

CACHING FOR WORDPRESS

BRANDON SAVAGE
MESSAGE SYSTEMS

WHO AM I?

- Regular blogger on the Wordpress platform.
- Employee of Message Systems.
- Developing PHP since 2004.

WHAT WE'LL DISCUSS

- What is caching?
- Why do we need caching?
- Improving code performance
- Improving content performance
- Available caching plugins for Wordpress

WHAT IS CACHING?

- Caching is the storage of repeatedly needed but infrequently updated data, so that it can be retrieved quickly.
- Cached data is usually complete, and requires minimal processing to improve request times.

**CACHING MAKES YOUR
SITE FASTER, YOUR
SERVER HAPPIER, AND
YOUR READERS LOVE
YOU.**

WHY IS CACHING IMPORTANT?

- Caching improves the speed of page load.
- Caching reduces server load.
- A basic Wordpress install makes 9,829 function calls per request!

```
9803      0.3184      0      ->
9804      0.3184      0      ->
9805      0.3184      0      -> wp_
9806      0.3185      0      -> c
9807      0.3185      0      ->
9808      0.3185      0      ->
9809      0.3186      0      -> wpdb->__destruct()
9810      0.3186      0      -> WP_Object_Cache->__
9811      0.3187      0      -> shutdown_action_hoc
9812      0.3187      0      -> do_action('shutdo
9813      0.3187      0      -> is_array(array
9814      0.3187      0      -> is_array('') /l
9815      0.3188      0      -> func_num_args()
9816      0.3188      0      -> ksort(array (1
9817      0.3188      0      -> reset(array (1
9818      0.3188      0      -> current(array (
9819      0.3189      0      -> is_null('wp_ob_
9820      0.3189      0      -> array_slice(arr
9821      0.3189      0      -> call_user_func
9822      0.3189      0      -> wp_ob_end_flu
9823      0.3189      0      -> ob_end_flush
9824      0.3190      0      -> ob_end_flush
9825      0.3190      0      -> next(array (1 =
9826      0.3190      0      -> array_pop(array
9827      0.3190      0      -> wp_cache_close()
9828      0.3192      0      -> WP_Object_Cache->__
9829      0.3194      0      -> wpdb->__destruct()
9830      0.3212      0
9831  TRACE END  [2009-05-16 01:21:37]
```

**WORDPRESS BY ITSELF
IS VERY INEFFICIENT!**

**WEB DEVELOPMENT
PARADOX: THE MORE
FEATURES YOU ADD,
THE SLOWER YOUR
APPLICATION RUNS.**

IT'S NOT NECESSARILY WORDPRESS' FAULT!

- More features mean more code is needed to execute those features.
- PHP's "share nothing" nature means requests can't be shared.
- Caching helps solve this paradox.

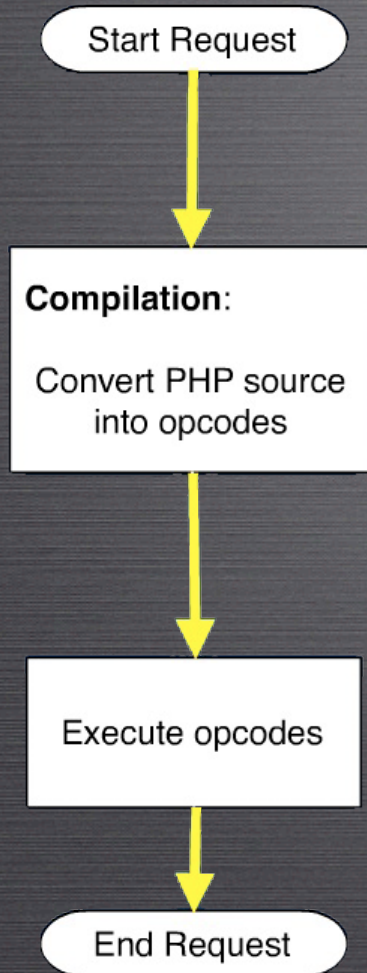
CACHING ALLOWS US
TO “SHARE
SOMETHING”
BETWEEN REQUESTS.

CACHING THE CODE

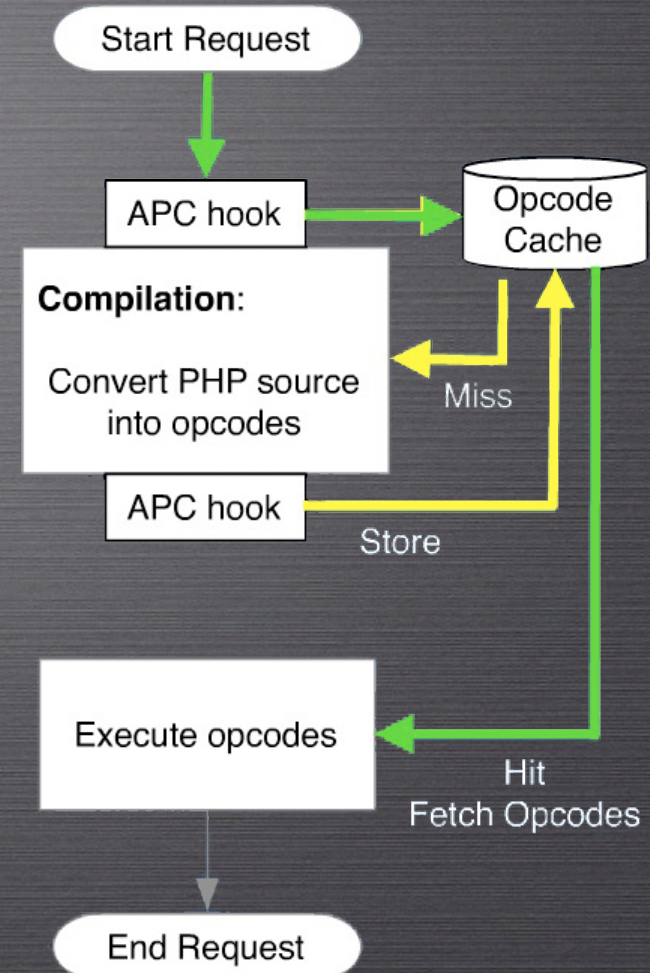
OPCODE CACHE

- PHP is not a compiled language; code is compiled and executed at run time.
- Compiling code takes extra time, extra CPU, and extra resources.
- Opcode caches take the processed code (opcode) and store it for easy retrieval.

PHP

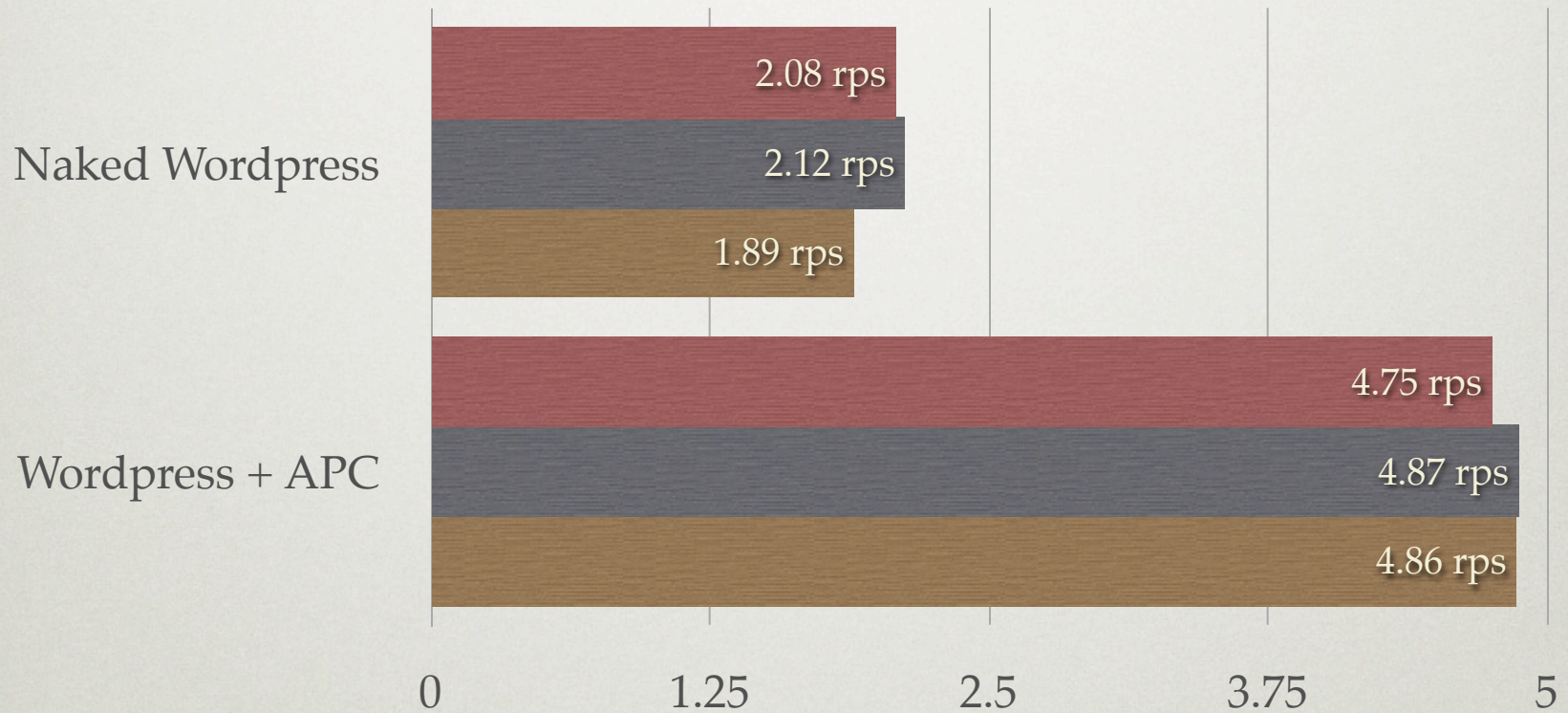


APC



(BORROWED FROM APC@FACEBOOK)

APC + WORDPRESS



ADDING APC
IMPROVES
WORDPRESS'
PERFORMANCE 237%

**(4.82 REQUESTS PER
SECOND IS STILL SLOW
THOUGH!)**

CACHEING THE CONTENT

WHY CACHE CONTENT?

- Most blogs are read-heavy - the content doesn't change very frequently.
- Cached content can reduce the load on your server and increase request response times.
- When high traffic shows up, you'll be ready.

DRAWBACKS TO CACHING

- The more cached your site, the less you're truly using Wordpress.
- Cached content introduces the possibility of stale content being presented.

Any sufficiently complex caching infrastructure is indistinguishable from black magic

7:46 AM Apr 9th via web

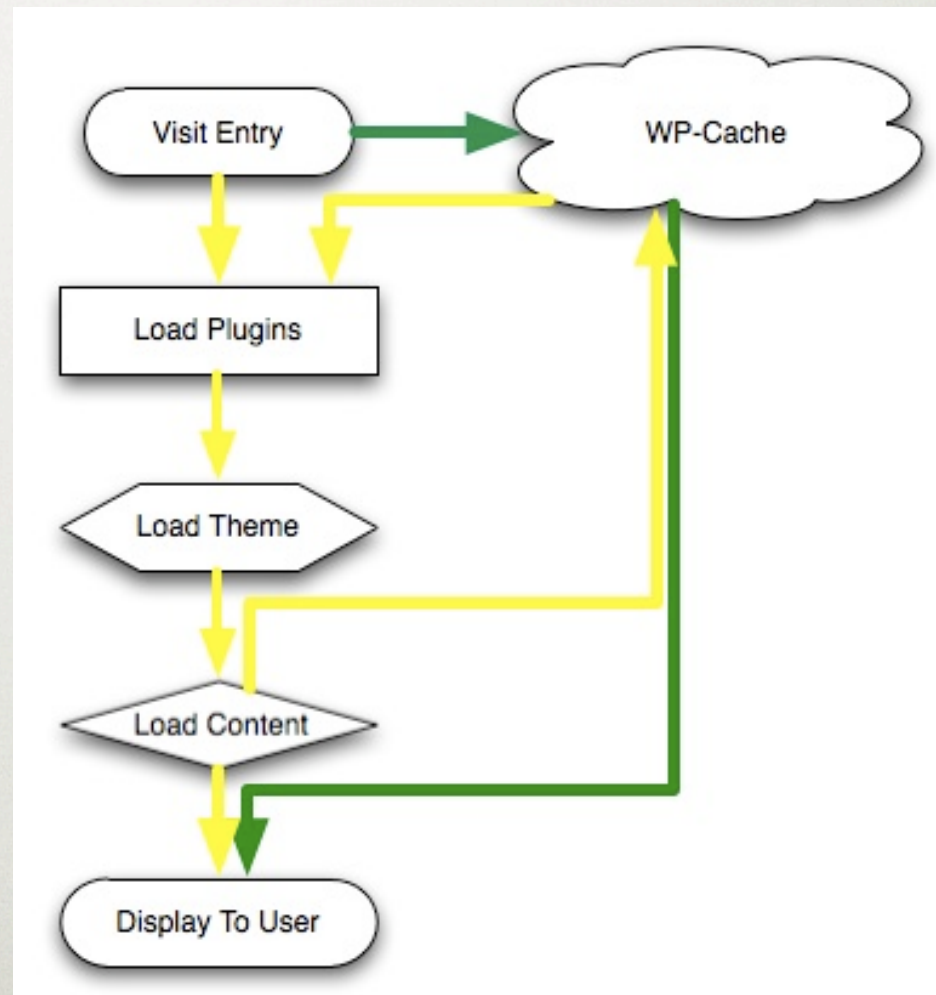
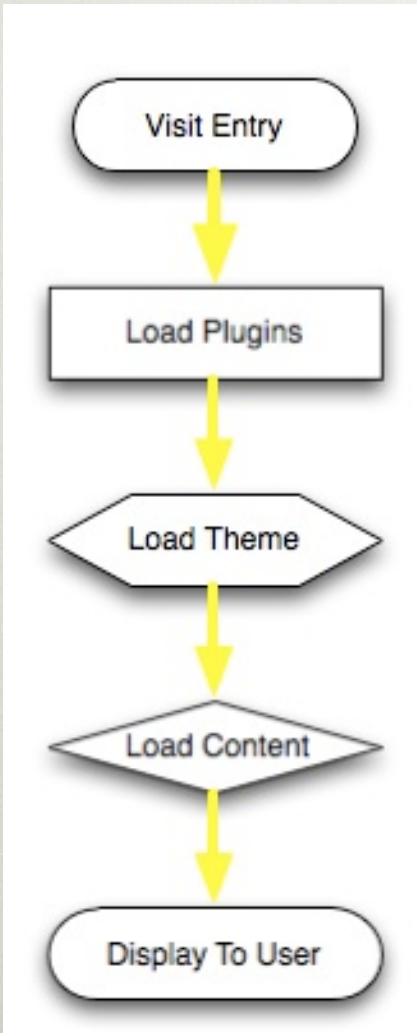


[lukewelling](#)

Luke Welling

IN CACHING THE
CONTENT, WE'RE
REDUCING THE ROLE
OF PHP IN OUR
APPLICATION.

REGULAR PROCESS Vs. CACHING



**THREE MAJOR
CACHING PLUGINS
FOR WORDPRESS**

1 : WP-CACHE

WP-CACHE: ADVANTAGES

- Original plugin for Wordpress; well tested.
- Captures output, stores it in files for delivery to future viewers.
- On each request, makes a determination as to whether or not a file exists. If it does, it is served; if it doesn't, it generates a file for the future.

WP-CACHE: DISADVANTAGES

- Invokes PHP on each request to determine cache validity and file existence.
- Does not play well with APC by default (must disable the config file from being cached by APC)

#2. WP-SUPER-CACHE

WP-SUPER-CACHE: ADVANTAGES

- Combines WP-Cache and a static file cache.
- Static files mean it's possible to serve files directly from the file system.
- Doesn't necessarily invoke PHP on each request.

WP-SUPER-CACHE: DISADVANTAGES

- Does not cache included files like Javascript, CSS, etc.
- Since it's based on WP-Cache, it has the same issue with APC.

#3. W3 TOTAL CACHE

W3 TOTAL CACHE: ADVANTAGES

- Written by Mashable to enhance their own instance of Wordpress.
- Provides options to cache database objects, assets, include files and others in a variety of caches like APC, Memcached and S3.

W3 TOTAL CACHE: DISADVANTAGES

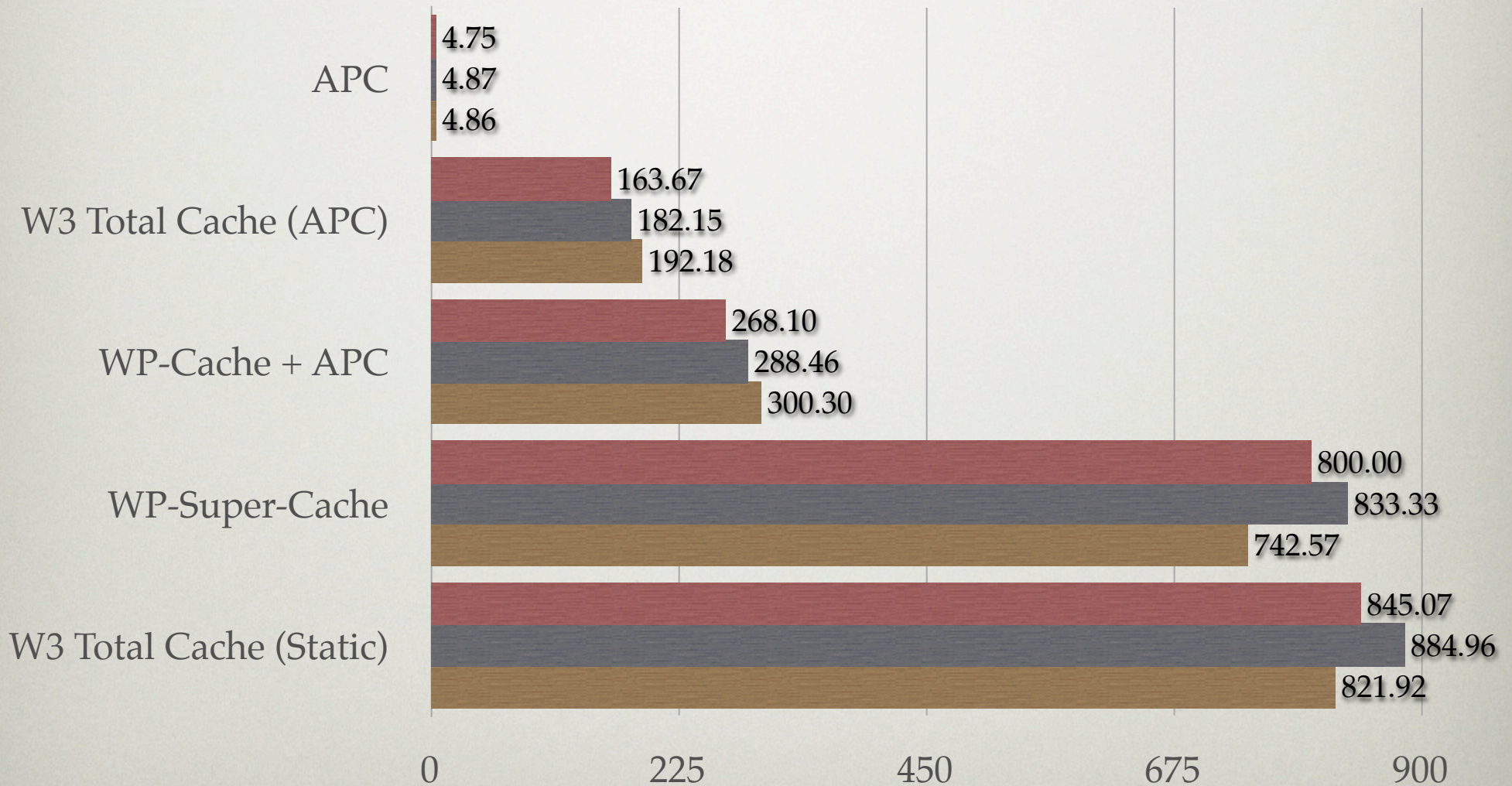
- Configuration is complex. There are many options. It's overkill for most blogs and is easy to misconfigure.
- Encourages use of memory-based caches, which must invoke PHP on each request.

COMPETITION OF THE CACHES

BENCHMARK METHODOLOGY

- 1GB Ubuntu server. Wordpress blog was a copy of BrandonSavage.net.
- Apache Bench, with 3000 requests (non concurrent).
- Benchmarking server and web server on same machine.
- Standard benchmark disclaimers apply.

BENCHMARKS



SO THE WINNER?

IT REALLY DEPENDS.

THE WINNER IS...

- If you want to have ways to cache files with S3 and database objects, but not use static files, W3 Total Cache with APC is a decent option.
- If you want to use some static files but don't care about APC or S3, use WP-Cache.

THE WINNER IS...

- For sites that want to withstand the “Digg effect,” WP-Super-Cache is perfectly capable.
- For heavy duty caching, with S3, database caching, minifying, compression and APC / Memcached, use W3 Total Cache.

**BOTTOM LINE: USE
THE TOOL THAT
SOLVES YOUR
PROBLEM.**

HOW TO REACH ME

- I'm a regular blogger. My blog is <http://www.brandonsavage.net/>
- I can be found on Twitter
@brandonsavage

QUESTIONS?