

WEB CRAWLING – DIE ERSCHLIESSUNG DES WEBS

Ronny Harbich

Otto-von-Guericke-Universität

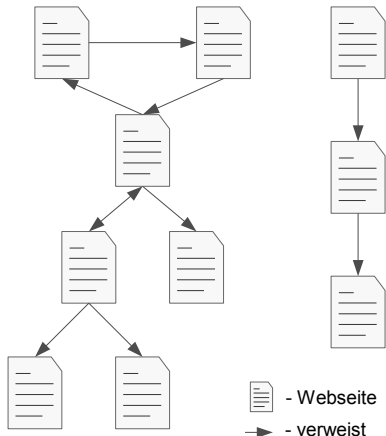
5. Dezember 2007

ÜBERSICHT

- 1 EINFÜHRUNG
- 2 ENTWURF EINES WEBCRAWLERS
 - Aufbau
 - Die Frontier
 - Der Downloader
 - Der Parser
- 3 SICHERHEIT
- 4 AUSBLICK UND LITERATUR

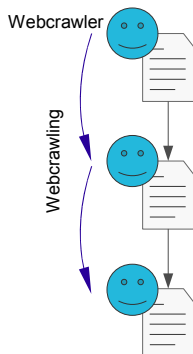
DAS WORD WIDE WEB

- System von untereinander referenzierten Webseiten
- Abstrakt: gerichteter Graph
 - Knoten repräsentieren Webseiten
 - gerichtete Kante von Knoten a nach Knoten b genau dann, wenn Webseite a auf Webseite b verweist
- WWW ist nicht zusammenhängend



DER WEBCRAWLER

- Programm, das Webseiten mit Hilfe der Verweise (*URLs*) durchläuft und herunterlädt
- Abstrakt: Webcrawler traversiert Graphen des Webs



ANWENDUNGEN

WEBCRAWLER FÜR

- Web-Suchmaschinen
- Datenschürfung (data mining)
- Web-Messungen (webometrics)
- Verweis- und HTML-Validierung
- E-Mail-Harvester (für E-Mail-Spam)

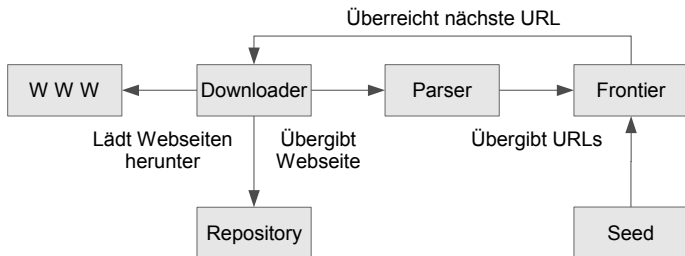
ÜBERSICHT

- 1 EINFÜHRUNG
- 2 ENTWURF EINES WEBCRAWLERS
 - Aufbau
 - Die Frontier
 - Der Downloader
 - Der Parser
- 3 SICHERHEIT
- 4 AUSBLICK UND LITERATUR

ÜBERSICHT

- 1 EINFÜHRUNG
- 2 ENTWURF EINES WEBCRAWLERS
 - Aufbau
 - Die Frontier
 - Der Downloader
 - Der Parser
- 3 SICHERHEIT
- 4 AUSBLICK UND LITERATUR

AUFBAU EINES WEBCRAWLERS



- *Frontier*: System, das nicht-besuchte URLs enthält
- Start-URLs (*seed*) an die Frontier übergeben
- Frontier übergibt URL an *Downloader*
- Downloader übergibt Webseite an *Repository* und *Parser*
- Parser übergibt gefundene URLs an Frontier

ÜBERSICHT

- 1 EINFÜHRUNG
- 2 ENTWURF EINES WEBCRAWLERS
 - Aufbau
 - **Die Frontier**
 - Der Downloader
 - Der Parser
- 3 SICHERHEIT
- 4 AUSBLICK UND LITERATUR

EIGENSCHAFTEN DES WEBS

GRÖSSEN

- Januar 2005: mehr als 11,5 Milliarden Webseiten durch Web-Suchmaschinen indexiert
- Dynamische Webseiten-Generierung: $n \in \mathbb{N}$ Query-Variablen $\Rightarrow n!$ verschiedene URLs

EINSCHRÄNKUNG

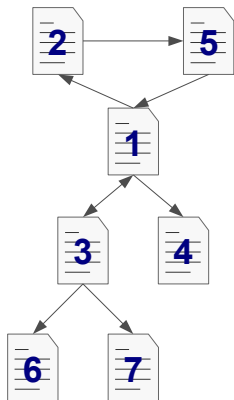
- Webcrawler können nicht das ganze Web durchsuchen.

FOLGERUNG

- Die Frontier muss zu besuchende URLs auswählen.

URL-AUSWAHLVERFAHREN

- ausschließliches Herunterladen von Webseiten durch Überprüfen des HTTP-Headers und der Dateieindung im URL
- doppelte URLs durch Hash-Tables vermeiden
- Beitensuche im Web-Graphen
⇒ Implementierung der Frontier als Warteschlange (queue)
- neue URLs durch *path-ascending crawling*

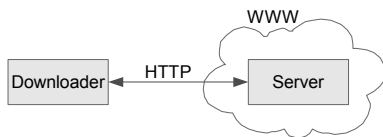


ÜBERSICHT

- 1 EINFÜHRUNG
- 2 ENTWURF EINES WEBCRAWLERS
 - Aufbau
 - Die Frontier
 - **Der Downloader**
 - Der Parser
- 3 SICHERHEIT
- 4 AUSBLICK UND LITERATUR

ARBEITSWEISE

- Downloader erhält URL von der Frontier
- Downloader ist HTTP-Client
- sendet HTTP-Anfrage an Server in der URL
- erhält HTTP-Antwort vom Server mit Webseite
- Metadaten im HTTP-Header evtl. wichtig für spätere Analyse



BESCHRÄNKUNGEN

RESSOURCEN EFFIZIENT NUTZEN

- Kleinen *timeout* wählen
- Größe der herunterzuladende Webseite beschränken
- Server-seitige Einschränkung durch *Robots Exclusion Standard* (robots.txt)

BEISPIEL: ROBOTS.TXT

User-agent: FooCrawler
Disallow: /private/

ÜBERSICHT

- 1 EINFÜHRUNG
- 2 ENTWURF EINES WEBCRAWLERS
 - Aufbau
 - Die Frontier
 - Der Downloader
 - Der Parser
- 3 SICHERHEIT
- 4 AUSBLICK UND LITERATUR

URL-EXTRAKTION

EXTRAKTIONS-METHODEN

- Parser muss URLs in Webseiten vom Downloader finden
- URLs stehen im a-Tag des HTML-Dokuments
- Finden der URLs mittels *regulären Ausdrucks* oder
- mittels *(X)HTML-beziehungsweise XML-Parsers*
- relative URLs beachten

BEISPIEL: A-TAG IN HTML-SEITE

```
<a href="http://foo.de/">  
    Zur Foo-Webseite  
</a>
```

BEISPIEL: REGULÄRER AUSDRUCK

```
(?<=href=") .+?(?=")
```

URL-NORMALISIERUNG

NORMALISIERUNG

- URL-Normalisierung zur Vermeidung mehrfachen Webseiten-Besuchs
- Z.B.: `http://Foo.de/` und `http://foo.de/` zeigen auf gleiche Webseite

BEISPIEL: EINIGE REGELN

- Kleinschreibung des *Schema*- und des *Host-Namens*
- Entfernung des *Ankers* aus der URL
- Sortierung von Query-Variablen
- Entfernen der *Port-Nummer*, wenn es sich um HTTP-Standard-Port 80 handelt

ÜBERSICHT

- 1 EINFÜHRUNG
- 2 ENTWURF EINES WEBCRAWLERS
 - Aufbau
 - Die Frontier
 - Der Downloader
 - Der Parser
- 3 SICHERHEIT
- 4 AUSBLICK UND LITERATUR

SCHÄDLICHE SERVER

- *spider trap*: Erstellen „unendlicher“ Verzeichnis-Strukturen \Rightarrow Webcrawler verschwendet Ressourcen, wird „ausgebremst“



SCHÄDLICHE WEBCRAWLER

SCHÄDIGUNGEN

- Verschleiern des Webcrawlers durch Falschangabe (z.B. als bekannter Webbrowser) der User-Agent-Eigenschaft im HTTP-Header \Rightarrow Server kann nicht auf Webcrawler reagieren
- *E-Mail-Harvester*: extrahieren E-Mail-Adressen von Webseiten zu Spam-Zwecken

BEISPIEL: EXTRAKTION VON E-MAIL-ADRESSEN

Folgender regulärer Ausdruck erkennt E-Mail-Adressen:

```
[A-Z0-9._%+-]+?
```

```
@
```

```
[A-Z0-9.-]+?
```

```
\.
```

```
[A-Z]{2,4}
```

ÜBERSICHT

- 1 EINFÜHRUNG
- 2 ENTWURF EINES WEBCRAWLERS
 - Aufbau
 - Die Frontier
 - Der Downloader
 - Der Parser
- 3 SICHERHEIT
- 4 AUSBLICK UND LITERATUR

AUSBLICK

- Erschließung des *deep webs* problematisch
- In Zukunft: genauere Analysen der HTML-Dokumente; enger mit Wissensdatenbanken zusammenarbeiten ⇒ effizientere, zielgerichtete Navigation durchs Web
- Webcrawling bleibt Gegenstand aktueller Forschung . . .

LITERATUR



Pant, G. ; Srinivasan, P. ; Menczer, F.:

Crawling the Web.

(2003).

<http://dollar.biz.uiowa.edu/~pant/Papers/crawling.pdf>



Gulli, Antonio ; Signorini, Alessio:

The Indexable Web is More than 11.5 billion pages.

(2005).

[http:](http://www.cs.uiowa.edu/~asignori/web-size/size-indexable-web.pdf)

[//www.cs.uiowa.edu/~asignori/web-size/size-indexable-web.pdf](http://www.cs.uiowa.edu/~asignori/web-size/size-indexable-web.pdf)



Lother, Mathias ; Wille, Cornelius ; Zbrog, Fritz ; Dumke, Reiner:

Web Engineering.

Pearson Studium, Addison Wesley, 2003

LITERATUR



Network Working Group:
Hypertext Transfer Protocol – HTTP/1.1.
<http://tools.ietf.org/html/rfc2616>.
Version: 1999



Network Working Group:
Uniform Resource Identifier (URI): Generic Syntax.
<http://tools.ietf.org/html/rfc3986>.
Version: 2005



SELFHTML e.V.:
robots.txt – Robots kontrollieren.
<http://de.selfhtml.org/diverses/robots.htm>.
Version: 2007



W3C:
HTML 4.01 Specification.
<http://www.w3.org/TR/html4/>.
Version: 1999