Tài liệu giảng dạy môn Khai thác dữ liệu Web

WEB CRAWLING

TS. Nguyễn Ngọc Thảo – ThS. Lê Ngọc Thành

Bộ môn Khoa học Máy tính, FIT HCMUS, VNUHCM

Thành phố Hồ Chí Minh, 02/2019

Nội dung bài giảng

- Định nghĩa Web Crawler
- Giải thuật Crawler cơ bản
- Các vấn đề thực thi
- Phân loại crawler và đánh giá
- Vấn đề đạo đức và xung đột
- Một số hướng phát triển mới

Web crawler là gì?

- Chương trình máy tính tự động tìm kiếm và tải về trang Web
- **Từ khóa:** ant, automatic indexer, bot, Web spider, Web robot, v.v. hay web scutters (trong cộng đồng FOAF)
- Web là môi trường động và phát triển với tốc độ cực nhanh
  - Crawler giúp cập nhật trang và liên kết được thêm, xóa, di chuyển hay chỉnh sửa.
  - Nếu Web tĩnh → tất cả các trang chỉ cần được tải về một lần và được phân tích cho lần sử dụng sau → không cần Crawler

1st We crawl!

May 27 Update:

41,404,250,804 pages

86,691,236 root domains

41 Billion URLs!!

And 86 million domain names!!

Ứng dụng của Web crawler

- Crawler viếng thăm rất nhiều trang để tập hợp thông tin cho việc xử lý về sau bởi một cỗ máy tìm kiếm.

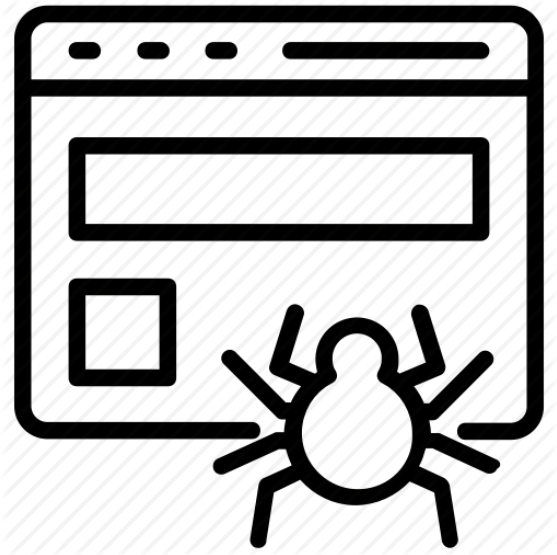
- Các tổ chức kinh doanh sử dụng crawler để tập hợp thông tin về các đối thủ và mối hợp tác tiềm năng.

- Crawler còn được dùng để tự động bảo trì Website.
  - Ví dụ, kiểm tra sự tồn tại của các liên kết và tính hợp lệ của mã HTML.

*Spiders are the only web developers in the world that enjoy finding bugs ^_^*

Ứng dụng của Web crawler

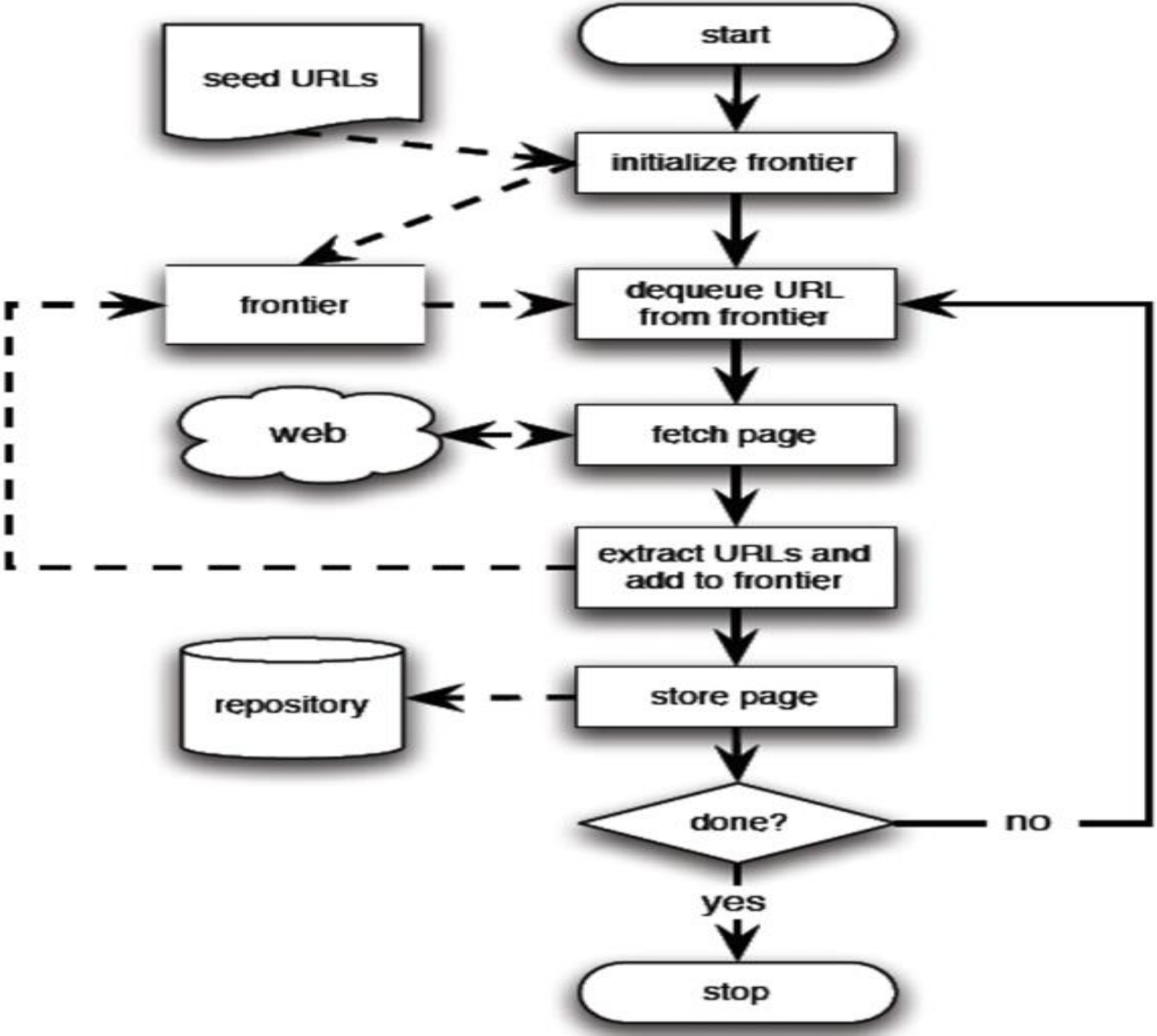
- Crawler cũng có những ứng dụng nguy hiểm.
- Spammer thu thập địa chỉ email/thông tin cá nhân để dùng trong quảng cáo, lừa đảo, tấn công mạng , trộm cắp, v.v.



Giải thuật Crawler cơ bản

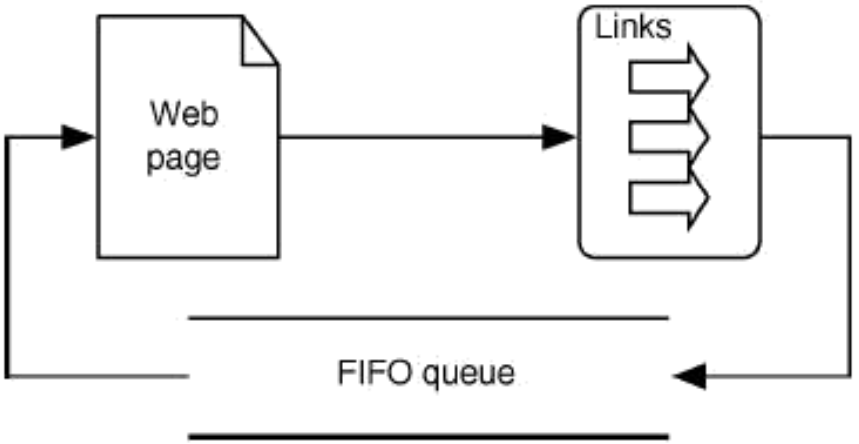
Giải thuật cơ bản

- Giải thuật bắt đầu với một danh sách URL, gọi là các hạt giống (seed)
- Mỗi URL hạt giống lần lượt được viếng thăm để trích xuất các siêu liên kết mới trong mỗi trang
- Các URL mới được thêm vào một danh sách chờ, gọi là bộ định trước (frontier)
- Quá trình lặp lại nhiều lần cho đến khi đạt tiêu chí dừng.


start
seed URLs
initialize frontier
frontier
dequeue URL from frontier
web
fetch page
extract URLs and add to frontier
repository
store page
done?
no
yes
stop

Bộ định trước (frontier)

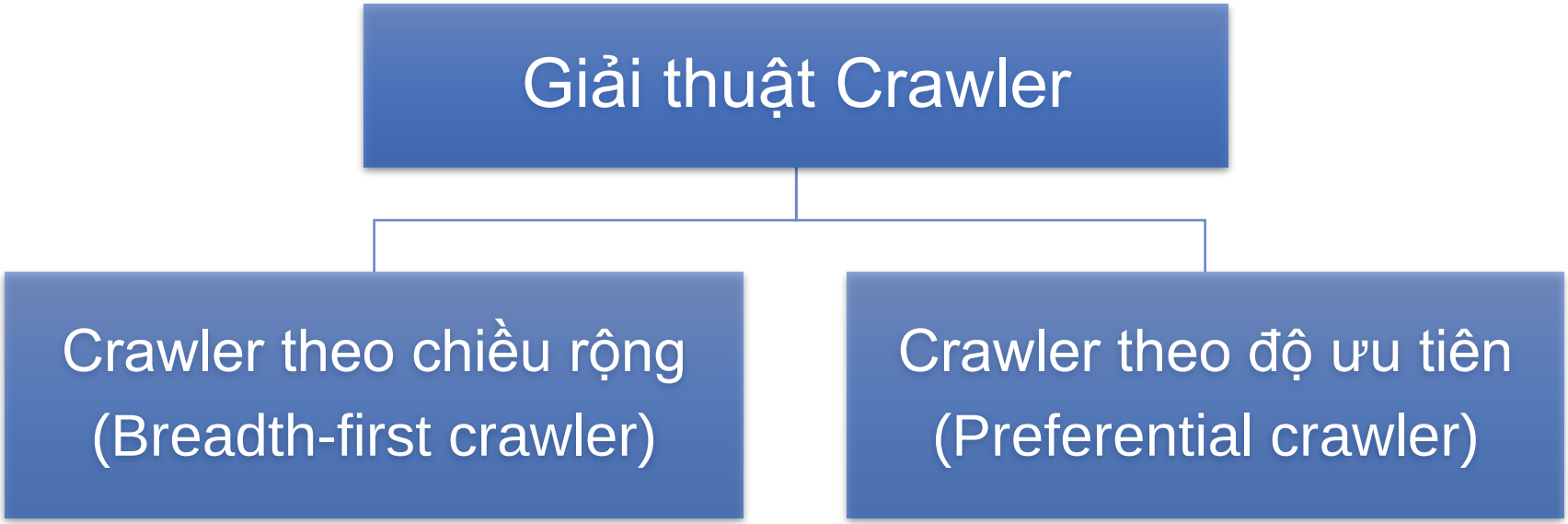
- Cấu trúc dữ liệu (thường là hàng đợi FIFO) dùng để chứa URL của các trang chưa được viếng thăm.
- Nếu frontier đầy, crawler phải quyết định bỏ đi URL nào có độ ưu tiên thấp.
  - Có thể duy trì một bảng băm độc lập để kiểm tra một URL đã có trong bộ định trước hay chưa



Lịch sử viếng thăm (Crawl history)

- Danh sách các URL được gắn kèm thời gian
  - URL đi vào lịch sử viếng thăm chỉ sau khi trang này đã được lấy về.
- Sử dụng để tránh viếng thăm lại những trang đã viếng thăm hay tránh lãng phí không gian trong bộ định trước
- Lịch sử này có thể được sử dụng cho phân tích và đánh giá kết quả crawling.

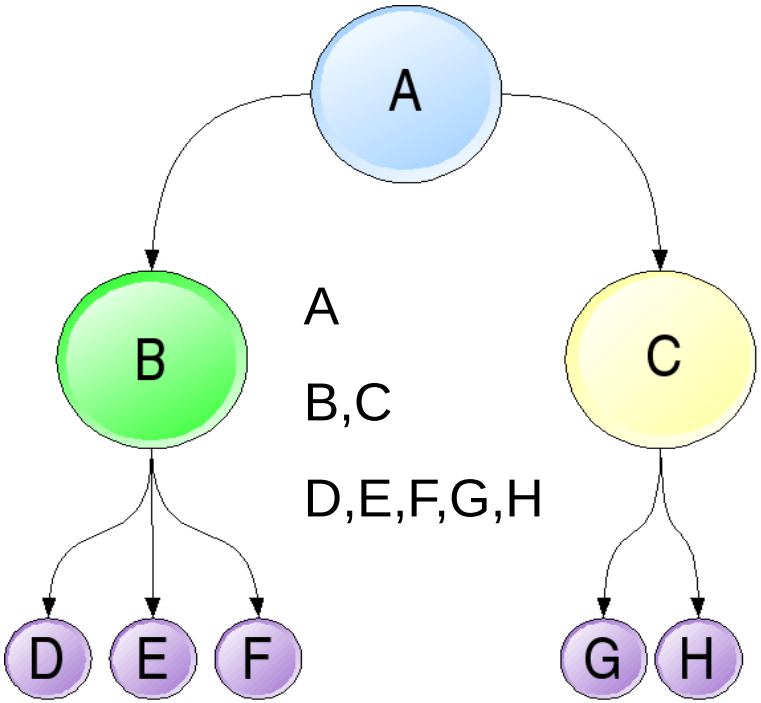
Giải thuật Crawler cơ bản



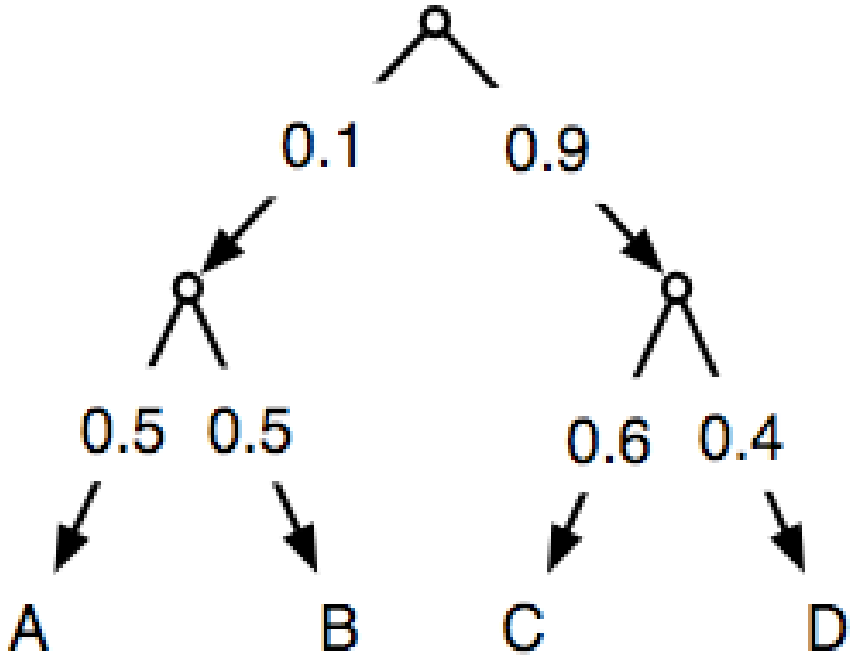
- Crawler theo chiều rộng tải tất cả các trang từ cùng một mức trước khi chuyển sang trang kế tiếp.
- Crawler theo độ ưu tiên gán cho mỗi liên kết chưa được viếng thăm điểm ưu tiên dựa trên ước lượng giá trị nào đó.
  - Bộ định trước được thực thi như hàng đợi ưu tiên (priority queue)

Minh họa giải thuật Crawler

Crawler theo chiều rộng

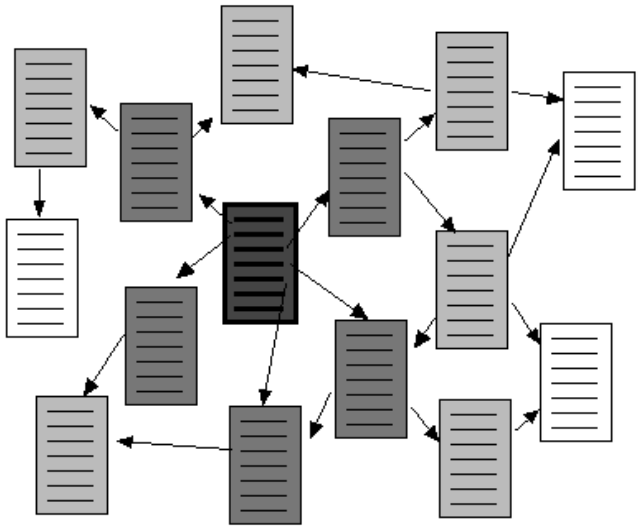


Crawler theo độ ưu tiên



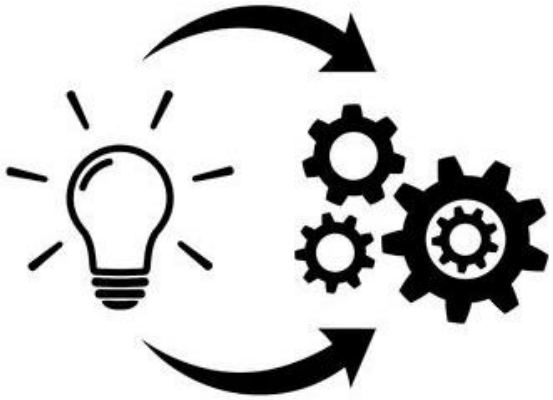
Crawler theo chiều rộng

- Các trang phổ biến thường có nhiều liên kết đến → thu hút các crawler theo chiều rộng
- Thứ tự trang được viếng thăm bởi crawler theo chiều rộng liên quan với giá trị bậc trong/Page Rank của chúng.
- Topical crawler: Trang thuộc các liên kết hàng xóm của trang hạt giống liên quan nhiều đến bản than hạt giống hơn là trang được chọn ngẫu nhiên



Crawler theo độ ưu tiên

- Ước lượng để xác định độ ưu tiên có thể được dựa trên thuộc tính của đồ hình, thuộc tính nội dung, hay bất kì sự liên kết đặc trưng nào khác có thể đo đạt.
  - Ví dụ, bậc trong của trang hay sự giống nhau giữa truy vấn và trang
- Best-first crawler: Các trang được viếng thăm theo thứ tự ưu tiên trong bộ định trước



Vấn đề thực thi

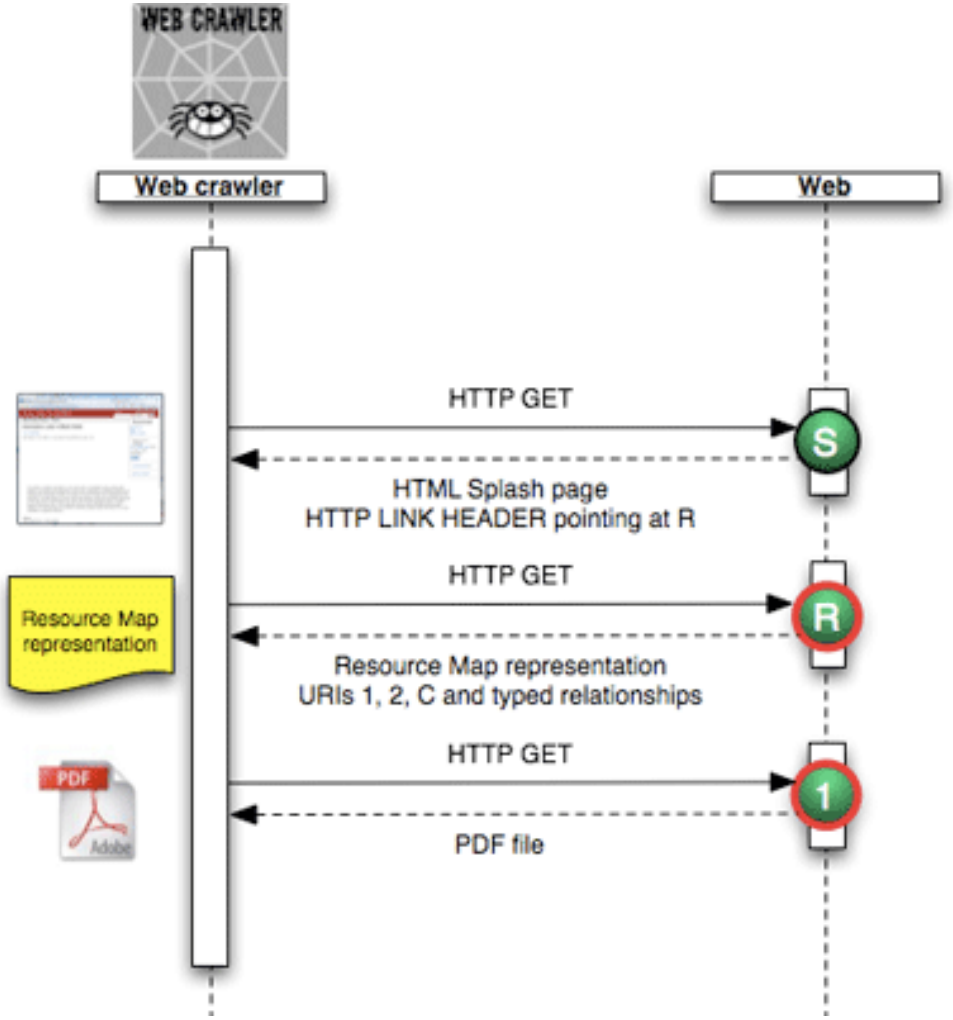
Vấn đề thực thi

- Crawler có thể gặp phải nhiều vấn đề khi triển khai thực tế.
  1. Vấn đề lấy các trang về (fetching)
  2. Vấn đề phân tích các trang (parsing)
  3. Vấn đề loại bỏ stopword và stemming
  4. Trích xuất liên kết và chuẩn quy tắc
  5. Bẫy nhện (spider trap)
  6. Vùng chứa trang
  7. Tính song song

Vấn đề lấy các trang về

1

- Crawler hành động như một Web client, gửi một HTTP request đến máy chủ chứa trang và phân tích các hồi đáp.



Vấn đề lấy các trang về

1

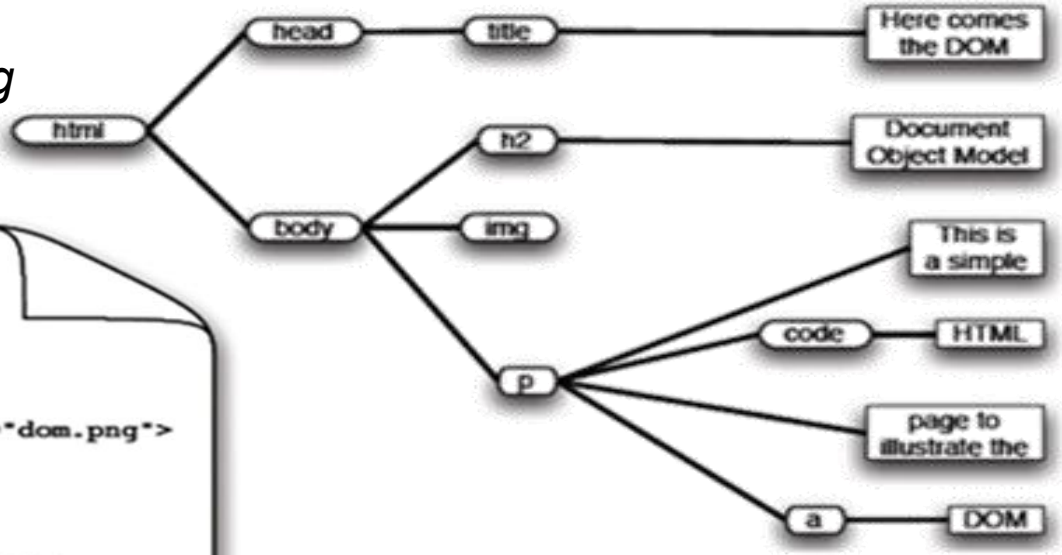
- Crawler phân tích header để lấy mã trạng thái và redirection.
- Crawler cần kết nối timeout để tránh lãng phí thời gian đợi hồi đáp từ máy chủ chậm hay đọc các trang dữ liệu lớn.
  - Thống kê timeout và mã trạng thái có thể được dùng để xác định vấn đề hay tự động điều chỉnh giá trị timeout
- Crawler phân tích và lưu header được điều chỉnh lần cuối để xác định tuổi của trang → thông tin có thể không đủ tin cậy
- Kiểm tra lỗi và bắt các ngoại lệ cũng quan trọng trong suốt tiến trình lấy các trang.
  - Tình trạng mã giống nhau tiềm ẩn ở hàng triệu server.

Vấn đề phân tích trang

- Crawler phân tích nội dung của một trang đã tải về.
- Trích xuất thông tin để hỗ trợ ứng dụng chính của crawler (ví dụ, đánh chỉ mục trang)
- Trích xuất các liên kết trong trang và đưa vào bộ định trước

Minh họa cây DOM được xây dựng từ trang HTML

```
<html>
  <head>
    <title>Here comes the DOM</title>
  </head>
  <body>
    <h2>Document Object Model</h2>
    <img align="right" alt="dom pict" src="dom.png">
    <p>
      This is a simple
      <code>HTML</code>
      page to illustrate the
      <a href="http://www.w3.org/DOM/">DOM</a>
    </p>
  </body>
</html>
```



Vấn đề phân tích trang

2

- Không thể đảm bảo trang được đăng ở trạng thái hoàn hảo
  - Các tag cần thiết bị thiếu, có tag mở mà không có tag đóng, tag được đặt lồng nhau không thích hợp, v.v.
  - Tên và giá trị bị thiếu hay sai lỗi chính tả, thiếu dấu nháy xung quanh các giá trị thuộc tính, v.v.
  - Các kí tự đặc biệt không được đánh dấu

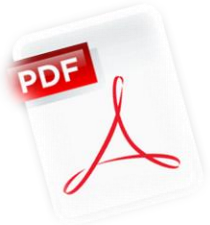
```
XML Parsing Error: not well-formed
Location: http://www.blog.web6.org/sitemap.xml
Line Number 2955, Column 13:

                <priority       <url>
--------------------------------^
```

Vấn đề phân tích trang

- Ngoài định dạng HTML, trang Web còn ở rất nhiều định dạng mở và độc quyền.
  - Ví dụ, văn bản thuần (plain text), PDF, Microsoft Word, Microsoft PowerPoint, Flash …
- Các chuẩn mới đang ngày một phổ biến.
  - Ví dụ, đồ họa vector co giãn (SVG), mã Javascript bất đồng bộ, XML (AJAX) và các ngôn ngữ dựa trên XML.

Gây khó khăn cho Crawler khi phân tích liên kết hay nội dung nguyên bản.



Loại bỏ stopword và stemming 3

- Viêc tính điểm cho các URL mới được rút ra từ một trang thường đòi hỏi loại bỏ stopword và stemming.
- Stopword thường gây trở ngại cho việc suy xét các trang dựa trên nội dung.
- Một số crawler tính điểm liên kết dựa trên sự giống nhau giữa trang nguồn và truy vấn → stemming giúp cải thiện độ trùng khớp giữa hai tập và độ chính xác của hàm tính điểm.

Trích xuất liên kết

4

- Để trích các siêu liên kết từ một trang, ta sử dụng một bộ phân tích để tìm ra các tag anchor (<a>) và lấy giá trị của thuộc tính href tương ứng.

HTML Code:

If you would like a free canvas portrait you can Click Here to Order from Canvas People and create one in just 5 minutes!

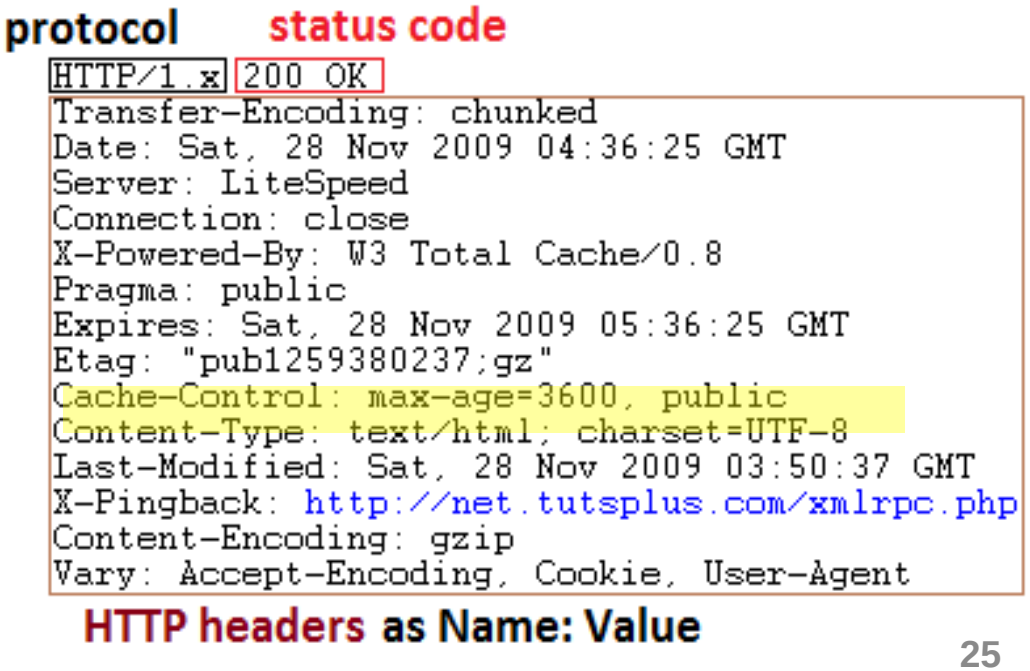
Result:

If you would like a free canvas portrait you can Click Here to Order from Canvas People and create one in just 5 minutes!

Trích xuất liên kết

- **Vấn đề 1:** Một số loại URL không thể phân tích và phải bỏ đi
  - Danh sách trắng (ví dụ, text/html) vs. danh sách đen (ví dụ, pdf)
  - Việc xác định loại tập tin có thể dựa vào phần mở rộng → thường ít tin cậy, đôi khi thiếu hoàn toàn.
  - Không thể đợi tải một tài liệu rồi mới quyết định nó có cần hay không

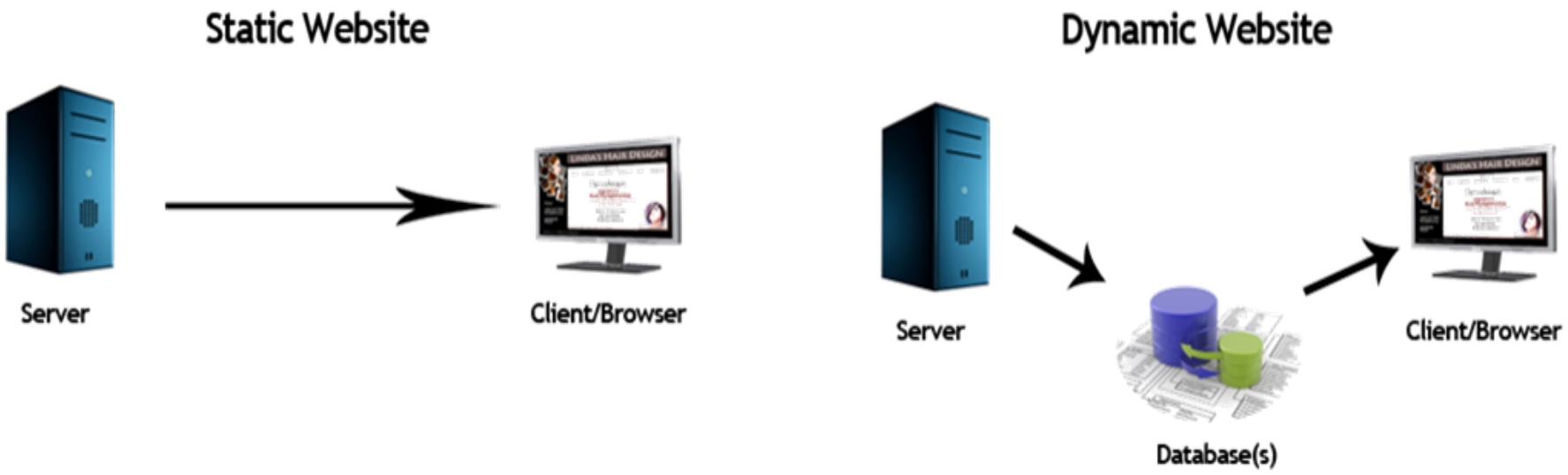
- **Giải pháp:** Crawler gửi yêu cầu HTTP HEAD và theo dõi header trả về loại nội dung → thường là các nhãn đủ tin cậy



Trích xuất liên kết

4

- **Vấn đề 2:** bộ lọc phải xét tính động của trang được liên kết.
  - Một trang động (ví dụ, phát sinh bởi mã CGI) có thể là giao diện truy vấn cơ sở dữ liệu hay ứng dụng khác mà crawler không quan tâm.
- Tính động rất khó để nhận ra khi xem xét chuỗi URL.



Chuẩn quy tắc

4

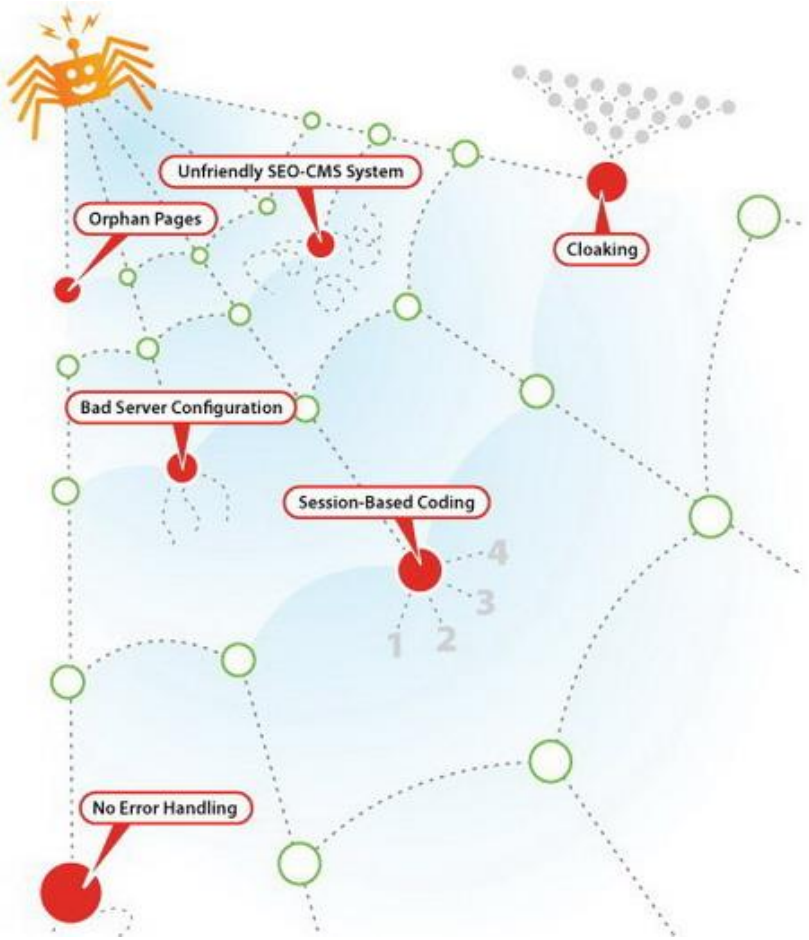
- URL tương đối phải chuyển đổi thành URL tuyệt đối trước khi được thêm vào bộ định trước.
  - Ví dụ, URL tương đối news/today.htm nằm trong trang http://www.somehost.com/index.html được chuyển thành thể tuyệt đối http://www.somehost.com/news/today.html

Chuẩn quy tắc

4

Description and transformation	Example and canonical form
Default port number Remove	http://cs.indiana.edu:80/ http://cs.indiana.edu/
Root directory Add trailing slash	http://cs.indiana.edu http://cs.indiana.edu/
Guessed directory* Add trailing slash	http://cs.indiana.edu/People http://cs.indiana.edu/People/
Fragment Remove	http://cs.indiana.edu/faq.html#3 http://cs.indiana.edu/faq.html
Current or parent directory Resolve path	http://cs.indiana.edu/a/./../b/ http://cs.indiana.edu/b/
Default filename* Remove	http://cs.indiana.edu/index.html http://cs.indiana.edu/
Needlessly encoded characters Decode	http://cs.indiana.edu/%7Efil/ http://cs.indiana.edu/~fil/
Disallowed characters Encode	http://cs.indiana.edu/My File.htm http://cs.indiana.edu/My%20File.htm
Mixed/upper-case host names Lower-case	http://CS.INDIANA.EDU/People/ http://cs.indiana.edu/People/

Bẫy nhện (spider trap)

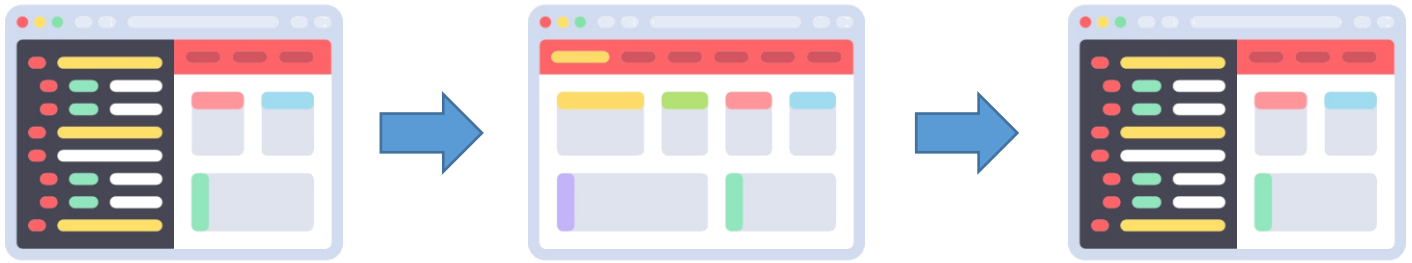


- Crawler có thể rơi vào bẫy nhện trong quá trình đi theo các siêu liên kết trong một trang.
- **Bẫy nhện** có thể là các đoạn mã được nhúng trong URL mà từ đó phát sinh động một số lượng lớn các URL chỉ đến cùng một trang.

Bẫy nhện (spider trap)

5

- URL động là một dạng của bẫy nhện.
  - Địa chỉ chứa dấu "?" cho phép nhúng nội dung truy vấn trong URL, ví dụ, www.google.com/search?q=dynamic+urls
- URL có thể được tạo động dựa trên tuần tự các hành động gây ra bởi người duyệt (hay crawler).
  - Ví dụ, Amazon.com sử dụng URL để mã hóa chuỗi sản phẩm người dùng đã xem, mỗi lần người dùng nhấp vào một liên kết, server ghi nhận lại thông tin chi tiết cách hành xử mua sắm.



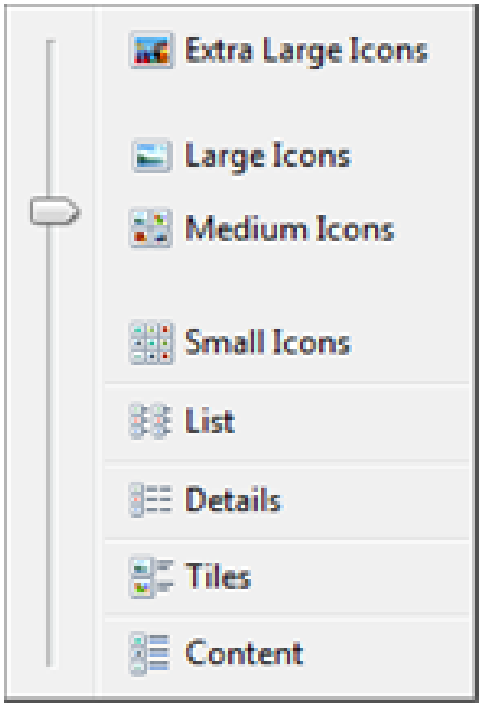
Page A

Page B

Same page A, but with a different URL

Bẫy nhện (spider trap): Ví dụ 5

- Một trang hiển thị hình ảnh trực tuyến đơn giản cung cấp tùy chọn đến người dùng qua tham số HTTP GET trong URL.
- Giả sử người dùng được lựa chọn 4 cách để sắp xếp hình, 3 loại kích thước thu nhỏ, 2 định dạng tập tin và chức năng lọc nội dung.



- Tập nội dung giống nhau này có thể được truy xuất với 48 URL khác nhau, tất cả chúng có thể được liên kết trên site.

Bẫy nhện (spider trap)

5

- Làm cho trang xuất hiện vô tận đối với một crawler.
  - Càng nhiều liên kết được theo, càng nhiều URL mới được tạo.
  - Tuy nhiên những liên kết giả mới này không dẫn đến nội dung mới.

```
http://foo.com/bar/foo/bar/foo/bar/foo/...
```

- Crawler khi mắc kẹt trong bẫy nhện có thể điền đầy cơ sở dữ liệu phía máy chủ với các phần tử giả.
  - Máy chủ bị lãng phí băng thông và dung lượng lưu trữ.
  - Cơ sở dữ liệu có thể bị đầy → website bị vô hiệu hóa do crawler vô tình tạo ra cuộc tấn công từ chối dịch vụ (DDOS).
- **Giải pháp:** giới hạn kích thước URL (ví dụ, 256 ký tự) hoặc giới hạn số trang được lấy từ một domain

Vấn đề vùng chứa trang

6

- Giả sử mỗi trang được lưu trữ thành một tập tin trên đĩa.
- Kích thước tập tin thường nhỏ → crawler quy mô lớn sẽ tốn thời gian đáng kể và không gian đĩa để quản lý một số lượng rất lớn các tập tin nhỏ

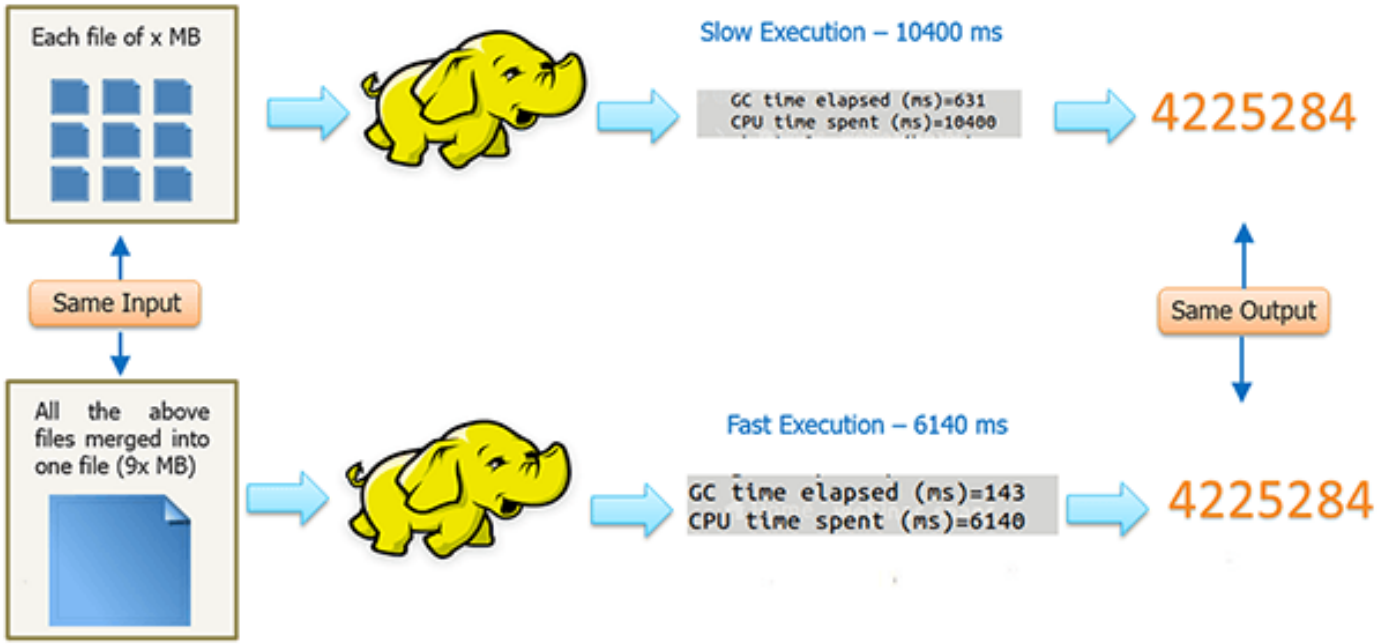
THỜI GIAN NHẢY ĐĨA

SỐ LƯỢNG FILE TỐI ĐA MỘT HĐH QUẢN LÝ

THÔNG TIN LƯU MỖI FILE

Vấn đề vùng chứa trang

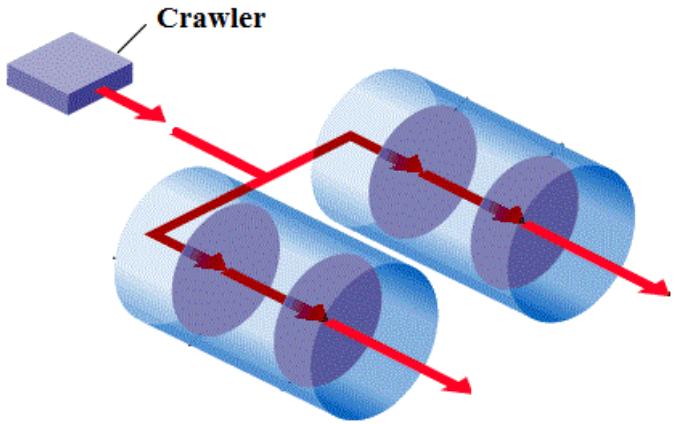
- **Giải pháp 1:** tập hợp nhiều trang (~ 1000) thành một tập tin đơn, quy ước dấu hiệu đặc biệt xác định ranh giới các trang.

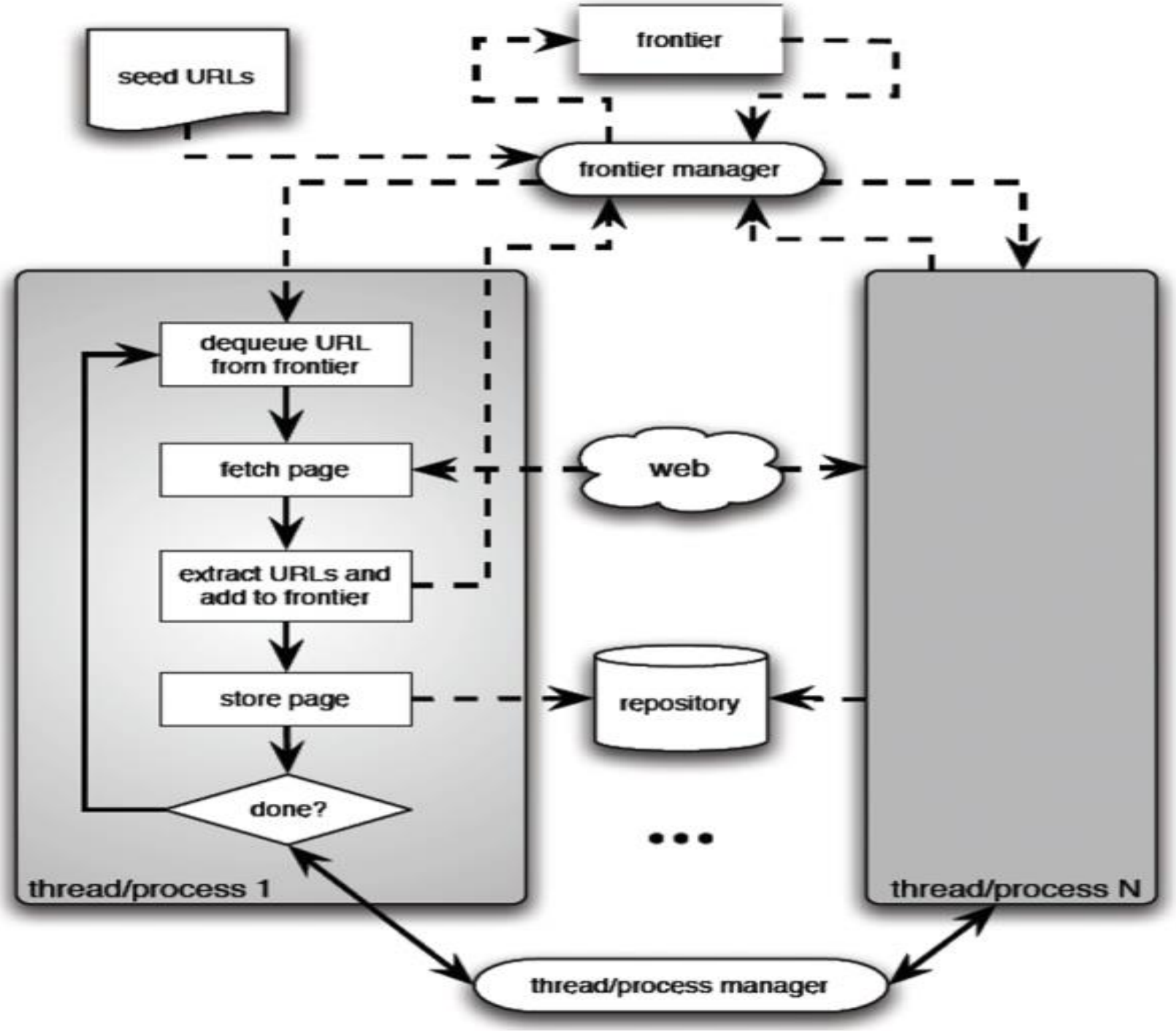


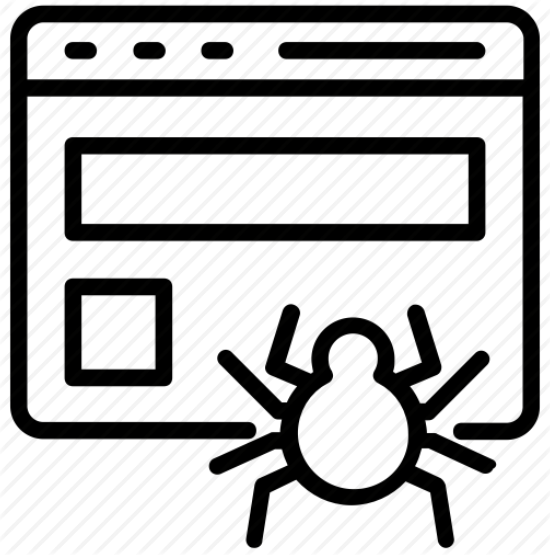
- **Giải pháp 2:** sử dụng hệ quản trị cơ sở dữ liệu để chứa và đánh chỉ mục.

Vấn đề tính song song

- Crawler sử dụng ba tài nguyên chính: mạng, CPU và đĩa
  → có thể gặp vấn đề "thắt cổ chai" (bottleneck)
- Crawler tuần tự có thể không tận dụng hiệu quả tài nguyên
  - Vào bất kì thời điểm nào, hai trong ba tài nguyên sẽ rảnh.
- **Giải pháp:** xử lý song song với đa luồng (multithreading) hay đa tiến trình (multiproccessing)
  - Phải xét tính đồng bộ dữ liệu (khóa và bỏ khóa truy xuất frontier, v.v.)




seed URLs
frontier
frontier manager
dequeue URL from frontier
fetch page
extract URLs and add to frontier
store page
done?
web
repository
thread/process 1
thread/process N
thread/process manager



Phân loại Crawler

Crawler phổ dụng

- Crawler phổ dụng (universal crawler) là một dạng crawler theo chiều rộng có xử lý song song.
- Khả năng của crawler được nhấn mạnh ở các khía cạnh
  - **Hiệu quả thực thi:** phân tích và xử lý hàng ngàn trang mỗi giây
  - **Chinh sách:** bao phủ càng nhiều trang quan trọng trên Web càng tốt, đồng thời phải luôn làm mới chỉ mục.
- Tức là crawler phổ dụng cần giải quyết các bài toán
  - Khả năng mở rộng (scalability)
  - Độ bao phủ (coverage)
  - Tính cập nhật (freshness)

Crawler phổ dụng: Khả năng mở rộng

- Yêu cầu cải tiến kiến trúc để đáp ứng quy mô dữ liệu và nhu cầu tính toán ngày càng phát triển
- Một số giải pháp được đề xuất
  - Mô hình song song sử dụng socket bất đồng bộ (quan trọng nhất)
  - Quản lý bộ định trước bằng nhiều hàng đợi
  - Phân giải tên miền bằng giao thức truyền tải UDP, gửi phân tích tên miền trước khi vào bộ định trước.
  - Đa kết nối mạng đến nhiều ISP, đa nhập xuất

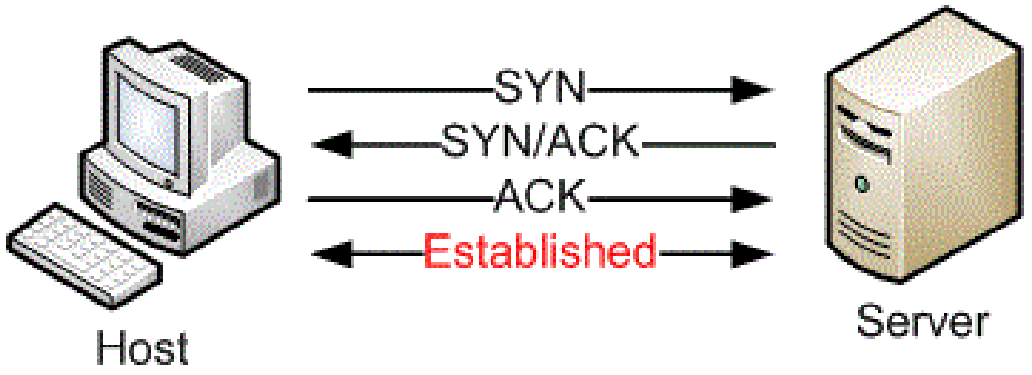
Crawler phổ dụng: Khả năng mở rộng

- Socket bất đồng bộ cho phép mỗi tiến trình hay luồng có thể giữ hàng ngàn các kết nối mạng mở một cách đồng thời.
  - Tận dụng tốt băng thông mạng
  - Loại bỏ sự xung đột các tài nguyên và nhu cầu khóa.

Crawler phổ dụng: Khả năng mở rộng

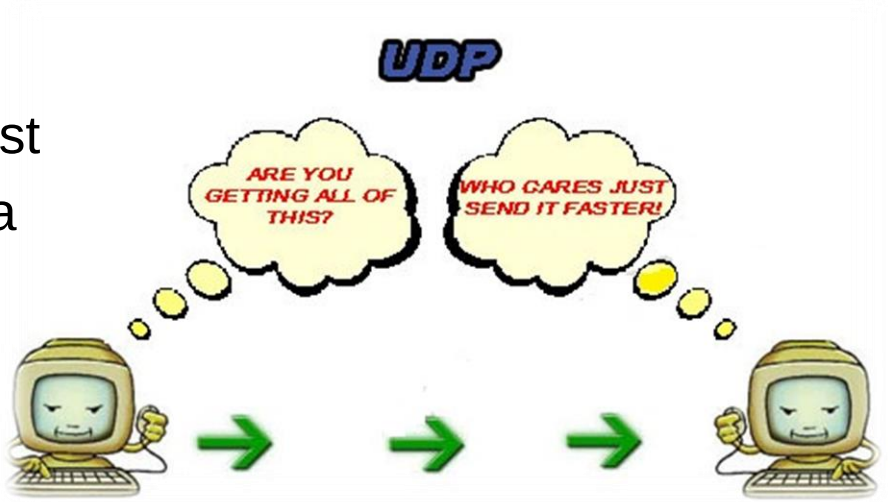
- Quản lí bộ định trước bằng nhiều hàng đợi song song, trong đó các URL ở mỗi hàng đợi chỉ đến một server duy nhất.
  - Cho phép duy trì các kết nối đến các server, do vậy tối thiểu hóa sự quá tải do các bắt tay mở và đóng TCP.

TCP Three-Step Handshake



Crawler phổ dụng: Khả năng mở rộng

- Sử dụng UDP thay vì TCP cho DNS request do tốc độ nhanh hơn.
  - UDP không đảm bảo và request có thể bị mất nhưng ít khi xảy ra

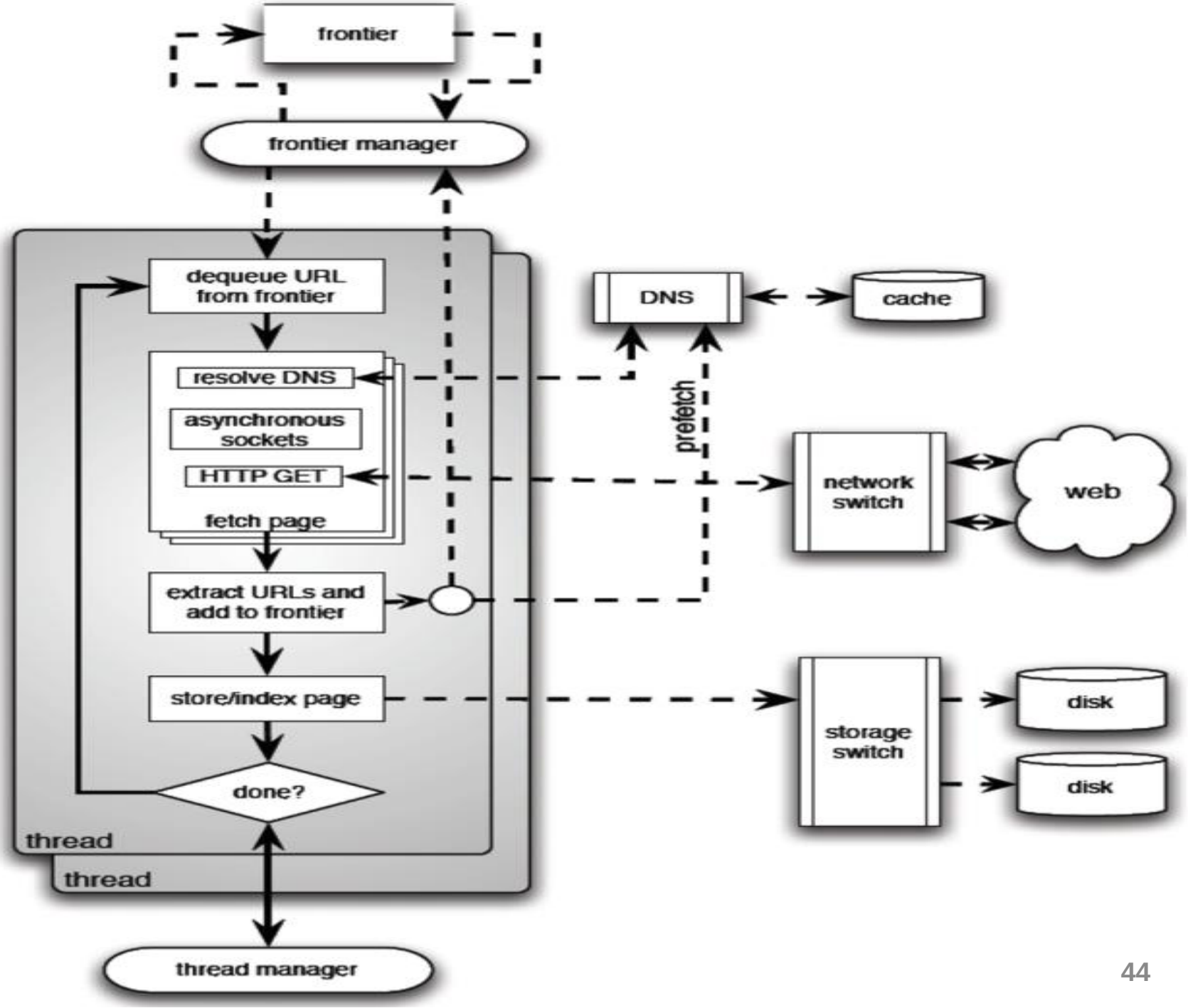


- Gửi DNS request khi một URL vừa mới được trích ra và trước khi vào bộ định trước.
  - Khi một URL được xử lý, địa chỉ IP host đã được tìm và lưu trữ trong cache của DNS trước đó.

Crawler phổ dụng: Khả năng mở rộng

- Tăng cường băng thông mạng bằng cách sử dụng đa kết nối mạng chuyển đến đa router
  - Tối ưu hóa các mạng của nhiều nhà cung cấp dịch vụ Internet (ISP)

- Quá trình nhập xuất đĩa có thể được cải thiện thông qua một mạng lưới vùng lưu trữ sử dụng truyền tải cáp quang.


frontier
frontier manager
dequeue URL from frontier
resolve DNS
asynchronous sockets
HTTP GET
fetch page
extract URLs and add to frontier
store/index page
done?
thread
thread
thread manager
DNS
cache
prefetch
network switch
web
storage switch
disk
disk

Crawler phổ dụng: Độ bao phủ

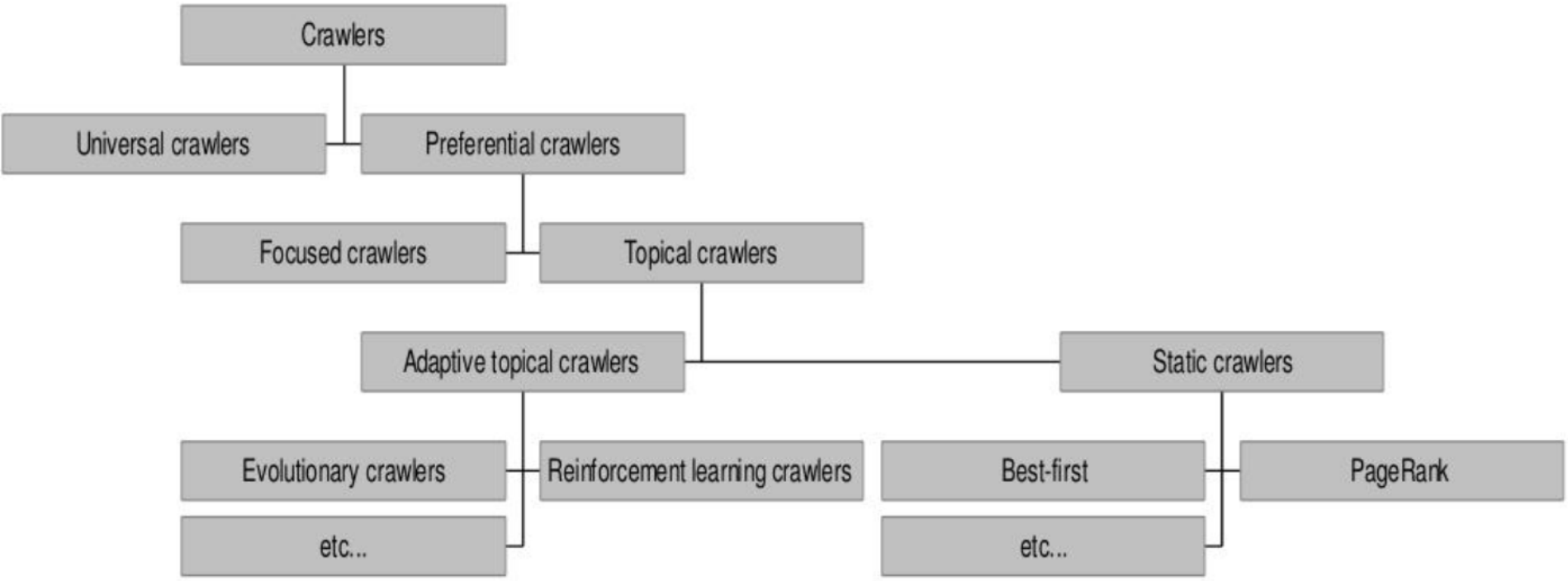
- Kích thước Web vô cùng khổng lồ
- Việc đánh chỉ mục tất cả các nội dung có thể truy xuất được cũng không khả thi thậm chí đối với các crawler loại to.
  - Ví dụ, Google, Yahoo, hay Bing đánh chỉ mục khoảng 10^{10} (~10 tỷ) trang (2011).
- Các cỗ máy tìm kiếm thường hướng đến trang quan trọng
  - Tính quan trọng dựa trên rất nhiều nhân tố như độ đo tính phổ biến của liên kết (bậc trong hay PageRank).

Crawler phổ dụng: Tính cập nhật

- Các trang được thêm, xóa và sửa thường xuyên.
  - Các trang mới được tạo ở tỉ lệ khoảng 8% mỗi tuần, trong đó chỉ khoảng 62% nội dung thực sự mới vì các trang thường sao chép từ một trang đang có.
  - Cấu trúc liên kết của Web có nhiều tính động, với khoảng 25% liên kết mới được tạo mỗi tuần. Hầu hết các thay đổi trên Web do việc thêm và xóa nhiều hơn là điều chỉnh.
- Crawler cần viếng thăm lại các trang đã đánh chỉ mục để giữ cho chỉ mục luôn được cập nhật.

Crawler phổ dụng: Tính cập nhật

- Các chiến lược viếng thăm lại
  - Theo dõi các trang dựa trên tần số thay đổi (frequency of change)
  - Theo dõi các trang dựa trên bậc của thay đổi (degree of change)
- Do tính bao phủ và tính mới mâu thuẫn nhau nên cần phải có chiến lược cân bằng tốt giữa hai mục tiêu này.


Crawlers
Universal crawlers
Preferential crawlers
Focused crawlers
Topical crawlers
Adaptive topical crawlers
Static crawlers
Evolutionary crawlers
Reinforcement learning crawlers
Best-first
PageRank
etc...
etc...

Preferential crawlers

- Không crawl toàn bộ web (URL) mà chỉ theo một những trang “quan tâm”
  - Trang từ .jp domain
  - Trang về thể thao
  - Trang về bóng đá
  - Trang có PageRank lớn.
- Nếu có công thức để ước lượng độ quan trọng của mỗi trang I(p) => crawl theo thứ tự I(p) giảm dần
- => ***Priority Queue*** được áp dụng cho frontier dựa vào I(p)

Preferential crawlers

- Tiêu chí đánh giá độ ưu tiên của trang:
  - Liên quan tới chủ đề chính
  - Gần trang seed
  - Độ phổ biến/Pagerank
  - Tần số cập nhật lớn
  - …

Preferential crawlers

- Focused Crawler:
  - Đã có dữ liệu mẫu – Supervise learning
- Topical Crawler:
  - Chưa có dữ liệu mẫu – Supervise learning

Focused Crawler

- Chỉ crawl những page thuộc những chủ đề nhất định.
- Cần dùng một ***classifier*** để chọn trang phù hợp
  - Naïve-bayes
  - SVM
  - Neural networks
- Dựa trên Anchor text (URL text) trước khi thực sự download page.

Ví dụ Focused Crawlers

- Dùng Bayes để phân lớp chủ đề cho trang p
- $Pr(c|p)$:xác suất trang p thuộc chủ đề c
- Điểm cho trang p được tính:

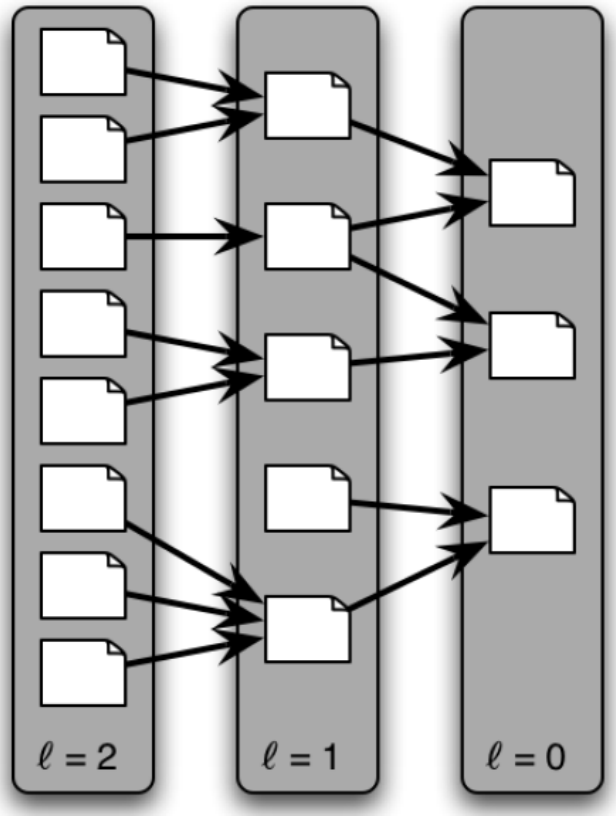
$$R(p) = \sum_{c \in c^*} \Pr(c \mid p).$$

- **Soft focus** hoặc **hard focus** được sử dụng để thêm URLs từ p vào frontier

Ví dụ Focused Crawlers

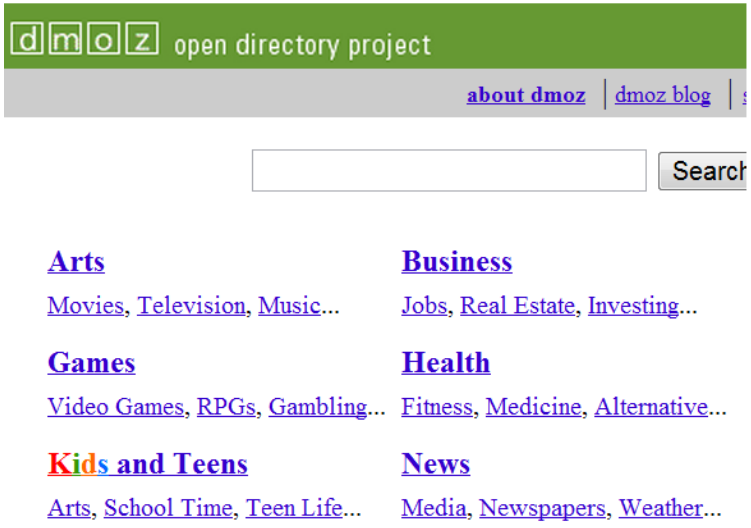
Context-Focused Crawler (CFC):

- Cùng ý tưởng phân lớp/dự đoán như Focused Crawlers
- Nhưng thay vì dự đoán chủ đề, CFC dự đoán khoảng cách link
  - L0: chủ đề quan tâm.
  - L1: link đến chủ đề quan tâm
  - L2: …
- Độ ưu tiên trong frontier dựa trên khoảng cách ước lượng từ mục tiêu
- Tốt hơn focused crawler thông thường

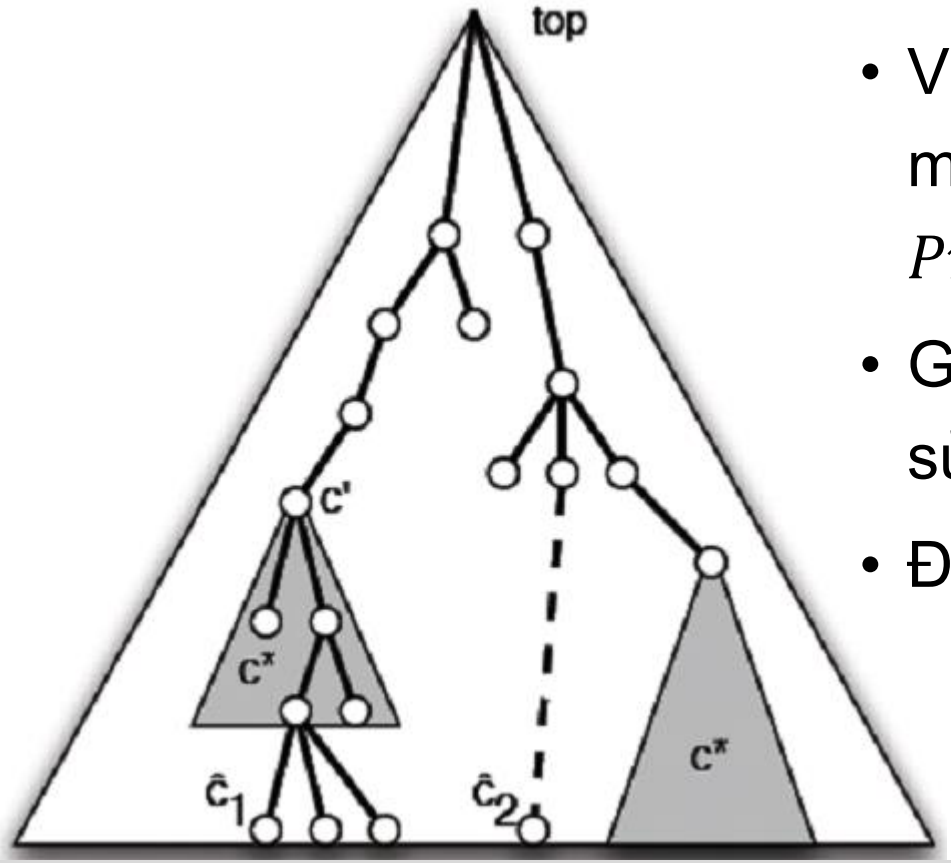


Crawler tập trung

- Crawler tập trung (focused crawler) là một dạng crawler theo độ ưu tiên, hướng đến việc thu thập các trang trong từng phân loại cụ thể mà người dùng quan tâm.
- Chakrabarti và cộng sự đề xuất một crawler tập trung dựa trên bộ phân lớp (classifier).
  - Sử dụng mẫu học là trang được đánh nhãn rút ra từ nhiều loại trong thư viện phân loại (ví dụ, ODP)
  - Crawler ưu tiên được định hướng chọn từ bộ định trước những trang thuộc về loại được quan tâm.



Crawler tập trung



Một cây phân loại cho Crawler tập trung. Khu vực xám thể hiện loại được quan tâm c. Các liên kết thuộc về nhóm c* này và nhóm con $\hat{c}_1$ được thêm vào bộ định trước, nhóm $\hat{c}_2$ sẽ bỏ qua hoặc có thứ hạng thấp hơn.*

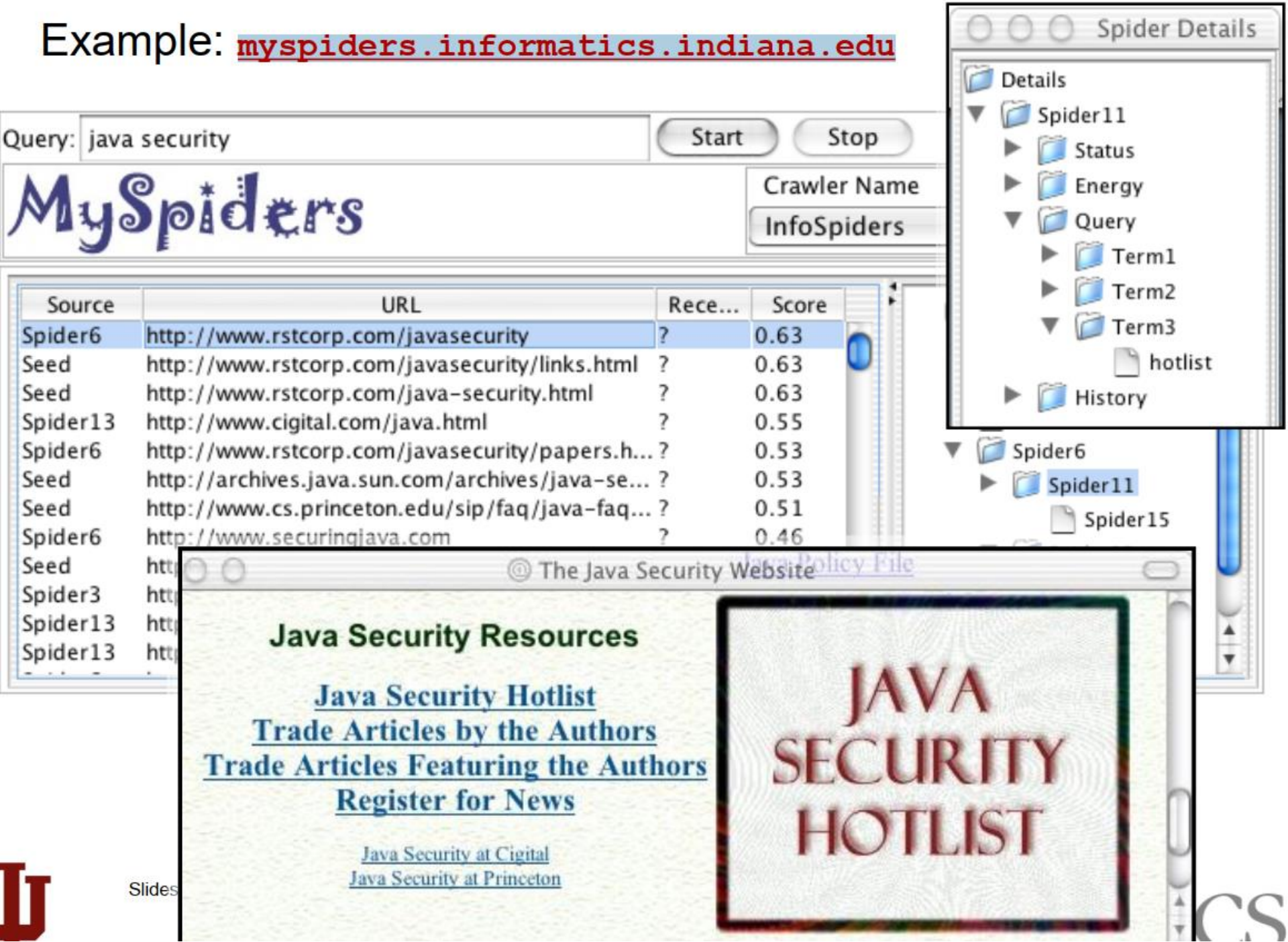
- Với mỗi loại c, tính xác suất một trang p thuộc về loại c $Pr(c|p)$ (với $Pr(top|p) = 1$)
- Gọi c^* là nhóm các loại người sử dụng quan tâm.
- Điểm của trang được tính là

$$R(p) = \sum_{c \in c*} Pr(c|p)$$

Crawler chủ đề

- Các mẫu được đánh nhãn sẵn thường không có hoặc không đủ để huấn luyện mô hình phân lớp.
- Crawler chủ đề (topical crawler) là một dạng crawler theo độ ưu tiên, bắt đầu với một tập nhỏ trang hạt giống kèm theo mô tả về chủ đề đang quan tâm (có thể ở dạng câu truy vấn)

Example: myspiders.informatics.indiana.edu



Ví dụ về Topical crawlers

- Naïve Best-First
  - Dựa theo mức độ tương đồng về nội dung
- Best-N-First
  - Tương tự Naïve Best-First, nhưng lấy N trang cùng lúc

Adaptive Topical Crawlers

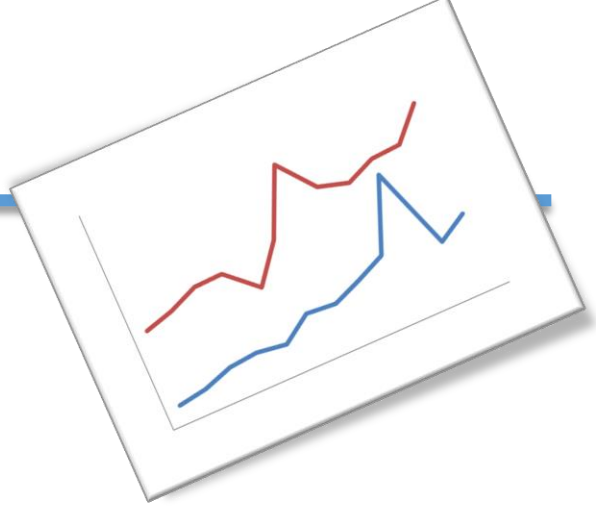
- Có khả năng thay đổi cách xếp hạng dựa vào quá trình crawl thực tế.

- Việc học tăng cường giúp cho crawler tương thích tốt với quá trình real-time

Crawler chủ đề

- Crawler chọn N trang (xếp theo độ ưu tiên) ở một thời điểm từ bộ định trước và lấy tất cả chúng về.
- Danh sách sắp xếp trang được ràng buộc bởi người dùng theo điểm số tương tự hay tính mới.
  - Điểm số tương tự có thể là giá trị cosine giữa nội dung của một trang và nội dung truy vấn.
  - Tính mới được ước lượng bởi lần điều chỉnh head cuối.
- Khi tất cả N trang được viếng thăm, các URL mới trích ra được sắp xếp trộn vào trong hàng đợi ưu tiên.
- Chu trình tương tự được lặp lại.

Đánh giá crawler



- So sánh giữa các crawler là vấn đề phức tạp vì còn phụ thuộc vào từng mục tiêu của mỗi crawler.
- Một số cách đánh giá cơ bản
  - Nếu khi thu thập, trong 500 trang đầu tiên có 100 trang liên quan được tìm thấy thì tỉ lệ là 20%
  - Đánh giá dựa trên chiều dài tìm kiếm, nghĩa là số trang được thu thập trước khi phần trăm nào đó các trang liên quan được tìm thấy.

Vấn đề đạo đức và xung đột

Vấn đề server quá tải

- Crawler gửi liên tiếp nhiều yêu cầu trang đến server với tốc độ nhanh có thể gây sự quá tải cho server và dễ dẫn đến một cuộc tấn công từ chối dịch vụ.

- Crawler cần xác định tỉ lệ cực đại các yêu cầu gửi đến server, hoặc phân phối chúng đến nhiều server

Thông tin crawler và chặn crawler

- Crawler sử dụng header HTTP User-Agent để khai báo với phía server về nhân dạng của mình
  - Thông tin có thể gồm tên, phiên bản, nguồn gốc, email liên lạc, v.v.

```
Host: www.[redacted].com
Accept-Language: en-US
Cache-Control: no-cache
Connection: keep-alive
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
From: googlebot(at)googlebot.com
User-Agent: Mozilla/5.0 (compatible; Googlebot/2.1; +http://www.google.com/bot.html)
Accept-Encoding: gzip,deflate,br
```

Vấn đề đạo đức và xung đột

- Crawler sẽ gây ra vấn đề, dù cho không cố ý, nếu không thiết kế một cách cẩn thận để cho “polite” và “ethical”
- VD: send quá nhiều request ➔ DoS
  - Server admin và người dùng sẽ nổi giận
  - Crawler bị blacklist

Đạo đức crawler (important!)

- Báo danh chính mình
  - Sử dụng 'User-Agent' HTTP header để báo danh crawler, website chứa thông tin crawler và cách liên lạc với người phát triển crawler
  - Sử dụng 'From' HTTP header để xác định email của người phát triển crawler
  - Không ngụy trang crawler như browser bằng cách sử dụng chuỗi 'User-Agent'
- Luôn luôn kiển tra HTTP requests đã thành công, và trong trường hợp lỗi, sử dụng HTTP error code để xác định và lập tức sửa lỗi.
- Chú ý đến bất kỳ thứ gì có thể dẫn đến quá nhiều request đến một server dù không cố ý:
  - redirection loops
  - spider traps

Đạo đức crawler (important!)

- Phân tán tải, không overwhelm server
  - Bảo đảm rằng không sử dụng quá một số lượng nhất định request đến một server tại một thời điểm nhất định, như < 1/second
- Tôn trọng Robot Exclusion Protocol
  - Server có thể xác định phần nào trong trang web cho phép crawling trong file 'robots.txt' đặt ở HTTP root directory, e.g. http://www.indiana.edu/robots.txt
  - Crawler luôn luôn phải kiểm tra và tuân thủ file này trước khi send requests đến server
  - Xem thêm:
    - http://www.google.com/robots.txt
    - http://www.robotstxt.org/wc/exclusion.html

Thông tin crawler và chặn crawler

- Nhà quản trị website có thể không muốn crawler truy xuất tài nguyên của mình.
- Tập tin "robots.txt" đặt trong thư mục gốc của Web server sẽ cung cấp các điều lệ truy xuất đối với crawler.
  - Các crawler xác định bởi trường User-agent.
  - Ví dụ: không cho bất kì Crawler nào truy cập server:

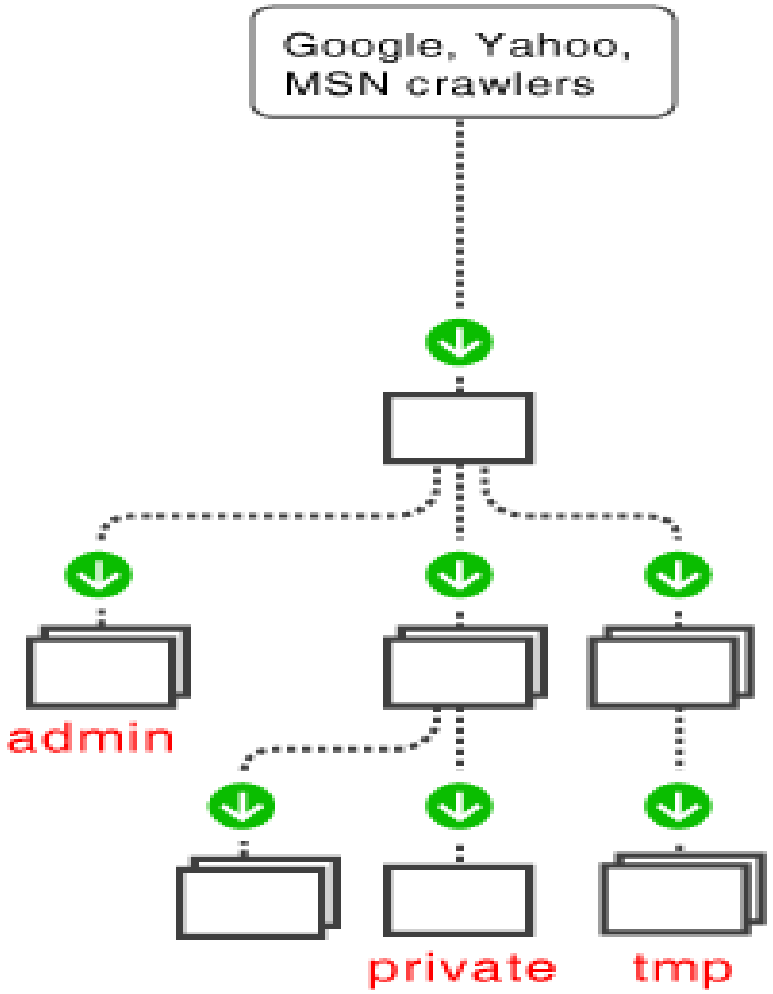
    *User-agent: **

    Disallow: /
- Crawler đạo đức phải lấy và phân tích tập tin robots.txt trước khi thâm nhập vào server.

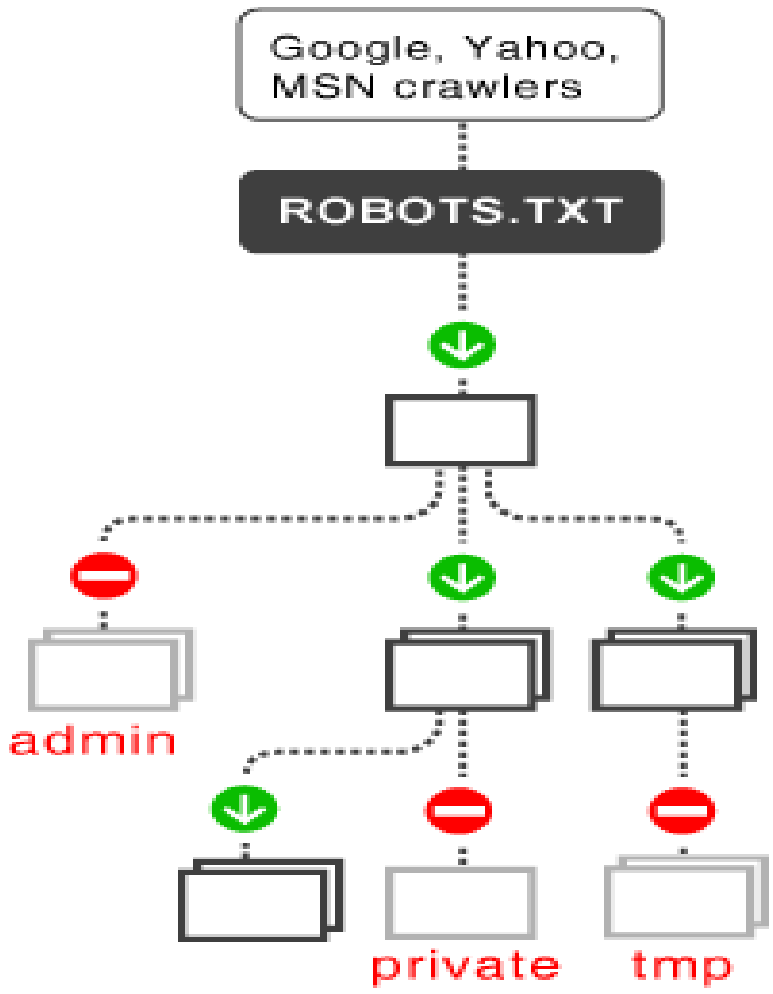
Robots.txt File Explained

Use the robots.txt file to restrict search engine crawlers from indexing selected areas of your website.

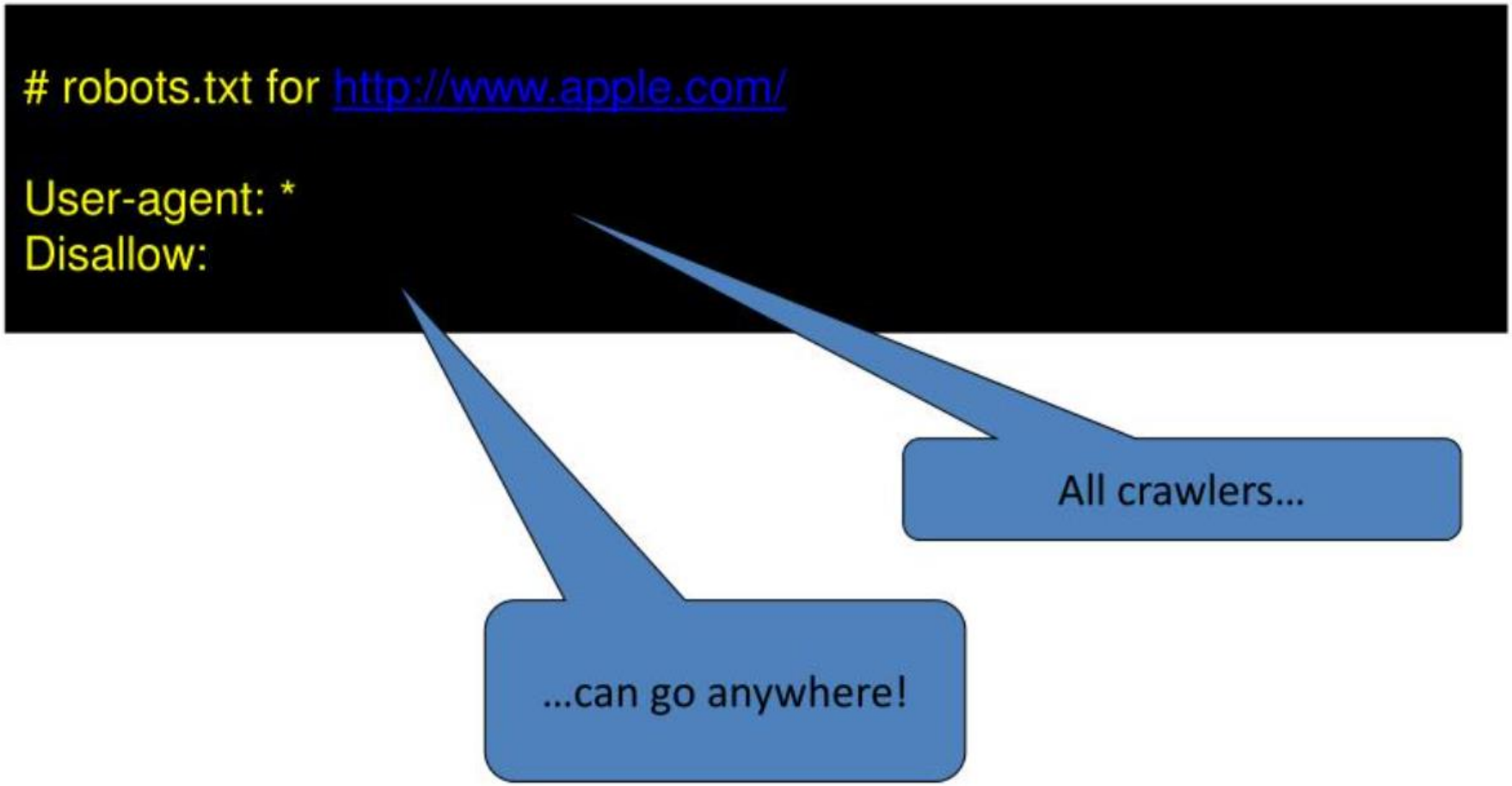
Site without Robots.txt



Site with Robots.txt



www.apple.com/robots.txt



www.microsoft.com/robots.txt

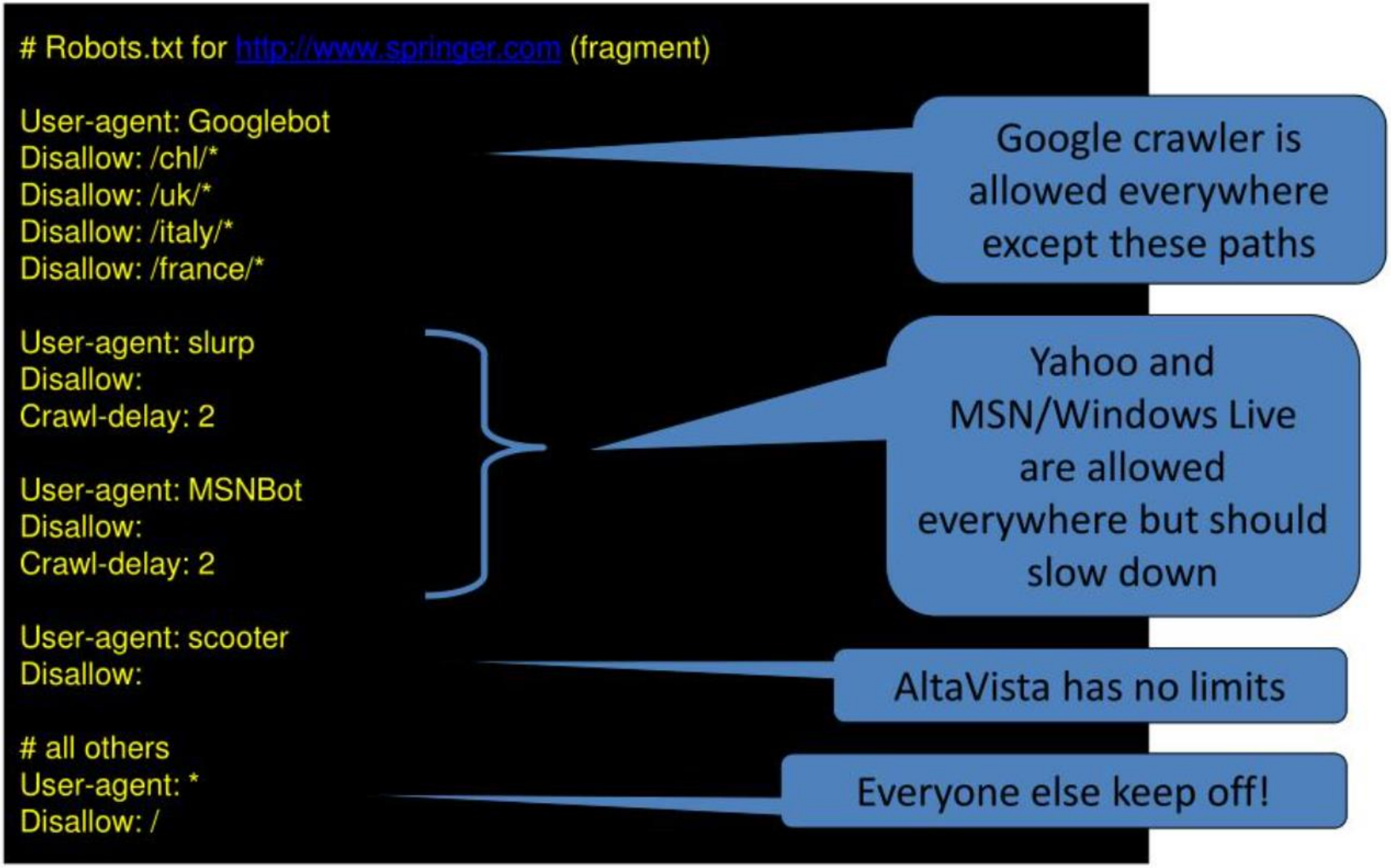
```
# Robots.txt file for http://www.microsoft.com

User-agent: *
Disallow: /canada/Library/mnp/2/aspx/
Disallow: /communities/bin.aspx
Disallow: /communities/eventdetails.mspx
Disallow: /communities/blogs/PortalResults.mspx
Disallow: /communities/rss.aspx
Disallow: /downloads/Browse.aspx
Disallow: /downloads/info.aspx
Disallow: /france/formation/centres/planning.asp
Disallow: /france/mnp_utility.mspx
Disallow: /germany/library/images/mnp/
Disallow: /germany/mnp_utility.mspx
Disallow: /ie/ie40/
Disallow: /info/customerror.htm
Disallow: /info/smart404.asp
Disallow: /intlkb/
Disallow: /isapi/
#etc…
```

All crawlers…

…are not allowed in these paths…

www.springer.com/robots.txt



Vấn đề đạo đức chung

- Tuân thủ nghi thức là vấn đề đạo đức.
  - Sự bằng lòng là tình nguyện và tập tin robots.txt không thể ép buộc được điều này.
- Các crawler cố tình vi phạm có thể bị xa lánh bởi cộng đồng.

- Crawler hiểm độc đang trở nên thông minh, không chỉ spam mà còn đánh cắp thông tin cá nhân và lừa gạt người dùng.
- Do đó, crawler chính thống cũng cần tăng cường tính phức tạp để ngăn chặn các thể tinh vi của spam, tránh làm ô nhiễm và phá hoại môi trường Web.

Một số hướng phát triển mới

Mức độ chuyên biệt

- Crawler phổ dụng tập trung vào các trang quan trọng nhất hướng về người dùng mức "trung bình", coi nhẹ tính khác biệt trong sự quan tâm của người dùng.
  - Ví dụ, trang về toán học có lẽ không thu hút đối với người dùng trung bình bằng trang về ngôi sao âm nhạc.

- Crawler cần học cách đánh giá chất lượng cho các trang liên quan đến cộng đồng nhỏ.

Google scholar

Hợp tác với mạng xã hội

- Mạng xã hội đang nhận được nhiều sự chú ý của người dùng.
- Các cơ chế như đánh tag (del.icio.us và flickr.com), bình chọn (stumbleupon.com), bình bầu (digg.com) và phân tầng (givealink.org) trở thành tài nguyên chia sẻ và tích hợp quan trọng.
- Crawler phải được thiết kế để tập hợp thông thông tin qua mạng xã hội như trên.

Mạng ngang hàng

- Mạng ngang hàng P2P đang được nhìn nhận như một kiến trúc mạnh trong việc chia sẻ giữa nhu cầu cá nhân và phục vụ cộng đồng.

- Crawler có thể được cài đặt trên các máy ngang hàng và các láng giềng của nó.
- Crawler phổ dụng được bổ sung một nhóm các crawler cá nhân phục vụ nhu cầu thông tin đặc biệt của người dùng cá nhân và mạng xã hội tự tổ chức động P2P.

Tài liệu tham khảo

- Bing Liu. 2007. *Web Data Mining-Exploring Hyperlinks, Contents, and Usage Data.* Springer Series on Data-Centric Systems and Applications. **Chapter 8**.
- http://en.wikipedia.org/wiki/Web_crawler
- http://www.eiao.net/webmining/teachers/presentations2007/webandcrawling/webandcrawling.html
- http://www.wepapers.com/Papers/112872/Web_Crawler.ppt