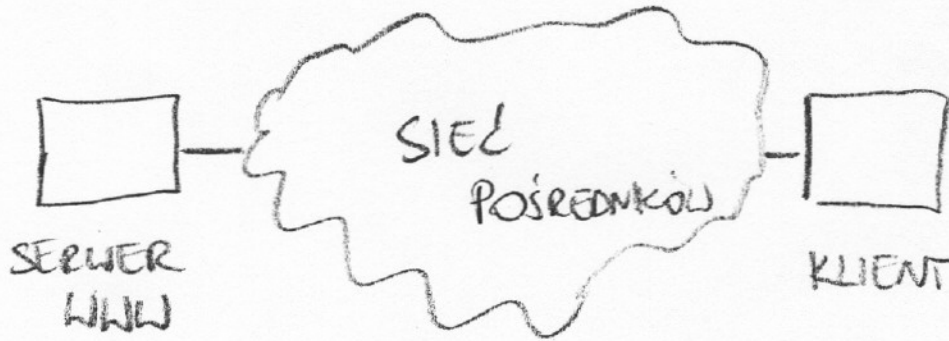


POŚREDNIKI SPAMIĘTUJĄCE - KOMUNIKACJA

1. WSTĘP



PODSTAWOWE ZAGADNIENIA

- ROZMIESZCZENIE POŚREDNIKÓW
- JAKIE DANE ZAPAMIĘTYWAĆ?
- JAKIE DANE WYMIENIAĆ MIĘDZY POŚREDNIKAMI?
- JAKIE DANE SPRAWDZIĆ Z WPRZEDZIEMEM?
- JAKIE DANE WPYCHAĆ DO SIECI?
- W KTÓRE MIEJSCE WPYCHAĆ DANE?
- JAK USUWAĆ DANE Z PAMIĘCI POŚREDNIKA?
- JAK ZAPEWNIĆ SPÓDNOŚĆ DANYCH?

2. PODSTAWOWE PROTOKOŁY WYMIANY INFORMACJI MIĘDZY POŚREDNIKAMI

- ICP (INTERNET CACHE PROTOCOL), HTTP (HYPER TEXT CACHING PROTOCOL)

• DZIAŁANIE:..

- + ŻĄDANIE - ODPOWIEDŹ
- + ŚCIĄGNIĘCIE Z WPRZEDZIEMEM

• GEOUNA WADA: MAKA SKALOWALNOŚĆ

- CACHE - DIGEST

• DZIAŁANIE:

FILTRY BLOOMA:

PARA: WEKTOR v ZŁOŻONY Z m BITÓW ORAZ
K NIEZALEZNYCH FUNKCJI HASZUJĄCYCH $h_1, \dots, h_k: U \rightarrow \{0, \dots, m-1\}$
(rozmiar U dużo większy od m)

OPERACJE

* INICJALIZACJA

x USTAWIENIE WSZYSTKICH BITÓW v NA 0

* WSTAWIENIE ZASOBU $x \in U$ DO FILTRA:

x OBLICZAMY $h_1(x), \dots, h_k(x)$

x USTAWIAMY $v[h_1(x)], \dots, v[h_k(x)]$ NA 1

* SPRAWDZENIE, CZY $x \in v$

x OBLICZAMY $h_1(x), \dots, h_k(x)$,

x JEŚLI DLA KTÓREGOŚ i $v[h_i(x)] = 0$, TO x NIE MA W FILTRZE,

x JEŚLI DLA WSZYSTKICH i $v[h_i(x)] = 1$, TO x JEST W FILTRZE
(TAK. Z DRYM PRAWDOPODOBIEŃSTWEM BYŁ TAM WSTAWIONY)

PRZYKŁAD:

$m=5$ $k=2$

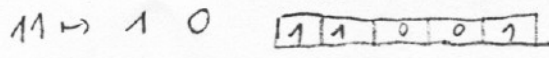
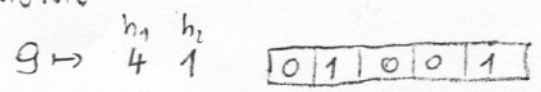
$h_1(x) = x \text{ mod } 5$

$h_2(x) = (2x+3) \text{ mod } 5$

* INICJALIZACJA



* WSTAWIENIE



* SPRAWDZENIE

15 \rightarrow

| | |
|-------|-------|
| h_1 | h_2 |
| 0 | 3 |

 NIE MA

16 \rightarrow

| | |
|-------|-------|
| h_1 | h_2 |
| 1 | 0 |

 JEST

DLA FILTRÓW BLOOMA ZACHODZI:

3

$$P_n(\text{FILTR SIĘ POMYLIŁ PO } n \text{ OPERACJACH DODAJANIA}) \approx (1 - e^{-\frac{k \cdot n}{m}})^k$$

• REALIZACJA CACHE - DIGEST

UZYWAMY 4 FUNKCJI HASZUJĄCYCH DAJĄCYCH 32 BITOWE WYNIKI - DZIELIMY 16 BASTOUE MDS Z URLA NA 4 BLOKI

$$k = 4$$

$$m = 2^{32}$$

ZADOKŁADAJĄCE WYNIKI DLA $n \approx 744261117$
(OPTIMALNA LICZBA FUNKCJI HASZUJĄCYCH)

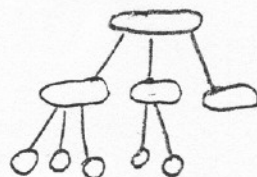
3. LOSOWE DRZEWA - UNIKANIE ZALANIA

PROBLEM: CZASAMI JEDEN ZASOB STAJE SIĘ NAGLE KARDIO POPULARNY. JAK POWINIEN ZOSTAĆ ZORGANIZOWANY SYSTEM POŚREDNIKÓW, ABY NIE SPOWODOWAĆ ZABLOKOWANIA DOSTĘPU?

IDEA: ROZPROSZYĆ ZLECENIA PRZEZ LOSOWE ODWOŁYWANIE SIĘ DO POŚREDNIKÓW.

JAK TO ZREALIZOWAĆ: POŚREDNIKI BĘDĄ WYPEŁNIAĆ SIĘC NASTĘPUJĄCEGO KSZTAŁTU - JEST TO DRZEWO O C WIERZCHOŁKACH (C TO LICZBA POŚREDNIKÓW + 1) MAJĄCE STOPIEŃ ROZGĄĘZIENIA d . WSZYSTKIE POZIOME DRZEWA SĄ, W PEŁNI ZAPEŁNIONE ZA WYJĄTKIEM, BYĆ MOŻE, OSTATNIEGO POZIOMU.

PRZYKŁAD:



$$\text{DLA } d = 3 \\ C = 3$$

NASZ PROTOKÓŁ DLA USTALONEGO URL-A PRZYPISUJE
WIERZCHOŁKOM DRZEWA ADRESY POŚREDNIKÓW. REALIZOWANE
TO JEST PRZEZ UMIESZCZENIE WE WSZYSTKICH PRZEGLĄDARKACH
ORAZ POŚREDNIKACH FUNKCJI HASZUJĄCEJ $h: U \times \{2, \dots, C\} \rightarrow C$, GDZIE
 C JEST ZBIOREM ADRESÓW POŚREDNIKÓW, ZAŚ U ZBIOREM URL-ÓW

DODATKOWO JESZCZE WPROWADZAMY PARAMETR q - POŚREDNIK ZAPAMIĘTANIA
ZASÓB z , GDY WJRZY GO q RAZY.

PROTOKÓŁ DLA DRZEŁ LOSOWYCH WYGLĄDA TAK:

PRZEGLĄDARWA: GDY PRZEGLĄDARWA CHCE SIĘ ODWOŁAĆ DO
ZASOBU z , TO WYBIERA DOWOLNY LIŚĆ W DRZEWIE, PRZYPISUJE
WIERZCHOŁKOM DRZEWA MASZYNĄ ZGODNIE Z FUNKCJĄ h ,
A NASTĘPNIE WYSYŁA ZADANIE DO MASZYNY PRZYPISANEJ
WYBRANEMU LIŚCIOWI. ZADANIE SKŁADA SIĘ Z:

- ADRESU PRZEGLĄDARKI,
- ADRESU ŚWIĄGANEGO ZASOBU z ,
- ŚCIEŻKI OD WYBRANEGO LIŚCIA DO KORZENIA (IMPLICITNIE),
- WNIKU DZIAŁANIA h DLA z I DRZEWA (IMPLICITNIE).

POŚREDNIK: GDY POŚREDNIK OTRZYMUJE ZADANIE, TO SPRAWDZA,
CZY ZASÓB, KTÓREGO ZADANIE DOTYCY ZNAJDUJE SIĘ
W PAMIĘCI ORAZ, CZY WŁAŚNIE JEST ŚWIĄGANYM. GDY
ZACHODZI PIERWSZA SYTUACJA, TO PODAJE ZAPAMIĘTANY ZASÓB.
GDY ZACHODZI DRUGA, TO CIEKA NA ŚWIĄGNIĘCIE ZASOBU I
PODAJE OTRZYMANY ZASÓB.

JEŚLI ZASOBU NIE MA W PAMIĘCI, TO ZWIĘKSZA LICZNIK
ZASOBU I JEŚLI NIE JEST ON ŚWIĄGANYM, TO ZWIĘKSZA
LICZNIK ZASOBU I PROSI NASTĘPNĄ MASZYNĘ NA ŚCIEŻECIE O
ZASÓB. GDY LICZNIK OSIĄGNIE q , ZAPISUJE ZASÓB
W SŁOJCE PAMIĘCI. W KAŻDYM WYPADKU ZASÓB JEST PRZEKAZY-
WANY DO KOMPUTERA, KTÓRY GO ZARZĄDZA.

SERWER: PO OTRZYMANIU ZADANIA ODSYŁA ZASÓB DO MASZYNY, KTÓRA
O NIEGO PROSIŁA

+ OPÓŹNIENIE $\leq 2\delta \lceil \log_d C \rceil$

(δ - OGRANICZENIE NA OPÓŹNIENIE MIĘDZY MASZYNAMI)
 PRZY CIĄGU ODWOKAŃ DŁUGOŚCI R ,

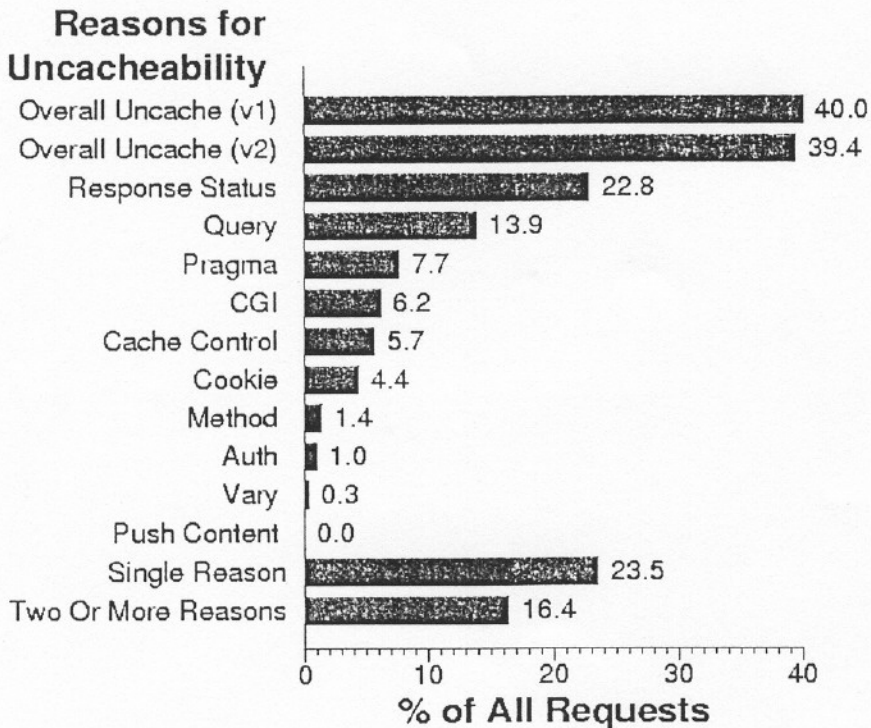
+ DLA DOWOLNEGO N PRAWDOPODOBIEŃSTWO, ŻE WSZYSTKIEMU
 POŚREDNIK DOSTAŃIE ZADANIE WIĘCEJ NIŻ

$$p \left(2 \log_d C + O\left(\frac{\log N}{\log \log N}\right) \right) + O\left(\frac{d \log N}{\log\left(\frac{d \log N}{p} \log N\right)} + \log N\right)$$

JEST MNIEJSZE NIŻ $1/N$ (GDZIE $p = \frac{R}{C}$)

+ POWIĘKSZE OGRANICZENIE JEST OPTIMALNE.

4. POWODY NIE ZAPAMIĘTYWANIA



5. CO ROBIĆ, ZEBY ZAPAMIĘTYWAĆ WIĘCEJ?

6

A) ZAPAMIĘTYWANIE AKTYWNYCH DANYCH

- + DANE DYNAMICZNE ZMIENIAJĄ SIĘ, RAZ NA JAKIŚ CZAS,
- + TYLKO CZĘŚĆ STRONY ZAWIERA DYNAMICZNE DANE.

B) ZAPAMIĘTYWANIE W POŚREDNIKU PAR: DANE WEJŚCIOWE, WYNIK

C) AKTYWNE POŚREDNIKI

- + POŚREDNIK ZAPAMIĘTUJE SERULET W JAVIE

D) WZMOCNIENIE MECHANIZMÓW ZAPEWNIANIA SPÓJNOŚCI

- + WYSYŁANIE UNIEWAZNIEŃ,
- + SYSTEMY WYNADZMOWA NA ZASOBÓW.

E) SPROWADZANE Z WPRZEDZENIEM

6. ZACHOWANIE SPÓJNOŚCI DANYCH

A) USTALANIE CZASU ŻYCIA,

B) UNIEWAZNIANIE (PROBLEMY ZE SKALOWALNOŚCIĄ),

C) WYNADZMOWANIE ZASOBÓW (ODŚWIEŻANIE ŚWIADCZENIA WYNADZMU),

CIEKAWY PRZYKŁAD: SKALOWALNA ARCHITEKTURA SPÓJNEGO WWW

* POŚREDNIKI SĄ ZORGANIZOWANE W DRZEWO (DAG),

* W DÓŁ DRZEWA POSYŁANE SĄ OKRESOWO "TYKNIĘCIA"

* OKRES WYNADZMU T MIERZONY JEST W "TYKNIĘCIACH" NP.

T OKRESÓW TYKNIĘCIA

* KAŻDY SYN OKRESOWO SPRAWDZA CZAS, JEŚLI OD OSTATNIEGO "TYKNIĘCIA"

UPYMNIEŁO WIĘCEJ NIŻ T , TO WSZYSTKIE STRONY OD OJCA SĄ UNIEWAZNIANE

- * RAZEM Z "TYKNIĘCIAMI" PRZESYŁANE SĄ INFORMACJE O NIEWAZNYCH STRONACH, INFORMACJE TE SĄ PROPAGOWANE W GŁĘB, (7)
- * SERWER DOŁĄCZA SIĘ W DOWOLNYM MIEJSW HIERARCHII,
- * POŚREDNIKI PRZEKAZUJĄ INFORMACJE O DOŁĄCZENIU SERWERA U GÓRY HIERARCHII,
- * RAZ NA JAKIŚ CZAS SERWER WYSYŁA INFORMACJE O NIEWAZNOŚCI STRON
- * TA INFORMACJA JEST PROPAGOWANA W GÓRĘ HIERARCHII
- * JAK ZA DŁUGO (T) NIE PRZYCHODZI ŻADNE WIADOMOŚCI Z SERWERA, TO JEST USUWANY I UNIEWAZNIANE SĄ JEGO WSZYSTKIE STRONY