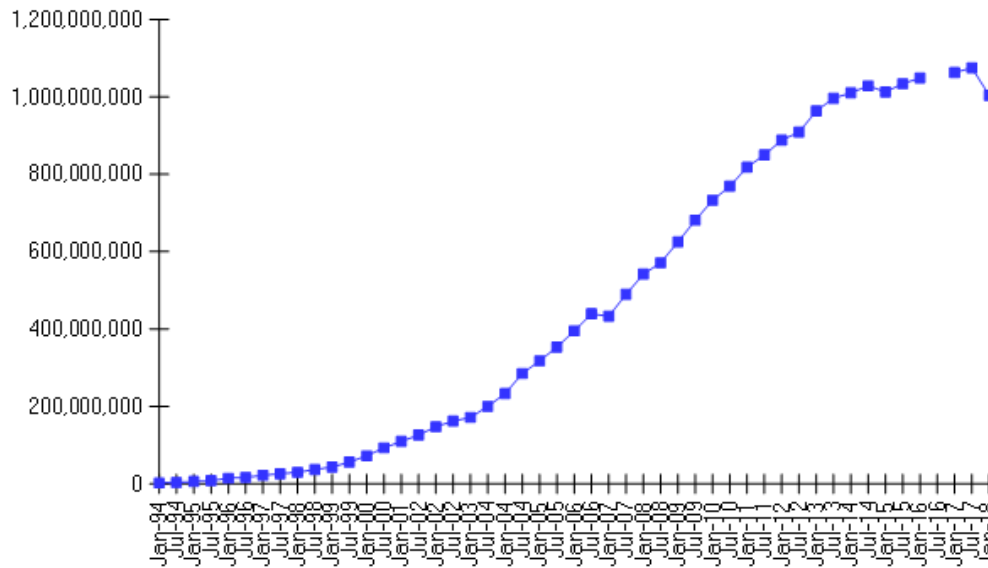


Websuchmaschinen

Wachstum des Internets und des Web

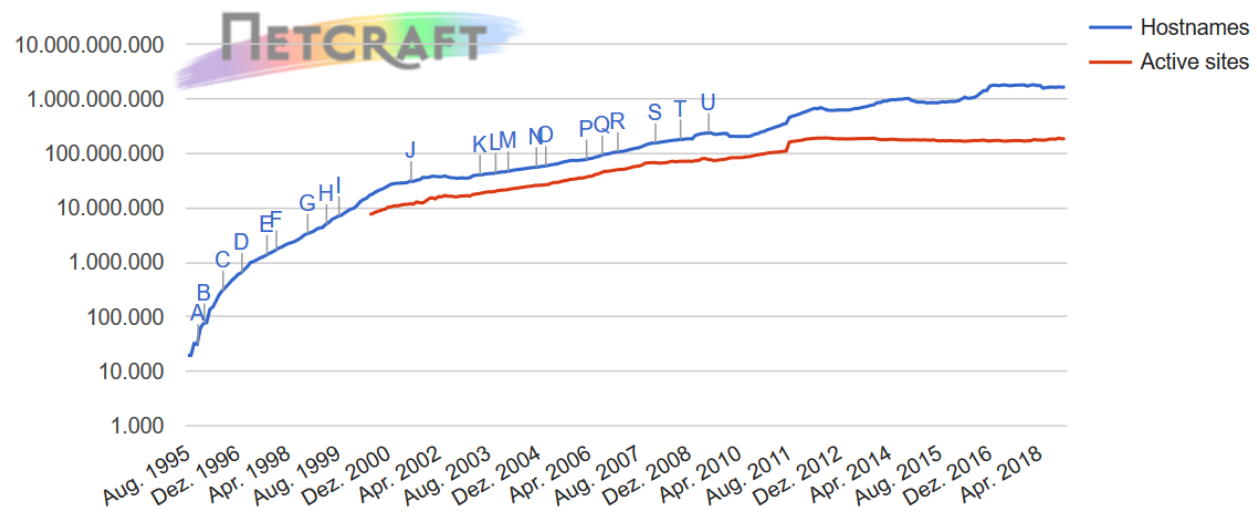
Quelle: <https://www.isc.org/network/survey/>

Internet Domain Survey Host Count



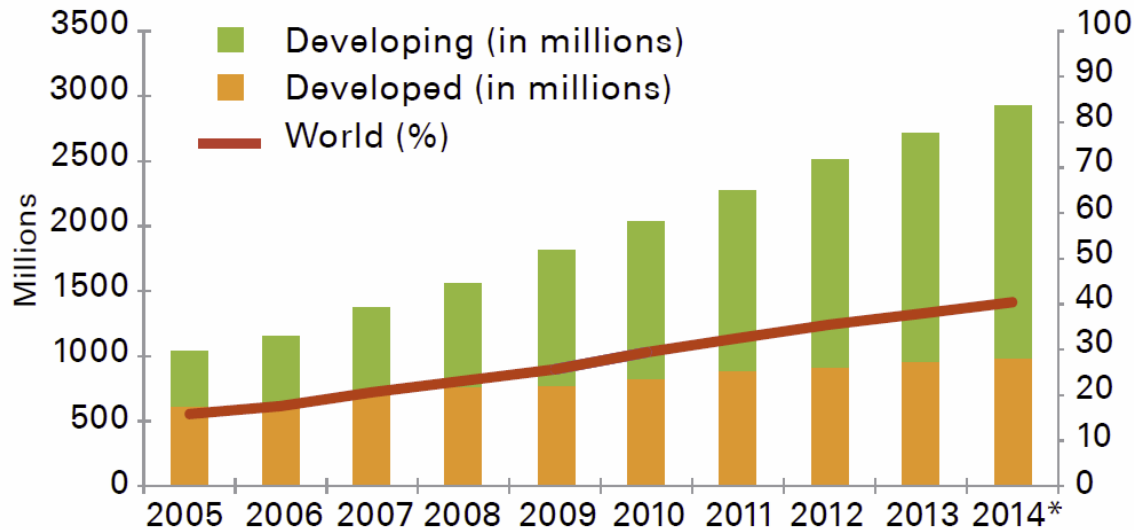
Source: Internet Systems Consortium (www.isc.org)

Total number of websites (logarithmic scale)



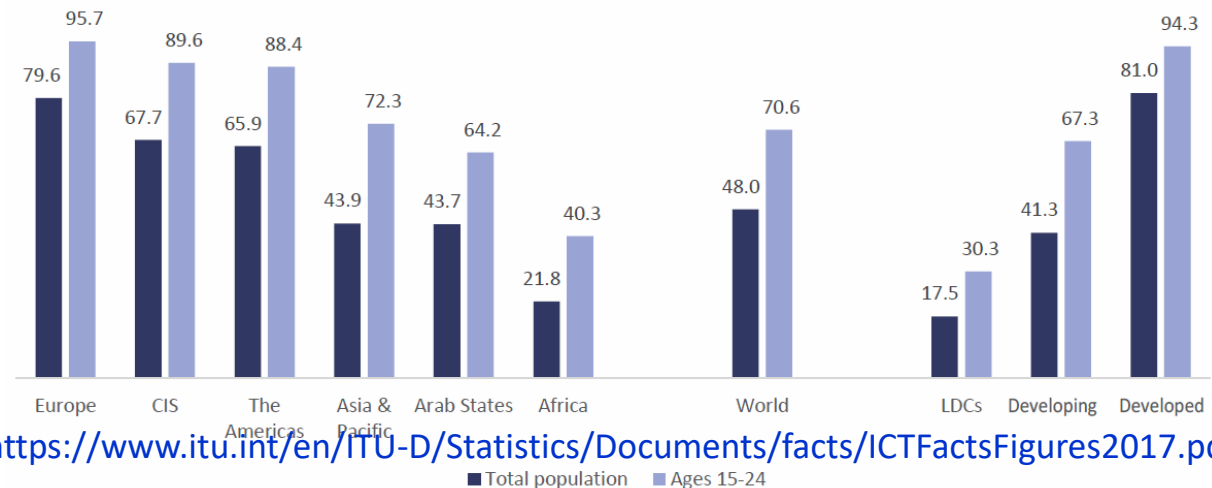
Quelle:
<https://news.netcraft.com/archives/2018/12/17/december-2018-web-server-survey.html>

Wachstum der Benutzerzahlen



Quelle: <http://www.itu.int/en/ITU-D/Statistics/Documents/facts/ICTFactsFigures2014-e.pdf>

Proportion of individuals using the Internet, by age, 2017*



Quelle: <https://www.itu.int/en/ITU-D/Statistics/Documents/facts/ICTFactsFigures2017.pdf>

Folgen des Wachstums

Internet und Nutzerzahlen steigen rasant an:

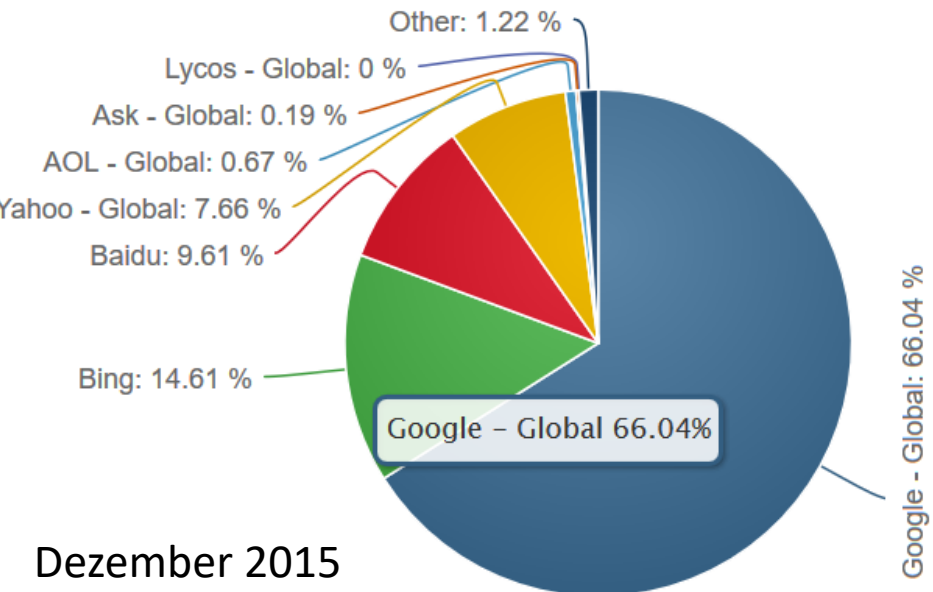
- Viele **verschiedene Nutzer** mit sehr **unterschiedlichen Interessen**
- **Gewaltige**, nicht mehr überschaubare, **Informationsmenge**

Folge: Bestimmte Informationen direkt zu finden ist praktisch unmöglich!

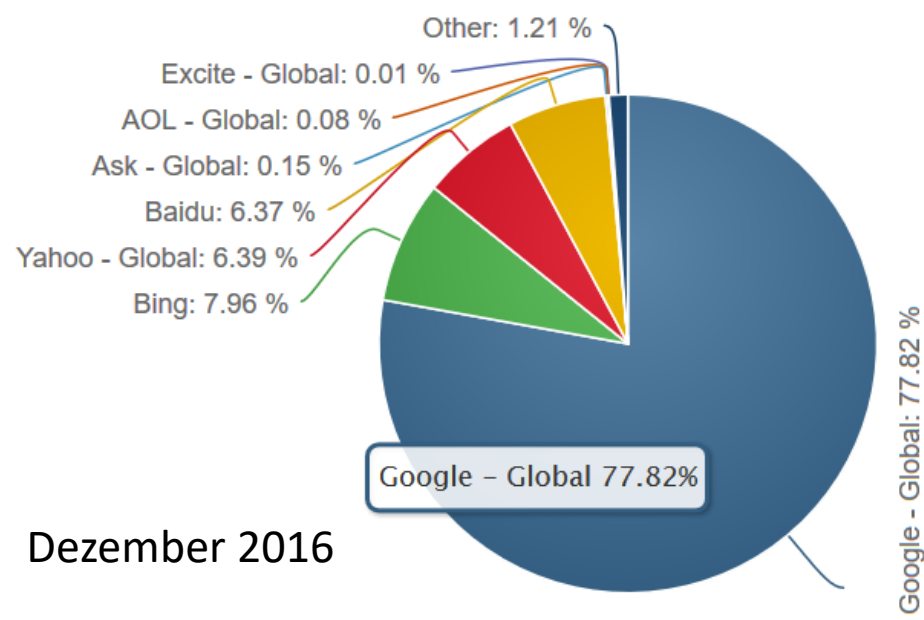
Ansätze für die Informationsfindung

- **Verlinkung** thematisch ähnlicher Seiten
 - Vorteil: Verlinkte Seiten sind oft sehr hilfreich
 - Nachteil: Einschränkung auf verlinkte Seiten
 - Beispiel: Wikipedia
 - **In digitalen Bibliotheken: Zitate in Veröffentlichungen**
- Bildung **thematischer Indexe**
 - Vorteil: Großes Verzeichnis von ähnlichen Seiten
 - Nachteil: Für die thematische Einstufung und Ordnung von Webseiten sind Experten erforderlich
 - Beispiel: Yahoo, DMOZ
 - **In digitalen Bibliotheken: Metadatenkataloge, Konferenzverzeichnisse**
- **Suchmaschinen**
 - Vorteil: Automatisierte Erfassung sehr vieler Webseiten
 - Nachteil: Herausforderungen des Information Retrieval
 - Beispiel: Google, Bing
 - **In digitalen Bibliotheken: Volltextsuche**

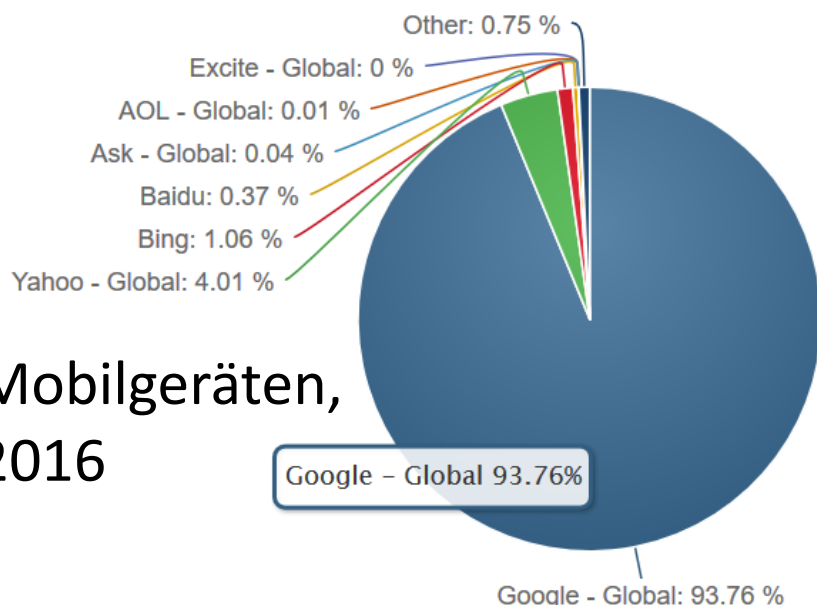
Marktanteile (weltweit)



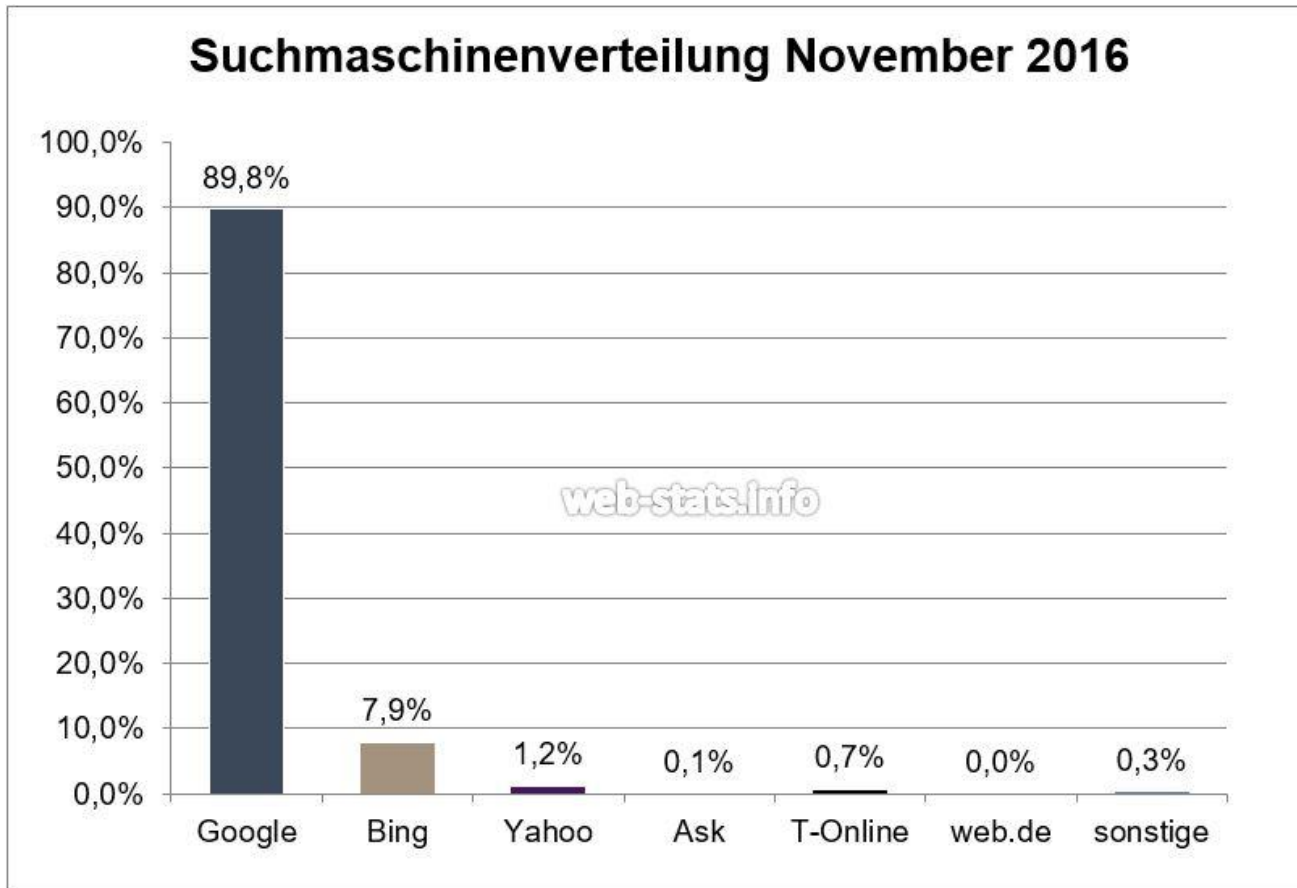
Suche von Desktop-PCs



Suche von Mobilgeräten, Dezember 2016



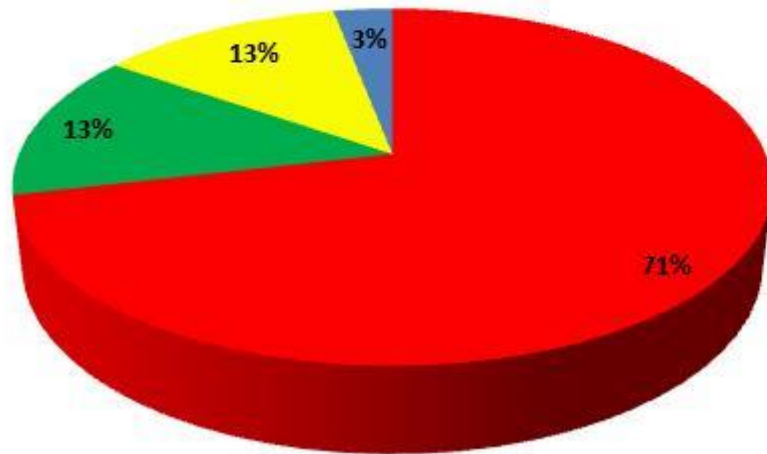
Markanteile (Deutschland, November 2016)



Quelle: <http://www.web-stats.info/>

Lokaler Marktführer: Baidu

Marktanteile Suchmaschinen China



Stand Oktober 2015

Quelle: <http://blog.xeit.ch/2015/11/baidu-zahlen-und-fakten-zur-suchmaschine-aus-china/>



Einige Gründe:

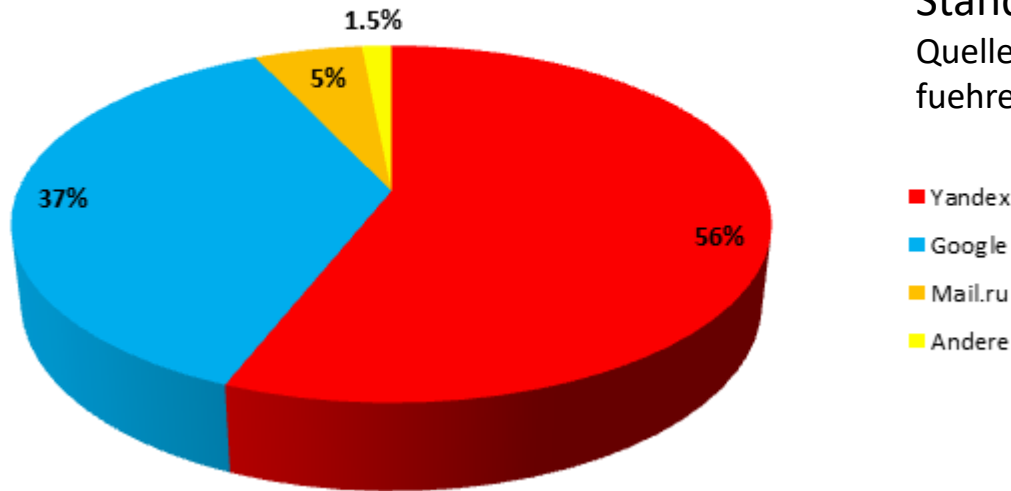
- Pionierunternehmen (First Mover Advantage)
- Downloads (insbesondere mp3s)
- Genaue Einhaltung der strikten Vorschriften der chinesischen Regierung
- Chinesische Sprache
- Rückzug von Google

Lokaler Marktführer: Yandex

Marktanteile Suchmaschinen Russland

Stand Januar 2016

Quelle: <http://blog.xeit.ch/2016/01/yandex-die-fuehrende-suchmaschine-russlands-zahlen-und-fakten/>



Einige Gründe:

- Pionierunternehmen (First Mover Advantage)
- Russische Sprache
- Erkennung von Flexionen (Wortbeugungen) in der Russischen Sprache

Archie – die erste „Suchmaschine“

Archie war die erste „Suchmaschine“:

- Werkzeug zur **Indizierung von FTP-Archiven**
- Geschrieben 1990 von Alan Emtage, Bill Heelan und J. Peter Deutsch, damals Studenten an der McGill Universität in Montreal
- Erste Version speicherte Dateilisten im lokalen Dateisystem, die mit dem Unixbefehl `grep` durchsucht wurden

*“The global collection of Archie servers process approximately **50,000 queries per day**, generated by a **few thousand users worldwide**. Every month or two of Internet growth requires yet another replica of Archie. A dozen Archie servers now replicate a continuously evolving 150 MB database of **2.1 million records**. While it responds in seconds on a Saturday night, it can take **five minutes to several hours** to answer simple queries during a weekday afternoon.”*

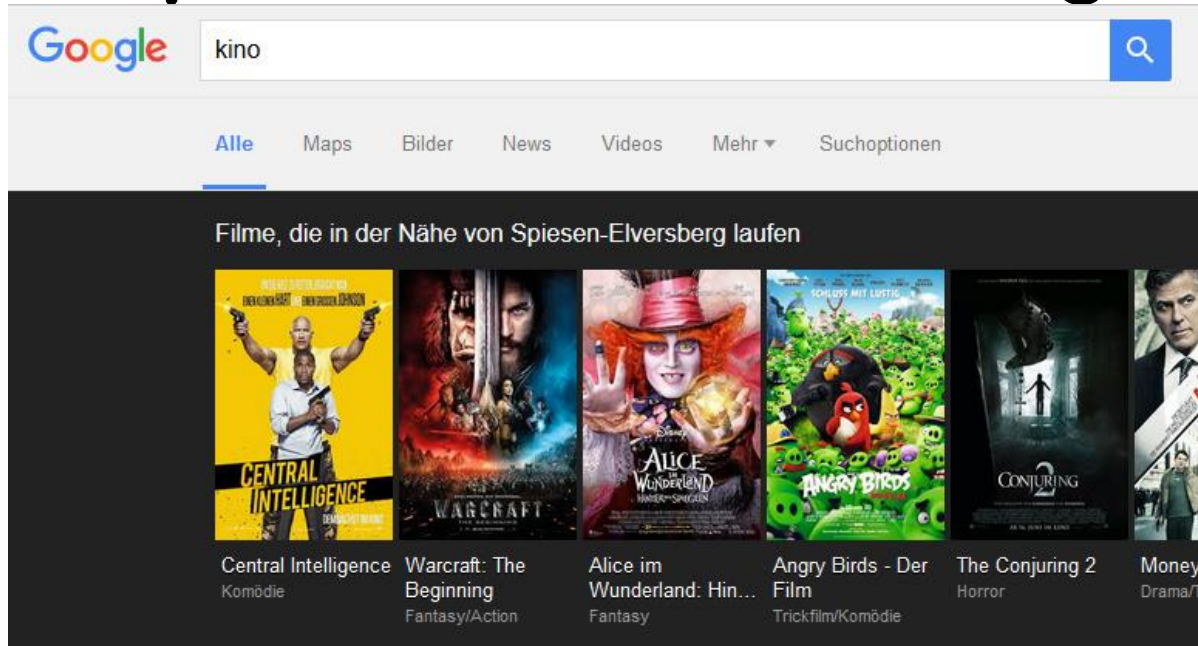
Quelle: Research Problems for Scalable Internet Resource Discovery (1993)

Herausforderungen an Websuchmaschinen

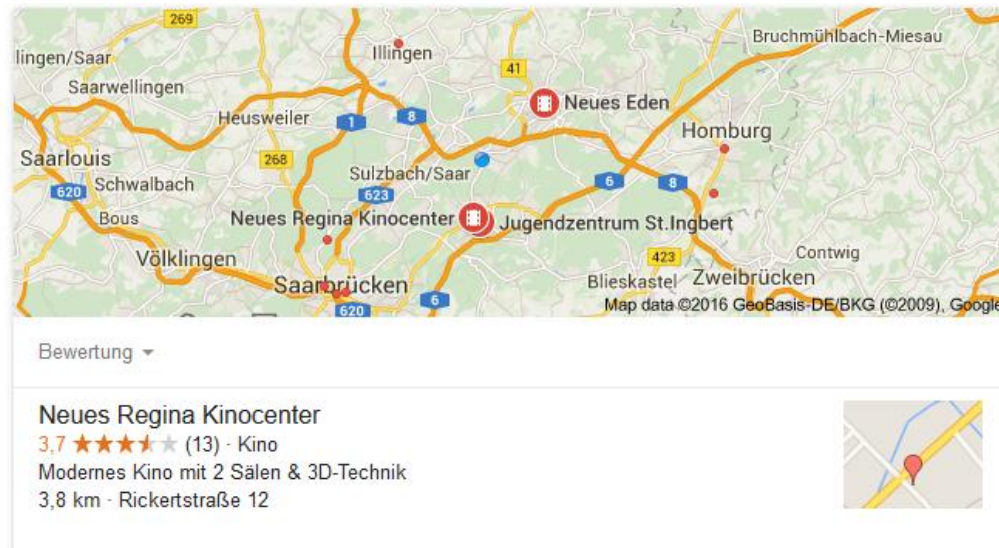
Größte Herausforderungen für Information Retrieval im World Wide Web:

- Sehr **großer Datenbestand**
- Sehr **dynamischer Datenbestand**
- Unterscheidung wichtiger und belangloser Webseiten
- Aussortierung **bösartiger** Webseiten
- Verarbeitung unterschiedlichster **Themengebiete** (z.B. Schlagworte, Kartendienste, Aktienkurse)
- Kontextualisierung (z.B. Geolokalisierung)
Beispiel: Welche Restaurants sind **in der Nähe**?
- Personalisierung
Beispiel: Welche Restaurants sind **für mich** zu empfehlen?


Beispiel für Geolokalisierung: Suche nach Kino



Hier könnte jetzt auch Personalisierung eine Rolle spielen



Beispiel für Geolokalisierung: Suche nach

Google 

[Alle](#) [News](#) [Maps](#) [Shopping](#) [Videos](#) [Mehr ▾](#) [Suchoptionen](#)

Ungefähr 142.000.000 Ergebnisse (0,40 Sekunden)

66583 Spiesen-Elversberg
Freitag, 12:00
Bewölkt

 **15** °C | °F

Niederschlag: 5%
Luftfeuchte: 83%
Wind: 18 km/h

[Temperatur](#) [Niederschlag](#) [Wind](#)

15 16 17 16

13:00 16:00 19:00 22:00 01:00

Fr.	Sa.	So.	Mo.	Di.
 17° 11°	 18° 9°	 19° 8°	 22° 12°	 22° 14°

Wo ist hier das Web?

Web als Quelle für
strukturierte Daten

[wetter.com: Wetter, Wettervorhersage, Wetterbericht, Reise | wetter.com](#)

[www.wetter.com/ ▾](#)

Aktuelles Wetter und 16-Tages Wettervorhersage für Ihren Ort. Mit Niederschlagsradar, Wetterwarnungen, Satellitenbildern und Spezialinformationen wie ...

[Wetter.com mobil](#) · [Berlin](#) · [Hamburg](#) · [Frankfurt am Main](#)

[Wetter Spiesen-Elversberg | wetter.com](#)

[www.wetter.com/deutschland/spiesen_elversberg/DE0010008.html ▾](#)

Wie wird das Wetter heute in Spiesen-Elversberg? Temperatur-, Wind- und ...

[Wetter.com mobil](#) · [Berlin](#) · [Hamburg](#) · [Frankfurt am Main](#)

[Wetter Spiesen-Elversberg | wetter.com](#)

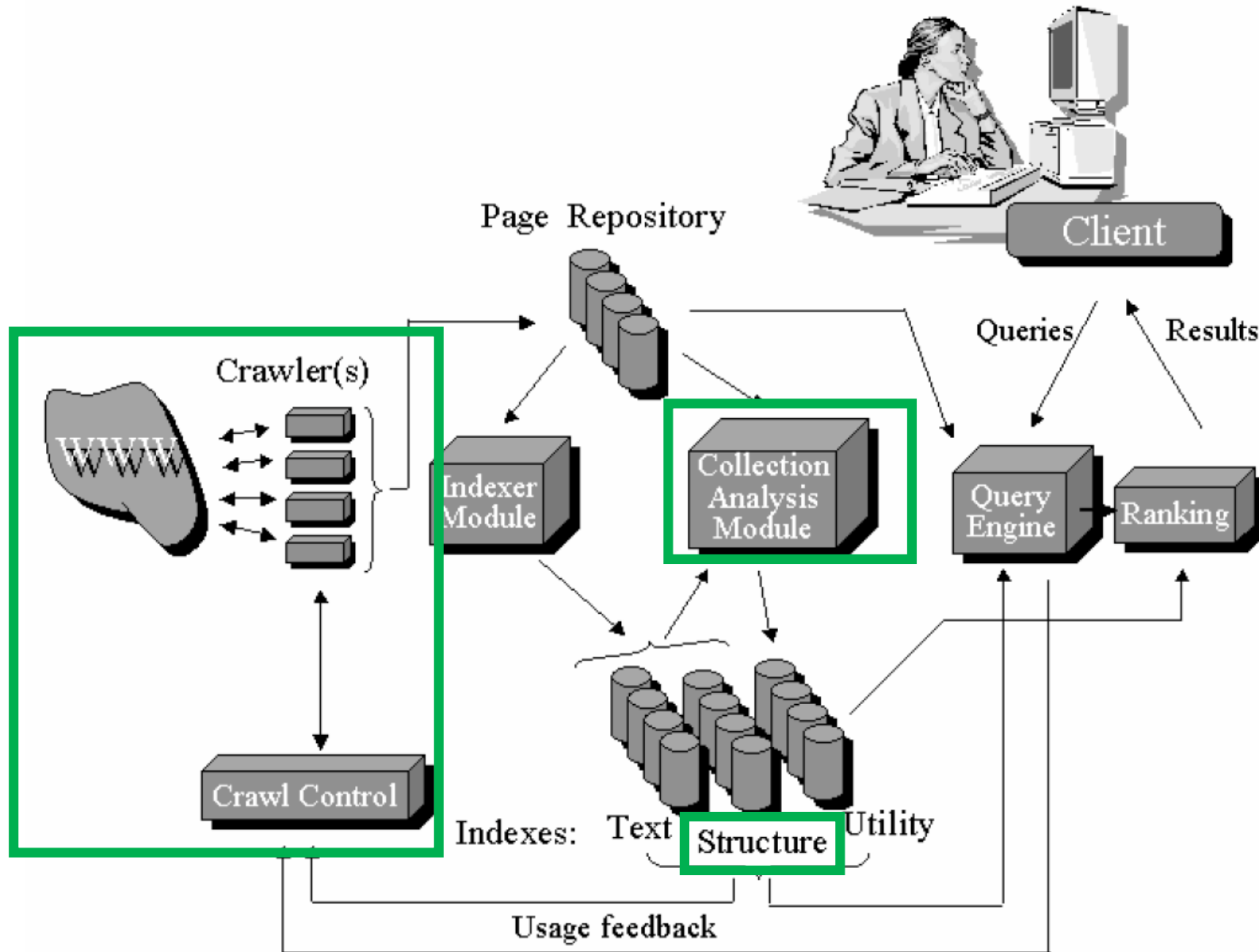
[www.wetter.com/deutschland/spiesen_elversberg/DE0010008.html ▾](#)

Wie wird das Wetter heute in Spiesen-Elversberg? Temperatur-, Wind- und ...

weather.com

[Feedback](#)

Grundarchitektur einer Suchmaschine



Crawler

Ein **Crawler** scannt das WWW nach Veränderungen ab. Seine Aufgabe ist es, **neue**, **modifizierte** und **gelöschte Webseiten** zu identifizieren. Seine Informationen gibt er an den Indexer weiter.

Crawler (auch Robots oder Spiders genannt) sind im Prinzip Programme, die auf Hosts ausgeführt werden und schlicht Webseiten im Internet anhand ihrer Verlinkungen aufrufen. Sie können zusätzlich auch von außen Aufforderungen zum Abarbeiten einer Webseite erhalten.

Crawler-Zugriffe
auf DBLP
im Dezember 2018

111 Zugriffe durch Suchmaschinen*	Zugriffe	Bytes	Letzter Zugriff
Sogou Spider	37,050,612+31	503.49 GB	31.12.2018 - 23:59
Googlebot	11,055,793+172	110.14 GB	31.12.2018 - 23:59
bingbot	3,259,142+1430	45.93 GB	31.12.2018 - 23:59
Unknown robot (identified by 'bot*')	1,586,622+27012	112.39 GB	31.12.2018 - 23:59
Yandex bot	885,527+364	9.34 GB	31.12.2018 - 23:59
Unknown robot (identified by 'crawl')	705,297+3209	13.46 GB	31.12.2018 - 23:59
Exabot	598,640+39	7.03 GB	31.12.2018 - 23:59
yeti	494,828+79	6.86 GB	31.12.2018 - 23:59
Python-urllib	389,906+116	17.84 GB	31.12.2018 - 23:59
BaiDuSpider	340,198+97	100.47 GB	31.12.2018 - 23:59
Sonstige	1,373,577+24420	169.73 GB	

* Die Robots, die hier angezeigt werden, zeigen Treffer oder Traffic welchen Besucher "nicht gesehen" haben und sind in den übrigen Diagrammen nicht enthalten. Zahlen hinter + sind erfolgreiche Treffer auf die "robots.txt"-Datei

Anforderungen an Crawler

Robustheit gegen **Spidertraps**:

- Syntaxfehler in Webseiten
- Fehlerhafter Aufbau von Webseiten
- Dynamische Applikationen

Höflichkeit gegenüber **Webservers**:

- Implizite serverseitige Regeln für Crawler, z.B. Anfragehäufigkeit (“politeness”)
- Explizite serverseitige Regeln für Crawler, z.B. die „robots.txt“, rel="nofollow" für HTML-Links und das Metatag robots
- „Web Etiquette“

Empfehlungen für Crawler

- Verteiltheit
- Skalierbarkeit, Effizienz
- Webseiten mit guter Qualität bevorzugen
- Aktualität der indizierten Webseiten gewährleisten
- Erweiterbarkeit (z.B. neue Datenformate, neue Protokolle)

Aktualisieren von Webseiten

Beispielszenario:

- n Webseiten
- Neuindizierung alle δ Zeiteinheiten
- Wann sollte der Crawler welche Seite aktualisieren?

Aktualität von Webseiten im Index

Die **Aktualität** F (Freshness) einer Webseite w_i im Index zum Zeitpunkt t sei bestimmt durch:

$$F(w_i, t) := \begin{cases} 1 & \text{falls } w_i \text{ zum Zeitpunkt } t \text{ aktuell ist} \\ 0 & \text{sonst} \end{cases}$$

Die **Aktualität** einer Menge von n Webseiten W im Index zum Zeitpunkt t sei die durchschnittliche Aktualität dieser Webseiten:

$$F(W, t) := \frac{1}{n} \sum_{i=1}^n F(w_i, t)$$

Alter von Webseiten im Index

Das **Alter (Age)** einer Webseite w_i im Index zum Zeitpunkt t sei definiert als:

$$A(w_i, t) := t - \max\{s | s \leq t, F(w_i, s) = 1\}$$

Das Alter einer Menge von n Webseiten W sei das durchschnittliche Alter dieser Webseiten:

$$A(W, t) := \frac{1}{n} \sum_{i=1}^n A(w_i, t)$$

Aktualisieren von Webseiten

Szenario:

- n Webseiten
- Neuindizierung alle δ Zeiteinheiten

Wann sollte der Crawler welche Seite aktualisieren?

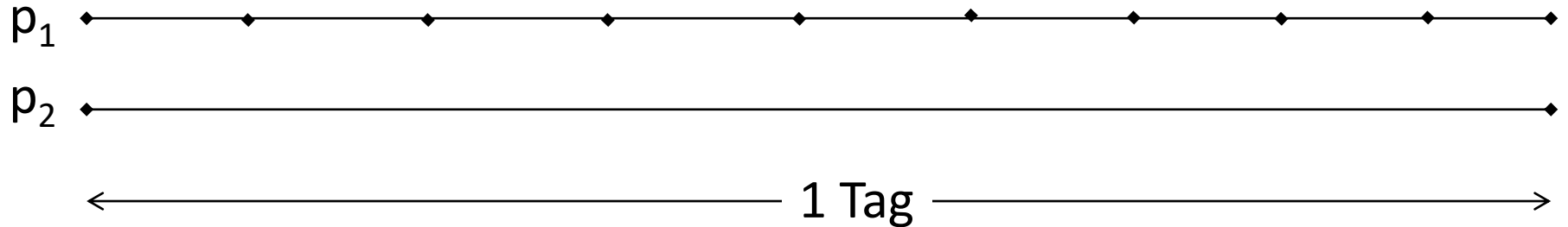
Vorgehen: z.B. **Minimierung des Alters** über die Zeit:

$$\text{Minimiere } \int_{t_1}^{t_2} A(W, t) dt = \int_{t_1}^{t_2} \sum_{i=1}^n A(w_i, t) dt$$

Dieses Problem ist allerdings nicht einfach zu lösen, selbst wenn die Änderungsraten der einzelnen Seiten bekannt sind.

Einfaches Beispiel: 2 Webseiten

Wir betrachten den einfachen Fall von zwei Webseiten p_1 und p_2 .



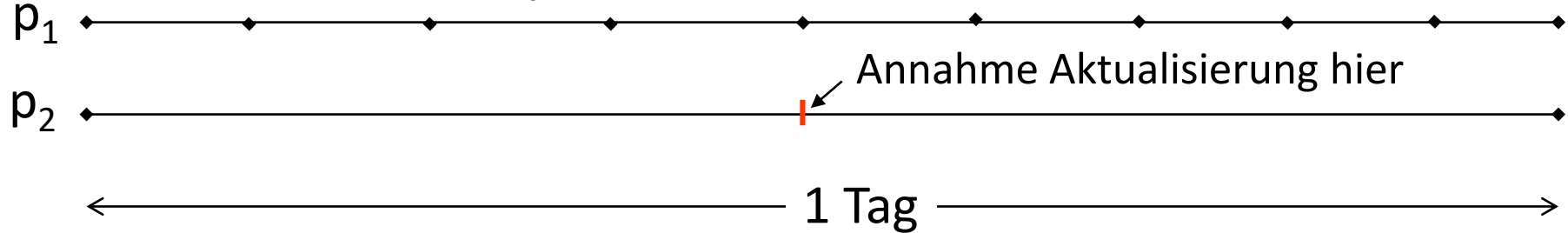
- p_1 ändert sich einmal in jedem gezeigten Intervall, also neunmal am Tag.
- p_2 ändert sich nur einmal am Tag.

Wenn wir nur **eine Aktualisierung pro Tag** machen können, welche Seite sollten wir aktualisieren, um die mittlere Freshness zu maximieren?

Die naheliegende Idee ist, die Seite p_1 , die sich häufiger ändert, zu aktualisieren. Tatsächlich ist es aber besser, p_2 zu aktualisieren.

Um dies zu zeigen, betrachten wir die Änderungsraten $\lambda_1 = 1/9$ und $\lambda_2 = 1$ der beiden Seiten und überlegen, welchen Benefit es für die (erwartete) Freshness bedeutet, eine der beiden Seiten zu aktualisieren.

Einfaches Beispiel: 2 Webseiten



Wie wäre der **erwartete Benefit** (also die Verbesserung der Freshness), wenn wir p_2 **einmal in der Mitte des Tages** aktualisieren würden?

- Mit Wahrscheinlichkeit $\frac{1}{2}$ ändert sich p_2 erst nach der Aktualisierung, also Benefit 0
- Mit Wahrscheinlichkeit $\frac{1}{2}$ hat sich p_2 vor der Aktualisierung geändert, also Benefit $\frac{1}{2}$
- Der **erwartete Benefit** ist also $\frac{1}{2} \cdot 0 + \frac{1}{2} \cdot \frac{1}{2} = \frac{1}{4}$

Analog (Aktualisierung in der Mitte eines Intervalls) können wir für p_1 den Benefit $\frac{1}{2} \cdot \frac{1}{18} = \frac{1}{36}$ bestimmen.

Es ist also besser, nur die Seite zu aktualisieren, die sich selten ändert, da wir mit einer Aktualisierung pro Tag die Dynamik von p_1 nicht nachvollziehen können.

Einfaches Beispiel: 2 Webseiten

Wie ändert sich diese Priorisierung, wenn wir mehr als eine Aktualisierung pro Tag durchführen können?

row	$f_1 + f_2$	f_1	f_2	benefit	best
(a)	1	1	0	$\frac{1}{2} \times \frac{1}{18} = \frac{1}{36}$	0 1
(b)		0	1	$\frac{1}{2} \times \frac{1}{2} = \frac{9}{36}$	
(c)	2	2	0	$\frac{1}{2} \times \frac{1}{18} + \frac{1}{2} \times \frac{1}{18} = \frac{2}{36}$	0 2
(d)		1	1	$\frac{1}{2} \times \frac{1}{18} + \frac{1}{2} \times \frac{1}{2} = \frac{10}{36}$	
(e)		0	2	$\frac{1}{3} \times \frac{2}{3} + \frac{1}{3} \times \frac{1}{3} = \frac{12}{36}$	
(f)	5	3	2	$\frac{3}{36} + \frac{12}{36} = \frac{30}{72}$	2 3
(g)		2	3	$\frac{2}{36} + \frac{6}{16} = \frac{31}{72}$	
(h)	10	9	1	$\frac{9}{36} + \frac{1}{4} = \frac{36}{72}$	7 3
(i)		7	3	$\frac{7}{36} + \frac{6}{16} = \frac{41}{72}$	
(j)		5	5	$\frac{5}{36} + \frac{15}{36} = \frac{40}{72}$	

In der Tabelle sind f_1 und f_2 die Aktualisierungsfrequenzen für p_1 und p_2 . Wir erkennen folgende Heuristiken:

- Wenn die Aktualisierungsfrequenz ($f_1 + f_2$) viel kleiner als die Änderungsfrequenz ($\lambda_1 + \lambda_2$) ist, lohnt es sich nicht, die Seite zu aktualisieren, die sich häufig ändert.
- Selbst wenn $f_1 + f_2 \approx \lambda_1 + \lambda_2$, ist eine Gleichverteilung der Zugriffe (5:5) besser als eine proportionale Verteilung (9:1).

Optimierung der Crawl-Reihenfolge

Mit geeigneten mathematischen Modellen für die Änderung von Seiten (z.B. Poisson-Prozesse) und Schätzungen für die Änderungsraten:

- Bestimmung der optimalen Verteilung der Aktualisierungen auf die einzelnen Seiten exakt numerisch bestimmen
- ggf. nach Gruppierung von Seiten, die sich ähnlich verändern.

Cho&Garcia-Molina:

- erzielbare mittleren Freshness- und Age-Werte für eine Auswahl von einer Milliarde Webseiten.
- Empirische Schätzung der Änderungsraten bei im Mittel einer Aktualisierung pro Monat.

	overall		com		gov	
	Freshness	Age	Freshness	Age	Freshness	Age
Proportional	0.12	400 days	0.07	386 days	0.69	19.3 days
Uniform	0.57	5.6 days	0.38	8.5 days	0.82	2.0 days
Optimal	0.62	4.3 days	0.44	7.4 days	0.85	1.3 days

Table VII. Freshness and age prediction based on the real Web data

Junghoo Cho, [Hector Garcia-Molina](#): Effective page refresh policies for Web crawlers. [ACM Trans. Database Syst. 28\(4\)](#): 390-426 (2003).
<http://dblp.org/rec/journals/tods/ChoG03>

Aktualisieren von Webseiten mit Wichtigkeit

Szenario:

- n Webseiten
- Neuindizierung alle Zeiteinheiten
- **Wichtigkeit** (z.B. Pagerank, Clickfrequenzen)

Wann sollte der Crawler welche Seite aktualisieren?

Vorgehen: z.B. Minimierung des **gewichteten Alters** über die Zeit:

$$\text{Minimiere } \int_{t_1}^{t_2} \sum_{i=1}^n A(w_i, t) \cdot PR(w_i, t) dt$$

Man kann hier ein ähnliches Optimierungsproblem lösen.

Indexer

Indexer-Modul:

- nimmt die Informationen von Crawlern entgegen
- baut daraus Indexe

Anforderungen an den Indexer:

- Verschiedene Dokumentformate
- Sprachliche Vorverarbeitung von Tokens
- Kompressionsmechanismen: verlustfrei und verlustbehaftet (z.B. Weglassen von Postings, Champion Lists, etc.)
- Verschiedene Indextypen (z.B. BM25-Indexe, Positionsindexe, Vokabularindexe für Wildcards, N-Gramm-Indexe, ...)

Searcher

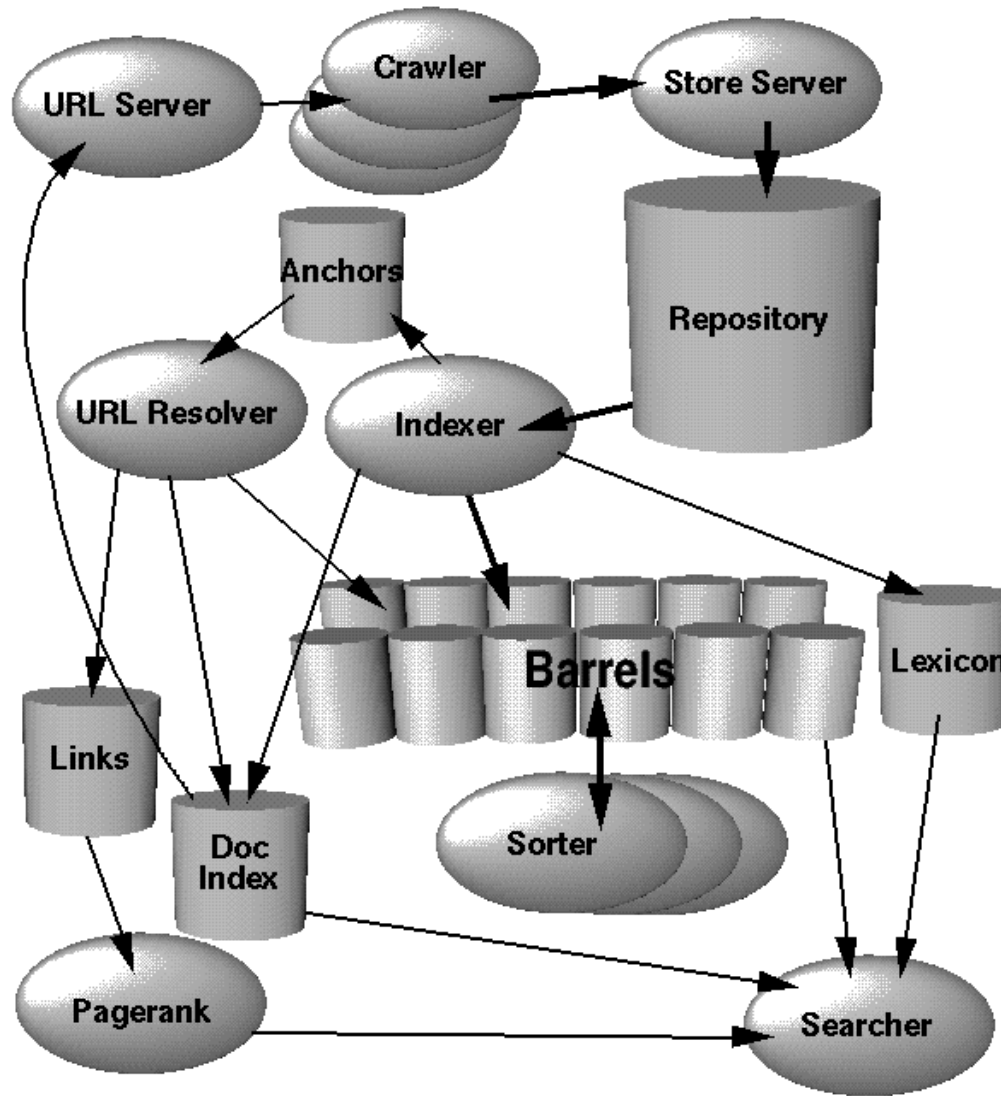
Searcher:

- nimmt Suchanfragen von Benutzern entgegen
- wertet diese anhand der vorhandenen Indexstrukturen aus
- führt die Relevanzschätzung zur Bewertung der Ergebnisse bzgl. der Suchanfrage durch

Anforderungen an den Searcher:

- Kommunikation mit Anwender
- Unterstützung des Anwenders bei Anfrageformulierung (z.B. Termvervollständigung, Anfragevervollständigung)
- Enge Zusammenarbeit mit dem Indexer: Verwendung der erstellten Indexe
- Relevanzschätzung, d.h. Ranking, von Dokumenten bzgl. Anfragen

Google-Architektur ca. 1997



Heute sind wesentliche Teile insbesondere der Anfrageverarbeitung und der Speicherung im Zuge der Skalierung und der Funktionserweiterung wesentlich anders aufgebaut.

Google-Crawler

Eigenschaften des **Crawlers** (**Googlebot**):

- Durchsucht täglich Milliarden Webseiten
- Die Updatehäufigkeit einer Webseite hängt von ihrer Relevanz (d.h. PageRank) ab
- Kann mittlerweile Flash-Animationen crawlen
- Laut Google keine kommerzielle Beeinflussungen

Google-Indexer

Indexarten:

- **Linkindex**

- Webgraph aus Knoten und Kanten
- Speichert insbesondere Nachbarschaftsinformationen


- **Textindex**


- Invertierter Index
- Lexikon
- Identifizierung gesuchter Webseiten

- **Relevanzindexe**

Die Indexierung unterstützt eine Vielzahl von Dateitypen.

Google-Searcher





Suchfeld

[Web](#) [Bilder](#) [News](#) [Maps](#) [Shopping](#) [Mehr ▾](#) [Suchoptionen](#)

Verticals

Ungefähr 208.000.000 Ergebnisse (0,46 Sekunden)

iMac Retina 5K Display
Anzeige www.apple.com/de ▾
Das ultimative Display für den ultimativen All-in-One. Mehr Infos.
[OS X Yosemite](#) · [MacBook Air](#) · [MacBook Pro](#) · [Mac mini](#)

Computer günstig kaufen - agando-shop.de
Anzeige www.agando-shop.de/computer ▾
4,6 ★★★★★ Bewertung für agando-shop.de
Erstelle Deinen Wunsch Computer! Perfekte Qualität, Made in Germany.
36 Monate Garantie · Pick-Up-&-Return Service · Sichere Spezialverpackung
Perfekte Fertig PCs · Der Kaufberater hilft · Riesiger Konfigurator

MEDION® Computer - Große Auswahl zu Herstellerpreisen
Anzeige www.medion.com/Computer ▾
4,5 ★★★★★ Bewertung für medion.com
Sichern Sie sich MEDION-Angebote!
Early Bird Deal - Bis 100€ Versand sparen - Jetzt 10% sparen
MEDION® 10% sparen - MEDION® AKOYA® P5501 D - Jetzt bestellen

Anzeigen



COMPUTER BILD: Tests, Downloads, Ratgeber & Kurse ...
www.computerbild.de/ ▾
Downloads - News - DSL-Speedtest - Tests

Ergebnisse

Computer – Wikipedia
de.wikipedia.org/wiki/Computer ▾
Die frühen Computer wurden auch (Groß-)Rechner genannt; deren Ein- und Ausgabe war zunächst auf die Verarbeitung von Zahlen beschränkt.
[Konrad Zuse](#) · [Zuse Z3](#) · [Personal Computer](#) · [Turing-Vollständigkeit](#)

Coreser - Computer Reparatur Service Passau
www.coreser.com/ ▾
Coreser Computer IT Service & Vertrieb Passau. Kostenlose Überprüfung Ihres Gerätes - egal ob Laptop oder PC. Rabatt für Studenten & Schüler!

Coreser IT Service & Vertrieb e. K. C...
www.coreser.com
2 Google-Bewertungen

Computer Service Taeuber
www.computer-taeuber.de
Google+ Seite

Lokale Ergebnisse


A Innstraße 71
Passau
0851 21376706

B Grünastraße 18
Passau
0851 7561781

Shopping-Ergebnisse

Google Shopping-Ergebnisse für computer **Anzeigen** ⓘ

**6.679,00 €**
Gamer Computer PC
-XTRAtec FULL ...
[tecstore.net](#)
Versand gratis

**329,00 €**
PC - CSL Sprint
Vision 6210
[CSL Computer](#)
+ 19,85 € Versand

**199,90 €**
Aufrüst-PC 677 -
FX-4300
[CSL Computer](#)
Versand gratis



Lokale Ergebnisse

Karte für computer

Anzeigen ⓘ

Chromebooks
www.google.de/Chromebook ▾
Schnell, sicher und einfach.
Der neue Computer - jetzt ab 249€.

DELL DE: Offizielle Seite
www.dell.com/de ▾
Riesen Auswahl an PCs & Notebooks
Mit Intel® Core™ Prozessor.

Anzeigen

Google-Searcher

Google

angela merkel

Suchfeld

- Web
- Bilder
- News
- Videos
- Maps
- Mehr
- Suchoptionen

Verticals

Fakten-Ergebnisse
aus dem Knowledge Graph

Ungefähr 66.200.000 Ergebnisse (0,39 Sekunden)

Angela Merkel

www.angela-merkel.de/

Die persönliche Internetseite der Vorsitzenden der CDU Deutschlands, Angela Merkel.

Angela Merkel – Wikipedia

de.wikipedia.org/wiki/Angela_Merkel

Angela Dorothea Merkel (* 17. Juli 1954 in Hamburg als Angela Dorothea Kasner) ist eine deutsche Politikerin. Bei der Bundestagswahl am 2. Dezember 1990 ...
Joachim Sauer - Merkel-Raute - Deutschlandkette - Lothar de Maizière

News zu angela merkel



Merkel sagt Wirtschaft niedrigere Rentenbeiträge zu

FAZ - Frankfurter Allgemeine Zeitung - vor 20 Minuten

Bundeskanzlerin Angela Merkel hat der Wirtschaft zugesagt, dass sich die große Koalition vermehrt um Investitionen und die Entlastung von ...

Angela Merkel auf Stippvisite in Greifswald

Nordkurier - vor 1 Tag

Freizügigkeit in der EU - Angela Merkel warnt David ...

STERN - vor 20 Stunden

Weitere Nachrichten für angela merkel

Angela Merkel - SPIEGEL ONLINE - Nachrichten

www.spiegel.de » Politik » Deutschland

Früher war sie "Kohls Mädchen", heute ist sie die mächtigste Frau Deutschlands: Angela Merkel hat sich an der Männerriege der Union vorbei an die Spitze der ...

Angela Merkel | Facebook

<https://de-de.facebook.com/AngelaMerkel>

Anlässlich des Reformationstages war Angela Merkel in der Maria-Magdalenen-Kirche in Templin zu Gast. In ihrer Rede zum Thema „Christsein und politisches ...



Mehr Bilder

Angela Merkel

Bilder

Bundeskanzlerin

Angela Dorothea Merkel ist eine deutsche Politikerin. Bei der Bundestagswahl am 2. Dezember 1990 errang Merkel, die in der DDR als Physikerin ausgebildet wurde und auch tätig war, erstmals ein ... Wikipedia

Geboren: 17. Juli 1954 (Alter 60), Hamburg

Größe: 1,65 m

Amt: Bundeskanzler seit 2005

Ehepartner: Joachim Sauer (verh. 1998), Ulrich Merkel (verh. 1977–1982)

Eltern: Horst Kasner, Herlind Kasner

Geschwister: Irene Kasner, Marcus Kasner

Wird auch oft gesucht

Über 15 weitere ansehen



Wladimir Putin



Joachim Sauer
Ehepartner



François Hollande



Barack Obama



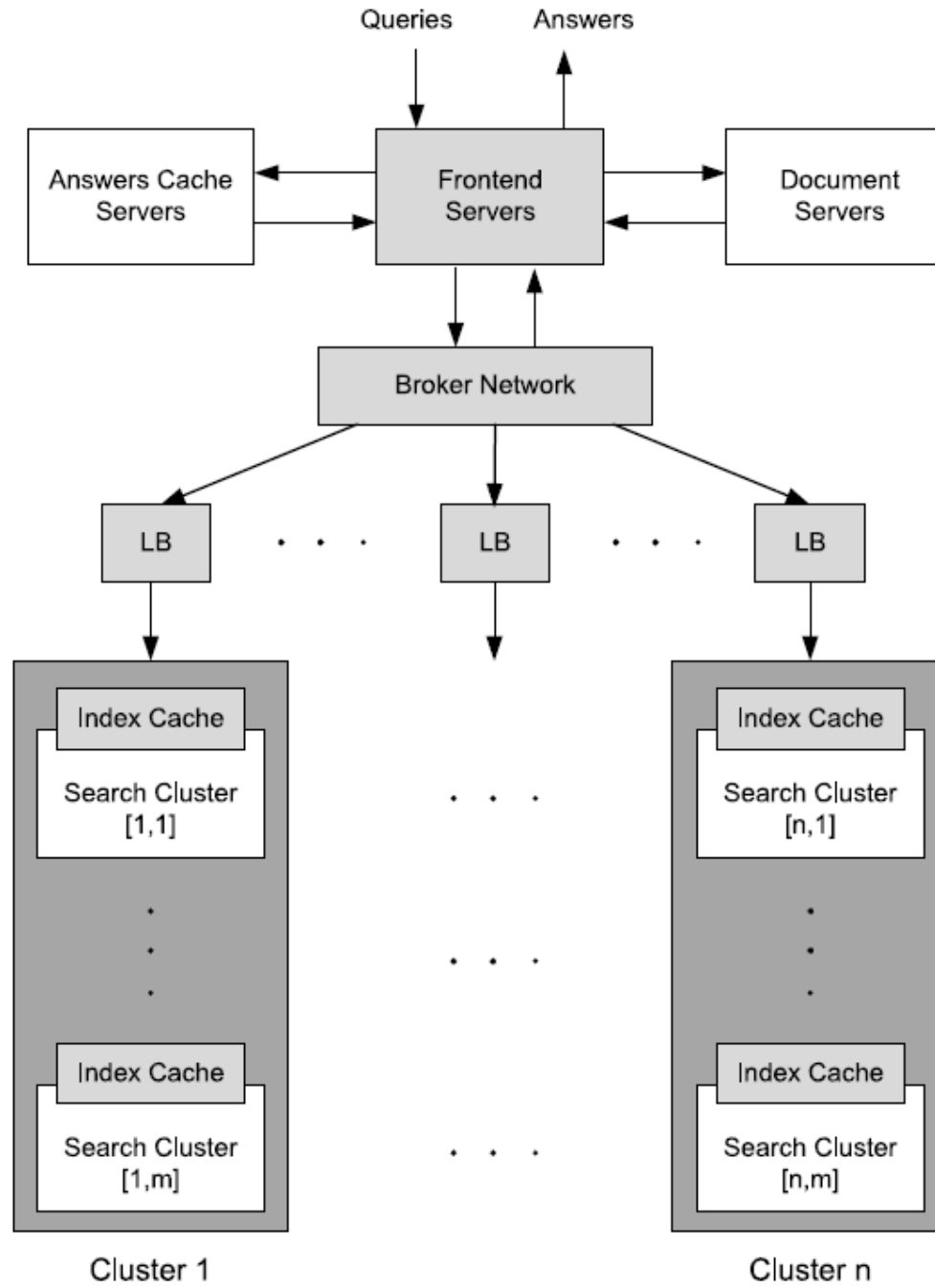
Joachim Gauck

Feedback geben

UI

7-33

Typische verteilte Architektur einer Suchmaschine

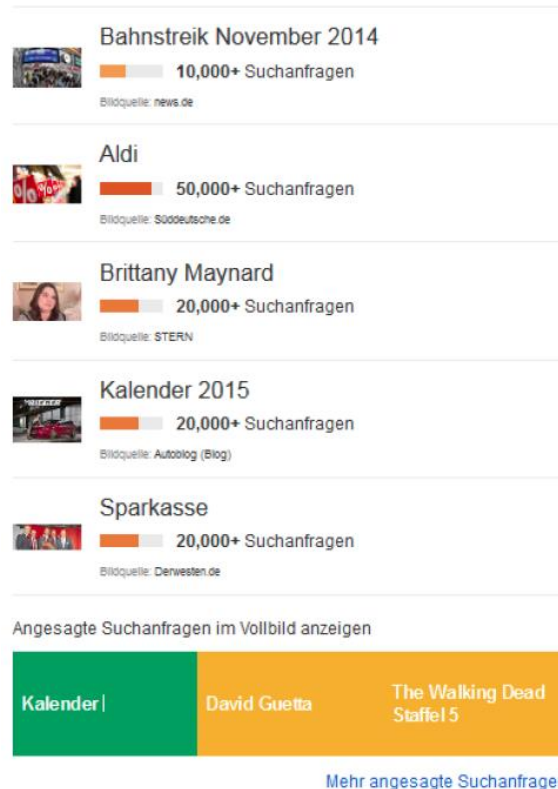


Quelle: Riccardo A. Baeza-Yates, Berthier A. Ribeiro-Neto:
Modern Information Retrieval - the concepts and technology behind search,
Second edition. Pearson Education Ltd., Harlow, England 2011, ISBN 978-0-
321-41691-9. <http://dblp.org/rec/books/aw/Baeza-YatesR2011>

Googles Anfragen

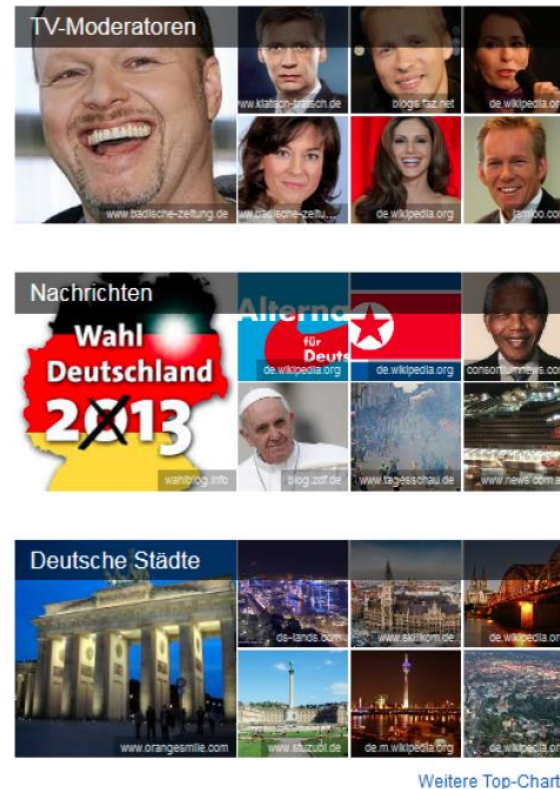
Aktuelle Trends

Deutschland

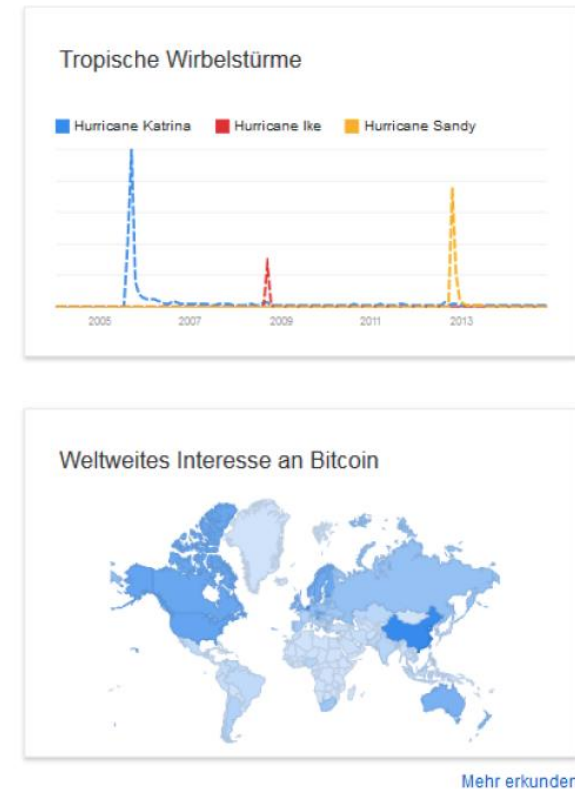


Charts für 2013

Deutschland



Ausführlicher Hintergrund



Einblick in Googles Arbeit: <http://www.google.de/trends>

- Aktuelle Trends für Anfragen
- Verteilung von Anfragen (geogr., hist., ...)

Websuchmaschinen

Ranking mit Pagerank

Ranking in Suchanfragen

Ranking ist ein zentraler Bestandteil bei Anfragen an Suchmaschinen:

Welche Dokumente benötigt der Anwender?

Probleme:

- Sehr unterschiedliche Dokumente und Inhalte
- Verlinkung von Webseiten bringt Zusatzinformation
- Bestimmung der Relevanz von Dokumenten
- Aufbau, Übersicht und Verständlichkeit von Webseiten sind auch relevant für den Anwender

Verlinkung im Web

Besonderheit des Webs: starke **Verlinkung**

Im frühen Web wurden Links **manuell** gesetzt. Man konnte also annehmen, dass der Linkersteller die verlinkte Seite für so wichtig hält, dass er darauf verweisen wollte. (Heute gilt das nicht immer, da viele Links **automatisch** gesetzt werden.)

Bei der Suche auf dem Web nutzt man diese Verlinkung der Seiten aus, um Seiten **hoher Qualität** zu identifizieren. Der so entstehende **Pagerank** war die wesentliche Zutat für den großen Erfolg der Google-Suchmaschine.

Webmodell

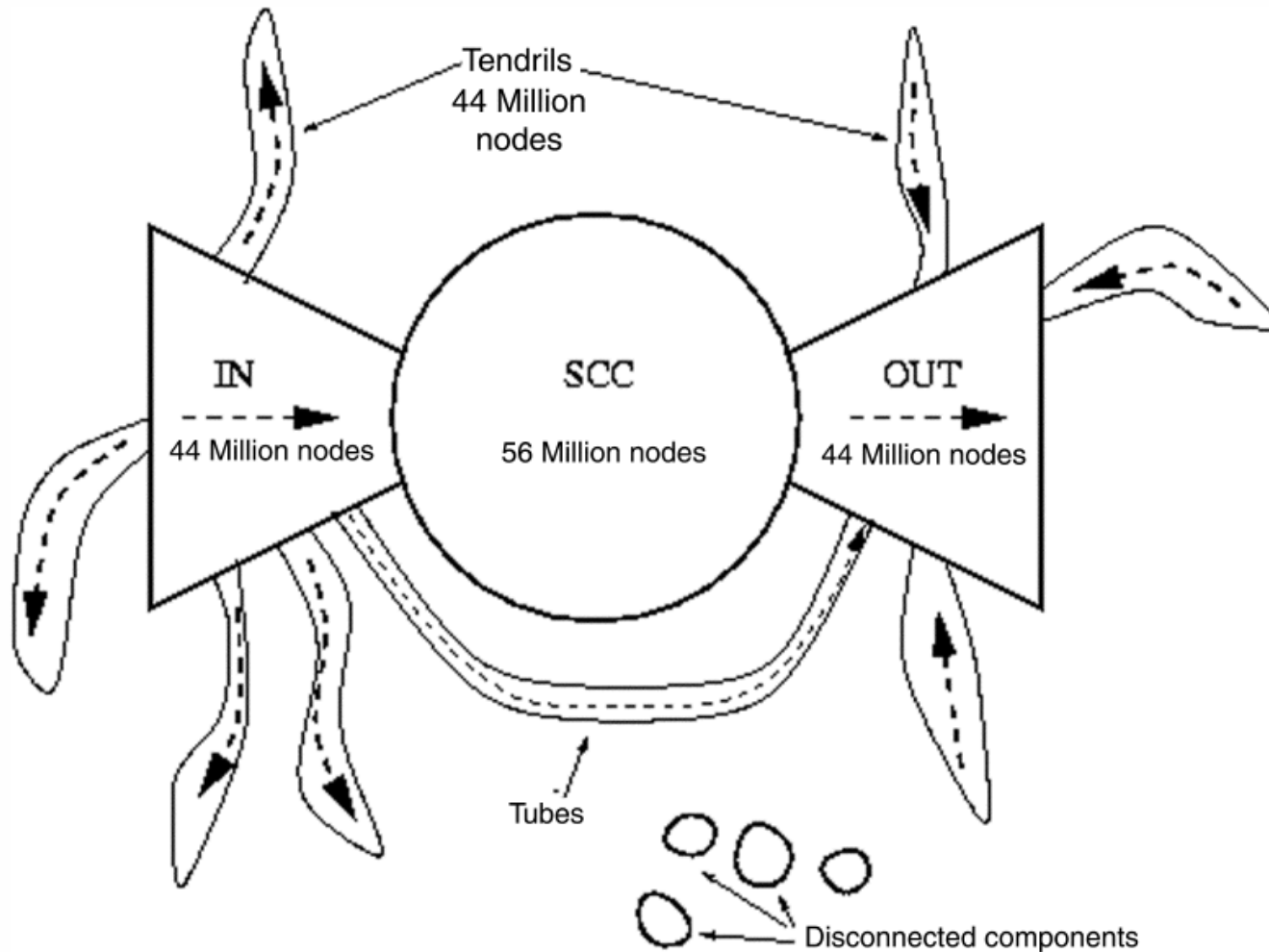
Der **Webpace W** ist ein gerichteter Graph $G = (V, E)$. Die Knoten V repräsentieren Webseiten im Webspace. Die Kanten E entsprechen der Verlinkung zwischen den Webseiten:

$$\exists e = (u, v) \in E \Leftrightarrow \text{Webseite } u \text{ enthält Link auf Webseite } v$$

Der Webspace bildet das Linkgeflecht der Webseiten im Internet ab. Der textuelle Inhalt der Webseiten wird dazu nicht berücksichtigt.

Graphstruktur des Webs

A. Broder et al. / Computer Networks 33 (2000) 309–320



Quelle: Andrei Z. Broder, Ravi Kumar, Farzin Maghoul, Prabhakar Raghavan, Sridhar Rajagopalan, Raymie Stata, Andrew Tomkins, Janet L. Wiener:
Graph structure in the Web. *Computer Networks* 33(1-6): 309-320 (2000).
<http://dblp.org/rec/journals/cn/BroderKMRSTW00>

Der Webspace sah im Jahr 2000 ungefähr aus wie eine Fliege (engl. Bowtie).

Nachbarschaft in Graphen

$N_{\text{out}}(v)$ bezeichne die **Out-Nachbarn**, d.h. die Menge aller Knoten (Webseiten), die durch eine Kante (einen Link) von v erreichbar sind:

$$N_{\text{out}}(v) := \{w \mid (v, w) \in E\}$$

$N_{\text{in}}(v)$ bezeichne die **In-Nachbarn**, d.h. die Menge aller Knoten (Webseiten), die eine Kante (einen Link) zu v besitzen:

$$N_{\text{in}}(v) := \{u \mid (u, v) \in E\}$$

Grundlegendes Modell: „Random Surfer“

Ein **Surfer** s wandert zufallsgetrieben durch den Webspace W , der durch den Graphen $G = (V, E)$ beschrieben wird. Er startet auf einer zufällig gewählten Webseite. Anschließend bewegt er sich zufällig von Knoten zu Knoten (Webseite zu Webseite) entlang der Kanten (Links) von G durch den Webspace (**Random Walk**).

Für die Bewegung des Surfers s gilt:

$$P(\text{Surfer startet bei } u) = \frac{1}{|V|}$$

$$P(\text{Surfer geht nach } v | \text{Surfer befindet sich in } u) = \begin{cases} \frac{1}{|N_{out}(u)|}, & \text{falls } (u, v) \in E \\ 0 & \text{sonst} \end{cases}$$

Prinzip des Rankings: Seiten, auf denen der Random Surfer häufig ist, sind „wichtig“.

Übergangsmatrix A

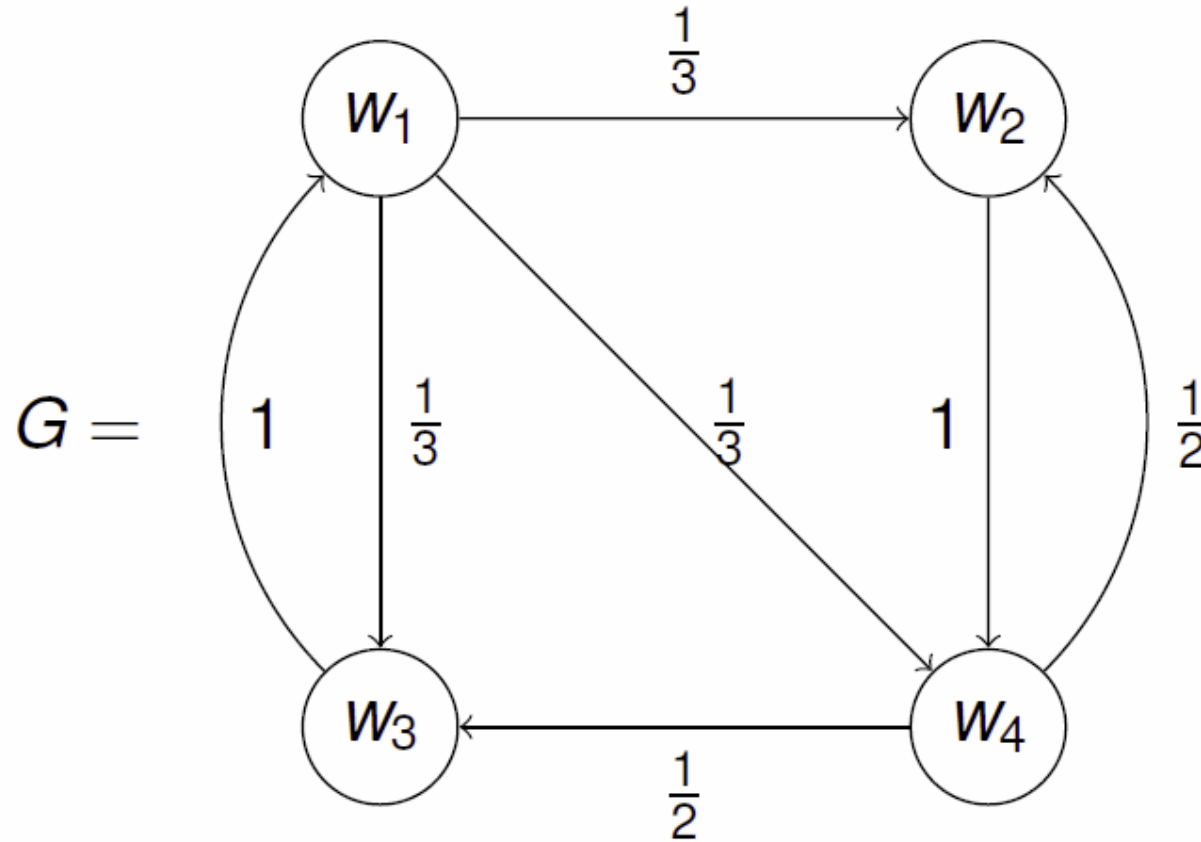
Die **Übergangsmatrix A** ist eine quadratische $(|V| \times |V|)$ -Matrix zur Darstellung der Verlinkung des Webspaces.

A ist wie folgt definiert:

$$A(v, u) = \begin{cases} \frac{1}{|N_{out}(u)|} & \text{falls } (u, v) \in E \\ 0 & \text{sonst} \end{cases}$$

Zu beachten ist, dass entgegen der Festlegung der Kanten als Knotenpaar (u, v) der zugehörige Eintrag in der Matrix A an Position (v, u) , d.h. Zeile v und Spalte u, auftritt.

Beispiel: Übergangsmatrix



$$A = \begin{pmatrix} 0 & 0 & 1 & 0 \\ \frac{1}{3} & 0 & 0 & \frac{1}{2} \\ \frac{1}{3} & 0 & 0 & \frac{1}{2} \\ \frac{1}{3} & 1 & 0 & 0 \end{pmatrix}$$

Übergangsmatrix

Aus der Definition des zufallsgetriebenen Surfers folgt für die Übergangsmatrix A :

- Der u -te Spaltenvektor enthält die **diskrete Wahrscheinlichkeitsverteilung** über alle ausgehenden Kanten des Knotens u .
- Der v -te Zeilenvektor benennt die möglichen **Quellen für Aufrufe von v** (Einträge > 0).

Die Komponentensumme eines Zeilenvektors ergibt nicht zwangsweise den Wert 1. Die Zeilenvektoren stellen daher auch keine diskrete Wahrscheinlichkeitsverteilung dar.

Vereinfachter PageRank

Der Surfer s wählt **zufällig und gleichverteilt** eine der vorhandenen Webseiten als **Startknoten**. Anschließend wählt s in jedem Schritt **zufällig** eine der **aktuell erreichbaren Webseiten** aus. Die Auswahl einer Webseite wird **gleichverteilt** über alle aktuell erreichbaren Webseiten getroffen.

Sei s ein Surfer, der sich im Webpace W entlang der Links bewegt. Der **PageRank $PR(v)$** einer Webseite v entspricht dem **Grenzwert der Auftrittswahrscheinlichkeit** von v nach unendlich vielen Bewegungen von s .

Vereinfachter PageRank

Sei W ein Webspace und $G = (V, E)$ der zugeordnete gerichtete Graph. Dann kann der **vereinfachte PageRank $PR(v)$** einer Webseite v approximiert werden mit

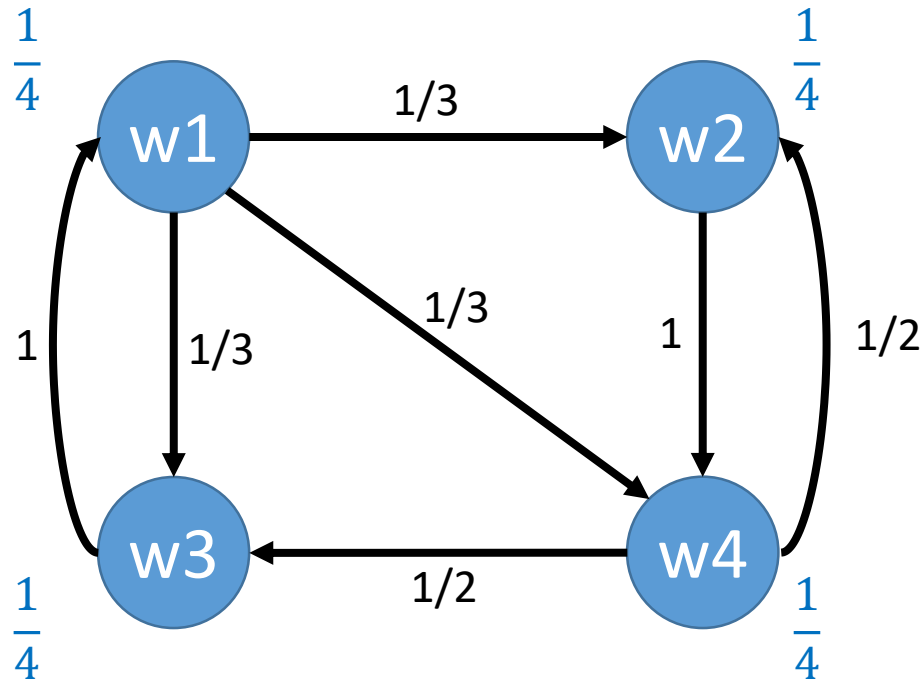
$$PR_1(v) = S(v) \quad (\text{z.B. } S(v) = \frac{1}{|V|})$$

$$PR_{n+1}(v) = \sum_{u \in N_{in}(v)} \left(\frac{1}{|N_{out}(u)|} \cdot PR_n(u) \right)$$

$$PR(v) = \lim_{n \rightarrow \infty} PR_n(v)$$

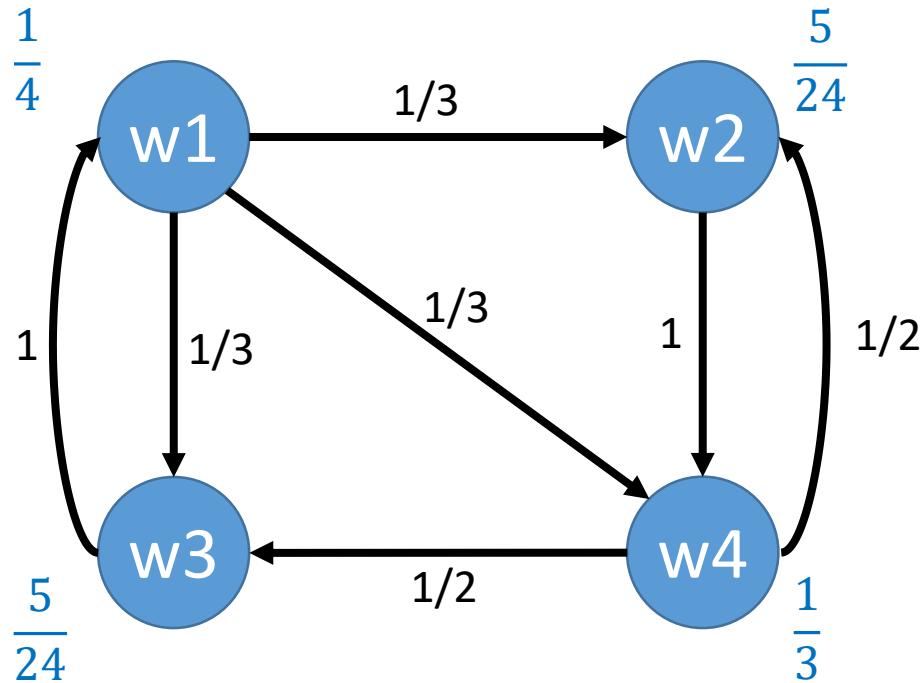
Für den initialen PageRank S muss gelten $\sum_{v \in V} S(v) = 1$

Vereinfachter PageRank: Beispiel



$$PR_1 = S = \begin{pmatrix} 1/4 \\ 1/4 \\ 1/4 \\ 1/4 \end{pmatrix}$$

Vereinfachter PageRank: Beispiel



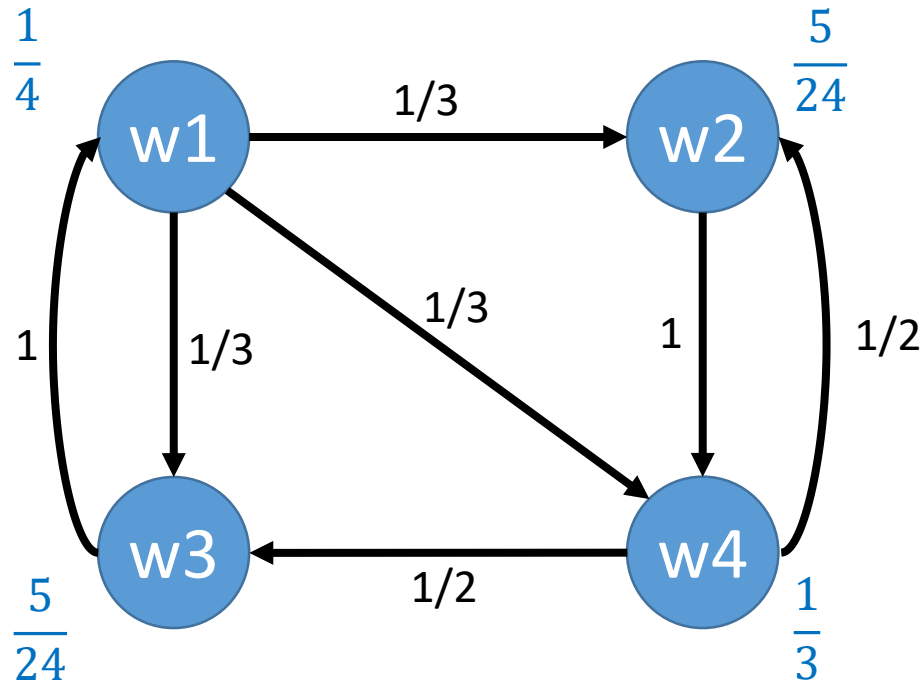
$$w_1 : \frac{1}{4} = 1 \cdot \frac{1}{4}$$

$$w_2 : \frac{5}{24} = \frac{1}{3} \cdot \frac{1}{4} + \frac{1}{2} \cdot \frac{1}{4}$$

$$w_3 : \frac{5}{24} = \frac{1}{3} \cdot \frac{1}{4} + \frac{1}{2} \cdot \frac{1}{4}$$

$$w_4 : \frac{1}{3} = \frac{1}{3} \cdot \frac{1}{4} + 1 \cdot \frac{1}{4}$$

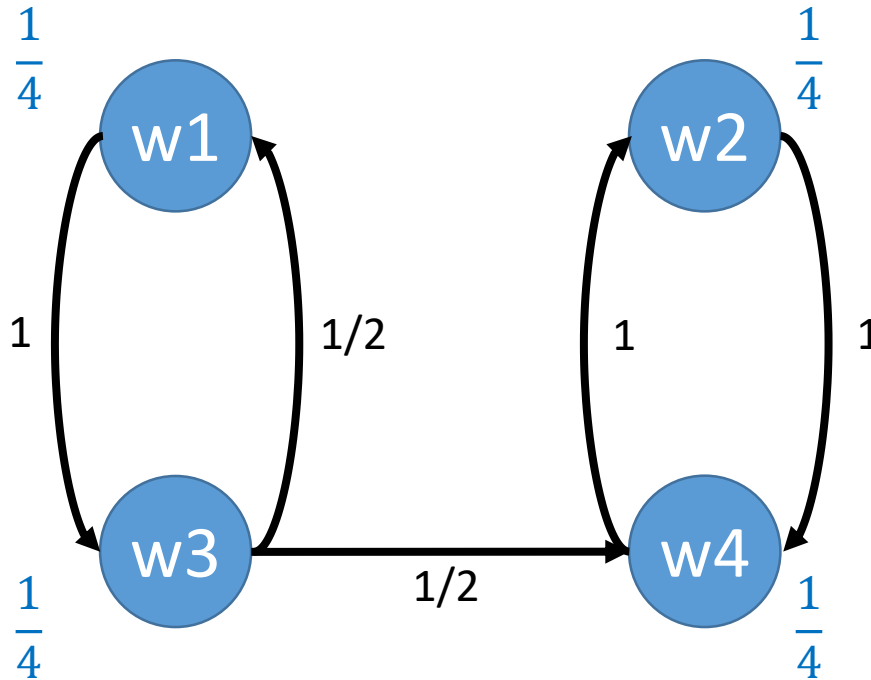
Vereinfachter PageRank: Beispiel



$$PR_2 = A \cdot S = \begin{pmatrix} 0 & 0 & 1 & 0 \\ \frac{1}{3} & 0 & 0 & \frac{1}{2} \\ \frac{1}{3} & 0 & 0 & \frac{1}{2} \\ \frac{1}{3} & 1 & 0 & 0 \end{pmatrix} \begin{pmatrix} \frac{1}{4} \\ \frac{1}{4} \\ \frac{1}{4} \\ \frac{1}{4} \end{pmatrix} = \begin{pmatrix} \frac{1}{4} \\ \frac{5}{24} \\ \frac{5}{24} \\ \frac{1}{3} \end{pmatrix}$$

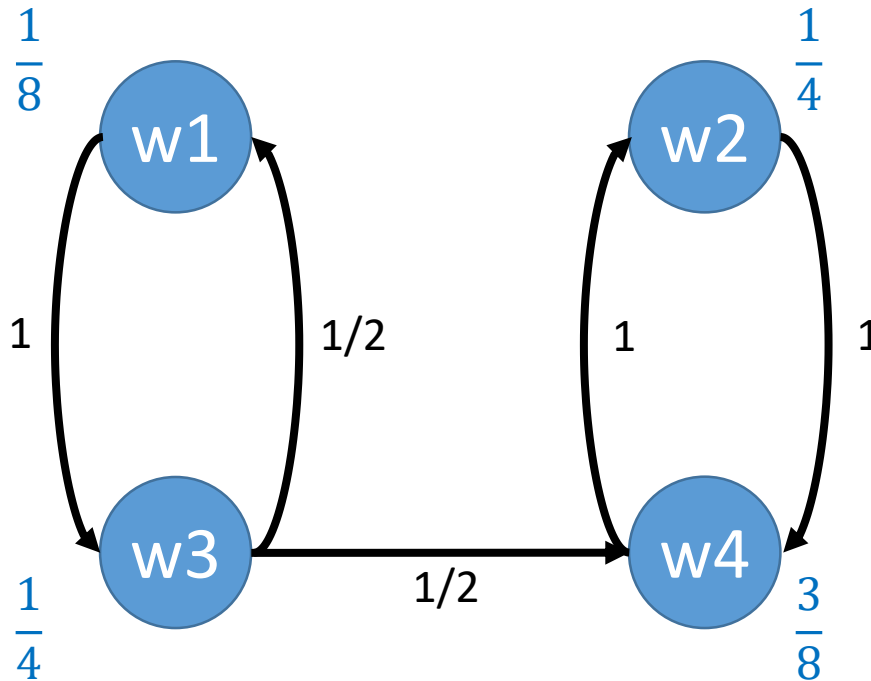
$$PR_{i+1} = A \cdot PR_i = A^i \cdot S \quad \text{und} \quad PR = \lim_{i \rightarrow \infty} A^i \cdot S$$

Vereinfachter PageRank: Zweites Beispiel



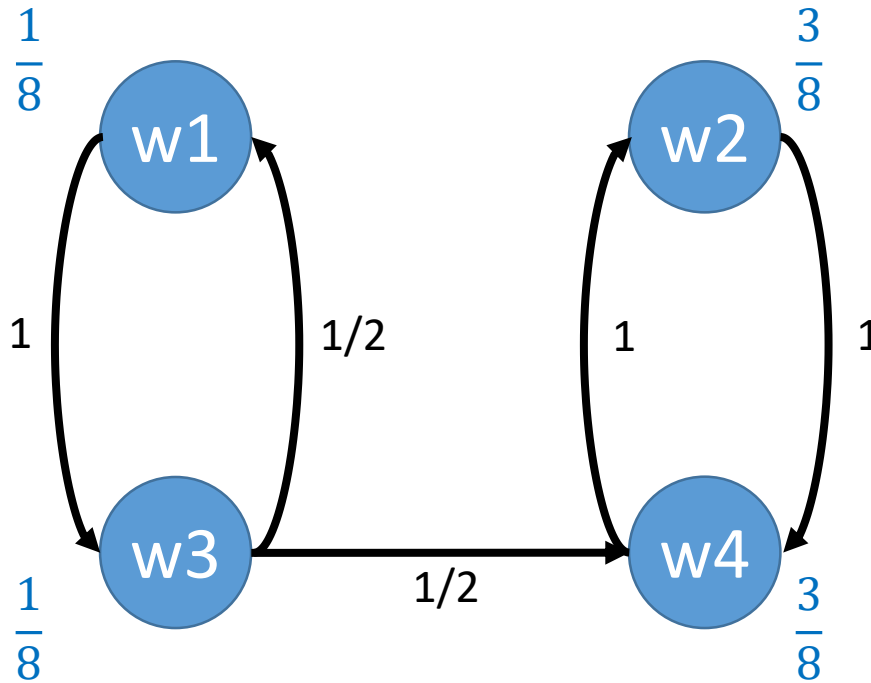
$$PR_1 = S = \begin{pmatrix} 1/4 \\ 1/4 \\ 1/4 \\ 1/4 \end{pmatrix}$$

Vereinfachter PageRank: Zweites Beispiel



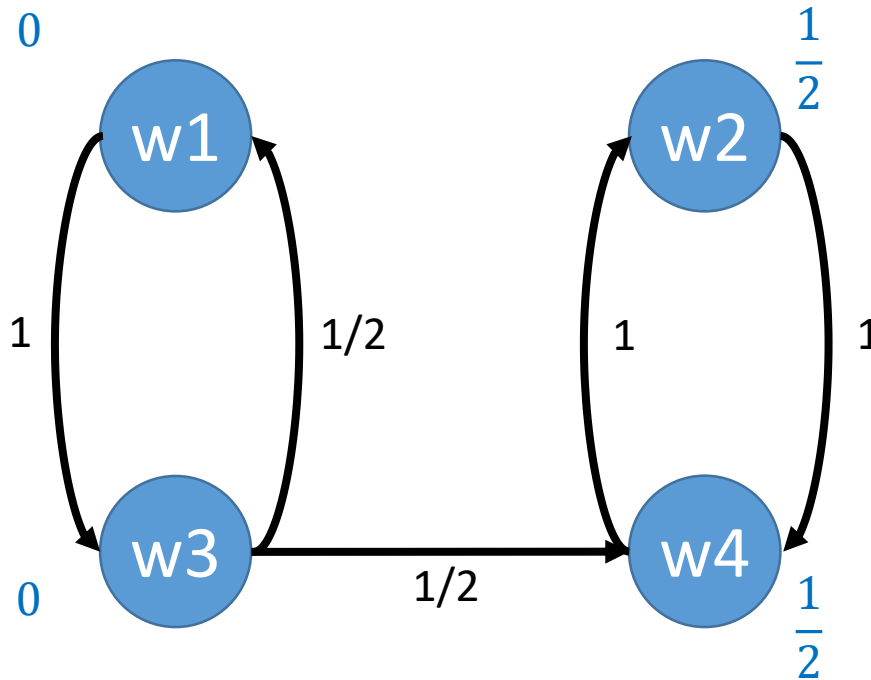
$$PR_2 = A \cdot S = \begin{pmatrix} 0 & 0 & 1/2 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 1/2 & 0 \end{pmatrix} \begin{pmatrix} 1/4 \\ 1/4 \\ 1/4 \\ 1/4 \end{pmatrix} = \begin{pmatrix} 1/8 \\ 1/4 \\ 1/4 \\ 3/8 \end{pmatrix}$$

Vereinfachter PageRank: Zweites Beispiel



$$PR_3 = A \cdot A \cdot S = \begin{pmatrix} 0 & 0 & 1/2 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 1/2 & 0 \end{pmatrix} \begin{pmatrix} 1/8 \\ 1/4 \\ 1/4 \\ 3/8 \end{pmatrix} = \begin{pmatrix} 1/8 \\ 3/8 \\ 1/8 \\ 3/8 \end{pmatrix}$$

Vereinfachter PageRank: Zweites Beispiel



$$PR = \lim_{i \rightarrow \infty} PR_i = \begin{pmatrix} 0 \\ 1/2 \\ 0 \\ 1/2 \end{pmatrix}$$

Rangsenken

Eine **Rangsenke** ist ein Zyklus bestehend aus Knoten Z , so dass gilt:

$$\forall u \in Z: (\neg v \in V \setminus Z: (u, v) \in E)$$

Die einfache PageRank-Berechnung verläuft fehlerhaft, wenn ein Knoten außerhalb des Zyklus einen Link zu einem Knoten im Zyklus besitzt. Die Knoten von Rangsenken **akkumulieren PageRank** aus dem Restgraphen.

Behandlung von Rangsenken

Rangsenken stellen einen **Verbund von Webseiten** dar, die nur aufeinander verlinken, aber **keine Links nach außerhalb** besitzen.

Ein Surfer hat somit nicht die Möglichkeit, aus dem Verbund auszubrechen, sobald er diesen einmal betreten hat. In der Realität wird der Surfer irgendwann einfach **eine andere Webseite direkt anspringen**.

Zur Modellierung erlaubt man dem Surfer in jedem Schritt mit Wahrscheinlichkeit α eine **Teleport-Operation** auszuführen, die ihn zu jeder beliebigen Webseite bringen kann.

Teleport-Operation

Die **Teleport-Operation** erlaubt es dem Surfer s , jede beliebige Webseite direkt anzuspringen. Sie wird mit Wahrscheinlichkeit α ausgeführt. Mit Wahrscheinlichkeit $1-\alpha$ folgt der Surfer weiter einer Zufallsbewegung entsprechend der ausgehenden Links.

$\alpha = P(\text{Surfer wählt zufällig eine beliebige Webseite aus})$

α wird auch als **Dämpfungsfaktor** bezeichnet. Die Berücksichtigung dieses Verhaltens führt zur finalen PageRank-Berechnung.

Normaler PageRank

Sei W ein Webspace und $G = (V, E)$ der zugeordnete gerichtete Graph. Dann kann der **PageRank-Vektor PR** über alle Webseiten approximiert werden mit

$$PR_1 = S$$

$$PR_{i+1} = (1 - \alpha)(A \cdot PR_i) + \alpha \cdot S$$

$$PR = \lim_{i \rightarrow \infty} PR_i$$

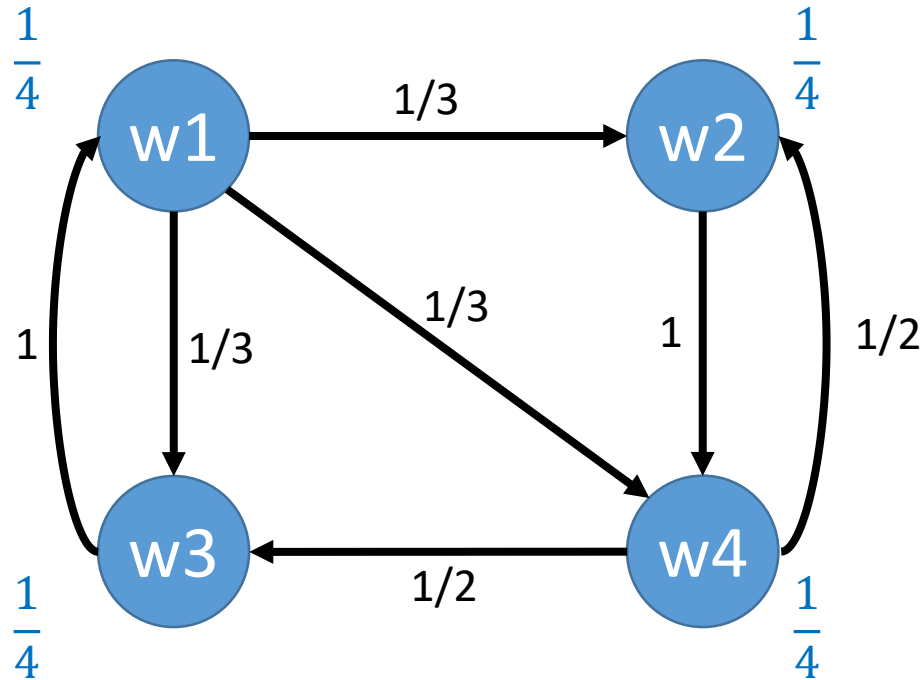
Mit **Grundrang** $S(v) = \frac{1}{|V|}$ und typischerweise $\alpha=0,15$

Normaler PageRank-Algorithmus

PAGERANK(Übergangsmatrix A , Grundrang S , Dämpfung α)

```
1
2   $i \leftarrow 0$ 
3   $PR_1 \leftarrow S$ 
4  while  $\delta > \epsilon$ 
5  do
6       $i \leftarrow i + 1$ 
7       $PR_{i+1} \leftarrow (1 - \alpha)(A \cdot PR_i) + \alpha \cdot S$ 
8
9      //  $\delta$  ist monoton fallend
10      $\delta \leftarrow \|PR_{i+1} - PR_i\|$ 
11 return  $PR_{i+1}$ 
```

Normaler PageRank: Ablauf



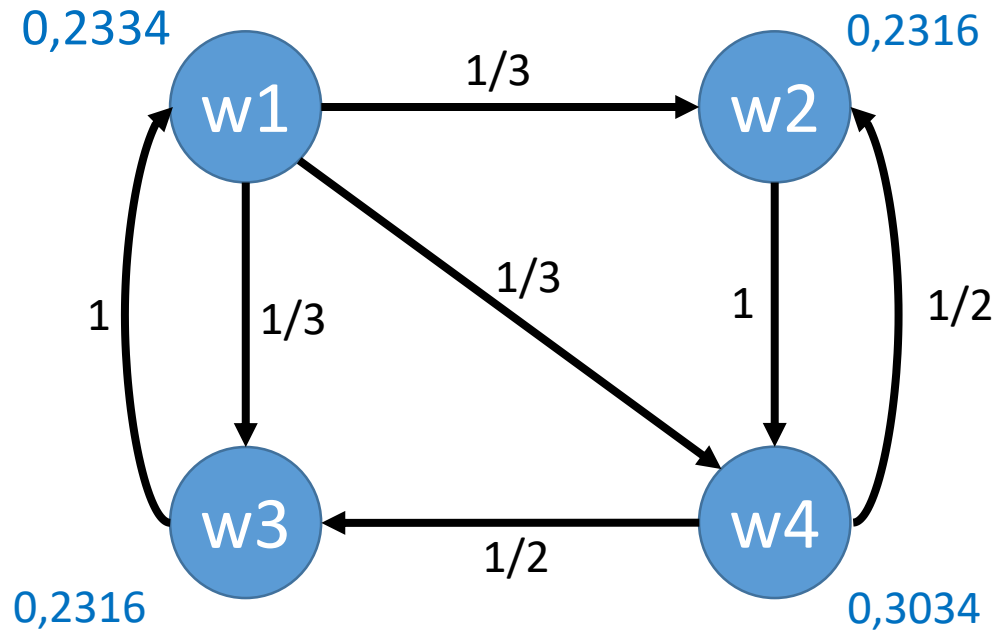
$$A = \begin{pmatrix} 0 & 0 & 1 & 0 \\ 1/3 & 0 & 0 & 1/2 \\ 1/3 & 0 & 0 & 1/2 \\ 1/3 & 1 & 0 & 0 \end{pmatrix} \quad S = \begin{pmatrix} 1/4 \\ 1/4 \\ 1/4 \\ 1/4 \end{pmatrix}$$

Normaler PageRank: Ablauf

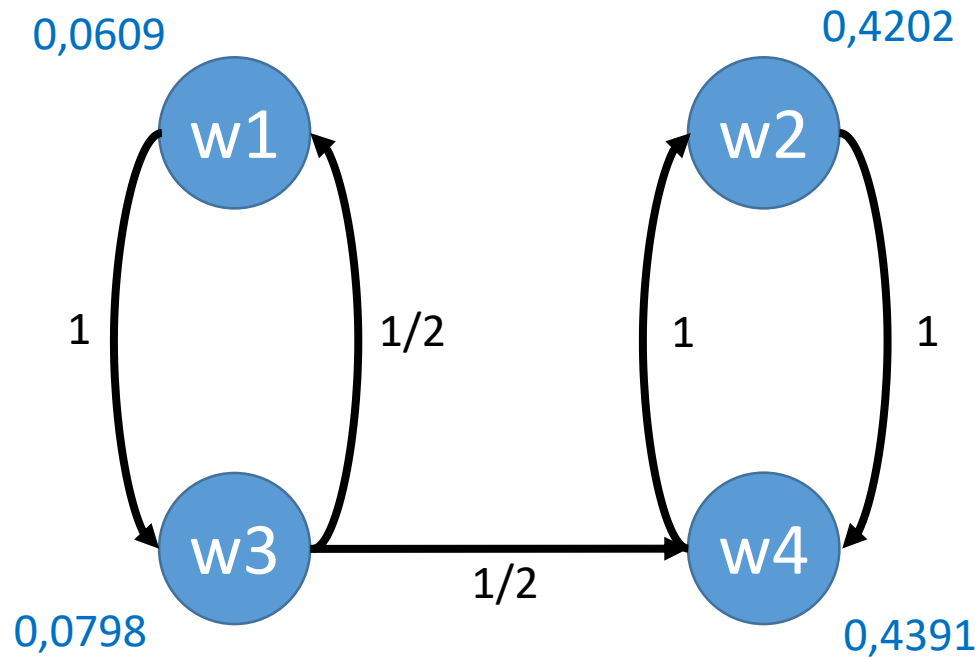
PR_i	w_1	w_2	w_3	w_4	δ
PR_1	0,2500	0,2500	0,2500	0,2500	—
PR_2	0,2500	0,2125	0,2125	0,3250	0,091856
PR_3	0,2163	0,2463	0,2463	0,2913	0,067500
PR_4	0,2466	0,2209	0,2209	0,3115	0,051123
PR_5	0,2238	0,2392	0,2392	0,2978	0,037015
PR_6	0,2402	0,2262	0,2262	0,3074	0,026416
PR_7	0,2286	0,2354	0,2354	0,3006	0,018778
PR_8	0,2369	0,2289	0,2289	0,3054	0,013334
...
PR_{29}	0,2334	0,2316	0,2316	0,3034	0,000009...

Annahmen: $\alpha = 0,1$, $S_k = 0,25$ für $k = 1 \dots |V|$, $\epsilon = 0,00001$

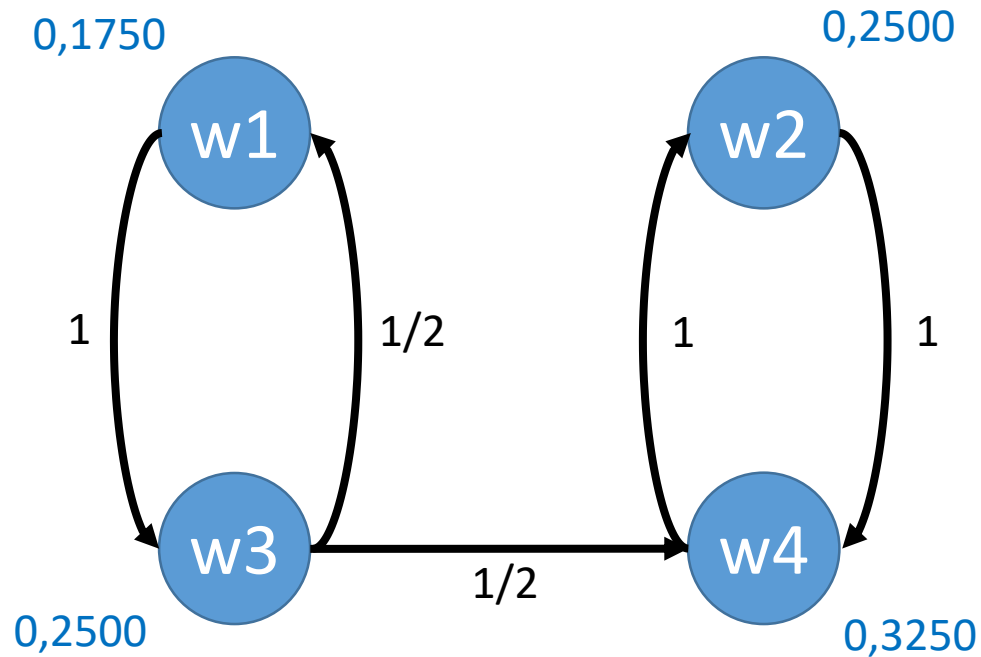
Normaler PageRank: Ablauf



Normaler PageRank: Zweites Beispiel, $\alpha=0,1$



Normaler PageRank: Zweites Beispiel, $\alpha=0,4$



Suche mit PageRank

Zur Suche nach Webseiten über PageRank gibt es 2 Ansätze:

Ansatz 1: PageRank für Sortierung

1. Suche alle zur Suchanfrage passenden Webseiten anhand eines IR-Verfahrens
2. Ordne die Webseiten in absteigender Reihenfolge entsprechend ihres PageRanks

Ansatz 2: PageRank für Relevanzbestimmung

1. Kombiniere PageRank mit einem IR-Verfahren zur Relevanzschätzung von Webseiten
2. Ordne die Webseiten in absteigender Reihenfolge entsprechend ihrer Relevanz

Suche mit PageRank

Der 1. Ansatz zur Suche mit PageRank ist einfach zu implementieren. Die Sortierung der Webseiten erfolgt aufgrund ihrer Wichtigkeit im Webspace **unabhängig von deren geschätzter Relevanz** für die Suchanfrage.

Der 2. Ansatz erfordert eine **komplexe Kombination von PageRank und IR-Scoring**. Die Sortierung der Webseiten erfolgt aufgrund ihrer Wichtigkeit im Webspace und unter Berücksichtigung von deren Relevanz für die Suchanfrage.

In der Praxis verwenden Suchmaschinen eine Variante des 2. Ansatzes.