PERFORMANCE STRATEGIES

Alexander Meindl @ Drupalcamp Wien 2009

ABOUT ME

- Freelancer for 13 years

- Linux experience for 18 years

- PHP experience for 9 years

- Drupal addicted since 2006

AGENDA

- use cases

- analyses

- strategies

  - shift load to the client (frontend optimization)

  - optimize the application (drupal)

  - server/infrastructure optimization

PERFORMANCE STRATEGIES

- Some kittens can die

SERVER QUICK WINS

- PHP OP code caching (xcache, eaccelerator, APC, Zend Server)

- MySQL Query Cache

- use mysqli with MySQL5

- AllowOverwrite = None (move php configuration to vhost file)

ANALYSES

- Analyses of the current status

  - Stress tests / simulation

  - Profiling (xdebug, drupal devel)

  - Log files (error logs, slow queries...)

USE CASES

1. complex functionality (e.g. large number of modules)

2. large number of concurrent anonymous users

3. large number of concurrent registered users

4. large amount of data

1. COMPLEX FUNCTIONALITY

- simplify it!

  - reduce modules (solve tasks with a more abstract solution - views, cck, panels, context?)

  - optimize modules (find bottlenecks)

- use asyncron jobs (drush is your friend)

- cache it (data, objects, ...)

1. COMPLEX FUNCTIONALITY

- use a different cache engine (default DB):

  - memcache

  - cacherouter (APC, eAccelerator, memcache, XCache)

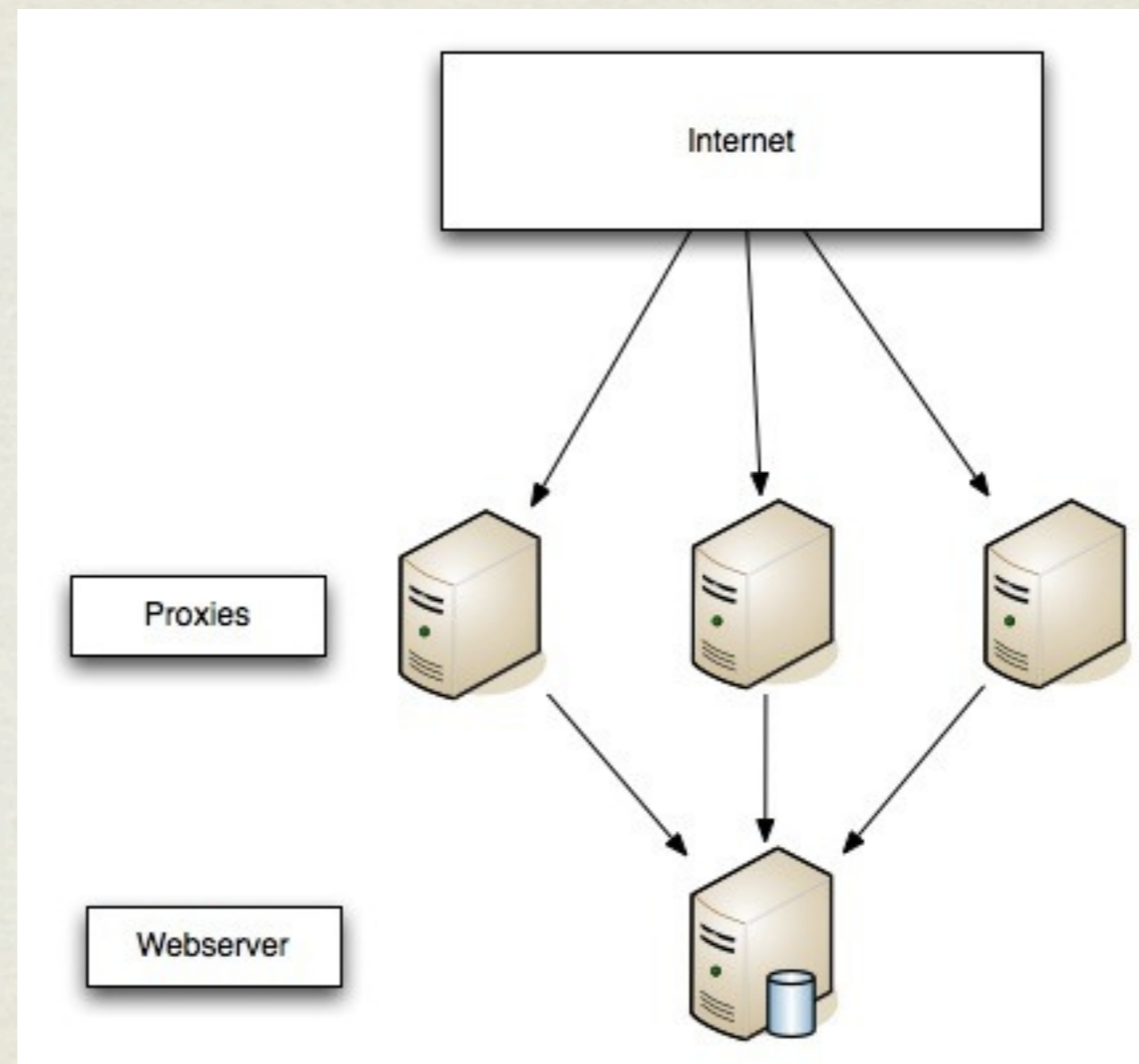- use cache_get/cache_set in your code

2. HIGH TRAFFIC

large number of concurrent anonymous users

- everyone can see the same page content

- page can be cached without complex logic behind
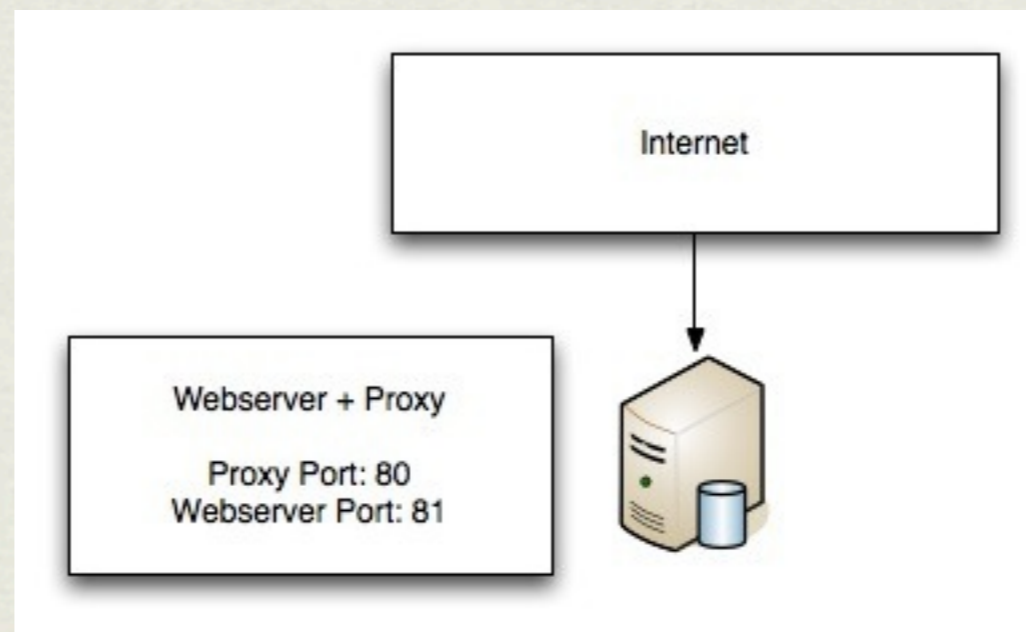
- reverse proxy

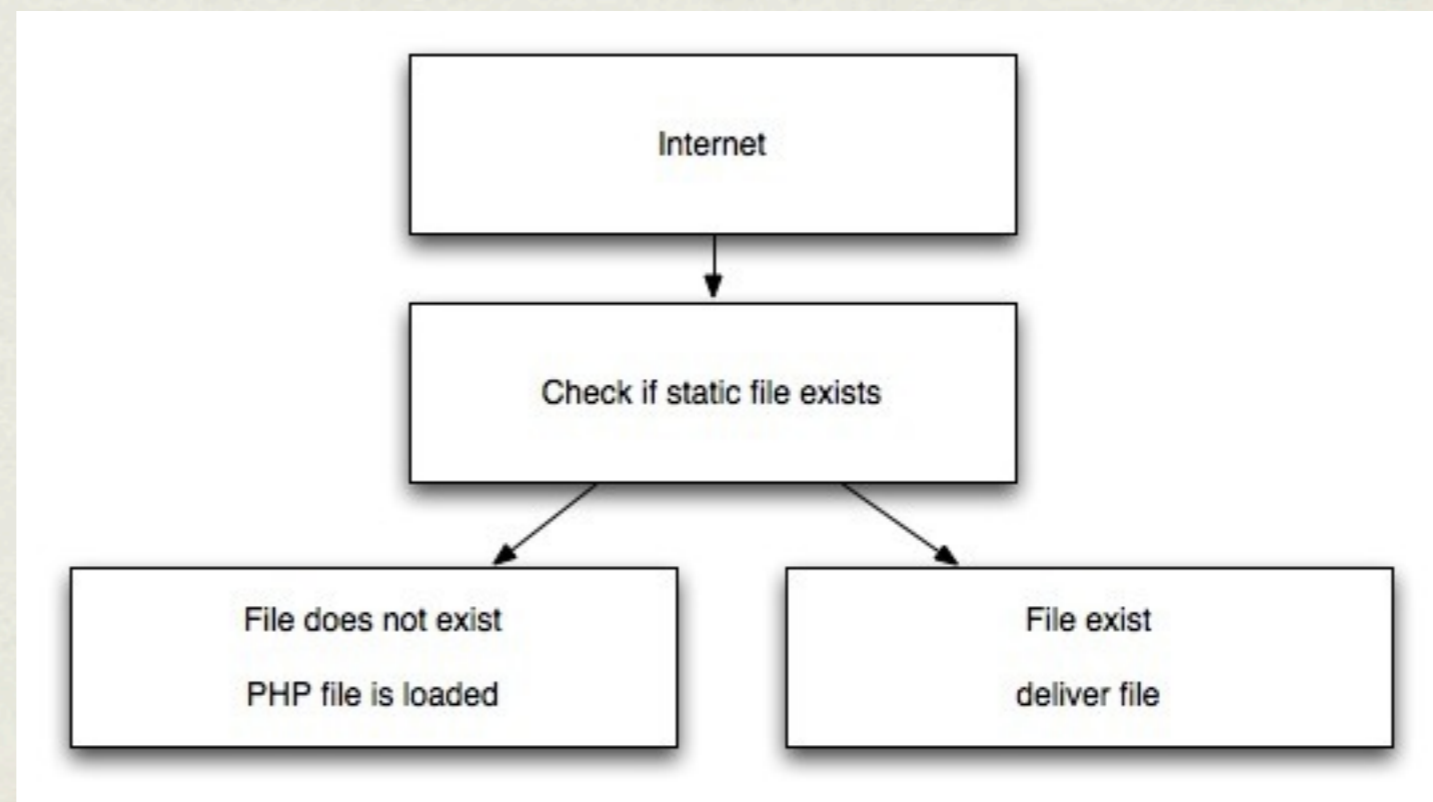- boost cache

REVERSE PROXY

- Squid / Varnish

REVERSE PROXY

- Squid / Varnish - Single Server

BOOST CACHE

- Files from file system without parsing PHP and a database connection

3. HIGH TRAFFIC

large number of concurrent registered users

problem:

- role based content

- user based content

-> lots of database queries

3. HIGH TRAFFIC

large number of concurrent registered users

- block based caching

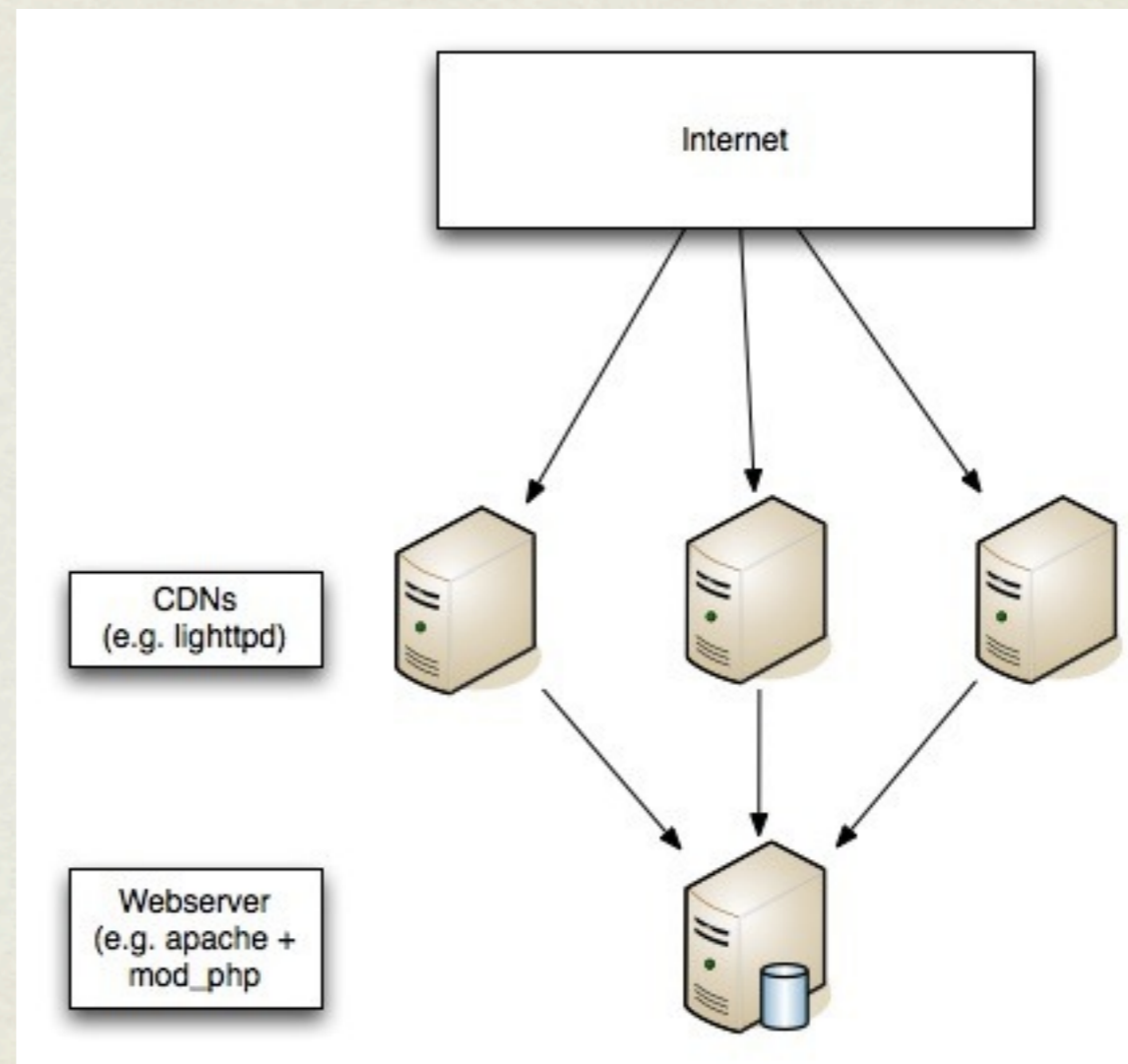 - drupal core can do it!

  - authcache

  - ESI

STATIC CONTENT - CDN

- Images, videos, CSS servered with lighttpd or nginx

  - Own server
    - less memory usage
    - faster client requests (2 request restriction)
    - cheaper for small projects

  - CDN provider
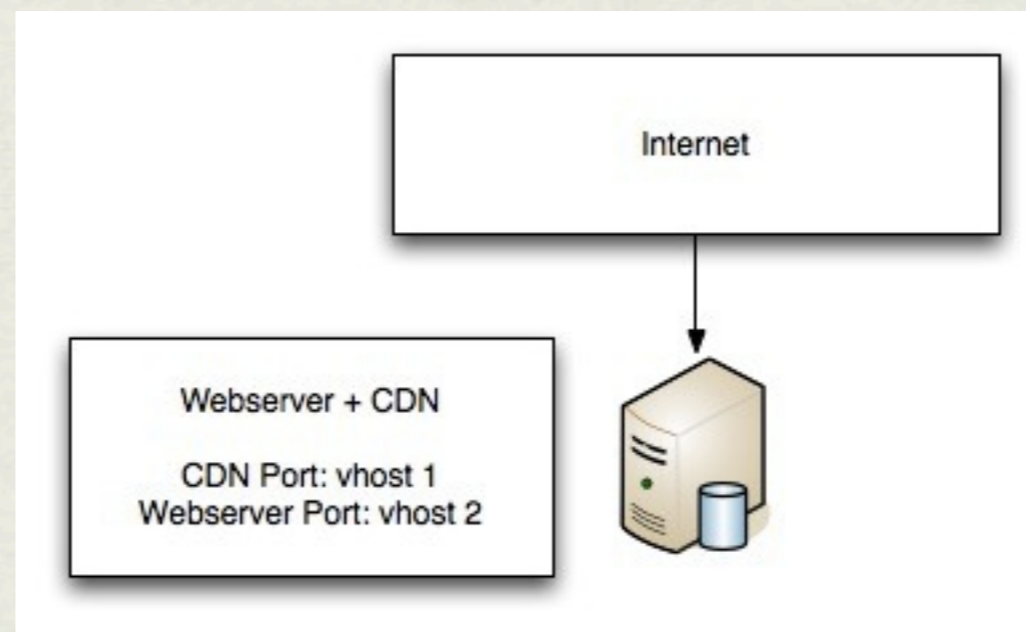   - less server load
    - less traffic
    - expensive

CONTENT DELIVERY NETWORK (CDN)

- Multiple CDN Server

CONTENT DELIVERY NETWORK (CDN)

- CDN+Webserver

CONTENT DELIVERY NETWORK (CDN)

- CDN+Webserver iptables rules:

```
#  Redirect lighttpd
# enable ip forward
echo 1 > /proc/sys/net/ipv4/ip_forward

/sbin/iptables --flush
/sbin/iptables -t nat --flush
/sbin/iptables -t mangle --flush
/sbin/iptables --policy INPUT ACCEPT
/sbin/iptables --policy OUTPUT ACCEPT
/sbin/iptables --policy FORWARD ACCEPT
/sbin/iptables -t nat --policy PREROUTING ACCEPT
/sbin/iptables -t nat --policy OUTPUT ACCEPT
/sbin/iptables -t nat --policy POSTROUTING ACCEPT
/sbin/iptables -t mangle --policy PREROUTING ACCEPT
/sbin/iptables -t mangle --policy OUTPUT ACCEPT


/sbin/iptables -t nat -A PREROUTING -i eth0 -p tcp -d 85.25.178.115 --dport 80 -j REDIRECT --to-port 81

# close port 81 from outside
/sbin/iptables -t nat -A PREROUTING -i eth0 -p tcp -m tcp -d 85.25.178.115 --dport 81 --syn -j DROP
```

4. LARGE NUMBER OF DATA

- lots of nodes and/or users

  - no application based solution

  - server based solution

    - database partitioning (MySQL >= 5.1)

    - Database replication/ cluster

IF PROJECT GROWTH

- scale you project!

    - database replication / cluster

    - external search (solar)

    - CDN and/or reverse proxies

    - external applications for expert solutions (e.g. forum, videos, ad server)

THINK BIG

20

THINK BIG

0.2

PRESSFLOW

- MySQL Replication

- Reverse proxy

- MySQL optimization

- PHP optimization (PHP5 required)

- compatible to contributed modules

SUMMARY

- Try to simplify complexity

- Cache whatever you can as long as you can

- source functionality out

INFORMATION

- http://www.2bits.com

- https://launchpad.net/pressflow

- http://tag1consulting.com/patches

QUESTIONS

- Alexander Meindl

  alex@squatlabs.com