

Apkrovimo paskirstymas

Projekto istorija augant iki 1.5 M vartotojų per dieną

Apie mane

- Sergej Kurakin
- <http://kurakin.info/>
- 10+ metų dirbu WEB srityje
- Paskutinius 5 metus 60% laiko skiriau vienam projektui
- Kol neišėjau į Net Frequency

Trumpai apie projektą

- Dėja, pasirašiau NDA, todėl pavadinimo nebus
- Projektas įkurtas 1999 metais
- Dauguma puslapių sudaro tik tekstai
- Piko metu budavo iki 1200 HTTP r/s (500 r/s), 80 Mb/s out (15 Mbps)
- Kasdien apsilankidavo apie 1.5M unikalių vartotojų ir atidaridavo 11M puslapių

Mano “pareigos” projekte

- Buvo neapibrėžtos...
- Pirminė veikla: naujų funkcijų programavimas su PHP + bet kas, kas padės kuo greičiau įvykdyti užduotį
- Nuotolinis serverių diegimas ir priežiūra
- Įvairus planavimas ir konsultavimas
- Gaisrų gesinimas ir užsikimšusių vamzdžių valymas

Projekto evoliucija

- Tik HTML
- Server-Side Includes (SSI)
- Perl + tekstinės bilos
- PHP 4 + MySQL 3/4 (jokio UTF-8)
- PHP 4 + MySQL 3/4 (UTF-8 į latin1)
- O po to prasidėjo eksperimentai ir linksmybės

Standartiniai žingsniai

- Dedikuotas serveris
- Atskiras serveris duomenų bazei
- Cache standžiajame diske, gzip → shm
- Viskas vyksta dar iki 2005 metų
(memcached atsirado tik 2003 metais)



Iškilusios problemos

- Lėtoki diskai dirbant su Cache
- Per mažas atminties kiekis
- Tikriausiai CPU našumo trūkumas

Pirmas žingsnis

- Round Robin DNS
- Primitiviausias variantas
- 2 ar daugiau IP adresų (A records)
- Labai mažas TTL



dig A example.com

```
example.com. 3600 IN A 203.0.113.01  
example.com. 3600 IN A 203.0.113.02
```




dig A example.com

```
example.com. 3600 IN A 203.0.113.01
example.com. 3600 IN A 203.0.113.02
```



Veikia \o/

- Padėjo, bet neilgam
- Lėtos SQL užklausos, užrakintos užklausos
- Lėtos HTTP užklausos
- Cache atnaujinimas ir užraktai

Antras žingsnis

- 2 Apache veikiantys per 80 ir 81 TCP portus
- Apache ant 80 porto atiduoda puslapius iš Cache, jei Cache tuščias, daromas nukreipimas į 81 portą
- Apache ant 81 porto daro ką galį tam, kad sukurtų puslapį



MySQL optimizacija

- Standartiniai veiksmai
- SQL užklausų analizė
- MySQL Query Slow Log
- EXPLAIN SELECT ...
- Index'u analizė
- MySQL Serverio optimizacijos

Veikia \o/

Taip, padējo..., bet staiga :-(

- Google: Per daug nukreipimų
- Google: Besidubliuojantis tūrinis
- Vartotojai: Dalis nepasiekė 81 porto

Trečias žingsnis

- Rinkomės tarp Lighttpd ir NGINX
- NGINX 0.6
- PHP 4
- PHP-FPM (2007 metai)
- 2 PHP-FPM grupės

Veikia \o/

- Kartais sugrįždavo “sena” liga – lėtos SQL užklausos
- Lėtos disko operacijos su Cache standžiajame diske, Cache atnaujinimo užraktai
- PHP versija priversdavo verkti

Kodėl PHP < 5.3?

- Red Hat Enterprise Linux 4 (2006?)
- Naudojama daug Deprecated PHP funkcijų
- Seni PHP 4 konstruktoriai
- Senos PHP 4 bibliotekos
- Beveik nėra jokių testų ir dokumentacijos
- Nedokumentuotas kelių žmonių kodas

Ketvirtas žingsnis

- Paskirstyme puslapius pagal tipus ir svarbą
- Paskireme kiekvienam tipui atskiras PHP-FPM grupes
- Paskireme atskiras Cache vietas (failai, NoSQL)
- Paskiriame kiekvienam tipui atskirą MySQL serverį (shard)

PHP-FPM grupės

- Su PHP 5.2 PHP-FPM turi labai nepatogų XML konfiguracionį failą
- PHP 5.2 PHP-FPM neturi jokios statistikos
- eAccelerator VS APC (facepalm)
- RHEL 4 ir PHP 5

NoSQL

- Redis (dėl to, kad greitai pasileido ant RHEL4)
- ~ 6GB duomenų
- Keli servisai viename serveryje
- Labai patogus ir greitas

NoSQL / Redis

- Viso puslapio Cache kopija
- Puslapio dalies Cache kopija
- Lētos/sudėtingos SQL užklauso rezultātų kopija
- ORDER BY RAND() perkėlimas



Atnaujinimas iki CentOS 5

- Red Hat Enterprise Linux 4 (2006?)
- Vyko per lėtai: 1 s/m
- Žymiai mažiau problemų su PĮ
- Geriau pradėjo naudoti techninę įrangą
- Taisiklingiau pradėjome kurti/skirstyti failų sistemą

Tinklas / LAN

- Jo niekas nestebėjo (facepalm)
- “Staiga” teko atsinaujinti nuo 100 mb/s iki 1000 mb/s
- O tai turi įtakos...

O kartais tereikia įmesti
keliasdešimt GB atminties ir viskas
gerokai pagreitės.

Stebėjimas

- Munin naudojamas jau kelis metus
- Monit tik kelis mėnesius (RHEL4 :()
- Nuosavos gamybos PHP skriptas MySQL replikacijai stebėti
- Prieš mėnesį sužinojome, kad serverinėje yra dar viena stebėjimo sistema (siurprizas)

Nuosava vidinė paieška

- MySQL LIKE '%<keyworg>%'
- Atskiras serveris
- MySQL Full-Text Index/Search
- Xapian paprastame standžiajame diske
- Xapian ant SSD
- Xapian in SHM (priverstinis eksperimentas)
- Indeksavimo problemos

Kokie buvo ateities planai

- Pilnai pereiti prie CentOS 5, CentOS 6
- Nuosavas Linux PĮ repozitorijus
- Atsinaujinti iki PHP 5.3 ir po to iki PHP 5.4
- Kodo refaktoringas
- Panaudoti naujausias bibliotekas
- Puppet konfigūracijai valdyti

Kokie buvo ateities planai

- NoSQL + MySQL (užbaigti migracija)
- Atsikratyti Round Robin DNS
- Linux-HA arba kitoks sprendimas
- Išmesti Xapian ir pereiti prie Sphinx
- “Atsikratyti” NFS (DRBD arba GFS)
- Atskirti tinklus, rezervinė tinklo įranga
- Ugniasienės

Pabaiga. Klausimai?

Kontaktams:
sergej@kurakin.info