science at Stanford University where he received his M.S. in 1995. He is a recipient of a National Science Foundation Graduate Fellowship. His research interests include search engines, information extraction from unstructured sources, and data mining of large text collections and scientific data.



**Lawrence Page** was born in East Lansing, Michigan, and received a B.S.E. in Computer Engineering at the University of Michigan Ann Arbor in 1995. He is currently a Ph.D. candidate in Computer Science at Stanford University. Some of his research interests include the link structure of the web, human computer interaction, search engines, scalability of information access interfaces, and personal data mining.

Brin & Page didn't want to leave grad school,

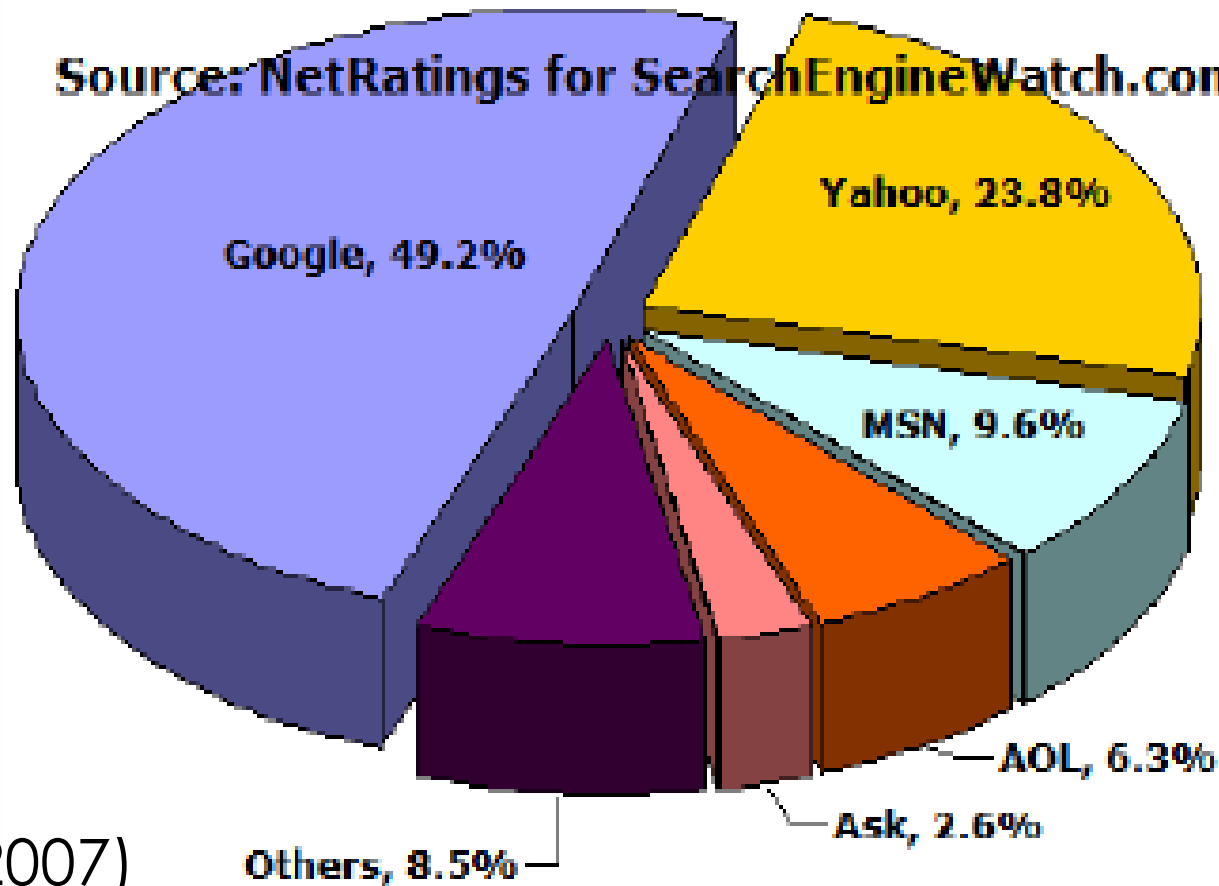
**BUT**

they shopped the idea around for \$1 million  
(including to Yahoo!) and no one bought!

...so they dropped out and started google.



Source: NetRatings for SearchEngineWatch.com



(2007)

## TODAY

Google: 75%

Baidu: 10%

Bing: 8%

Yahoo: 5%

What is pagerank?



Back to the big picture

## Back to the big picture

### Ranking uses

- Frequency of matching query words in the page
- Proximity of matching words to one another
- Location of terms within the page
- Location of terms within tags e.g. <title>, <h1>, link text, Body text
- Anchor text on pages pointing to this one
- Frequency of terms on the page and in general
- Click-through analysis: how often the page is clicked on
- How “fresh” is the page
- Graph structure – how “important” the page is

↑  
pagerank { Depends on heavy-tailed degrees,  
high clustering, small diameter to have  
-- large distinctions in ranks  
-- fast convergence of calculation

## Back to the big picture

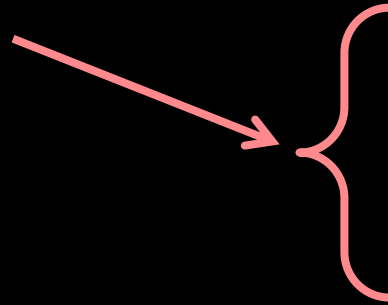
Four components of a search engine

1) Crawling

2) Indexing

3) Ranking

4) Display



Depends on large giant component,  
heavy-tailed degrees, small diameter

## Back to the big picture

Search engine goal: Make money

### Challenges:

- Need lots of users
- Need to serve ads effectively

Give users what  
they want (speed & quality)

Later in course

Data centers!



