The Story of MetaCrawler
June 1995 – Feb. 2001

Erik W. Selberg
University of Washington
February 8, 2001

1

Outline

- Motivation and Genesis
- MetaCrawler implementation
- Hurdles faced at UW
- Commercialization and Scale at Go2Net
- Lessons Learned

2

In Search of a Quals Project
Spring 1994

- **WebCrawler**, moving from UW to AOL
- **Lycos**, moving out of CMU
- **InfoSeek**, charging for more than 10 results
- **Open Text**, some guys out of Canada
- **Yahoo!**, Dave & Jerry's big adventure
- **Excite**, newer engine out of Stanford
- **Galaxy**, yet another spider + cheapo engine

3

Difficulties with Search Engines

- Flaky services
- Poor results
- Incomplete coverage
- Disjoint coverage
- Latency & Connection failures

4

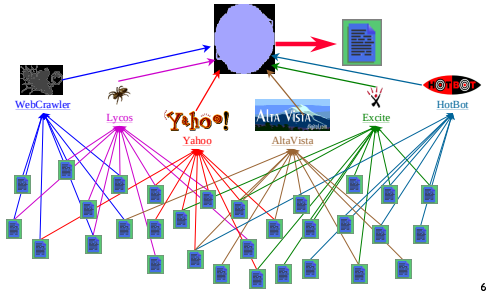
Solution: A Web Search Agent

User says "*What*,"
Agent determines "*How*" and "*Where*"

- Integrate multiple search resources
- Obtain precise and relevant information
- Satisfy real time constraints
- Build customizable interfaces

5

MetaCrawler



6

Search Improvements using MetaCrawler

- Smaller latency
- Better coverage
- Post-processing

Smart, but not ***too*** smart

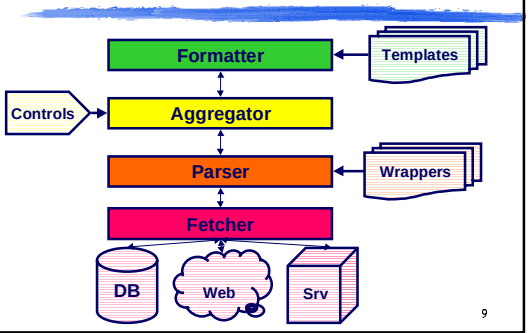
7

MetaCrawler v1.0 – June 1995

- Operational basics:
  - NCSA httpd (precursor to Apache)
  - Perl based CGI interface (back before mod_perl)
  - C/C++ binary
- Hardware:
  - edgar.cs.washington.edu (port 8080)
  - General purpose (spare) DEC Alpha, 133Mhz
  - Purchased Aug. 1993 for $9,066.

8

Architecture



9

MetaCrawler Implementation

- Formatter:
  - Hard-coded HTML
  - Server-push status updates for Netscape
- Aggregator:
  - Phrase search would download and verify documents on demand
  - NDS Algorithm
- Parser:
  - Hard-coded
  - Parameterized wrappers
- Fetcher: pthreads, hacked libwww

10

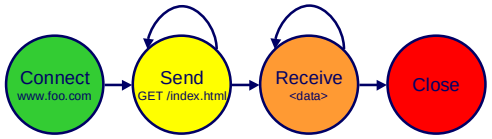
First problem: threads

- Naïve Perl: One process per GET
  - ~15 simultaneous queries on SPARC-2 (SS)
  - Bottlenecks: Context swapping, process limits
- Naïve Java/C++: One thread per GET
  - ~25 simultaneous queries on Alpha 5000 (MC)
  - Bottlenecks: Context swapping, process limits
    - Kills you if all pages are downloaded!
    - Don't forget user-level thread limit!
- Problem: context swapping kills at scale

11

Basic FSM for HTTP

- GET http://www.foo.com/index.html

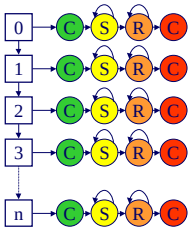


12

FSM Network

- Array of FSMs
- Cell index corresponds to Socket FD
- select() UNIX call
  - accepts array of sockets
  - non-blocking if activity
  - built in time-out

✔ One thread per query
✘ Non standard
✘ Painful to extend



13

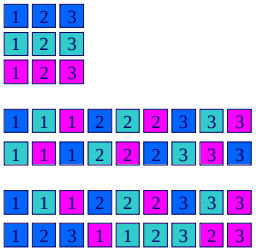
Fetcher Implementation

- FSMs for HTTP, FTP, Gopher, WAIS, DNS
- Connection-oriented protocols (like DNS):
  - DNS: www.metacrawler.com -> 140.142.13.201
  - Global "connect" and "close" state per server
  - Local "connect" and "close" state per request
  - Applicable for other database systems

14

Aggregator Fusion Algorithm

- Goal: merge URLs from sundry sources
- Strict interleave?
  - Problem: how are similar ranks ordered?
  - First arrival bias, random
- Normalized score?
  - Use reported score
  - Problem: score inflation



15

Normalize-Distribute-Sum Algorithm

- s_i is score of Doc i, 1000 is top score

① Normalize engine scores to [0 .. 1000]

② Distribute scores over [0 .. 1000]
  - $s_i' = s_i * (n - i) / n$
  - If n=10, then $s_0' = s_0$, $s_1' = .9 * s_1$, $s_2' = .8 * s_2$, ...

③ Sum scores from duplicate URLs

16

MetaCrawler at UW

- Tried Linux (1.0 kernel) on a 100Mhz Pentium
- MC running on four dedicated Alphas
- C&C ran dedicated fiber channel to Sieg 127
- 7x24x365 NOC provided by C&C (Argus)
  - "If you see this alert, call 543-7798, tell whomever answers to reboot the machine"

17

The Curse of a Good Quals Project

- Got some great early press coverage
  - Forbes cover story, Oct. '95
  - C|Net Best Engine (online only... ☺)
- Started to hit more than 100K queries / day
- Commercial search engines started to notice
  - and got cranky...
- I was doing operations, not doctorate work
- Plan: Commercialize this puppy

18

1996: The Year of the NetBot

- Oren Etzioni, Dan Weld, Bob Doorenboos, Cody Kwok, myself, & a couple other grads / ugrads formed NetBot
- Two main products: MetaCrawler and ShopBot
- What happened: We couldn't figure out how to make money with MetaCrawler.
- Thus, NetBot licensed MetaCrawler to Go2Net

19

How Go2Net makes money with MetaCrawler

- NetBot model: show ads (banners)
  - Couldn't figure out a way to cover costs of other engines
- Go2Net model: not all ads are banners
  - ~15 ads on MC results page
- New model: get search engines to pay
  - Goto.com, FindWhat, etc.

20

Go2Net, Jan 1997 – June 1999
(or, what they did without me)

- Made MetaCrawler engine into Apache module
- Implement real memory management
- Ported MC to Solaris and Linux (2.0 Kernel)
- Hooked it up to the ol' Ad Server
- Broke up MC functionality by machine
  - HTML servers, engine servers, image servers

21

Fixing the Template Problem

- Old: each template was a class
  - Changing output formats required recompilation
  - This is bad when you want to change ads quickly…
  - Also bad if your HTML person doesn't know C++…
- New: Use text file templates
  - Just HTML with special placeholders replaced at rendering time
  - `<a href="##url##">##title##</a>`

22

Fixing the Wrapper Problem

- Old: each wrapper was a class
  - Problems: had to recompile & republish for updates.
- New: generic wrapper engine, params in text files
  - Easy to fix wrappers
  - Just require publishing new text file
  - Easy to generate configs from other wrapper formats (e.g. Sherlock)

23

MetaCrawler on auto-pilot

- Non-standard ads lead to poor creative
- Lou's Law of Click Conservation
- Design began to look dated
- Is it www.metacrawler.com or www.go2net.com?
- ~7 second response time

24

Go2Net, June 1999 - June 2000
(Oren and I come back)

- Update the design
- Get rid of underperforming ad spaces
- Constraints around the ads
- www.metacrawler.com is MetaCrawler
- www.go2net.com is the Portal

25

7s -> 1.5s or bust

- Why was MetaCrawler so slow?
  - Queried ad server after results were returned
  - Ad server was overloaded
  - 5s default timeout for engines to come back
- Get rid of the timeout
- Call the ad server in parallel to requests

26

Quality vs Time

- More engines means more likely to hit timeout
- Return when "Good enough" subset returned?
- Sliding Return Criteria:
  - (< 1s + Gold Standard) OR (1s + Silver Standard)
  - (< 2.5s + Silver Standard) OR (Bronze Standard)
  - 5s

27

The Problems with Ads

- AdServer treated like any other engine
- Put AdServer in Bronze Standard?
  - Ads are requested serially, and 15 per page
  - Ad servers are shared across Go2Net
  - Ad servers are overloaded at peak time
- Don't require AdServer to complete
- Accept partial returns
- Order ad requests by CPM

28

MetaCrawler at Go2Net

- Represents a large chunk of Go2Net high-margin revenue
- Running on ~ 15 Linux boxes
  - 4 Web servers, 11 query boxes
  - Shared Go2Net image servers, ad servers
- ~2M queries / day
  - ~1 box => 100K queries / day
- Google: 7000 Linux boxes, 70M q/day
  - ~1 box => 10K queries / day

29

MetaCrawler at InfoSpace
July 2000 - present

- Both Oren and Erik left Go2Net to pursue other interests
- Go2Net acquired by InfoSpace in July 2000
- Most Go2Net management gone / leaving
- InfoSpace just laid off 250...
  - Search group just lost one

30

Lessons Learned: Construction

- Threads aren't always the right choice
- Efficiency leads to scale
- Do as little as you need to, but do it right
- Code rots when not constantly measured
- Look at the **system**, not just the **binary**

31

Lessons Learned: Commercialization

- Models are not the same as instantiations
- The details matter
- In business, quality and performance are often hard to quantify into dollars

32

Thank you!

- For more information:

Erik Selberg
selberg@cs.washington.edu
http://www.cs.washington.edu/homes/selberg

33