

PHP Scaling with Cal Henderson
Flickr.com

Best language: Homespring (<http://spuzz.net/projects/homespring/download.php>)

Best of a bad bunch - PHP is not the best

What makes something scalable?

- Can support a bigger dataset
- More traffic
- Maintainable

Scaling is not about performance.

PHP is slow, and it is not your bottle neck.

Horizontal scaling is best. Easy to add boxes.

PHP sessions == bad (/var/notshared/). Keep them in a database. Best to have no sessions at all. Pass big data by reference. Don't serialize() big things.

Make your date in small logical blocks. Sucks about namespaces. Classes are kinda like namespaces. Many includes == bad. APT = full paths + nostat == awesome. Set root path via .htaccess ENV.

Beauty is less important than making things work.

iamcal.com/talks

=====

PHP Scalability & Performance

Facebook.com
Lucas Nealan

Social platform.

Over 61 million users
50 pages per user daily
250,000 new users
Over 7,000 platform applications
35 billion page views a month.

Caching is king.

\$GLOBALS
APC - Alternative PHP cache
Memcached
Database
Browser cache
Third party CDN

APC = opcode cache. Hundreds of included libraries.

Normal 4 seconds
APC 135ms
apc.stat=0 128ms

Memcached is good.

What doesn't scale?

- Lots of OO code
- Tens of thousands of functions
- Socket communications
- define()s

Profiling code?

- Callgrind
- APC or xdebug
- KCachegrind
- Measure and monitor average page generation times.

ciso.org/talks

=====

Sarah Goleman
pollita@php.net
PHP for women
"Scaling and embedding PHP"

Do I care about speed?

Faster hardware is not a silver bullet.

What's slowing me down?

- Not using an Opcode cache (pickle install apc)
- PHP4; Completely re-written engine
- Stupidity

References are pointless. The engine is smarter than you. If you try to outsmart it, it will have to do more work than it has to do.

Don't use `__autoload`, but it doesn't matter.

Don't port PHP code to C.

Cache stable data. Once you find data that isn't going to change much, put it somewhere and cache it. Also help browsers cache JS and CSS. Parallelize external request.

Use the right tool for the right job.

- Database vs. filesystem (use filesystems)
- Binaries vs extensions (use extensions)
- Windows vs. Unix (DEFINITELY use Unix)

You are in it for the user. Server speed != user experience. Functionality can trump performance. Security trumps both.

=====

Andrei Zmievski
Outspark, Inc.
Smarty templates
PHP Cookbook
Release manager of PHP 4.2
Internationalization expert
Lead programmer of PHP 6

PHP ++
or, leveling up without the grind

I could explain Unicode to Paris Hilton

PHP 6 = PHP 5 + Unicode

What is PHP?
33% of the internet

What is Unicode?
mojibake - screwed up or fucked up symbols

Encoding soup
-provinciality - why isn't it in English?
-computer limitations - computers are dumb.
-inertia

Unicode handles all major living scripts. Version 5.0 has 99,000+ characters, and it could fit 1 million.

Unicode will not solve all your problems. Internationalization means that users in other countries want to see their date in other formats. Unicode will only solve the encoding soup problem.

PHP6 has Unicode support anywhere. In the engine, in the extensions, in the API. It's native and complete (not a add-on library).

Two sting types, Unicode and binary. Unicode strings are for the text (default for literals). Binary strings will contain bytes.

Unicode identifies are allowed! (I can't take notes on this because I don't know how to use my international keyboard features.)

Functions understand Unicode text. `strtoupper()` and friends do proper case matching. You can do `simplexml` on a Chinese RSS feed, and have a `preg_replace` match work with international punctuations.

pecl/intl?

Collator extension allows you to compare strings.

NumberFormatter - allows to format numbers as strings according to the localized format or given pattern or set of rules.

MessageFormatter - produces concatenated messages.

Future of PHP6

- APC bundled
- "taint" mode - where untrusted data is marked as such until you clean it up
- closures, aka "mind benders"
- a couple of syntactical sugar treats
- PCRE default regex engine
- 64-bit integer type
- general cleanup

gravitonic.com/talks