Търсещи машини

Извличане на информация – Information Retrieval (IR)

- “Извличането на информация е област, занимаваща се със структурата, анализа, организацията, съхраняването, търсенето и извличането на информация” (Gerard Salton, 1968).
- Общата дефиниция може да бъде приложена към много типове информация и приложения за търсене.
- Основният фокус на IR от 1950 е върху текста и документите.

Какво е документ?

- Примери: web страници, е-мейли, книги, текстови съобщения, MS Word, PowerPoint, PDF, постинги, патенти и др.
- Общи свойства:
  - Значително текстово съдържание
  - Определена структура (заглавие, автор, тематика, подател и др.)

Документи и записи в бази данни

- Записите в базите данни (tuples) типично се състоят от добре дефинирани полета (attributes):
  - Банков запис с номера на сметки, баланси, имена, адреси, дата раждане и т.н.
- Лесни за сравняване полета с добре дефинирана семантика за заявките с цел откриване на съвпадение.
- Текстът е много по труден за обработка.

Документи и записи в бази данни - пример

- Пример за запитване към банкова база данни:
  - “Намери записи в баланс > 500000 във фирми, намиращи се в Лондон”;
  - Съвпаденията се намират лесно чрез сравнение със стойностите на полетата в записа.
- Пример за запитване към търсеща машина:
  - “Намери банковите скандали с България”;
  - Този текст трябва да бъде сравнен с текста във всички нови публикувани истории.

Сравняване на текст

- Сравняването на текста от запитването с текста в документа и определянето на добро съвпадение е ключов момент в IR.
- Не е необходимо точно съвпадение на думи:
  - Различни начини за запишем едно и също нещо с естествен език;
  - Някои публикации ще имат по-добро съвпадение от други.

Други медии

- Нови приложения, изискващи нови медии:
  - Видео, музика, изображения, говор.
- Съдържанието, както текста, е трудно за описание и сравнение:
  - Текст може да се използва за представяне на съдържание (тагове).

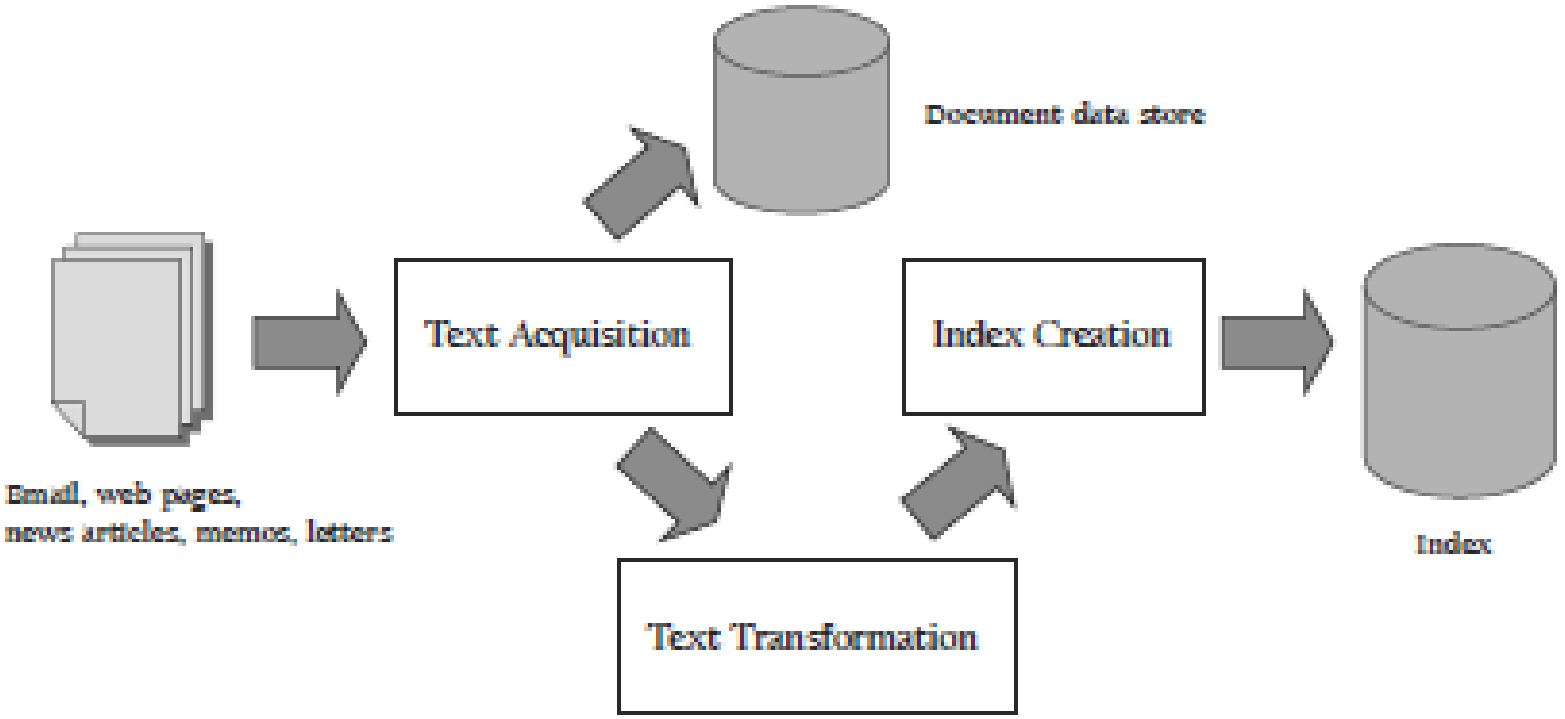
Архитектура на ТМ

- Софтуерната архитектура съдържа софтуерни компоненти, интерфейси, предоставени от компонентите и отношения между тях.
- Архитектурата се определя от 2 основни изисквания:
  - Ефикасност (време за отговор).
  - Ефективност (качество на резултатите).

Основни функции на ТМ

- Процес на индексиране.
- Процес на запитване.

Процес на индексиране



Изискване на текст

- Crawler:
  - Идентифицира и изисква документи за ТМ;
  - Различни типове – web, enterprise, desktop;
  - Web crawler-ите следват линковете за да откриват документи:
    - Трябва ефикасно да откриват огромен брой страници (coverage) и да ги поддържат актуални (freshness).

Изискване на текст

- Feeds
  - Потоци документи в реално време (блогове, жидео, радио, телевизия).
  - RSS reader е общ стандарт.
- Конвертиране
  - Конвертират се документи в консистентен текстов формат с метаданни (HTML, Word -> XML)
  - Конвертира се текстовото кодиране за различни езици.

Изискване на текст

- Хранилища на документи
  - Съхраняват текст, метаданни и друга съответстваща информация за документите
    - Метаданни – информация за документа (дата на създаване);
    - Друга информация (линкове, котви).
  - Предоставят бърз достъп до съдържанието на документите за всички компоненти на ТМ.
  - Могат да се използват релационни бази данни.

Трансформация на текст

- ***Парсер***
  - Обработване на последователност от текстови лексеми (tokens) за разпознаване на структурни елементи (заглавия, линкове и др.).
  - Думите се разпознават от *Tokenizer*.
  - Таговите езици (HTML, XML) често се използват за описание на структурата.

Трансформация на текст

- ***Stopping***
  - Премахване на общи думи.
  - Влияе на ефективността и ефикасността.
  - Може да е проблем при някои запитвания.

Трансформация на текст

- ***Stemming***
  - Групира думи, производни на общ корен.
  - Обикновено е ефективно, но не при всички запитвания.
  - Предимствата зависят от различните езици.

Трансформация на текст

- ***Анализ на връзки (Link analysis)***
  - Използва линковете и котвите в WEB страниците.
  - Анализът определя *популярността* (PageRank).
  - Текстът в котвите може значително да разшири обхвата на страниците, сочени от линковете.
  - Има значително влияние при търсенето в WEB.

Трансформация на текст

- ***Извличане на информацията***
  - Идентифицира класове на индексни термини, които са важни за определени приложения (класове като хора, компании, дати)

Трансформация на текст

- ***Класификатор***
  - Идентифицира класово-базирани метаданни за документите (присвоява етикети към документи)
  - Зависи от конкретното приложение.

Създаване на индекс

- ***Статистика за документите***
  - Събира се брой и позиция на думите и други свойства
  - Използва се в алгоритмите за рейтинговане.

Създаване на индекс

- ***Задаване на тегла (Weighting)***
  - Изчислява тегла за индексните термини.
  - Използва се в алгоритмите за рейтинговане

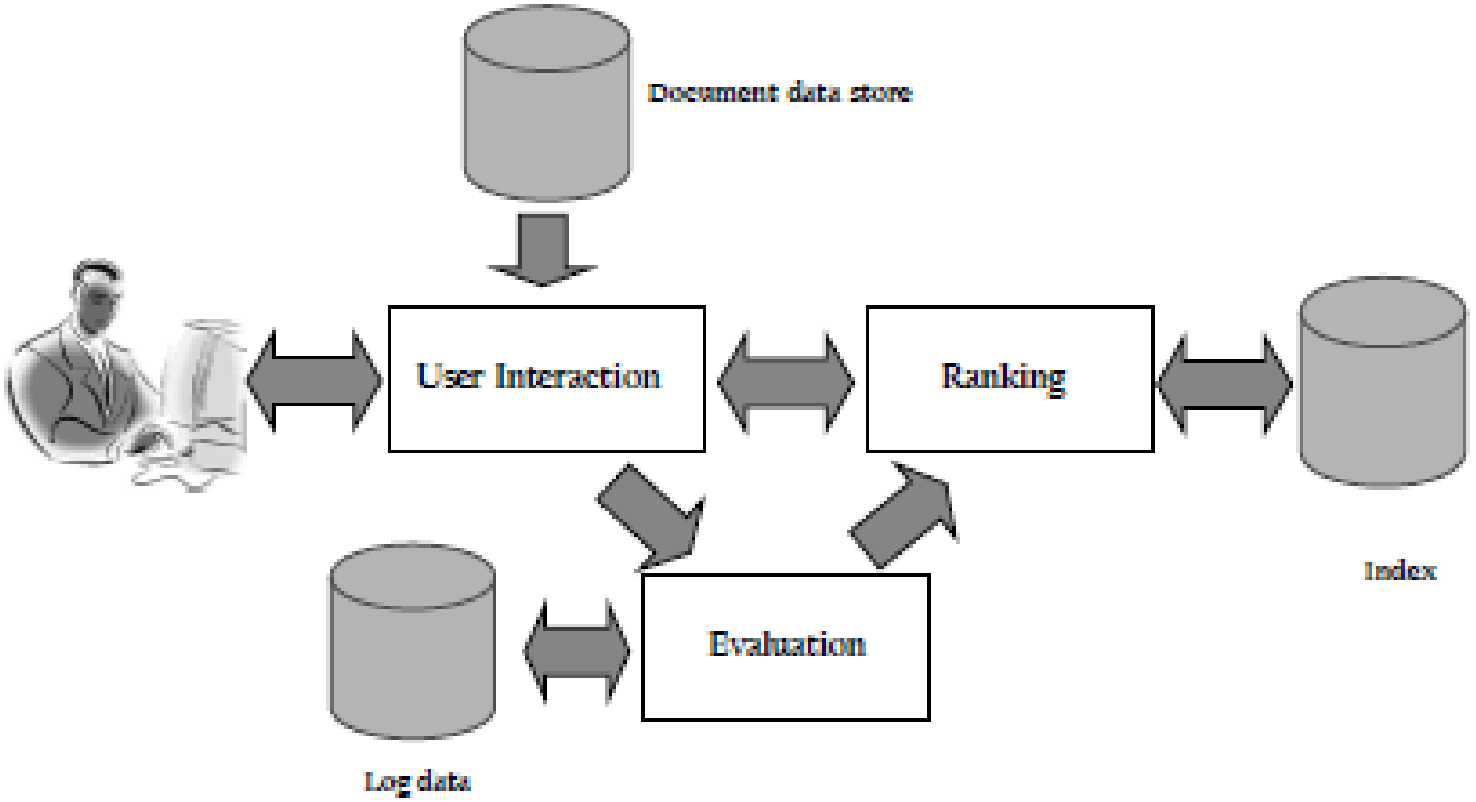
Създаване на индекс

- ***Инвертиране***
  - Същината на процеса на индексиране.
  - Преобразува информацията от вида документи-термини във вида термини-документи.
  - Форматът на инвертния файл е проектиран за бърза обработка на запитвания.

Създаване на индекс

- ***Разпределяне на индекса***
  - Разпределяне на индексите между множество компютри и/или сайтове.
  - Изключително важност за бърза обработка на запитвания с огромен брой документи.
  - Вариации (разпределян на документи, на термини, репликации).

Процес на запитване



Взаимодействие с потребителя

- ***Въвеждане на запитвания***
  - Предоставя интерфейс и парсер на езика за запитвания.
  - Повечето web запитвания са много опростени (кавички).
  - Някои приложения могат да използват форми.
  - За описание на сложни запитвания се използват езици за запитвания (query languages).

Взаимодействие с потребителя

- ***Трансформация на запитванията***
  - Подобрява началното запитване (и преди и след първоначалното търсене).
  - Използват се техники за трансформиране на текст, както при документите.
  - *Spell checking*, *query suggestion* предоставят алтернатива на оригиналните запитвания.
  - *Query expansion, relevance feedback* модифицират оригиналното запитване с допълнителни термини.

Взаимодействие с потребителя

- ***Извеждане на резултати***
  - Формиране на извеждане на рейтинговани документи за запитването.
  - Генериране на *snippets* за показване как запитванията съответстват на документите.
  - Подчертаване на важни думи и пасажи.
  - Получаване на подходящи реклами в много приложения.

Рейтинговане (Ranking)

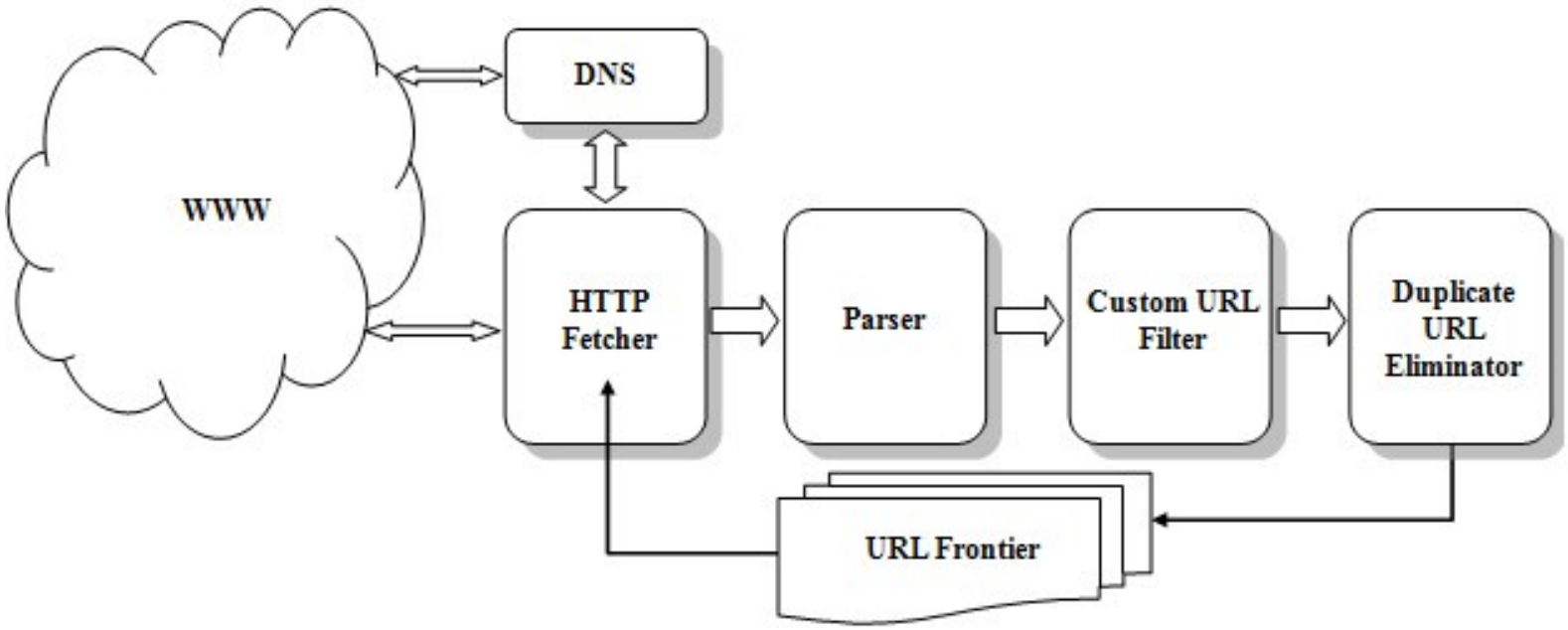
- ***Формиране на оценка***
  - Изчисляват се оценки за документите на базата на алгоритми за рейтинговане.
  - Ключов компонент на всяка ТМ.
  - Основен вид на оценката е

    $\Sigma\ q_i\ d_i$ – теглата на термините на запитването и на документа за термин i.

Оценяване

- ***Logging***
  - Съхраняване на потребителските запитвания и взаимодействие (*clickthrough data*, *dwell time*).
- ***Анализ на рейтингуването***
  - Измерване и настройване на ефективността на рейтингуването.
- ***Анализ на производителността***
  - Измерване и настройване на ефективността на система за търсене (тестови колекции).

Архитектура на Web crawler



Особености

- Налице е изчакване за получаване на отговора.
- Редуцирането на това време – нишки.
- Потенциално засипват сайтовете със заявки за страници (flooding).
- Избягване на това – изчакване между заявките към един и същ web сървър (*politeness policies)*.

Пример за нишка на crawler

```
procedure CrawlerThread(frontier)
    while not frontier.done() do
        website ← frontier.nextSite()
        url ← website.nextURL()
        if website.permitsCrawl(url) then
            text ← retrieveURL(url)
            storeDocument(url, text)
            for each url in parse(text) do
                frontier.addURL(url)
            end for
        end if
        frontier.releaseSite(website)
    end while
end procedure
```

Актуалност на страниците (freshness)

- Използва се специална заявка на протокола HTTP – HEAD.

Client request:
```
HEAD /csinfo/people.html HTTP/1.1
Host: www.cs.umass.edu
```

Server response:
```
HTTP/1.1 200 OK
Date: Thu, 03 Apr 2008 05:17:54 GMT
Server: Apache/2.0.52 (CentOS)
Last-Modified: Fri, 04 Jan 2008 15:28:39 GMT
ETag: "239c33-2576-2a2837c0"
Accept-Ranges: bytes
Content-Length: 9590
Connection: close
Content-Type: text/html; charset=ISO-8859-1
```

Deep Web

- Трудни за извличане web сайтове:
  - Частни сайтове;
  - Резултати от форми;
  - Скриптови страници.

Обработка на текст

- Парсване на текста (*parsing*) – разпознаване на съдържанието и структурата на документите.
- Tokenizing (lexical analysis) – формиране на думи.
- Съдържание на документ:
  - Думи (tokens);
  - Метаданни (автор, тагове).
- Синтактичен анализ

Проблеми

- Кратки думи могат да бъдат важни в някои запитвания (*xp, pm, world war II*).
- Има думи с и без тирета:
  - В някои случаи не са необходими (*e-bay, active-x, cd-rom*).
  - В други са част от думата или разделител (*mazda rx-7, e-cards, t-mobile*)

Проблеми

- Специални символи, важни за тагове, URL и код.
- Главни букви, имащи различно значение (*Bush, Apple*).
- Апострофите могат да са част от дума, притежание или грешка (*can’t, 80’s, o’donnell*).

Проблеми

- Числата имат важно значение (*Porsche 911, top 10 courses, Windows 10*).
- Точките могат да се появяват в числа, абревиатури, URL, край на изречение (*I.B.M., Ph.D. , cs.tu-varna.bg*)

Stemming

- Алгоритмични
- Базирани на речници

- Suffix-s stemmer

  cakes -> cake

  dogs -> dog

Но: *century – centuries* ?

Porter stemmer

- Последователност от стъпки
- На всяка стъпка се премахват или заменят окончания.

- Заменя *sses* с *ss* (*stresses -> stress*).

Krovetz stemmer

- Хибриден подход.
- Използва речник за определяне дали думата е валидна.
- Помощни списъци с деривационни суфикси.
- Предимство – корените в повечето случай са пълни думи.

Stemmers

Original text:

Document will describe marketing strategies carried out by U.S. companies for their agricultural chemicals, report predictions for market share of such chemicals, or report market statistics for agrochemicals, pesticide, herbicide, fungicide, insecticide, fertilizer, predicted sales, market share, stimulate demand, price cut, volume of sales.

Porter stemmer:

document describ market strategi carri compani agricultur chemic report predict market share chemic report market statist agrochem pesticid herbicid fungicid insecticid fertil predict sale market share stimul demand price cut volum sale

Krovetz stemmer:

document describe marketing strategy carry company agriculture chemical report prediction market share chemical report market statistic agrochemic pesticide herbicide fungicide insecticide fertilizer predict sale stimulate demand price cut volume sale

Други езици

kitab	*a book*
kitabi	*my book*
alkitab	*the book*
kitabuki	*your book* (f)
kitabuka	*your book (m)*
kitabuhu	*his book*
kataba	*to write*
maktaba	*library, bookstore*
maktab	*office*

ktb

Изграждане на индекси

- Проектирани са да поддържат *търсене.*

- Търсещите машини използва специфична форма на търсене – *ranking*
  - Документите се извличат в подреден ред в зависимост от оценката на документа.

Текст за индексиране

d1: The jaguar is a New World mammal of the Felidae family.

d2: Jaguar has designed four new engines.

d3: For Jaguar, Atari was keen to use a 68K family device.

d4: The Jacksonville Jaguars are a professional US football team.

d5: Mac OS X Jaguar is available at a price of US $199 for Apple's new "family pack".

d6: One such ruling family to incorporate the jaguar into their name is Jaguar Paw.

d7: It is a big cat.

Отделяне на думите

d1: the$_1$ jaguar$_2$ is$_3$ a$_4$ new$_5$ world$_6$ mammal$_7$ of$_8$ the$_9$ felidae$_{10}$ family$_{11}$

d2: jaguar$_1$ has$_2$ designed$_3$ four$_4$ new$_5$ engines$_6$

d3: for$_1$ jaguar$_2$ atari$_3$ was$_4$ keen$_5$ to$_6$ use$_7$ a$_8$ 68k$_9$ family$_{10}$ device$_{11}$

d4: the$_1$ jacksonville$_2$ jaguars$_3$ are$_4$ a$_5$ professional$_6$ us$_7$ football$_8$ team$_9$

d5: mac$_1$ os$_2$ x$_3$ jaguar$_4$ is$_5$ available$_6$ at$_7$ a$_8$ price$_9$ of$_{10}$ us$_{11}$ 199_{12}$ for$_{13}$ apple's$_{14}$ new$_{15}$ family$_{16}$ pack$_{17}$

d6: one$_1$ such$_2$ ruling$_3$ family$_4$ to$_5$ incorporate$_6$ the$_7$ jaguar$_8$ into$_9$ their$_{10}$ name$_{11}$ is$_{12}$ jaguar$_{13}$ paw$_{14}$

d7: it$_1$ is$_2$ a$_3$ big$_4$ cat$_5$

Нормализиране на думи

d1: the$_1$ jaguar$_2$ be$_3$ a$_4$ new$_5$ world$_6$ mammal$_7$ of$_8$ the$_9$ felidae$_{10}$ family$_{11}$

d2: jaguar$_1$ have$_2$ design$_3$ four$_4$ new$_5$ engine$_6$

d3: for$_1$ jaguar$_2$ atari$_3$ be$_4$ keen$_5$ to$_6$ use$_7$ a$_8$ 68k$_9$ family$_{10}$ device$_{11}$

d4: the$_1$ jacksonville$_2$ jaguars$_3$ be$_4$ a$_5$ professional$_6$ us$_7$ football$_8$ team$_9$

d5: mac$_1$ os$_2$ x$_3$ jaguar$_4$ be$_5$ available$_6$ at$_7$ a$_8$ price$_9$ of$_{10}$ us$_{11}$ 199_{12}$ for$_{13}$ apple$_{14}$ new$_{15}$ family$_{16}$ pack$_{17}$

d6: one$_1$ such$_2$ rule$_3$ family$_4$ to$_5$ incorporate$_6$ the$_7$ jaguar$_8$ into$_9$ their$_{10}$ name$_{11}$ be$_{12}$ jaguar$_{13}$ paw$_{14}$

d7: it$_1$ be$_2$ a$_3$ big$_4$ cat$_5$

Премахване на общите термини

d1: jaguar$_2$ new$_5$ world$_6$ mammal$_7$ felidae$_{10}$ family$_{11}$

d2: jaguar$_1$ design$_3$ four$_4$ new$_5$ engine$_6$

d3: jaguar$_2$ atari$_3$ keen$_5$ 68k$_9$ family$_{10}$ device$_{11}$

d4: jacksonville$_2$ jaguars$_3$ professional$_6$ us$_7$ football$_8$ team$_9$

d5: mac$_1$ os$_2$ x$_3$ jaguar$_4$ available$_6$ price$_9$ us$_{11}$ \$199$_{12}$ apple$_{14}$ new$_{15}$ family$_{16}$ pack$_{17}$

d6: one$_1$ such$_2$ rule$_3$ family$_4$ incorporate$_6$ jaguar$_8$ their$_{10}$ name$_{11}$ jaguar$_{13}$ paw$_{14}$

d7: big$_4$ cat$_5$

Индекси

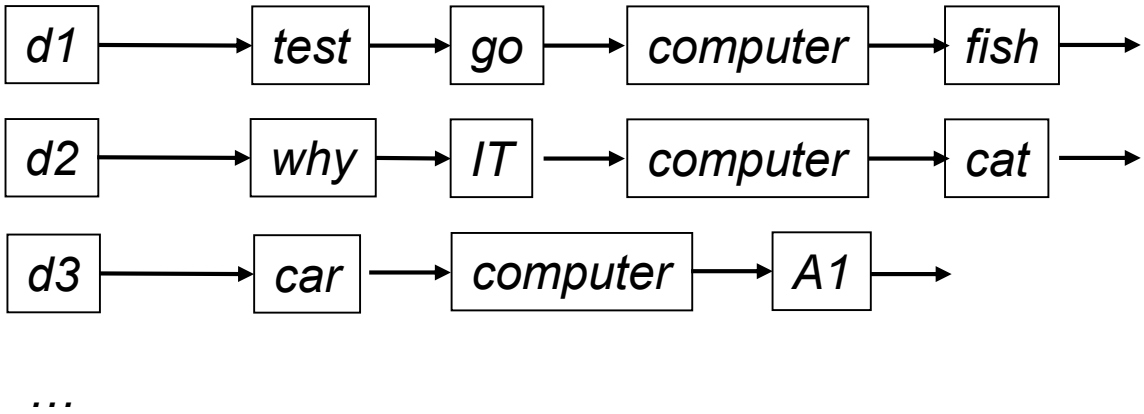
Forward index – списък от документи, във всеки документ се съдържат думи.

документи -> към -> думи

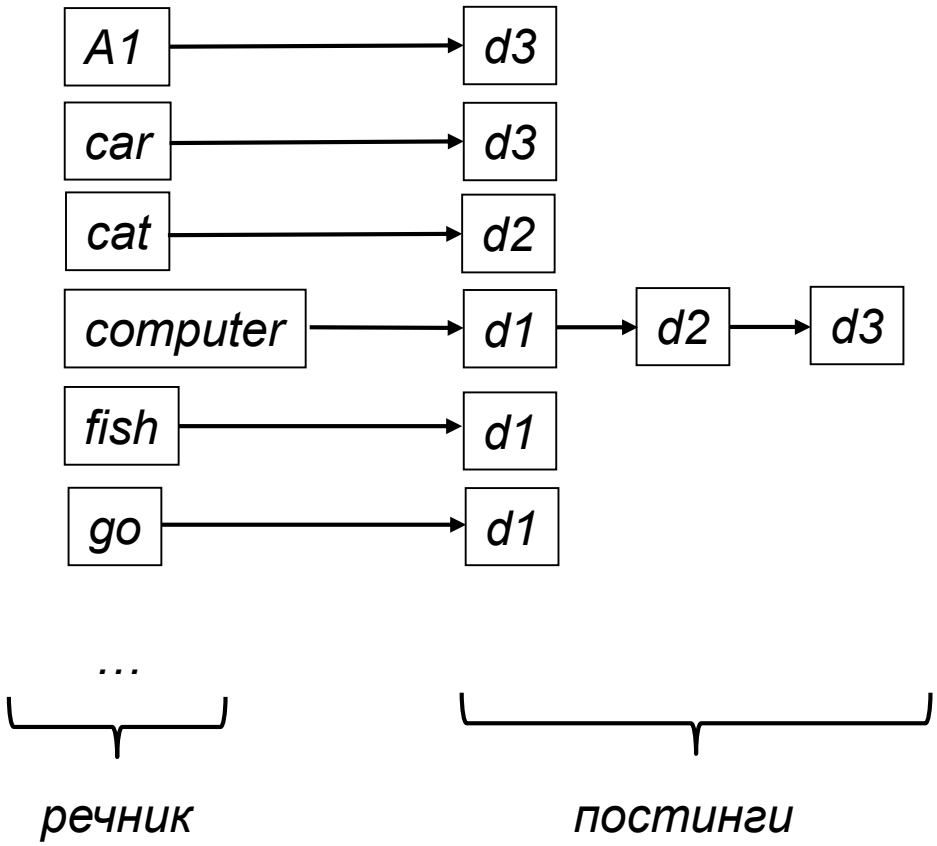
Inverted index – списък от думи и документи в които те се съдържат.

думи -> към -> документи

Forward index



Inverted index



Инвертен индекс - пример

d1: ***jaguar*** $_2$ ***new*** $_5$ ***world*** $_6$ ***mammal*** $_7$ ***felidae*** $_{10}$ ***family*** $_{11}$

d2: ***jaguar*** $_1$ ***design*** $_3$ ***four*** $_4$ ***new*** $_5$ ***engine*** $_6$

d3: ***jaguar*** $_2$ ***atari*** $_3$ ***keen*** $_5$ ***68k*** $_9$ ***family*** $_{10}$ ***device*** $_{11}$

d4: ***jacksonville*** $_2$ ***jaguars*** $_3$ ***professional*** $_6$ ***us*** $_7$ ***football*** $_8$ ***team*** $_9$

d5: ***mac*** $_1$ ***os*** $_2$ ***x*** $_3$ ***jaguar*** $_4$ ***available*** $_6$ ***price*** $_9$ ***us*** $_{11}$ ***$199*** $_{12}$ ***apple*** $_{14}$ ***new*** $_{15}$ ***family*** $_{16}$ ***pack*** $_{17}$

d6: ***one*** $_1$ ***such*** $_2$ ***rule*** $_3$ ***family*** $_4$ ***incorporate*** $_6$ ***jaguar*** $_8$ ***their*** $_{10}$ ***name*** $_{11}$ ***jaguar*** $_{13}$ ***paw*** $_{14}$

d7: ***big*** $_4$ ***cat*** $_5$

family	d_1, d_3, d_5, d_6
football	d_4
jaguar	$d_1, d_2, d_3, d_4, d_5, d_6$
new	d_1, d_2, d_5
rule	d_6
us	d_4, d_5
world	d_1
…	

Позиция на думата

family	*d_1/11, d_3/10, d_5/16, d_6/4*
football	*d_4/8*
jaguar	*d_1/2, d_2/1, d_3/2, d_4/3, d_5/4, d_6/8*
new	*d_1/5, d_2/5, d_5/15*
rule	*d_6/3*
us	*d_4/7, d_5/11*
world	*d_1/6*
…	

Релевантност

- ***Дефиниция**: Релевантният документ съдържа информацията, която потребителят търси, когато изпраща заявка към търсещата машина.*
- Редица фактори оказват влияние на преценката на потребителя кое е релевантно – съдържание, новост, стил и др.
- ***Тематична** релевантност* (същата тема) и ***Потребителска** релевантност* (всичко допълнително).

Оценка релевантността на документите

- Релевантността се измерва чрез присвояване на тегло към срещанията на термина в документа.
- Често срещан термин в документа е повече релевантен за индексиране на документа.
- Рядко срещан термин в колекция засилва релевантостта на документ.
- Често срещан термин в много документи е по-малко отличителен.

Оценка релевантността на документите

term frequency - inverse document frequency (*tf-idf*)

- **term frequency –** честотата на срещанията на термин към общия брой термини в документ

$$tf(t,d) = \frac{n_{t,d}}{\sum_{t'} n_{t',d}}$$

Оценка релевантността на документите

term frequency - inverse document frequency (*tf-idf*)

- **inverse document frequency –** важността на термин в колекция от документи

$$idf(t) = \log \frac{|D|}{|\{d' \in D | n_{t,d'} > 0\}|}$$

Оценка релевантността на документите

term frequency - inverse document frequency (*tf-idf*)

tf - idf

$$tfidf(t,d) = \frac{n_{t,d}}{\sum_{t'} n_{t',d}} \cdot \log \frac{|D|}{|\{d' \in D | n_{t,d'} > 0\}|}$$

Модифициран индекс

family	*d_1/11/.13, d_3/10/.13, d_5/16/.7, d_6/4/.8*
football	*d_4/8/.47*
jaguar	*d_1/2/.04, d_2/1/.04, d_3/2/.04, d_4/3/.04, d_5/4/.04, d_6/8/.04*
new	*d_1/5/.20, d_2/5/.24, d_5/15/.10*
rule	*d_6/3/.28*
us	*d_4/7/.30, d_5/11/.15*
world	*d_1/6/.47*
…	

Размер на индекса

Пример:
Колекция от е-мейли. Всеки мейл е с размер средно 1Kb и всеки мейл съдържа средно по 100 думи.

Колекцията съдържа 1 000 000 мейла в обем 1Gb.
Брой думи: 100×10^6
След парсване и токенизиране: 200 000 различни термина

1. Индексът съдържа 200 000 списъка
2. При допускане, че 20% от термините в документ се срещат 2 пъти, то всеки документ се среща в 80 списъка
3. Веки списък съдържа средно 400 записа
4. При представяне на ID на документ с 4byte integer, то средния размер на списък е 1600bytes
5. Целият индекс ще съдържа 400 x 200 000 = 80 000 000 записа
6. **Размерът на индекса е 320 Mb**

Размер на индекса

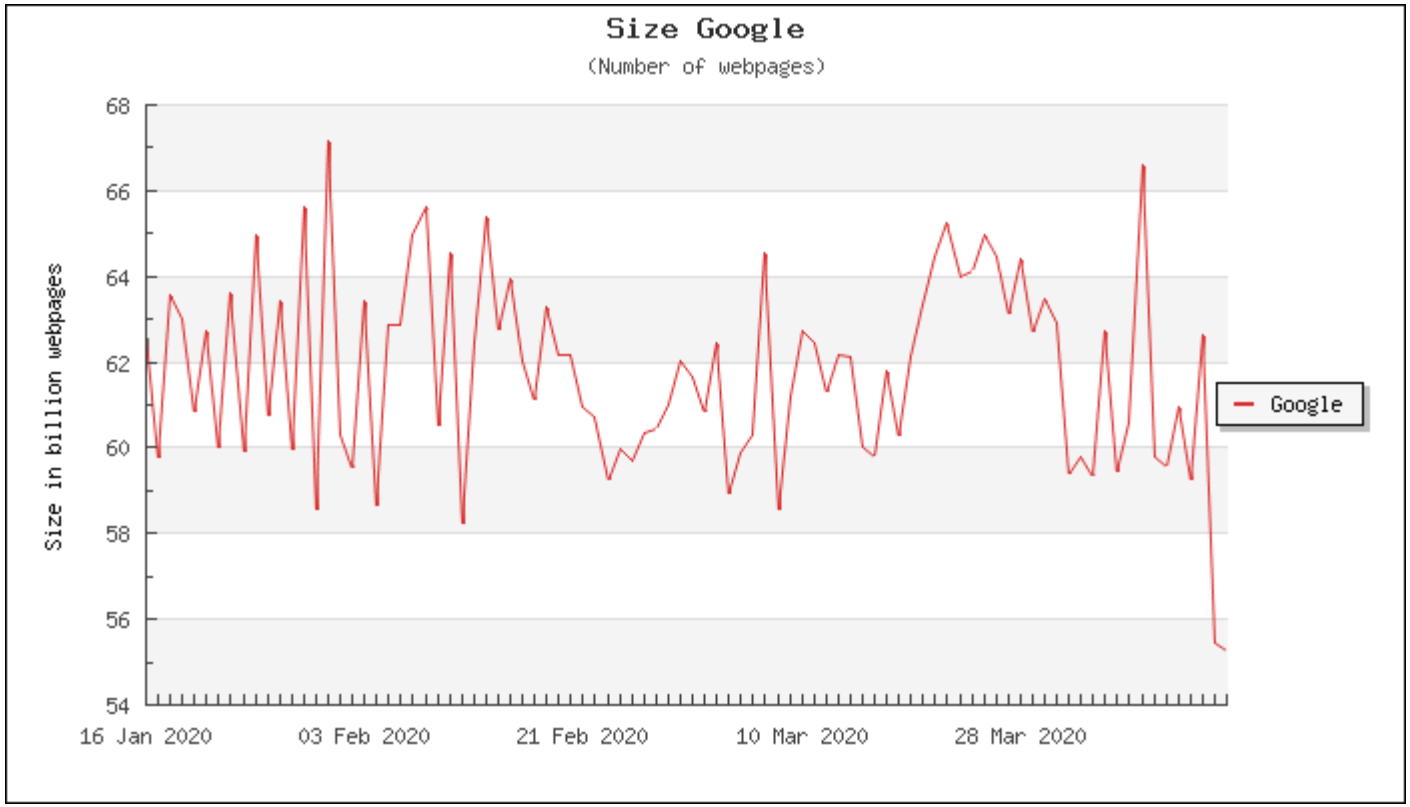
Добавяне на допълнителна информация към елементите в списъците:

- Позиция
- Тегло

Допълнително заета памет: 8 bytes

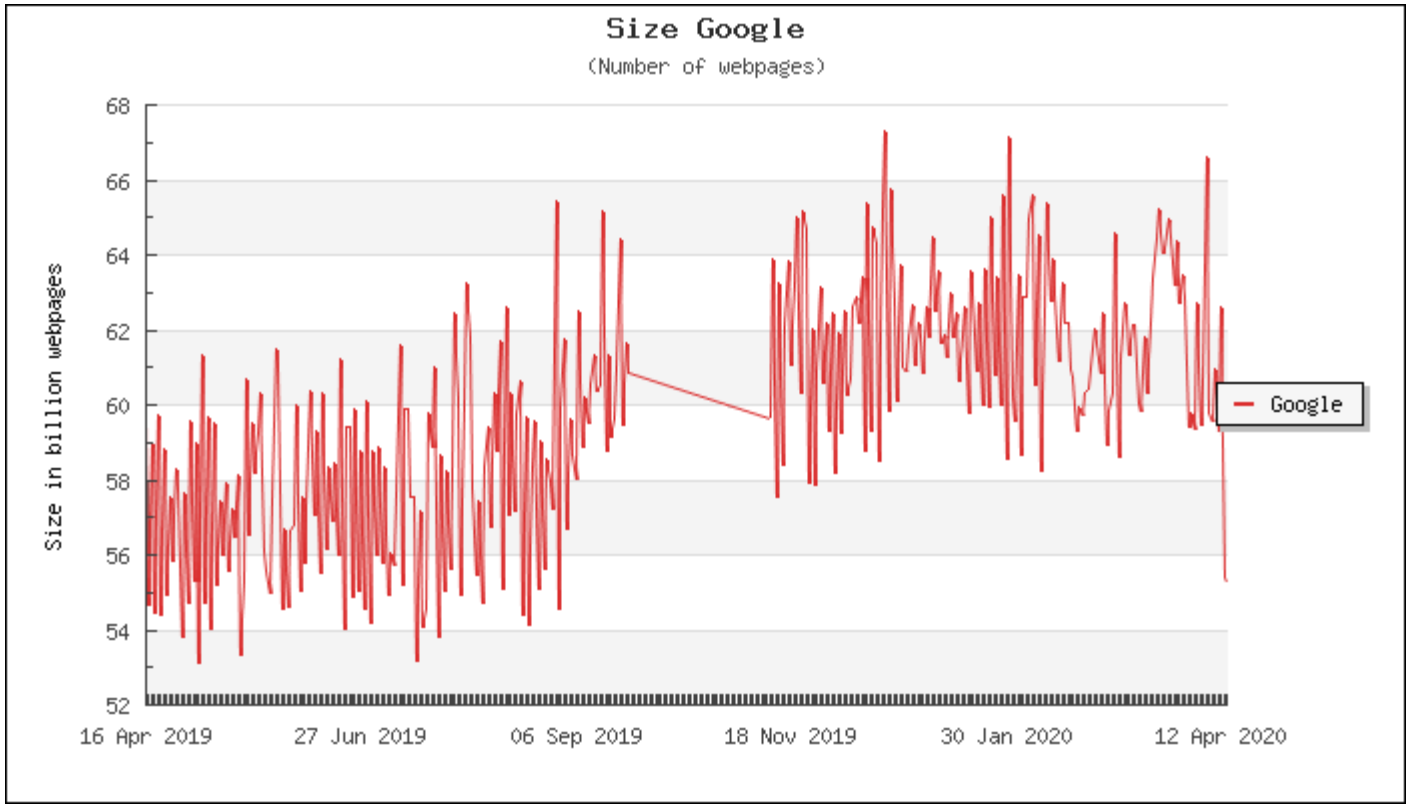
Целият индекс ще съдържа 80 x 12 x 10^6 = 960 Mb

Индексиране в WWW

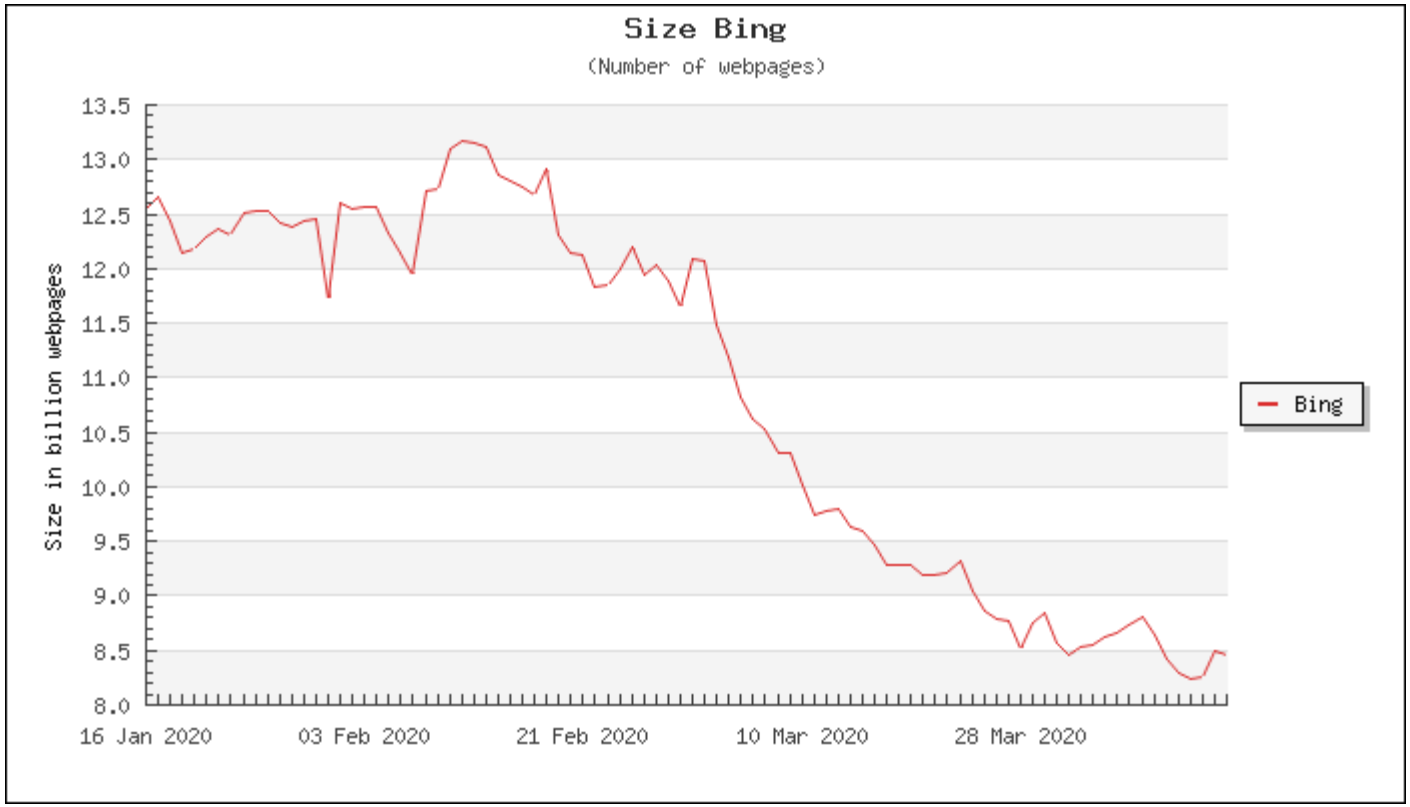


https://www.worldwidewebsize.com/

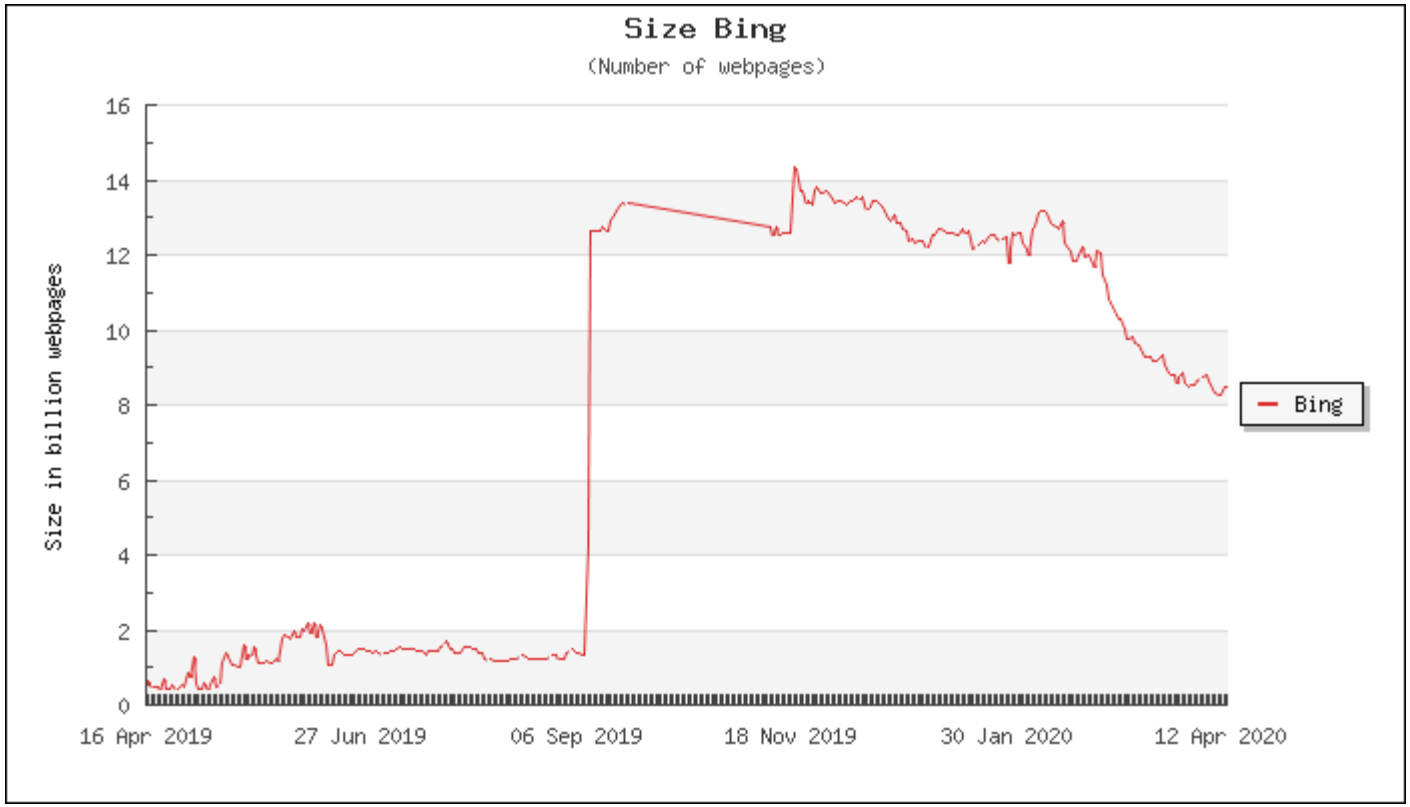
Индексиране в WWW



Индексиране в WWW



Индексиране в WWW



Анализ на връзките

- Линковете са ключов компонент на Web.
- Важни са за навигацията, но и за търсенето.

Линкове

```
<a href="http://example.com" >Example website</a>
```

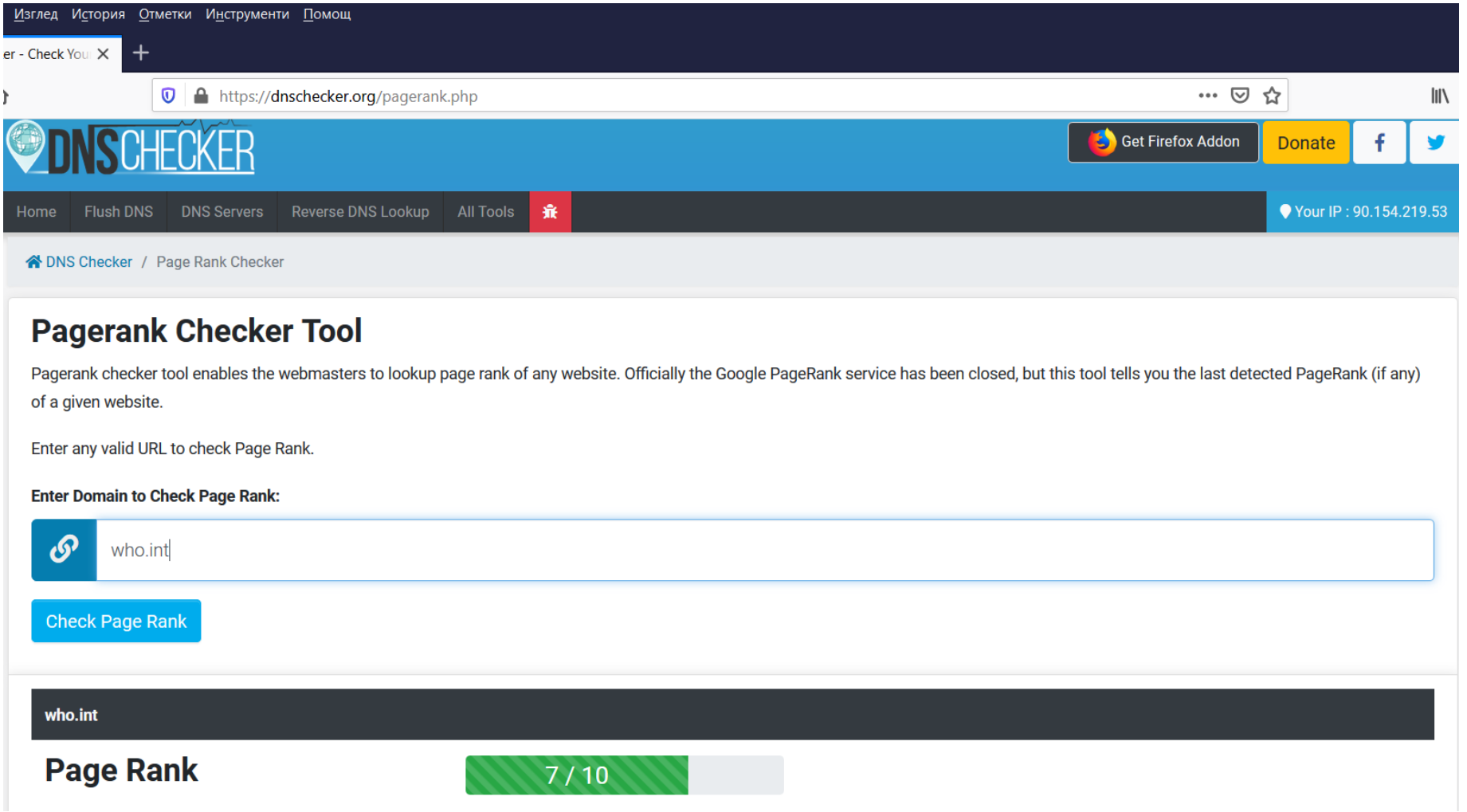
Оценка на връзките

- Милиарди страници, някои повече информативни от други;
- Линковете могат да се разглеждат като информация за популярността на web страница;
- Броят идващи линкове (*inlinks*) може да се използва като проста оценка;
- Алгоритми за анализ на връзките.

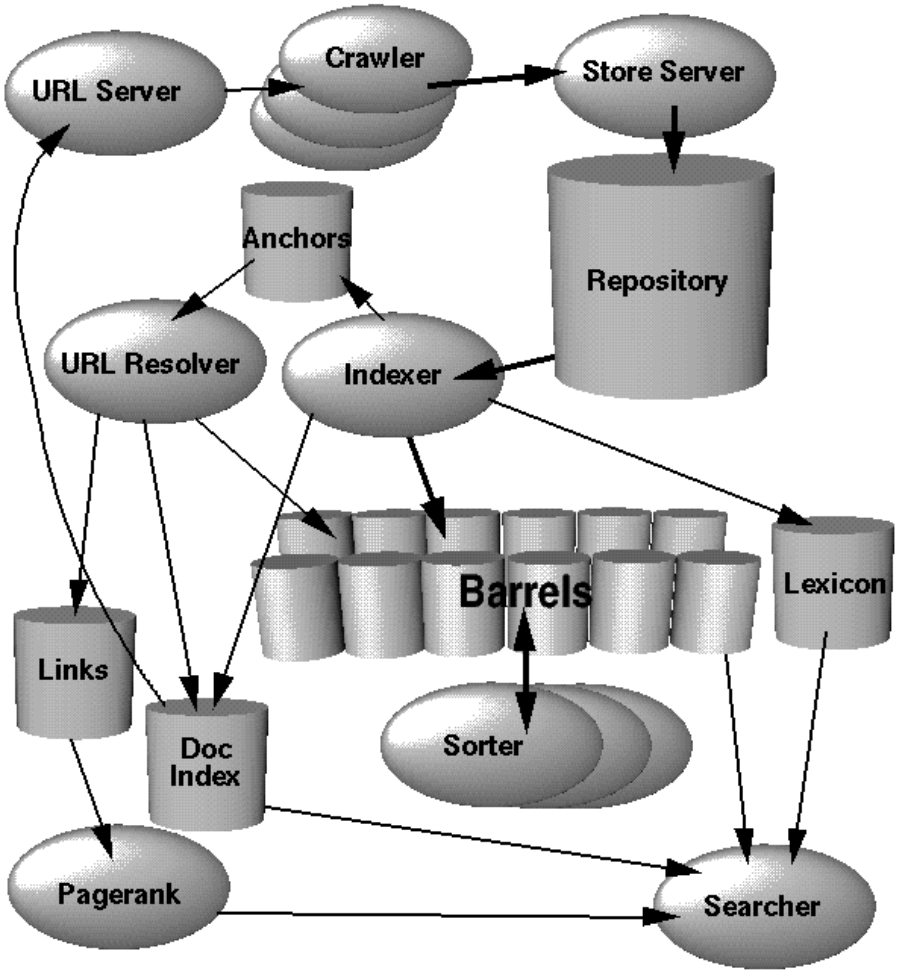
Page Rank

- Базира се на броя на „идващите“ линкове към страницата и популярността на страницата (weight) от която те идват.
- Колкото популярността на страниците, които сочат към определена, е по-голяма, толкова и PR на тази страница е по-голям.

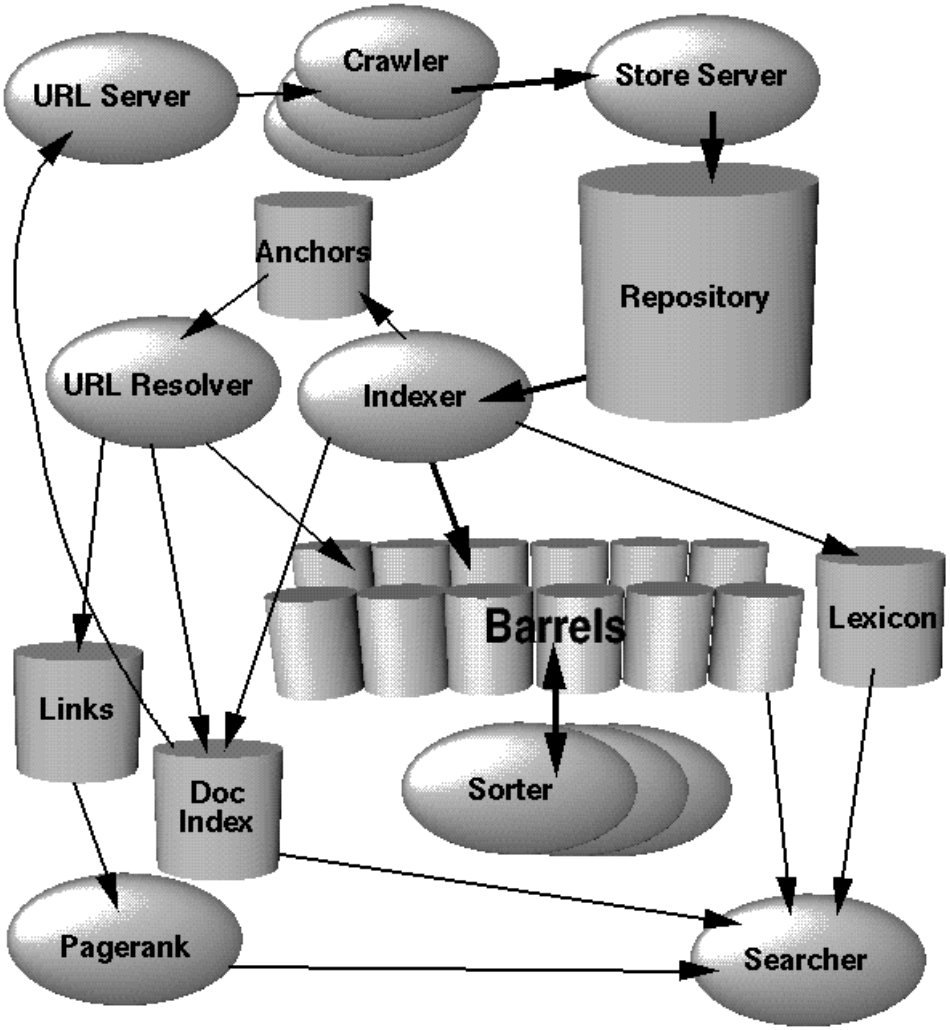
Page Rank Checker



Архитектура на Google

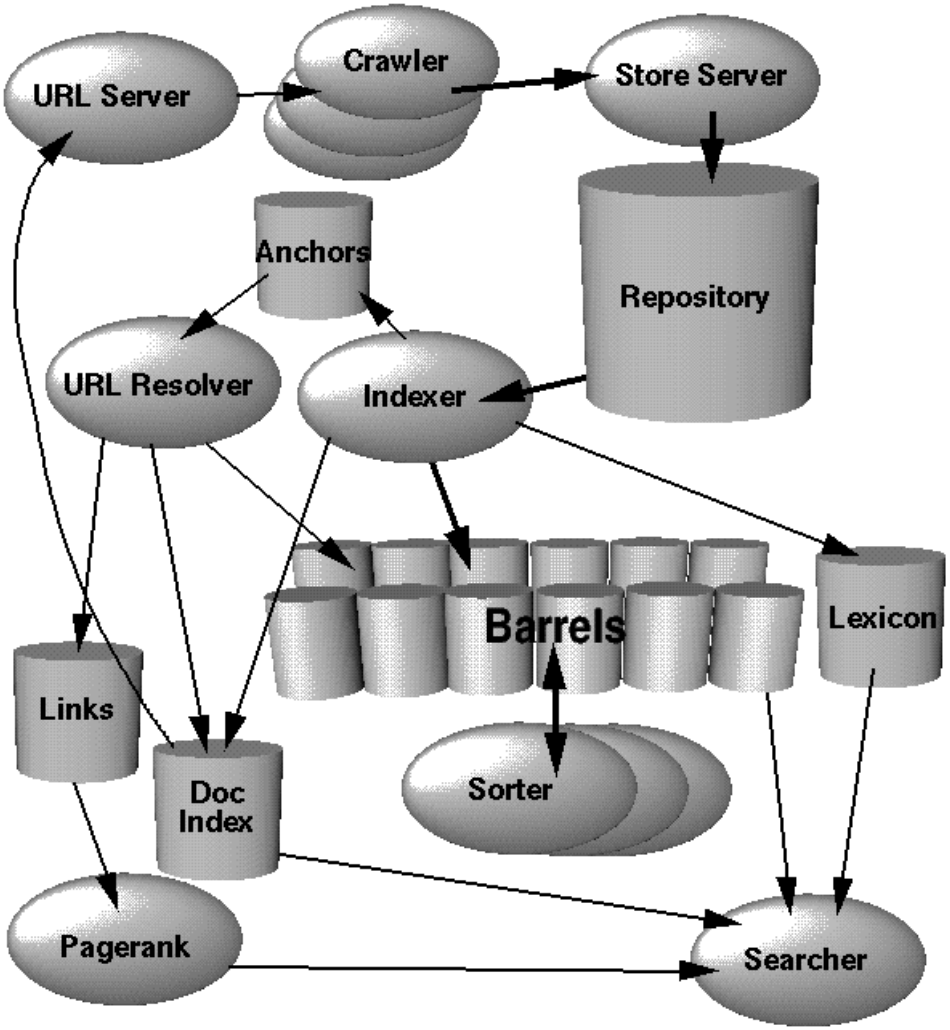


Компоненти - servers



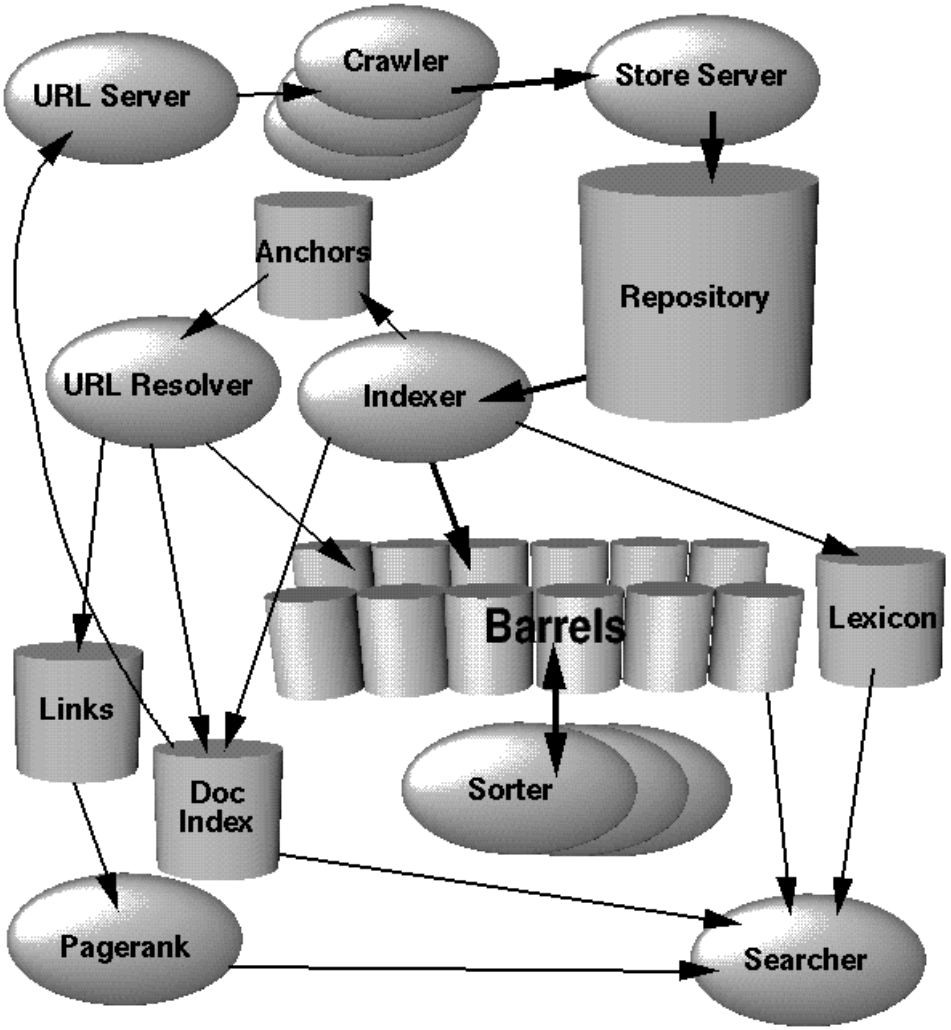
- URL server
- Store server

Компоненти - Repository



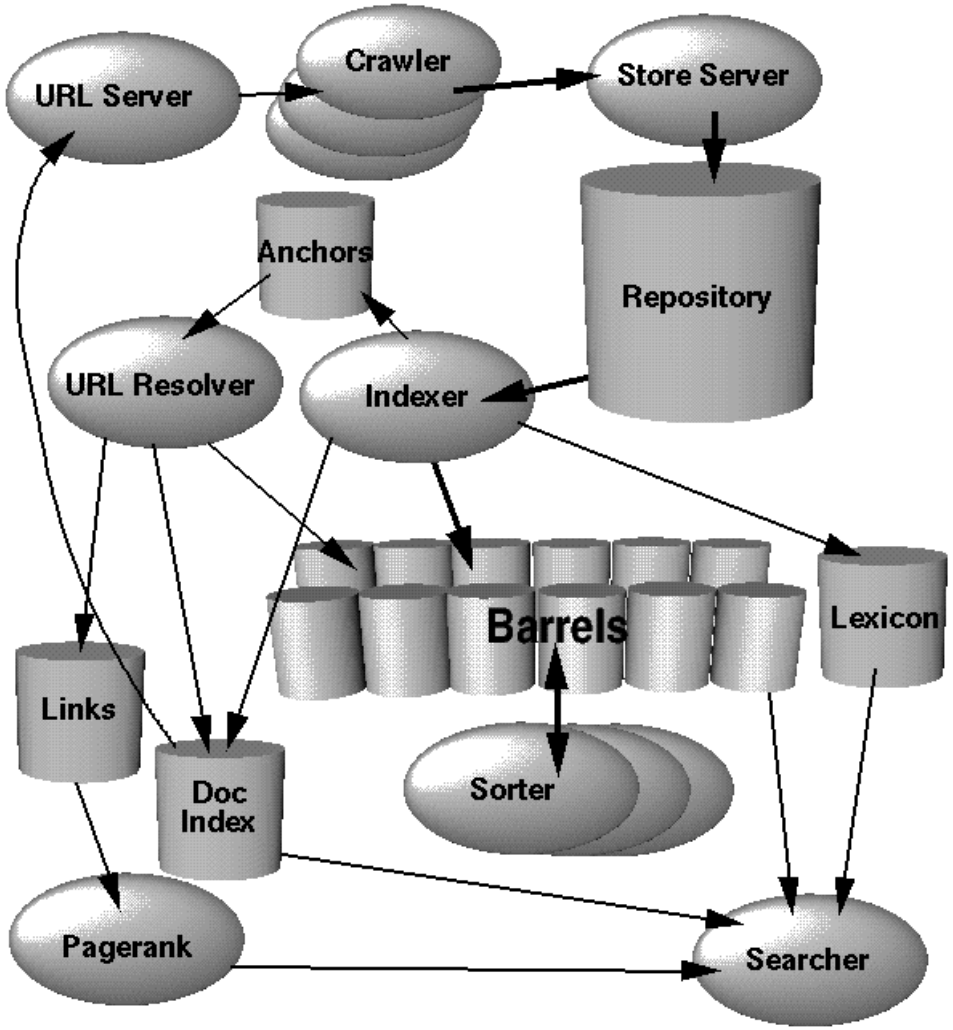
- Съдържа пълен HTML на всяка страница;
- Компресиране с zlib;
- Документите се съхраняват един след друг (docID, Len, URL, page)

Компоненти - Indexer



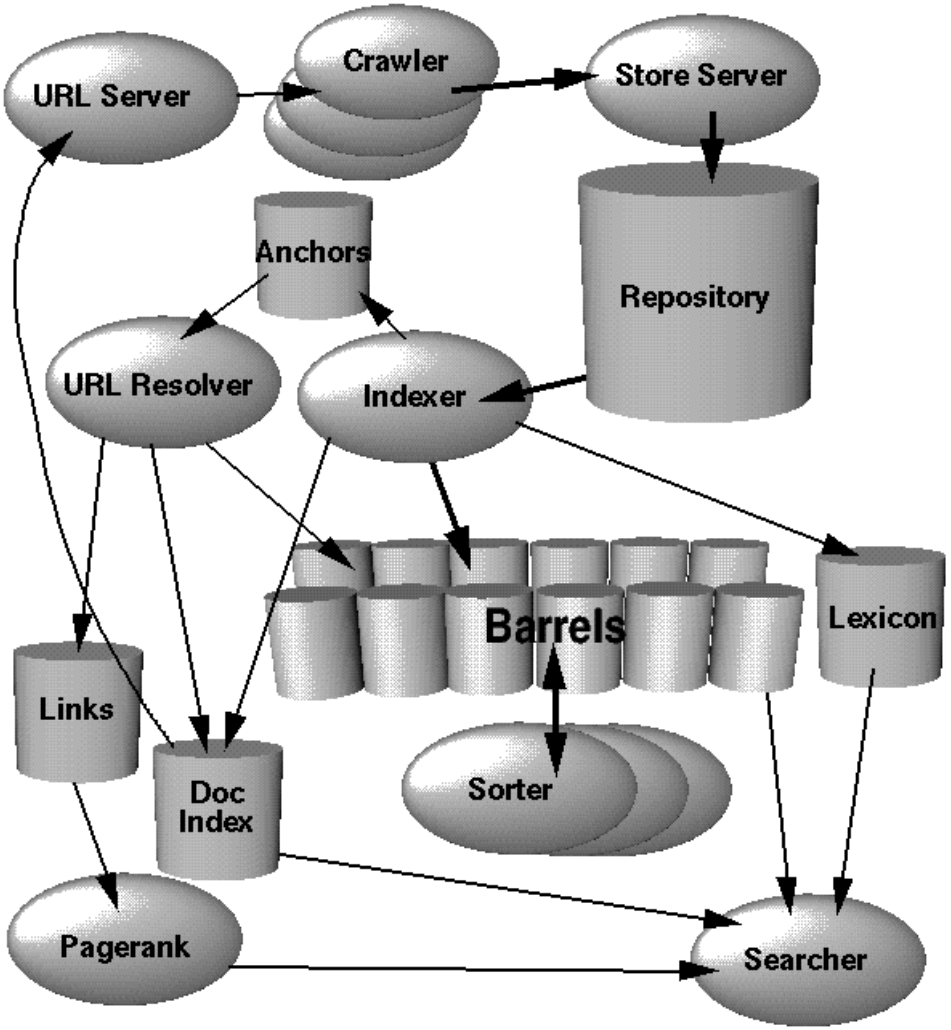
- Декомпресира документите.
- Всеки документ се конвертира в множество думи – *hits*.
- Hit – дума, позиция, размер шрифт, главни букви.
- Разпределят се в частично сортирани индекси – *barrels*.
- Парсва линковете в *anchor file*.

Компоненти – URL resolver



- Конвертира относителните URL в абсолютни;
- Поставя текста от котвата в индекс с docID, сочен от нея;
- Генерира база от линкове (линк-docID).

Компоненти – Sorter



- Сортира barrels по docID.
- Генерира инвертен индекс.
- Обновява лексикона.