DIY Drupal 7 Performance

Mike Carper @mcarper (mikeytown2)

Pacific Northwest Drupal Summit
October 2015

https://goo.gl/L4P3MA

Follow along

https://goo.gl/L4P3MA

https://goo.gl/L4P3MA

https://goo.gl/L4P3MA

About Mike

- November 2008 - Started using Drupal
- January 2009 - Discovered Boost module
- September 2009 - Boost 6.x-1.0 released
- November 2009 - Started @ Datasphere

https://goo.gl/L4P3MA

About Mike (@ Datasphere)

- Our team scaled a single Drupal 6 domain access multisite up to 1,300 different domains

- Our Drupal 7 site is 99% authenticated with a nothing shared between users, making caching very hard to do

- Created a lot of performance related modules that helped us out a lot

https://goo.gl/L4P3MA

"BEFORE YOU MARRY A PERSON YOU SHOULD FIRST MAKE THEM USE A COMPUTER WITH SLOW INTERNET TO SEE WHO THEY REALLY ARE."

WILL FERRELL

https://goo.gl/L4P3MA

https://goo.gl/L4P3MA

What can we do to make sure users enjoy using your site?

https://goo.gl/L4P3MA

Make sure it's FAST

For logged in & logged out users

https://goo.gl/L4P3MA

How?

https://goo.gl/L4P3MA

Throw Money At

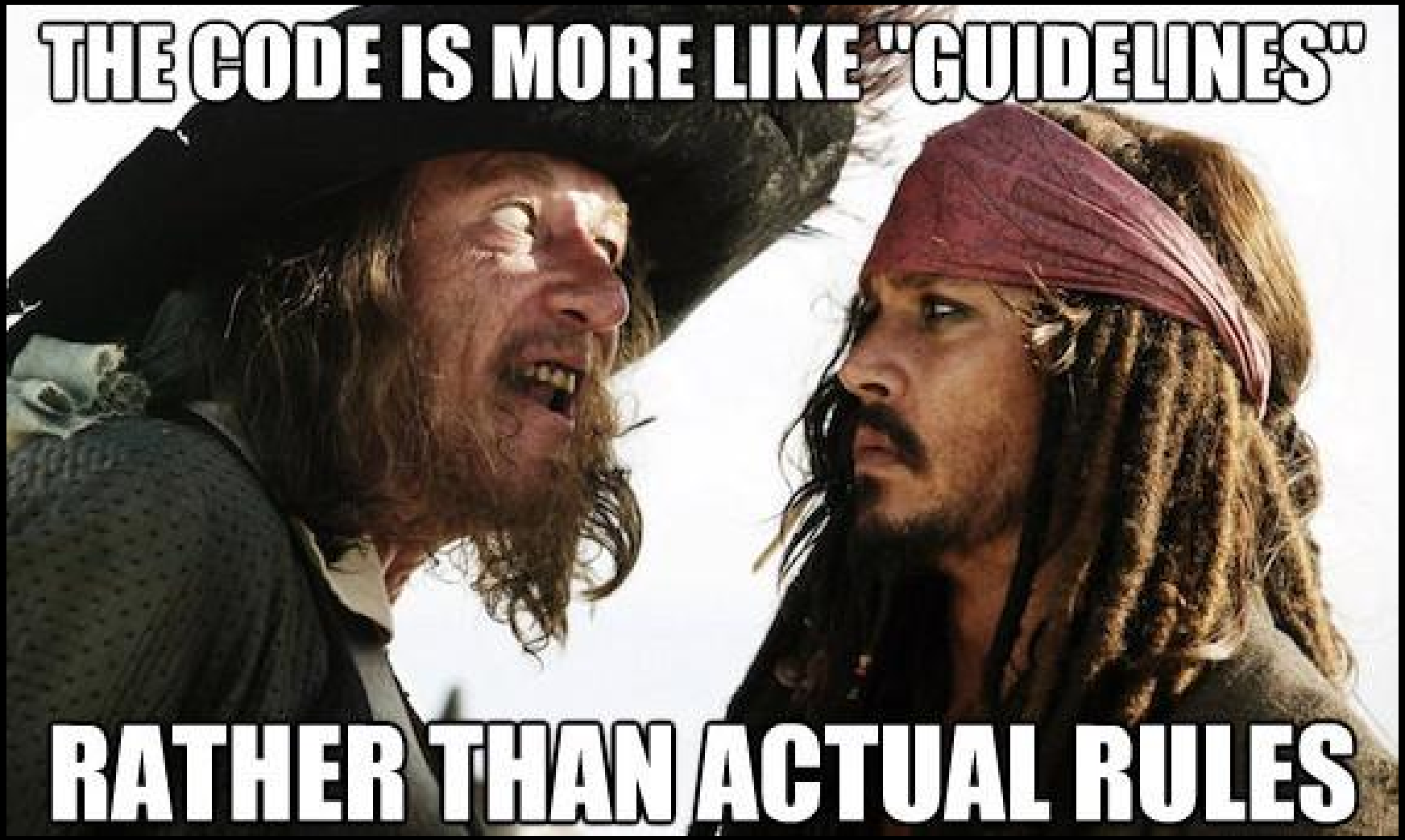
https://goo.gl/L4P3MA

Or you can DIY!

https://goo.gl/L4P3MA

Some rules to follow

https://goo.gl/L4P3MA



https://goo.gl/L4P3MA

General Guidelines

Anonymous visitor caching is easy.

Big performance gains come asynchronous/parallel and deferred code execution. Why do it now when you can do it later.

Less is better.

https://goo.gl/L4P3MA

General Guidelines (cont'd)

Big improvements can be found on the frontend.

If you have to pick scalability over performance 9 times out of 10 pick scalability; you'll thank yourself later on when you have more than 20 users logged in at once.

Measure changes - Was that good or bad?

https://goo.gl/L4P3MA

Page Caching

mainly useful for anonymous users

https://goo.gl/L4P3MA

Caching to the rescue!

https://goo.gl/L4P3MA

Popular Page Cache Options

- Drupal Core
- Boost
- Memcache/Redis & Varnish

https://goo.gl/L4P3MA

Drupal Core

Advantages:

- Comes with core
- Better than nothing

Disadvantages:

- Relies on the database
- Slow as PHP is needed

https://goo.gl/L4P3MA

https://goo.gl/L4P3MA

https://goo.gl/L4P3MA

Boost

Advantages:

- Works on almost all shared hosting
- Fast as no PHP is executed

Disadvantages:

- Need to copy in htaccess rules
- Cache is not easily shared with multiple servers

https://goo.gl/L4P3MA

https://goo.gl/L4P3MA

Memcache/Redis & Varnish

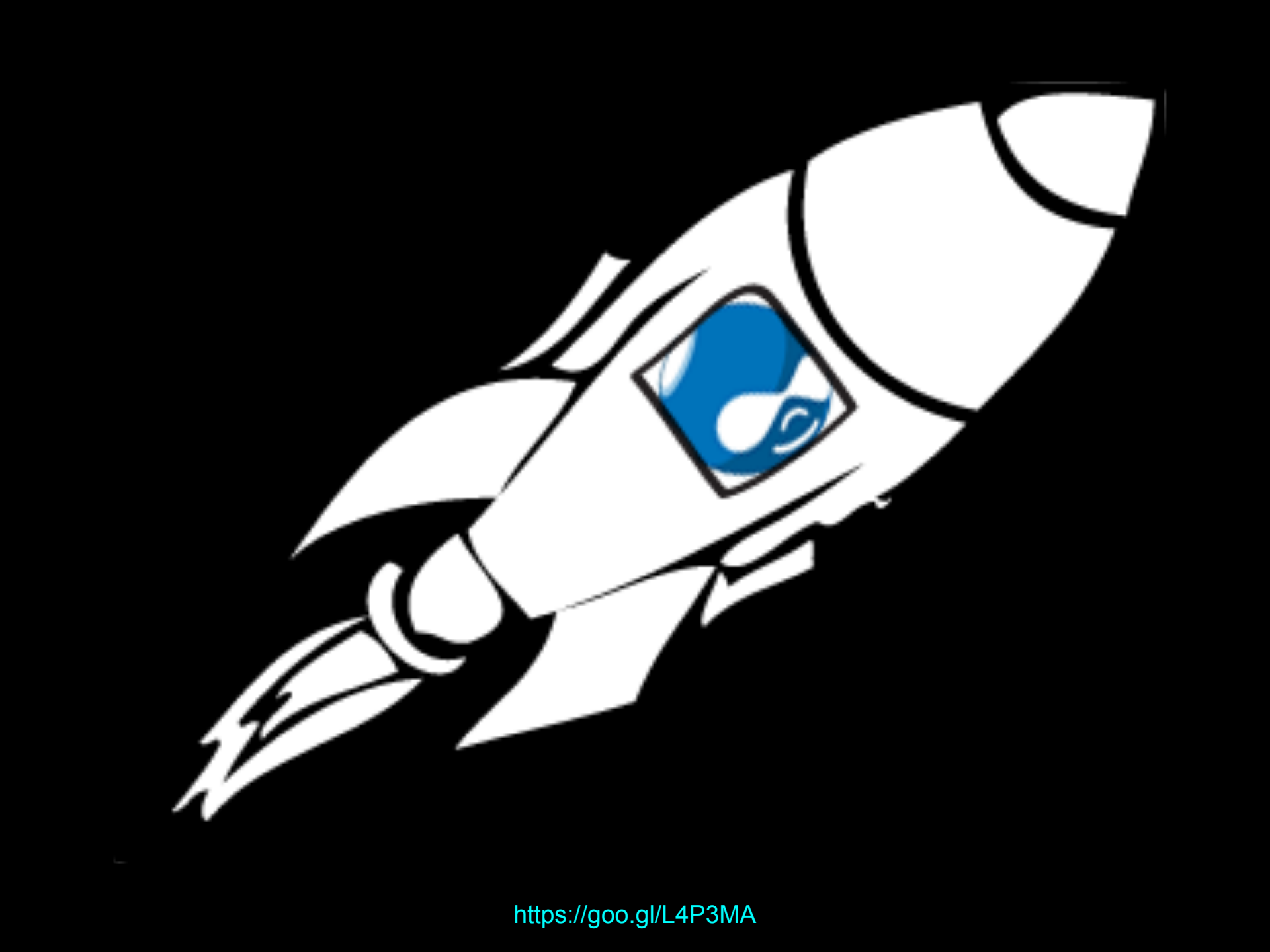
Advantages:

- Fast
- Can help logged in users as well

Disadvantages:

- Need shell access
- Not simple to setup
- High availability is hard

https://goo.gl/L4P3MA



https://goo.gl/L4P3MA

Optimization and Tuning of Drupal 7

https://goo.gl/L4P3MA

Optimization Overview

Core page and block caching

Core CSS & JS aggregation

Panels & views caching

Disable unnecessary modules

Modules that usually help

Poormanscron in core

Enable fast 404

Core patches

https://goo.gl/L4P3MA

Optimization Overview (cont'd)

Database Connection

Op Code

php.ini

PHP Notices/Warnings

Basic database tuning

Deadlock Detection

Diagnosing Stuck Queries

Slow Query Analysis

Cachegrind

https://goo.gl/L4P3MA

TLDR

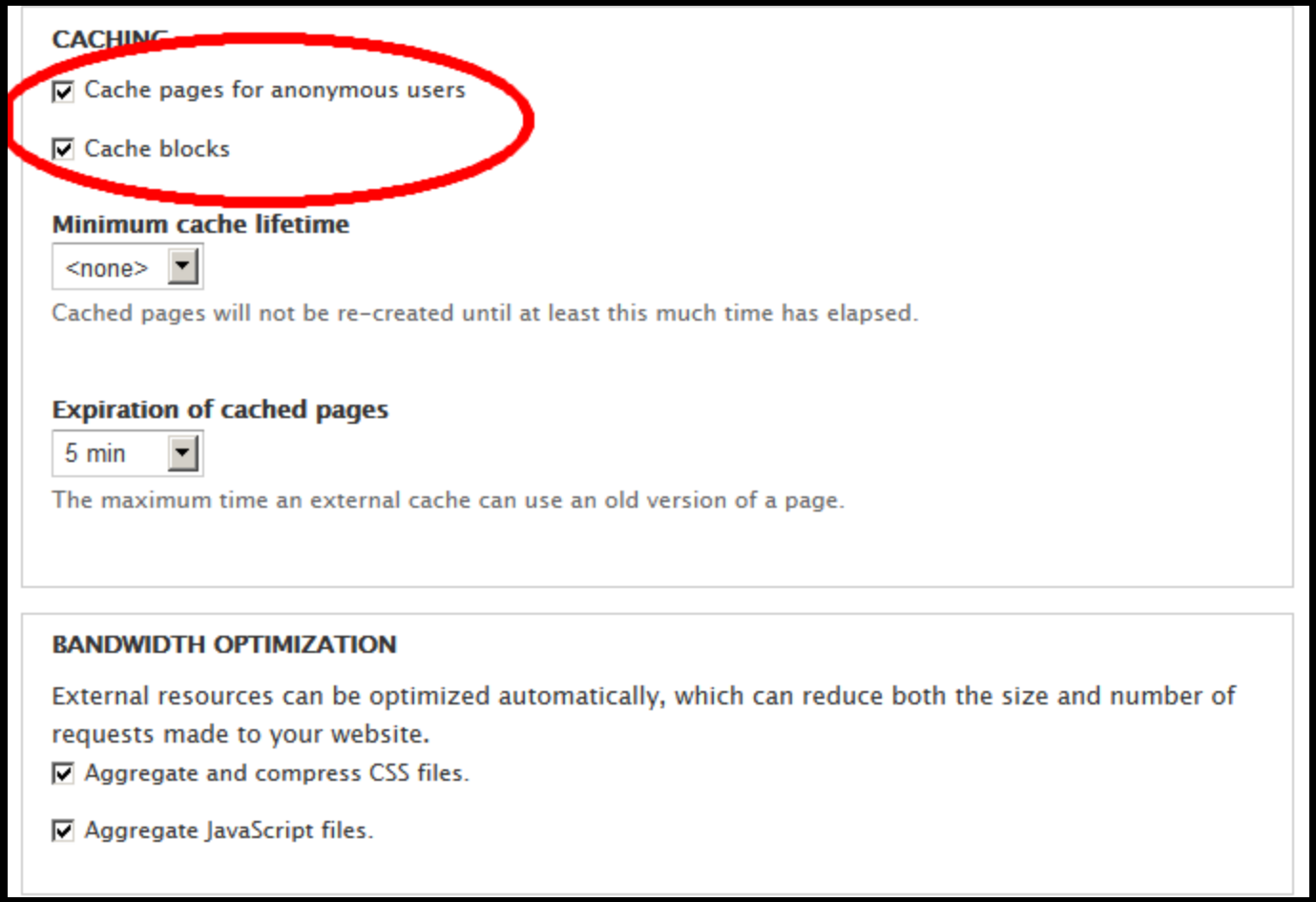
https://goo.gl/L4P3MA

Core and Block Caching

admin/config/development/performance

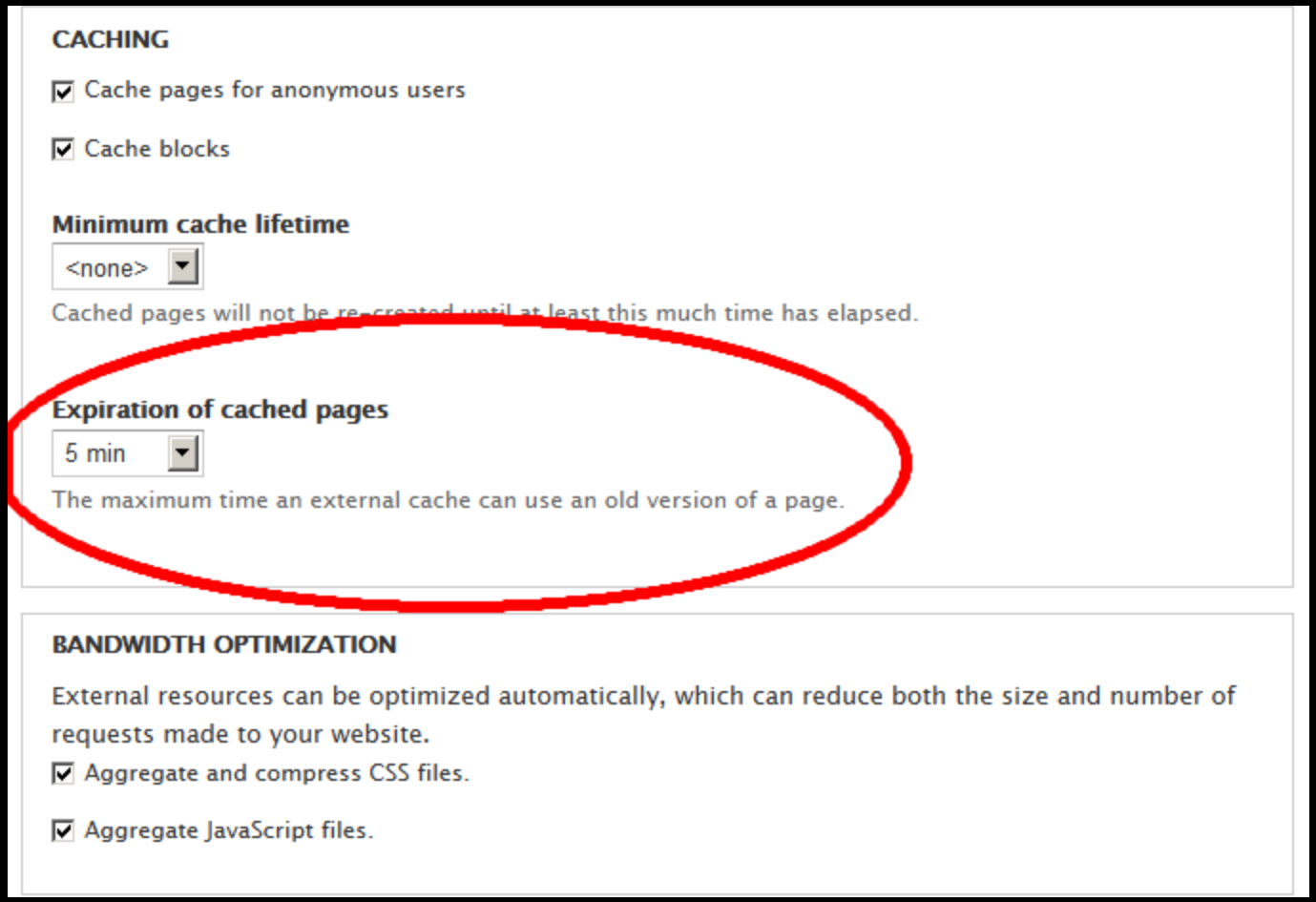
https://goo.gl/L4P3MA

admin/config/development/performance



https://goo.gl/L4P3MA

admin/config/development/performance



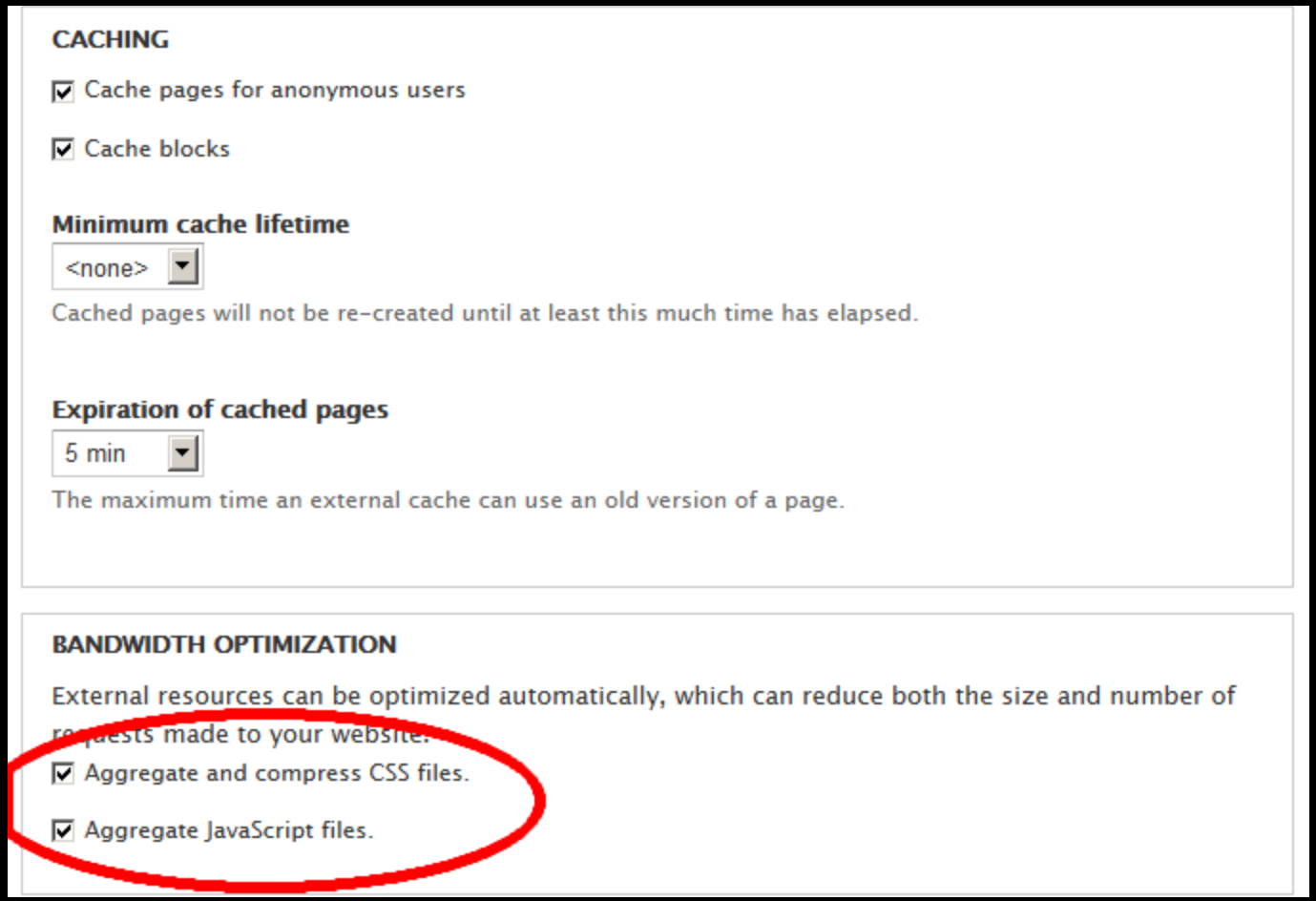
https://goo.gl/L4P3MA

Enable CSS & JS Aggregation

admin/config/development/performance

https://goo.gl/L4P3MA

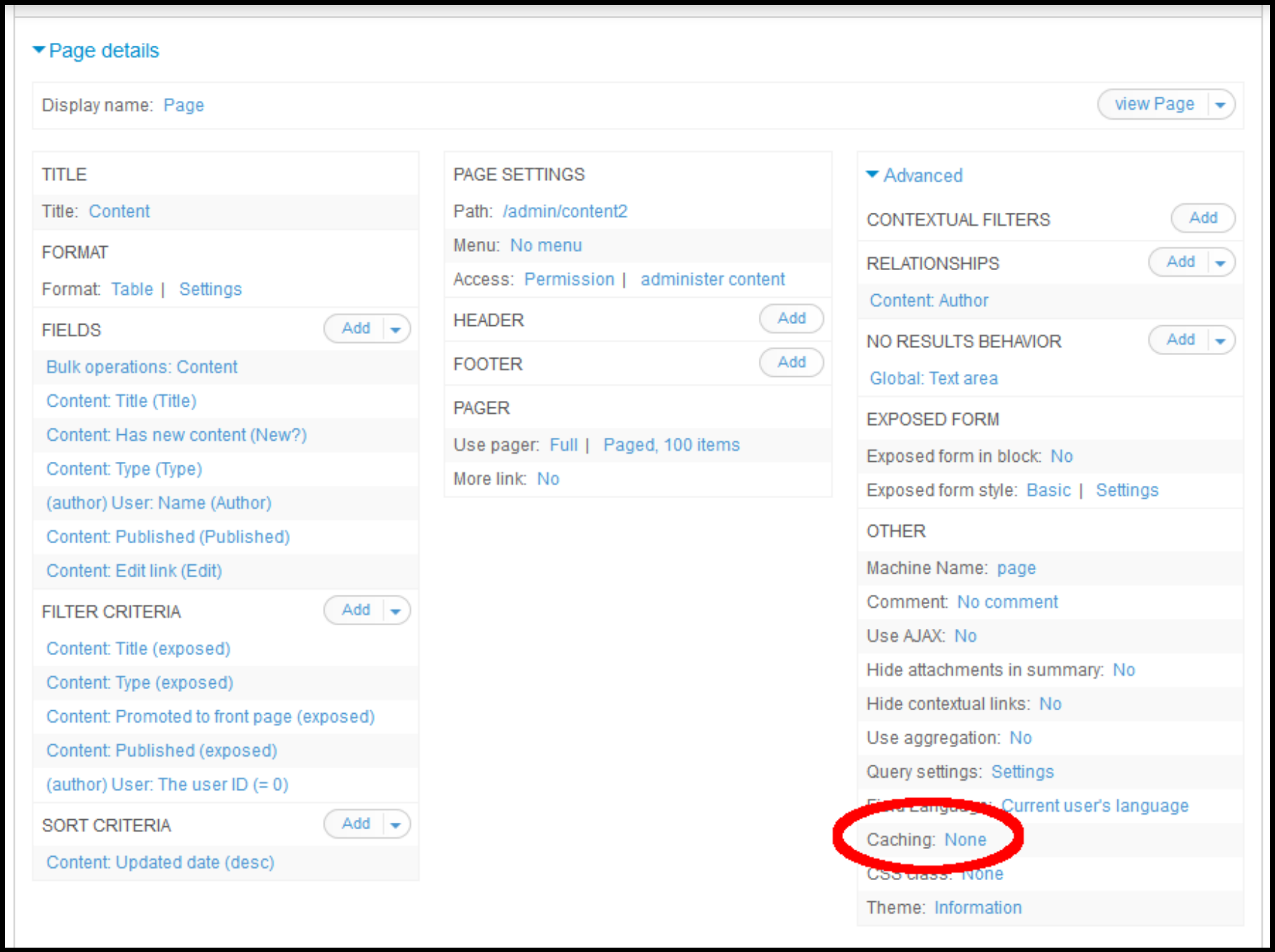
admin/config/development/performance



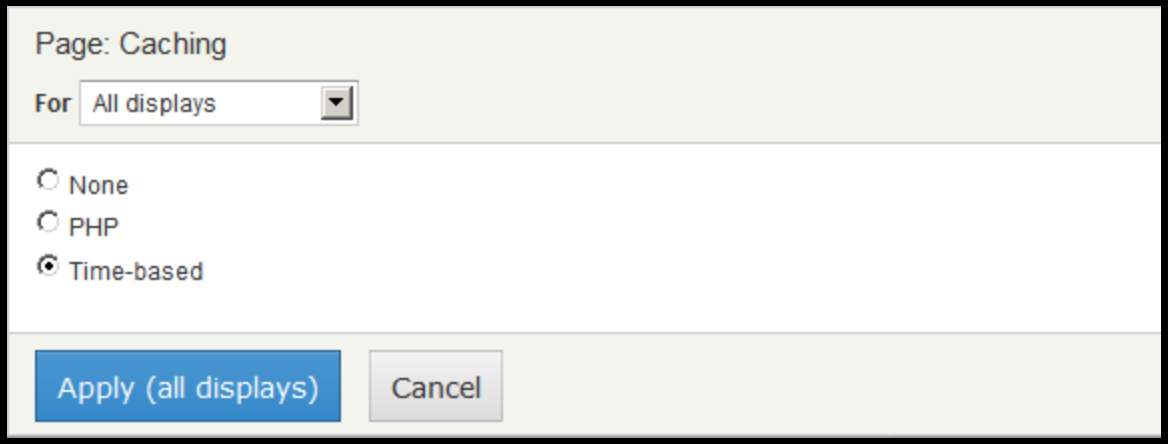
https://goo.gl/L4P3MA

Views Caching

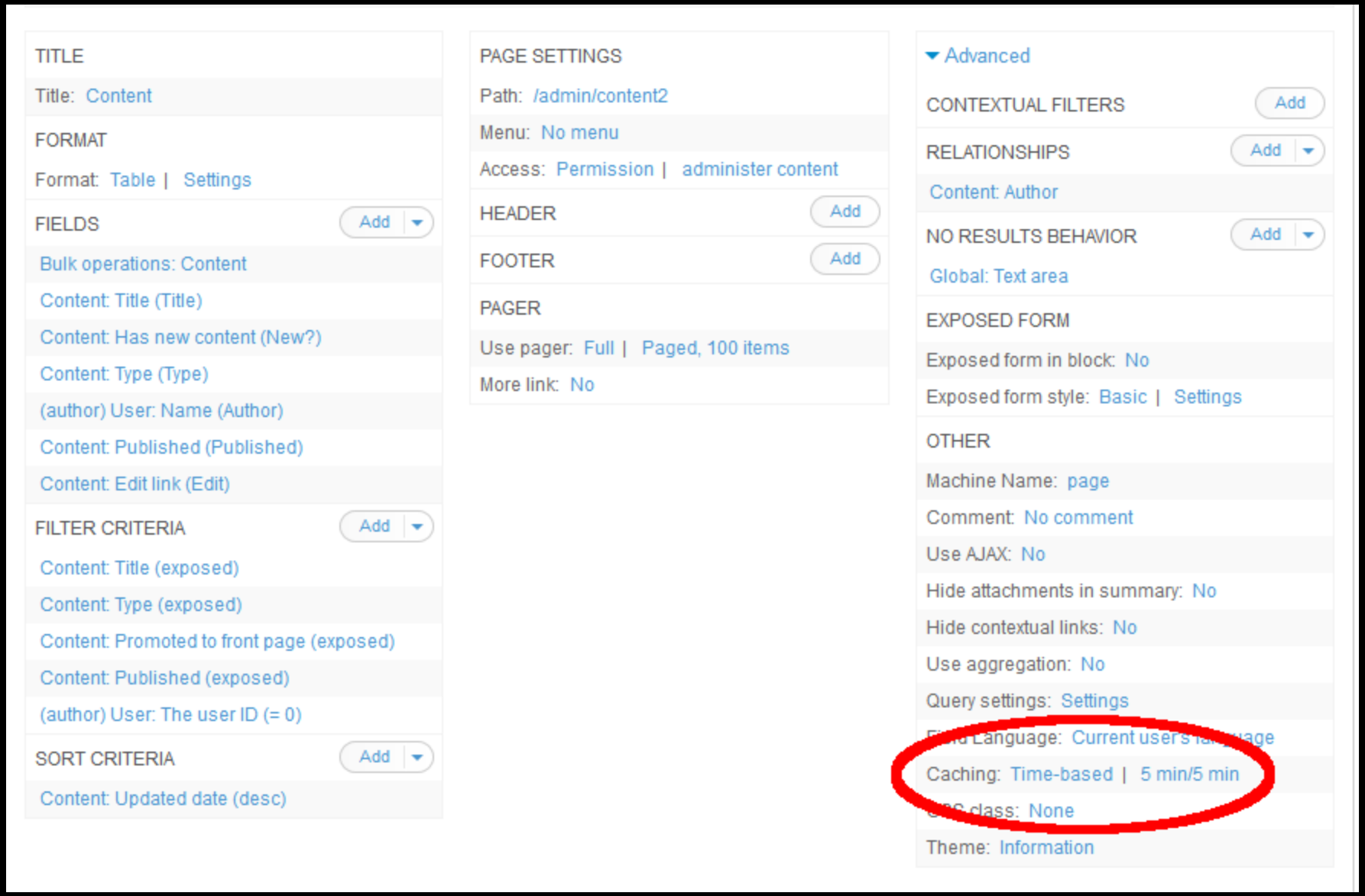
https://goo.gl/L4P3MA



https://goo.gl/L4P3MA



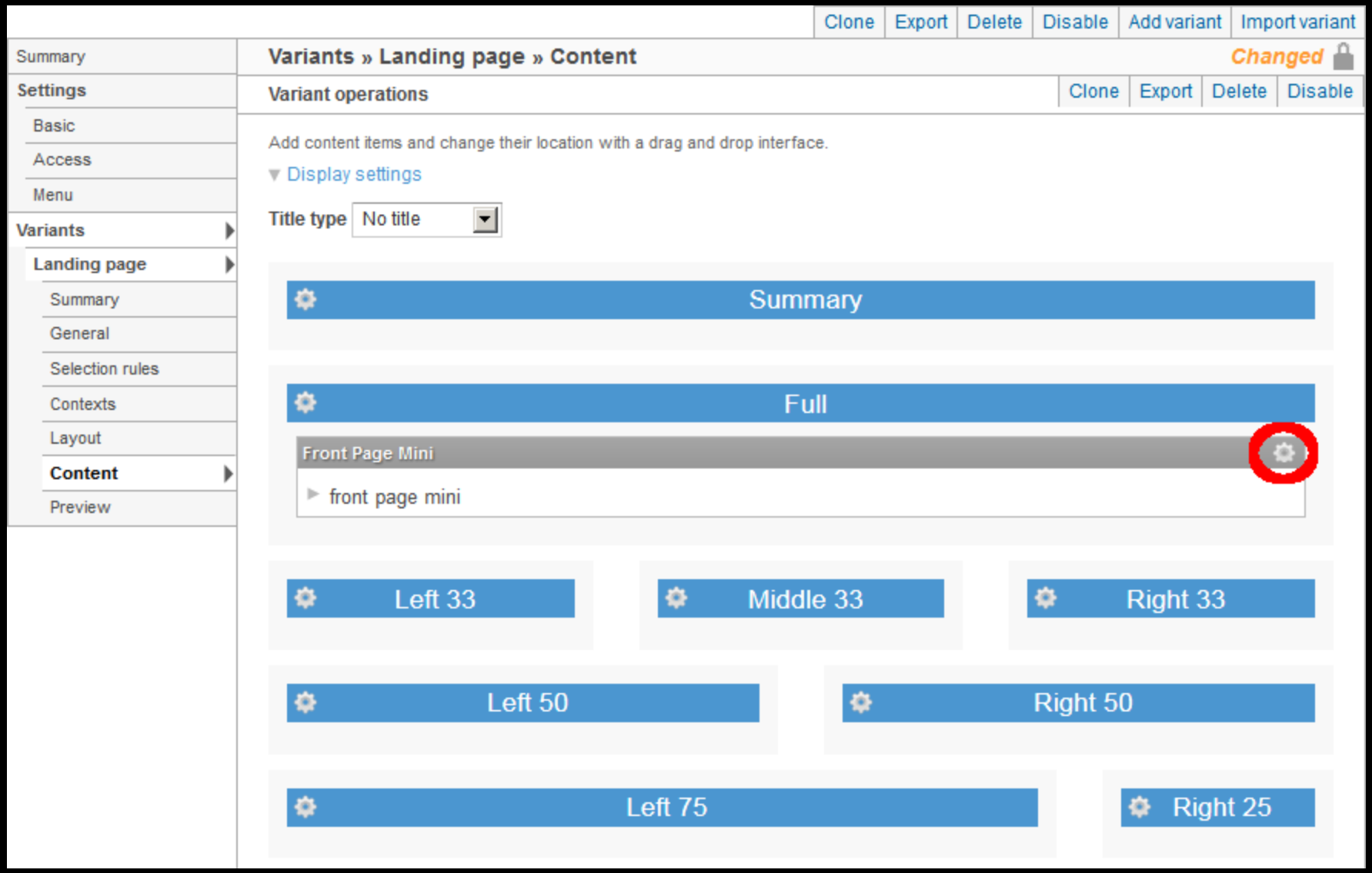
https://goo.gl/L4P3MA



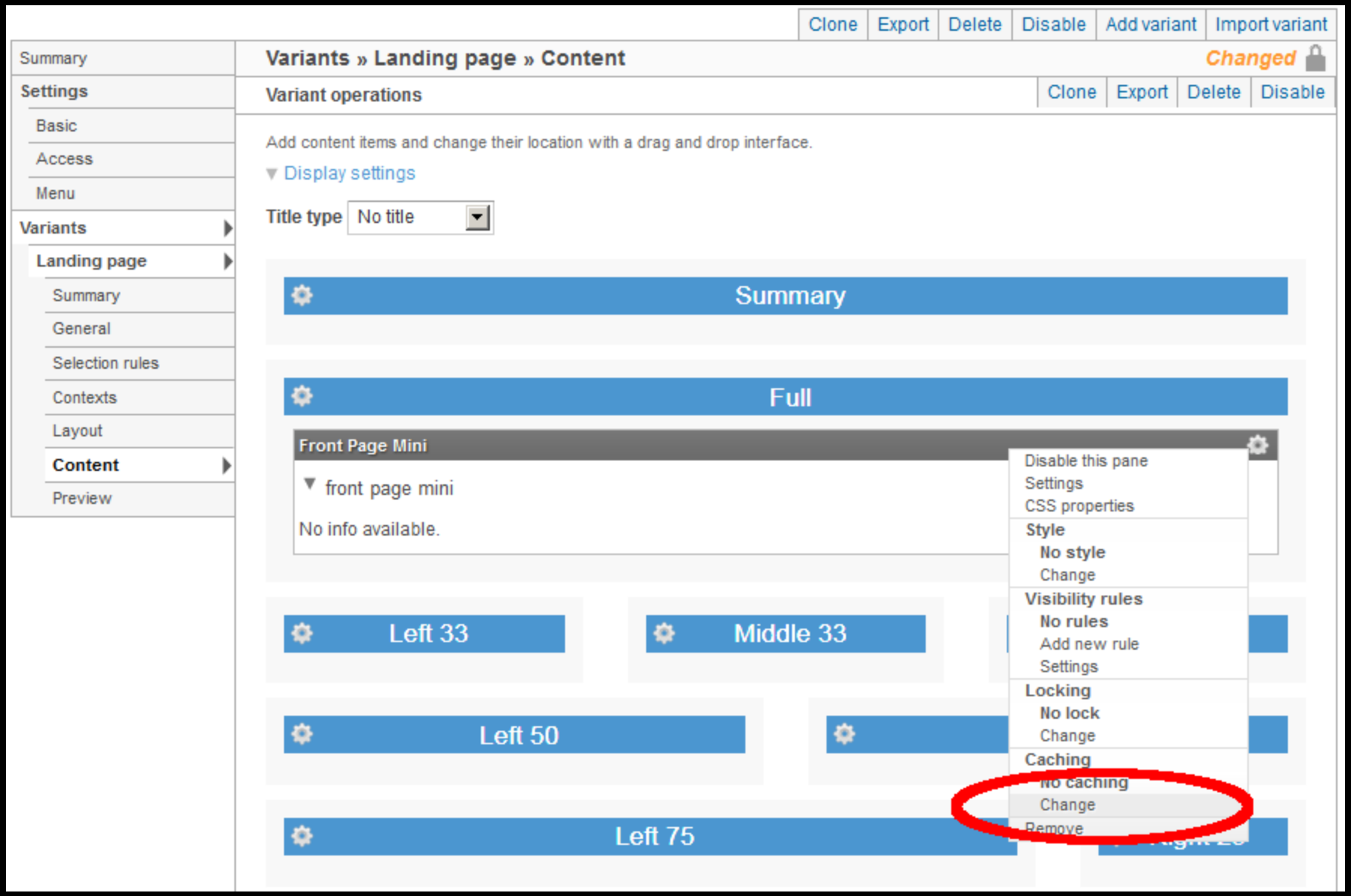
https://goo.gl/L4P3MA

Panels Caching

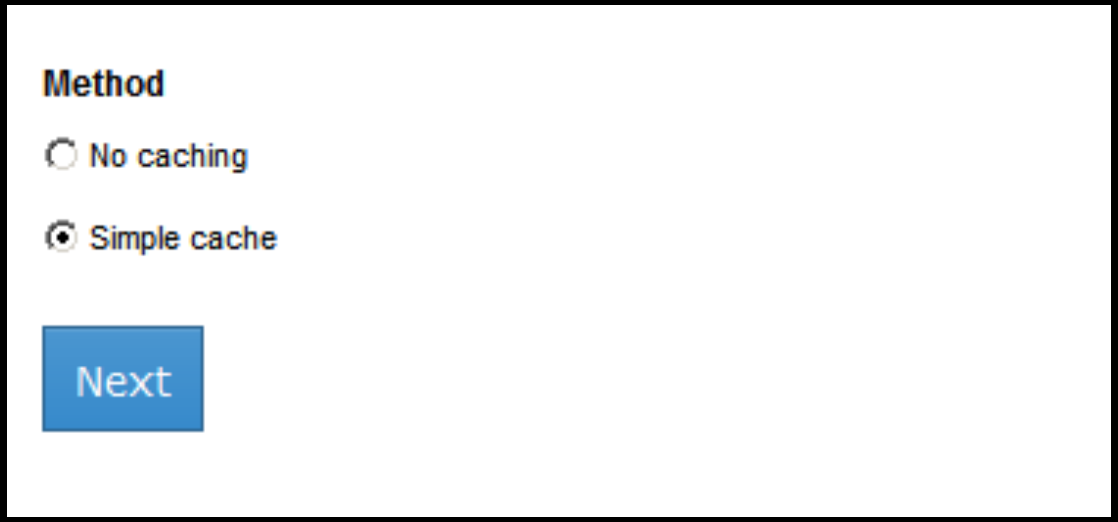
https://goo.gl/L4P3MA



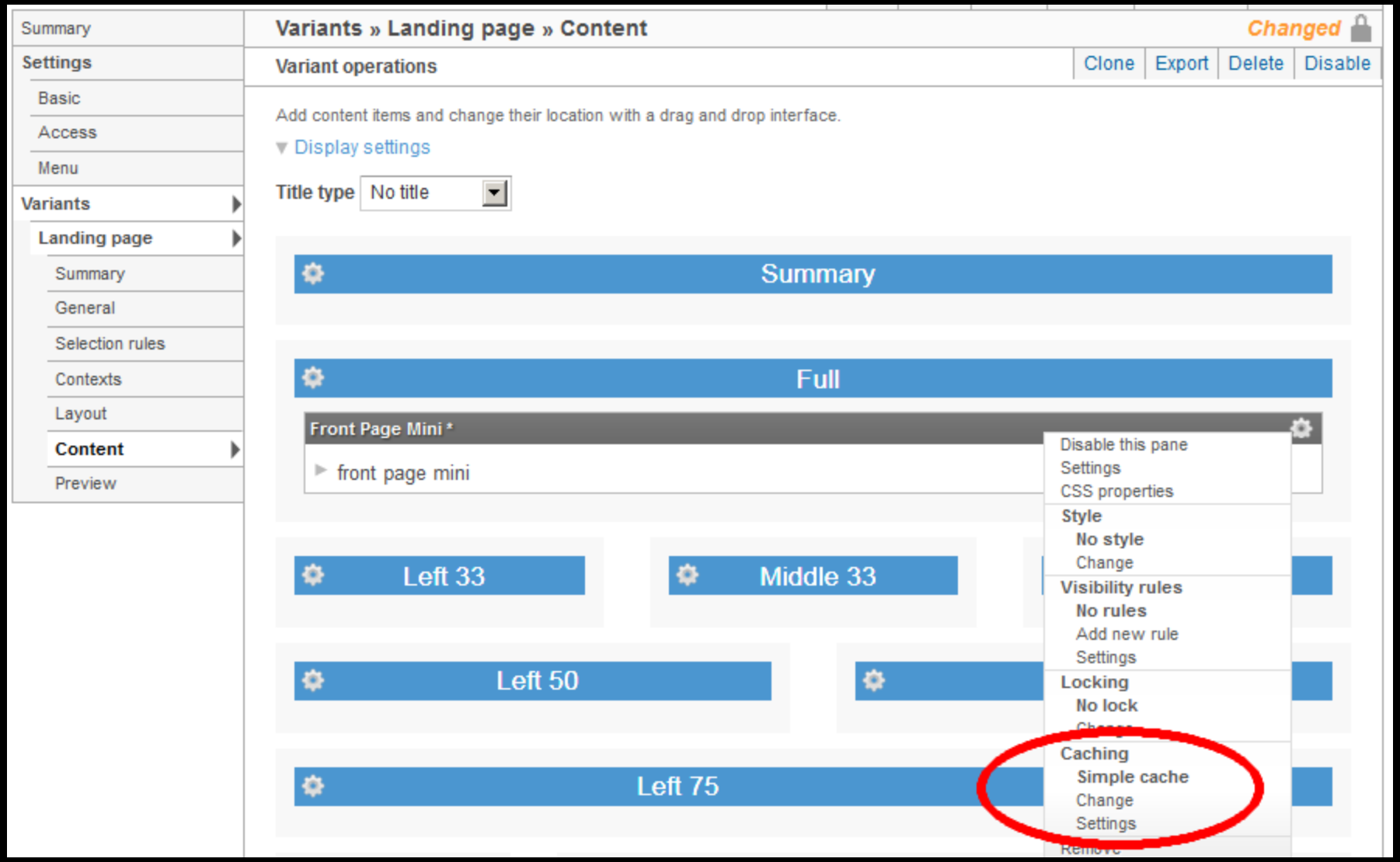
https://goo.gl/L4P3MA



https://goo.gl/L4P3MA



https://goo.gl/L4P3MA



https://goo.gl/L4P3MA

Disable unnecessary modules

https://goo.gl/L4P3MA

https://goo.gl/L4P3MA

https://goo.gl/L4P3MA

https://goo.gl/L4P3MA

https://goo.gl/L4P3MA

Most modules behave well; some do not. The poor performing modules are usually the ones without a lot of users.

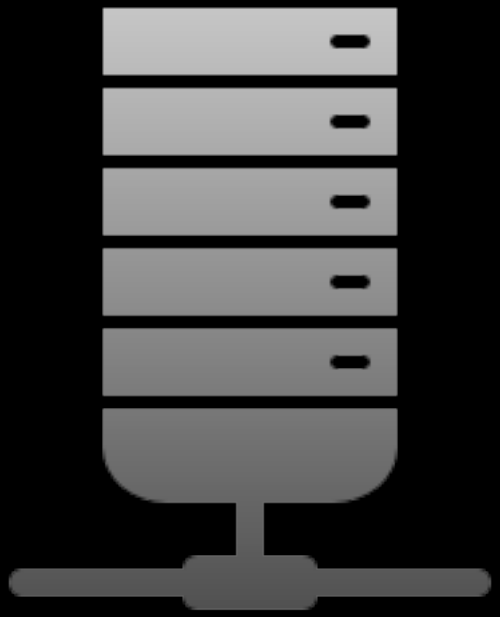
The next release of D7 (7.40) contains a big fix that should speed up most drupal installs by 5-10%; the module_implements() function has been improved in core.

https://goo.gl/L4P3MA

Modules that help with optimizing Drupal

https://goo.gl/L4P3MA

Backend



https://goo.gl/L4P3MA

Entity Cache Module

A lot more useful if using memcache
https://drupal.org/project/entitycache

https://goo.gl/L4P3MA

Render Cache

Caches rendered entities currently
https://www.drupal.org/project/render_cache

https://goo.gl/L4P3MA

Cache Expiration

Works with the page cache
https://www.drupal.org/project/expire

https://goo.gl/L4P3MA

Imageinfo Cache

Generate image styles on file upload

https://www.drupal.org/project/imageinfo_cache

https://goo.gl/L4P3MA

Asynchronous Prefetch Database Query Cache

Fixes everything that is wrong with the core database cache; MySQL only

https://www.drupal.org/project/apdqc

https://goo.gl/L4P3MA

HTTP Parallel Request & Threading Library

Doesn’t do much out of the box but has some awesome tools when creating custom code
https://www.drupal.org/project/httprl

https://goo.gl/L4P3MA

Frontend

https://goo.gl/L4P3MA

Frontend

I always test changes made with Google's PageSpeed Insights and webpagetest.org

I set the connection speed to 3G or slower when testing on webpagetest.org

https://goo.gl/L4P3MA

ImageAPI Optimize

Optimize the image when it is saved

https://www.drupal.org/project/imageapi_optimize

https://goo.gl/L4P3MA

Defer Image

Prevent images from slowing down the initial load and render of a page
https://www.drupal.org/project/defer_image

https://goo.gl/L4P3MA

Font Awesome SVG

Use font awesome? Use this to reduce the bytes downloaded.
https://www.drupal.org/project/fontawesome_svg

https://goo.gl/L4P3MA

AdvAgg

Faster CSS/JS aggregation

https://www.drupal.org/project/advagg/

https://goo.gl/L4P3MA

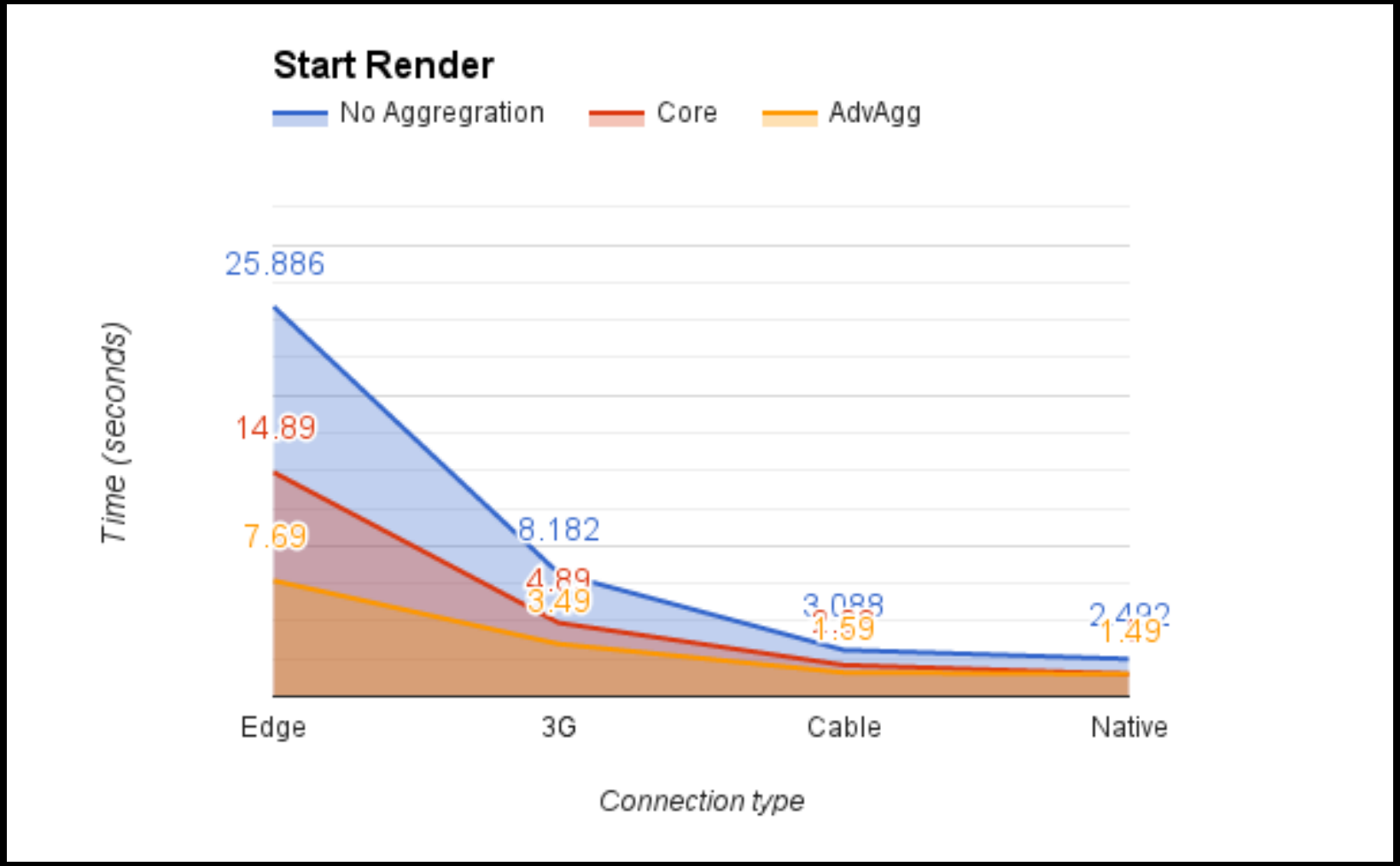
More info on AdvAgg

Drupal.org started using it this year. According to Google Analytics the average UC Browser page load time went from 13.3 seconds down to 5.4 seconds.

UC Browser is a mobile browser with a huge user base in China, strong adoption in India and continued growth in emerging regional markets.

https://goo.gl/L4P3MA

Stats from https://www.drupal.org/project/drupal
The slower the connection the bigger an effect AdvAgg has on improving the front end responsiveness.



https://goo.gl/L4P3MA

AdvAgg D.O Settings.

- Use Aggressive Render Cache
- Move all inline scripts to the bottom of the execution order & Move all external scripts to the top of the execution order
- Use DNS Prefetch for external CSS/JS
- Combine CSS files by using media queries
- Move JS to the footer - All but what is in the $all_in_footer_list
- Deferred JavaScript Execution: Add The defer Tag To All Script Tags - All but external scripts

https://goo.gl/L4P3MA

AdvAgg D.O Settings. (cont'd)

- Deferred inline JavaScript Execution: Put a wrapper around inline JS so it runs from a setTimeout call
- Remove ajaxPageState CSS and JS data if ajax.js is not used on this page
- Change CSS bundles to 2
- Change JS bundles to 5
- Move Google Analytics analytics.js code from inline to be a file
- Prefetch stats.g.doubleclick.net/robots.txt

https://goo.gl/L4P3MA

Other AdvAgg settings

- Use JSMin
- Use Async Font loading

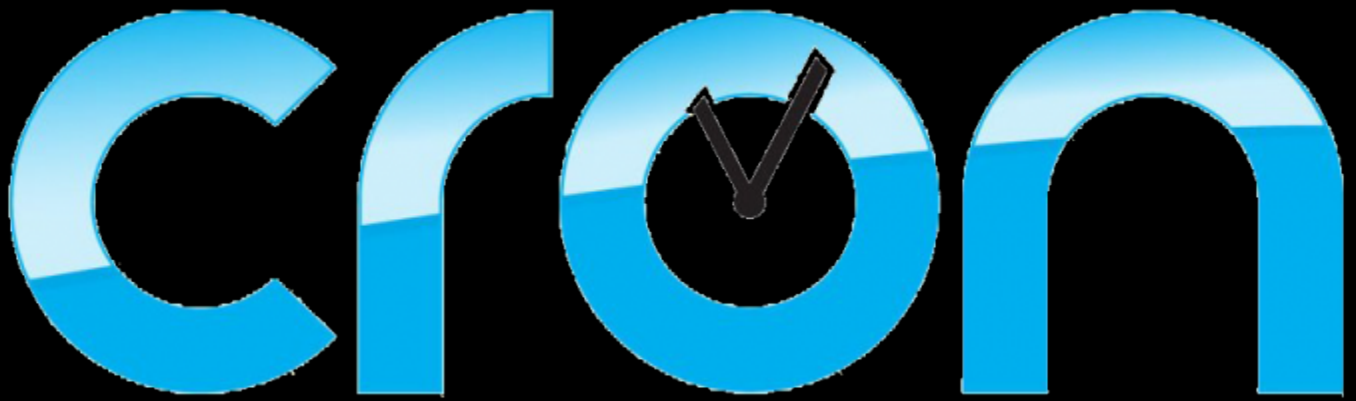
Future AdvAgg settings

- Inline critical css

https://www.drupal.org/node/2493801

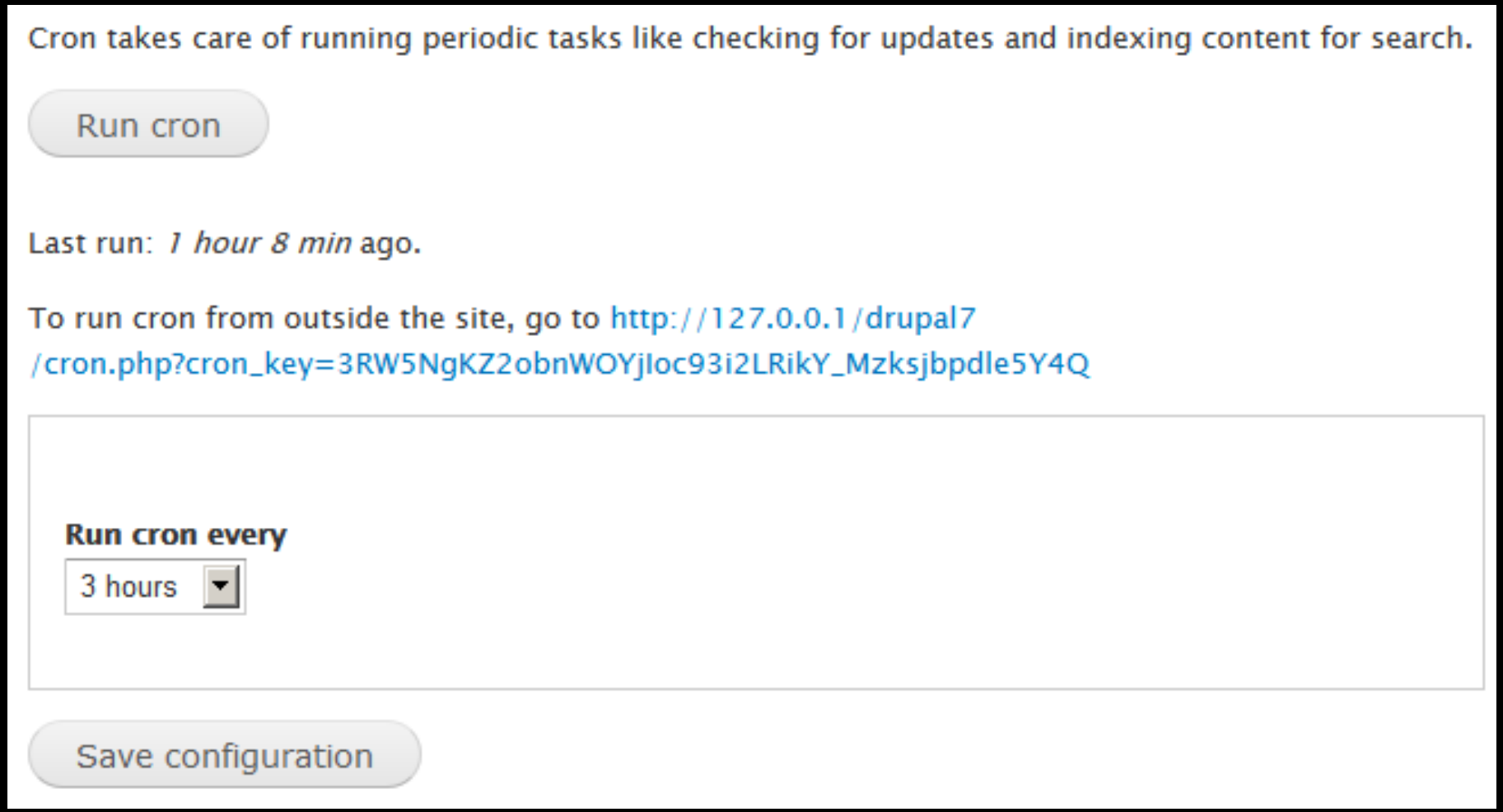
https://goo.gl/L4P3MA

Poormans cron in core



https://goo.gl/L4P3MA

admin/config/system/cron



https://goo.gl/L4P3MA

Follow a guide on how to setup cron on your server. Almost all shared hosting allows for cron jobs to be scheduled.

https://goo.gl/L4P3MA

Even bad hosts have instructions

https://goo.gl/L4P3MA

GoDaddy Support

> Help > Web Hosting > Databases > Accessing Your Cron Job Manager with Shared Hosting

Accessing Your Cron Job Manager with Shared Hosting

Cron is a standard Linux feature that lets you schedule tasks, called "Cron Jobs," to run unattended at a specified frequency.

To Create a Cron Job

1. Log in to your Account Manager.
2. Click **Web Hosting**.

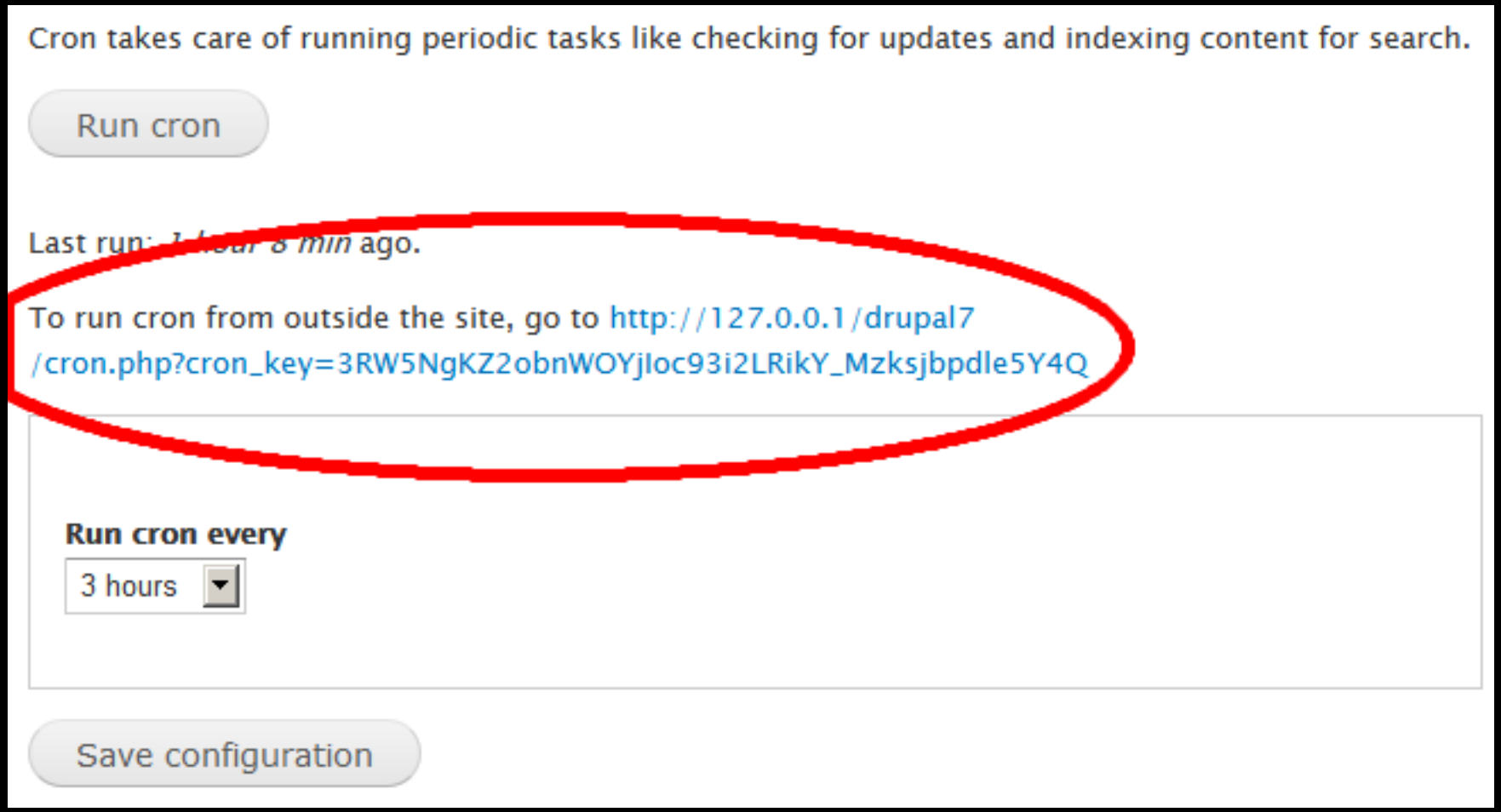
Click one of the following, based on the type of hosting you have. You can tell this by the second word in your hosting's description beneath your hosting account's domain name, e.g. *Deluxe Web* correlates to *Web/Classic (Hosting Control Panel)*.

Web/Classic (Hosting Control Panel)

1. Next to the hosting account you want to use, click **Launch**.
2. In the **Tools** section of the Hosting Control Panel, click the **Cron Job Manager** icon. This action displays the Cron Job Manager screen.
3. Click **edit** next to the **Email Address** heading in the left navigation area.
4. Enter the email address to which you would like to have all information sent regarding this Cron Job.
5. Click **Save**.
6. Click **Create Cron Job**.
7. Select and enter the options you want to use, or click **Custom** if you want more granular settings.
8. Click **Save**.

https://goo.gl/L4P3MA

admin/config/system/cron

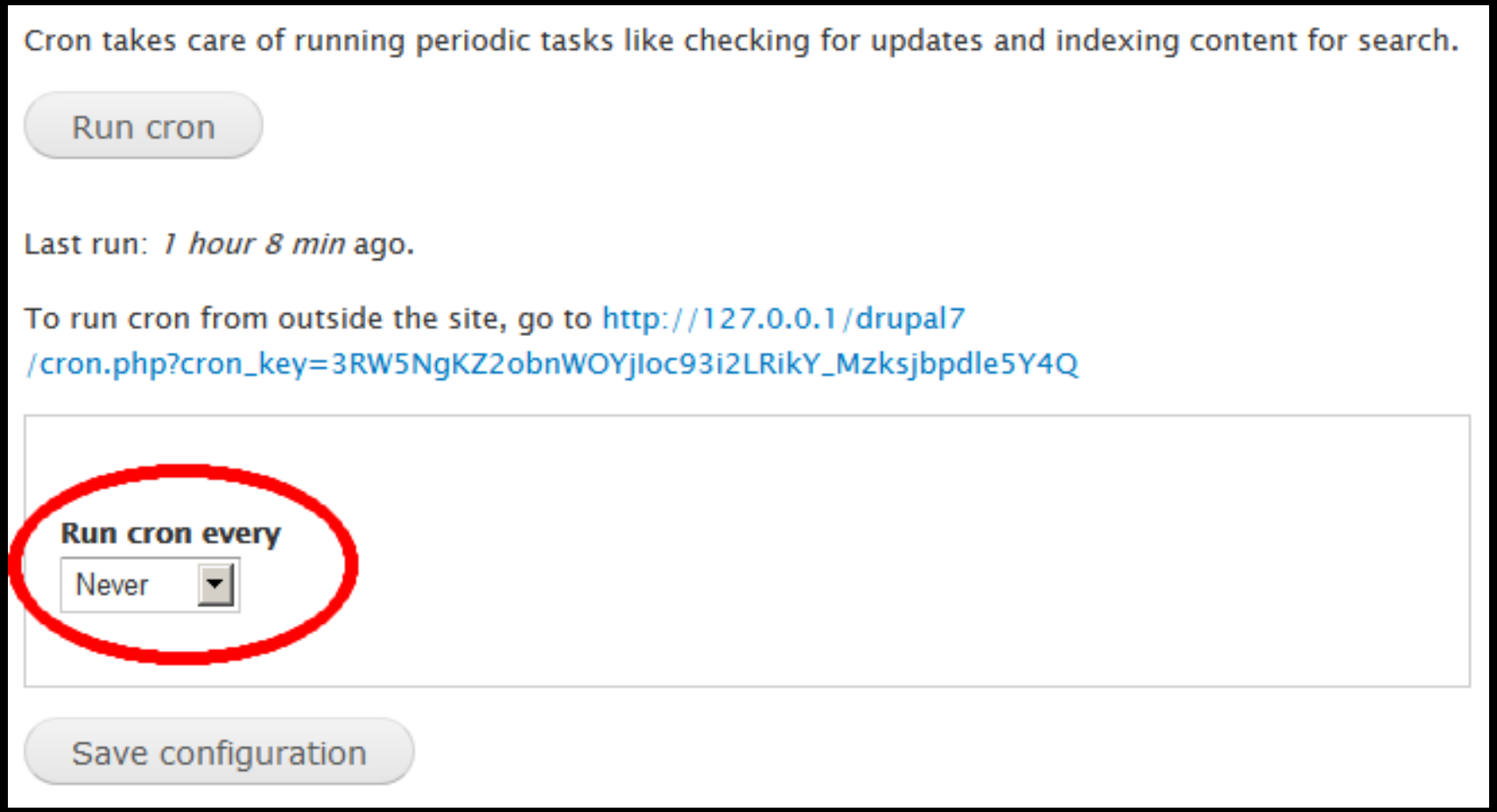


https://goo.gl/L4P3MA

Once you have your host running the drupal cron job you can disable the one that comes with core by default

https://goo.gl/L4P3MA

admin/config/system/cron



https://goo.gl/L4P3MA

Enable fast 404

Unless you're using the private file system

https://goo.gl/L4P3MA

Go to the sites/default directory
Open up to the setting.php file
Around line 500 uncomment this
drupal_fast_404();
by removing #

https://goo.gl/L4P3MA

Also use fast 404 to block urls commonly used by bots

Requests like `/postnuke/article.php` should be 404-ed ASAP. Do not waste server resources on serving a nice 404 to bots.

https://goo.gl/L4P3MA

While inside the settings.php file above where we enabled fast 404 are the settings that control it. add

```
^(postnuke|wp-content|mailman|phpBB)|
```

to the start of the $conf['404_fast_paths'] variable, right after the first `/`

This means that if any path starts with any of these strings we'll skip booting drupal and 404 right here taking very little server resources.

https://goo.gl/L4P3MA

In your settings.php Replace this:

```
$conf['404_fast_paths'] = '/\.(?:
txt|png|gif|jpe?
g|css|js|ico|swf|flv|cgi|bat|pl|dll|exe|asp)
$/i';
```

With this

```
$conf['404_fast_paths'] = '/^(postnuke|wp-
content|mailman|phpBB)|\.(?:txt|png|gif|jpe?
g|css|js|ico|swf|flv|cgi|bat|pl|dll|exe|asp)
$/i';
```

https://goo.gl/L4P3MA

You can of course add more paths to this as you discover the other paths used by bots.

Lets say you also want to fast 404 any path that starts with browserconfig.xml. This

```
(postnuke|wp-content|mailman|phpBB)
```

Now should be this

```
(postnuke|wp-content|mailman|phpBB|browserconfig\.xml)
```

Be careful when doing this; you can 404 a lot of paths.

https://goo.gl/L4P3MA

Core Patches

Hacking core for fun and profit
https://groups.drupal.org/node/210683

https://goo.gl/L4P3MA

Add static cache to module_load_include()

https://www.drupal.org/node/1443308#comment-10329229

Usually shaves 90ms off of a big site

If item is hidden in _menu_tree_check_access() skip it right away

https://www.drupal.org/node/1710656#comment-6304412

Usually shaves 50ms off of a big site

Avoid re-scanning module directory when a filename or a module is missing

https://www.drupal.org/node/1081266?page=1#comment-10434917

If you have one missing module then this can shave up to a second off of every page load

https://goo.gl/L4P3MA

Improve theme registry build performance

https://www.drupal.org/node/2339447#comment-10354975

Usually shaves 5-10 seconds off of a cache clear

Speed up drupal_parse_info_format()

https://www.drupal.org/node/2146643#comment-9592921

Reduces memory usage and shave a second off of cache clears

https://goo.gl/L4P3MA

inline file_uri_scheme() in file_stream_wrapper_uri_normalize()

https://www.drupal.org/node/1443342#comment-5613306

Shaves 150ms on a cache clear

Fix Notice: Trying to get property of non-object in image_style_deliver()

https://www.drupal.org/node/1762772#comment-7387274

Needed if the files directory is mounted on a NFS drive.

Database Connection

Use unix sockets if possible

https://goo.gl/L4P3MA

Use the APDQC Module

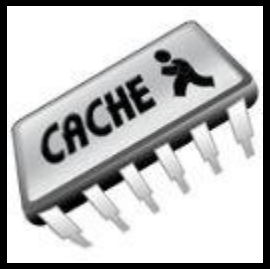
The status report has a section on setting up unix sockets or an IP instead of hostname.

	Drupal	7.36
⚠	APDQC - Database Connection	The connection to the database could be faster. Inside settings.php add this below the $databases array: `$databases['default']['default']['unix_socket'] = '/xampp/mysql/mysql.sock';`
✖	APDQC - MySQL Transaction Isolation Level	For best database performance, set the transaction isolation level to READ-COMMITTED. Current value: REPEATABLE-READ Inside settings.php add this below the $databases array `$databases['default']['default']['init_commands']['isolation'] = "SET SESSION tx_isolation='READ-COMMITTED'";` .
⚠	APDQC - MySQL innodb_lock_wait_timeout	wait_timeout should be a smaller value. Current value: 28800. Inside settings.php add this below the $databases array: `$databases['default']['default']['init_commands']['wait_timeout'] = "SET SESSION wait_timeout = 600";`

https://goo.gl/L4P3MA

Op Code Cache

Use the built in OPcache in php 5.5+



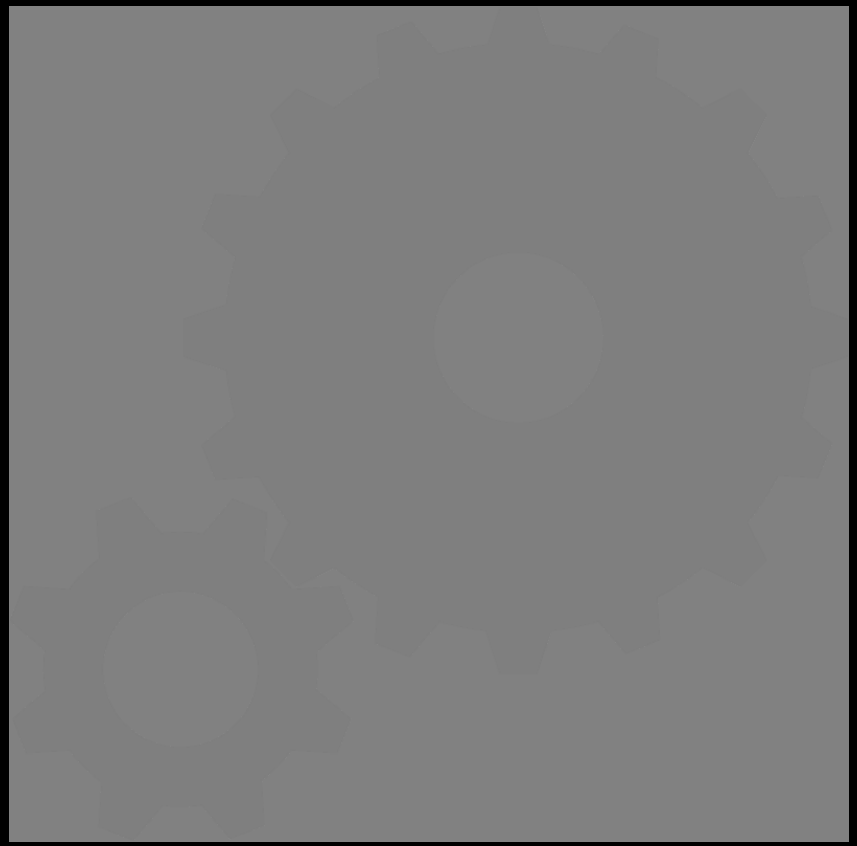
https://goo.gl/L4P3MA

OPcache settings

```
zend_extension=opcache.so
opcache.enable=1
opcache.memory_consumption=256
opcache.validate_timestamps=0
opcache.max_accelerated_files=100000
opcache.fast_shutdown=1
opcache.interned_strings_buffer=16
```

https://goo.gl/L4P3MA

php.ini



https://goo.gl/L4P3MA

For all

```
realpath_cache_size = 1M
realpath_cache_ttl = 3600
```

Only if memcache is being used

```
memcache.hash_strategy="consistent"
```

https://goo.gl/L4P3MA

PHP Notices/Warnings

Fix them as these slow down your site

https://goo.gl/L4P3MA

Put this in your settings.php file

```
<?php
if (!empty($_GET[‘errors’])) {
  // Show Errors in output
  ini_set('display_errors', '1');
  // Report all php errors.
  error_reporting(-1);
  // Display errors using dsm().
  $conf['error_level'] = 2;
}
?>
```

https://goo.gl/L4P3MA

Basic database tuning

https://goo.gl/L4P3MA

Use the APDQC module

Status report is your friend

https://goo.gl/L4P3MA

What it will check for

- That you're using the mysqlnd db driver
- using a unix socket if at all possible
- Verify tx_isolation is READ-COMMITTED
- max_allowed_packet is at least 32MB
- innodb_buffer_pool_size is big enough for your database to fit into memory
- Verify innodb_flush_log_at_trx_commit is 2
- Query cache is disabled
- innodb_lock_wait_timeout and wait_timeout are both sane values

https://goo.gl/L4P3MA

Note about the MySQL query cache

This should be disabled.

The built in MySQL query cache gets invalidated at the table level, any writes to that table will wipe out the query cache; it also has some nasty global locks on reads and writes.

https://goo.gl/L4P3MA

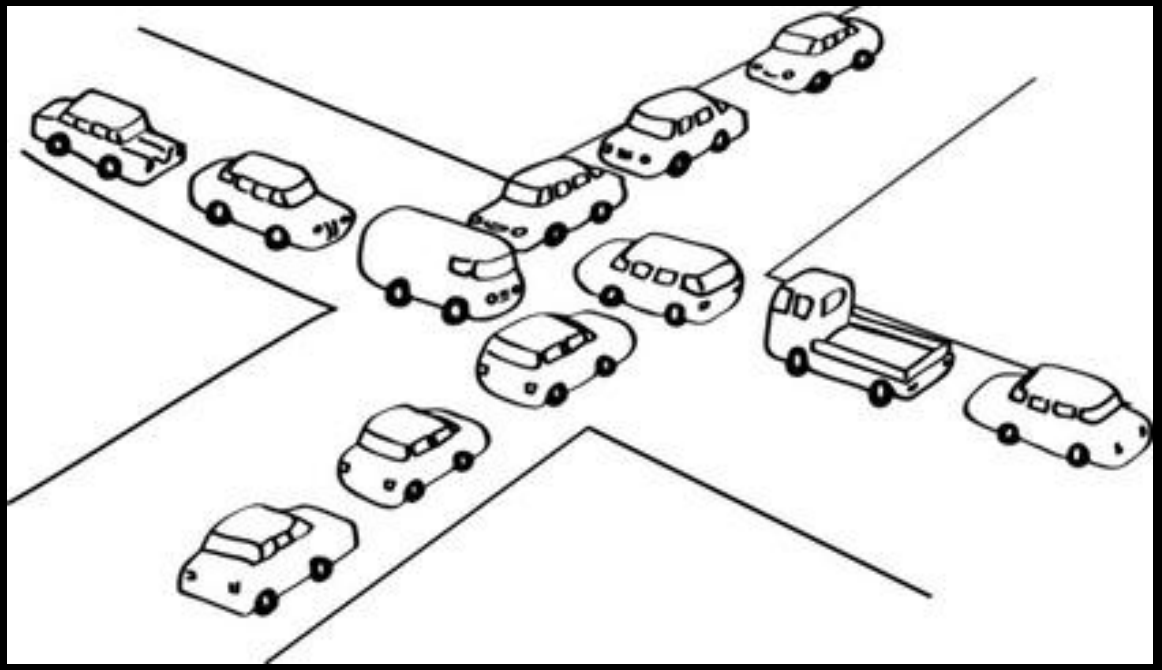
More info about MySQL & InnoDB

http://www.mysqlperformanceblog.com/2013/09/20/innodb-performance-optimization-basics-updated/

https://goo.gl/L4P3MA

Deadlock Detection

"deadlock found when trying to get lock try restarting transaction"



https://goo.gl/L4P3MA

`SHOW ENGINE innodb STATUS`

This command will output a bunch of information; the more interesting bit is under the “LATEST DETECTED DEADLOCK” section.

https://goo.gl/L4P3MA

Diagnosing Stuck Queries

What if SHOW PROCESSLIST shows a lot of simple queries that are taking too long

https://goo.gl/L4P3MA

Processes	ID	User	Host	Database	Command	Time	Status	SQL query ←T→
Kill	236	root	localhost:51083	d7	Sleep	3	---	---
Kill	237	root	localhost:51084	d7	Query	3	update	`INSERT INTO cache (cid, serialized, created, expire, data) VALUES ('node_types:en', '1', '1415747825`
Kill	238	root	localhost:51085	d7	Query	3	update	`INSERT INTO cache (cid, serialized, created, expire, data) VALUES ('node_types:en', '1', '1415747825`
Kill	239	root	localhost:51086	d7	Query	3	update	`INSERT INTO cache (cid, serialized, created, expire, data) VALUES ('node_types:en', '1', '1415747825`
Kill	240	root	localhost:51087	d7	Query	3	update	`INSERT INTO cache (cid, serialized, created, expire, data) VALUES ('node_types:en', '1', '1415747825`
Kill	241	root	localhost:51088	d7	Query	3	update	`INSERT INTO cache (cid, serialized, created, expire, data) VALUES ('node_types:en', '1', '1415747825`
Kill	243	root	localhost:51092	mysql	Query	0	init	`SHOW PROCESSLIST`

https://goo.gl/L4P3MA

95% of the time database transactions are to blame

https://goo.gl/L4P3MA

What to do?

https://www.drupal.org/project/apdqc Is the answer. Module if used correctly will fix the deadlock and metadata locking issues with core's db cache.

I want to monitor this

You can run this next query and you can then kill that mysql process id if you desire to. This only works in real time; every time you get a locked database, run the query and manually kill that process id.

https://goo.gl/L4P3MA

ID stuck thread that holds the lock

```
SELECT trx.trx_id AS trx_id, trx.trx_mysql_thread_id AS thread_id, trx.
trx_query AS query, trx.trx_tables_locked AS tables_locked, trx.
trx_rows_locked AS rows_locked, trx.trx_state AS trx_state, p.DB AS db, p.
STATE AS process_state, trx.trx_operation_state AS trx_op_state, p.COMMAND
AS command, p.TIME AS query_age, DATEDIFF( trx.trx_started, NOW( ) ) AS
trx_age, DATEDIFF( trx.trx_wait_started, NOW( ) ) AS wait_age, wr.
requested_lock_id AS requested_lock_id, wb.requested_lock_id AS
blocking_lock_id, trx.trx_isolation_level AS iso_level

FROM information_schema.innodb_trx AS trx

LEFT JOIN information_schema.processlist p ON trx.trx_mysql_thread_id = p.
ID

LEFT JOIN information_schema.innodb_lock_waits w ON trx.
trx_mysql_thread_id = p.ID

LEFT JOIN information_schema.innodb_lock_waits wb ON trx.trx_id = wb.
blocking_trx_id

LEFT JOIN information_schema.innodb_lock_waits wr ON trx.trx_id = wr.
requesting_trx_id

WHERE trx_tables_locked >0 OR trx_rows_locked >0

GROUP BY trx.trx_mysql_thread_id
```

https://goo.gl/L4P3MA

trx_id	thread_id	query	tables_locked	rows_locked	trx_state	db	process_state	trx_op_state	command	query_age	trx_age	wait_age	requested_lock_id	blocking_lock_id	iso_level
3805696	236	*NULL*	0	1	RUNNING	d7		*NULL*	Sleep	7	0	*NULL*	*NULL*	3805701:60646:24:3	REPEATABLE READ
3805697	237	INSERT INTO cache (cid, serialized, created, expir...	1	1	LOCK WAIT	d7	update	inserting	Query	7	0	0	3805697:60646:24:3	3805701:60646:24:3	READ UNCOMMITTED
3805698	238	INSERT INTO cache (cid, serialized, created, expir...	1	1	LOCK WAIT	d7	update	inserting	Query	7	0	0	3805698:60646:24:3	3805701:60646:24:3	READ UNCOMMITTED
3805699	239	INSERT INTO cache (cid, serialized, created, expir...	1	1	LOCK WAIT	d7	update	inserting	Query	7	0	0	3805699:60646:24:3	3805701:60646:24:3	READ UNCOMMITTED
3805700	240	INSERT INTO cache (cid, serialized, created, expir...	1	1	LOCK WAIT	d7	update	inserting	Query	7	0	0	3805700:60646:24:3	3805701:60646:24:3	READ UNCOMMITTED
3805701	241	INSERT INTO cache (cid,	1	1	LOCK WAIT	d7	update	inserting	Query	7	0	0	3805701:60646:24:3	*NULL*	READ UNCOMMITTED

https://goo.gl/L4P3MA

trx_id	thread_id	query	tables_locked	rows_locked	trx_state	db	process_state	trx_op_state	command	query_age	trx_age	wait_age	requested_lock_id	blocking_lock_id	iso_level
3805696	236	*NULL*	0	1	RUNNING	d7		*NULL*	Sleep	7	0	*NULL*	*NULL*	3805701:60646:24:3	REPEATABLE READ
3805697	237	INSERT INTO cache (cid, serialized, created, expir...	1	1	LOCK WAIT	d7	update	inserting	Query	7	0	0	3805697:60646:24:3	3805701:60646:24:3	READ UNCOMMITTED
3805698	238	INSERT INTO cache (cid, serialized, created, expir...	1	1	LOCK WAIT	d7	update	inserting	Query	7	0	0	3805698:60646:24:3	3805701:60646:24:3	READ UNCOMMITTED
3805699	239	INSERT INTO cache (cid, serialized, created, expir...	1	1	LOCK WAIT	d7	update	inserting	Query	7	0	0	3805699:60646:24:3	3805701:60646:24:3	READ UNCOMMITTED
3805700	240	INSERT INTO cache (cid, serialized, created, expir...	1	1	LOCK WAIT	d7	update	inserting	Query	7	0	0	3805700:60646:24:3	3805701:60646:24:3	READ UNCOMMITTED
3805701	241	INSERT INTO cache (cid,	1	1	LOCK WAIT	d7	update	inserting	Query	7	0	0	3805701:60646:24:3	*NULL*	READ UNCOMMITTED

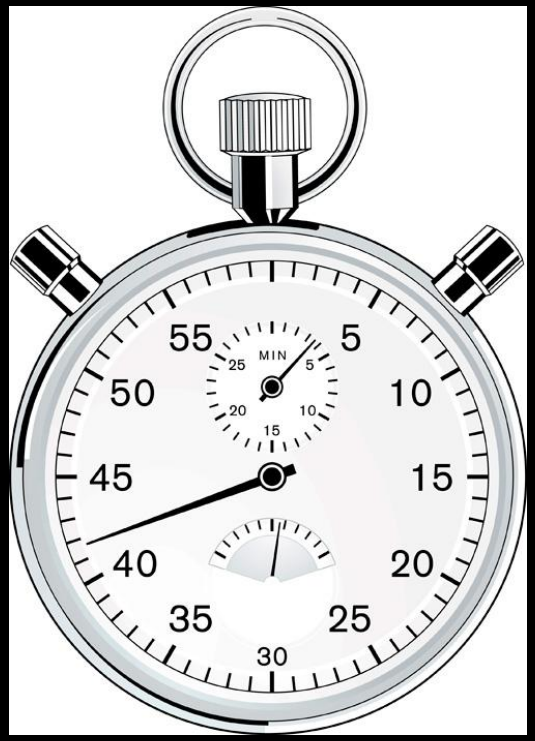
https://goo.gl/L4P3MA

Slow Query Analysis

Percona's toolkit has a nice perl script called pt-query-digest. Recommend using that.

https://goo.gl/L4P3MA

Basic query performance measurements with the devel module



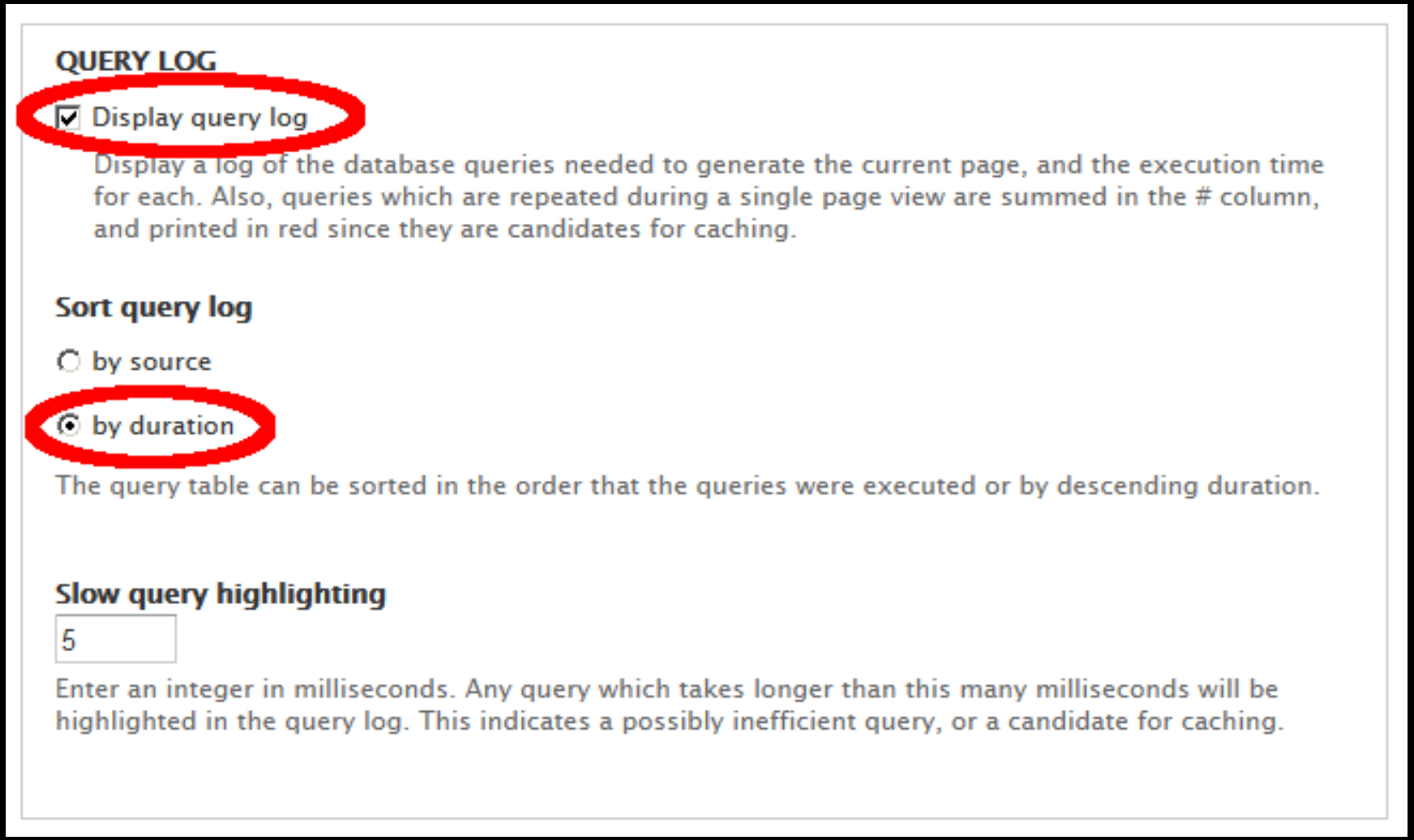
https://goo.gl/L4P3MA

https://drupal.org/project/devel

This module can tell you what queries are slow and how many ms is spent in the database and in php.

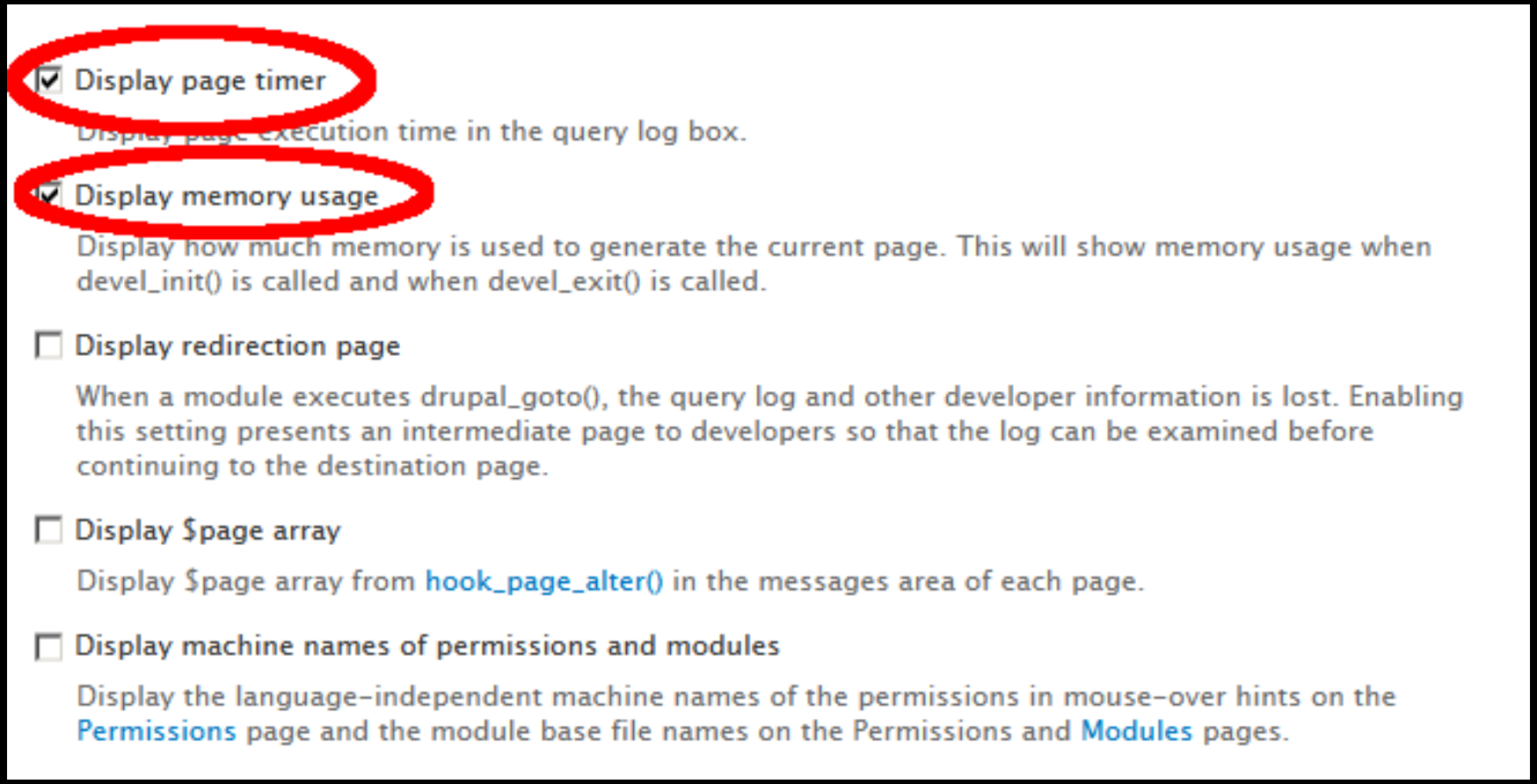
https://goo.gl/L4P3MA

admin/config/development/devel



https://goo.gl/L4P3MA

admin/config/development/devel



https://goo.gl/L4P3MA

Executed 34 queries in 6.86 ms. Queries exceeding 5 ms are **highlighted**. Page execution time was 123.86 ms. Memory used at: devel_boot()=**3.08** MB, devel_shutdown()=**15.14** MB, PHP peak=**15.25** MB.

ms	#	where	ops	query	target
0.66	3	menu_get_item	P A E	`SELECT * FROM menu_router WHERE path IN (:ancestors_0, :ancestors_1, :ancestors_2, :ancestors_3, :ancestors_4, :ancestors_5, :ancestors_6, :ancestors_7, :ancestors_8, :ancestors_9, :ancestors_10, :ancestors_11, :ancestors_12, :ancestors_13, :ancestors_14, :ancestors_15, :ancestors_16, :ancestors_17, :ancestors_18, :ancestors_19, :ancestors_20, :ancestors_21, :ancestors_22, :ancestors_23) ORDER BY fit DESC LIMIT 0, 1`	default
0.6	3	menu_get_item	P A E	`SELECT * FROM menu_router WHERE path IN ('admin/structure/block/manage/system/navigation', 'admin/structure/block/manage/system/%', 'admin/structure/block/manage/%/navigation', 'admin/structure/block/manage/%/%', 'admin/structure/block/%/system/navigation', 'admin/structure/block/%/system/%', 'admin/%/block/manage/%/%', 'admin/structure/block/manage/system', 'admin/structure/block/manage/%', 'admin/structure/block/%/system', 'admin/structure/block/%/%', 'admin/structure/%/manage/system', 'admin/structure/%/%/%', 'admin/%/block/%/system', 'admin/structure/block/manage', 'admin/structure/block/%', 'admin/structure/%/manage', 'admin/%/block/manage', 'admin/structure/block', 'admin/structure/%', 'admin/%/block', 'admin/structure', 'admin/%', 'admin') ORDER BY fit DESC LIMIT 0, 1`	default
0.57	3	menu_get_item	P A E		default

id	select_type	table	type	possible_keys	key	key_len	ref	rows	Extra
1	SIMPLE	menu_router	index	PRIMARY	fit	4		15	Using where

https://goo.gl/L4P3MA

Executed 34 queries in 6.86 ms. Queries exceeding 5 ms are highlighted. Page execution time was 123.86 ms. Memory used

ms	#	where	ops	query
0.66	3	menu_get_item	P A E	SELECT * FROM menu_router WHERE path IN :ancestors_6, :ancestors_7, :ancestors_8 :ancestors_15, :ancestors_16, :ancestors ORDER BY fit DESC LIMIT 0, 1
0.6	3	menu_get_item	P A E	SELECT * FROM menu_router WHERE path IN 'admin/structure/block/manage/%/navigati 'admin/structure/block/%/system/%', 'adm 'admin/structure/block/%/system', 'admin /block/%/system', 'admin/structure/block 'admin/structure/block', 'admin/structur
0.57	3	menu_get_item	P A E	

id	select_type	table
1	SIMPLE	menu_router

https://goo.gl/L4P3MA

evel_boot()=**3.08** MB, devel_shutdown()=**15.14** MB, PHP peak=**15.25** MB.

target

```
cestors_0, :ancestors_1, :ancestors_2, :ancestors_3, :ancestors_4, :ancestors_5,
ncestors_9, :ancestors_10, :ancestors_11, :ancestors_12, :ancestors_13, :ancestors_14,
 :ancestors_18, :ancestors_19, :ancestors_20, :ancestors_21, :ancestors_22, :ancestors_23)
```

default

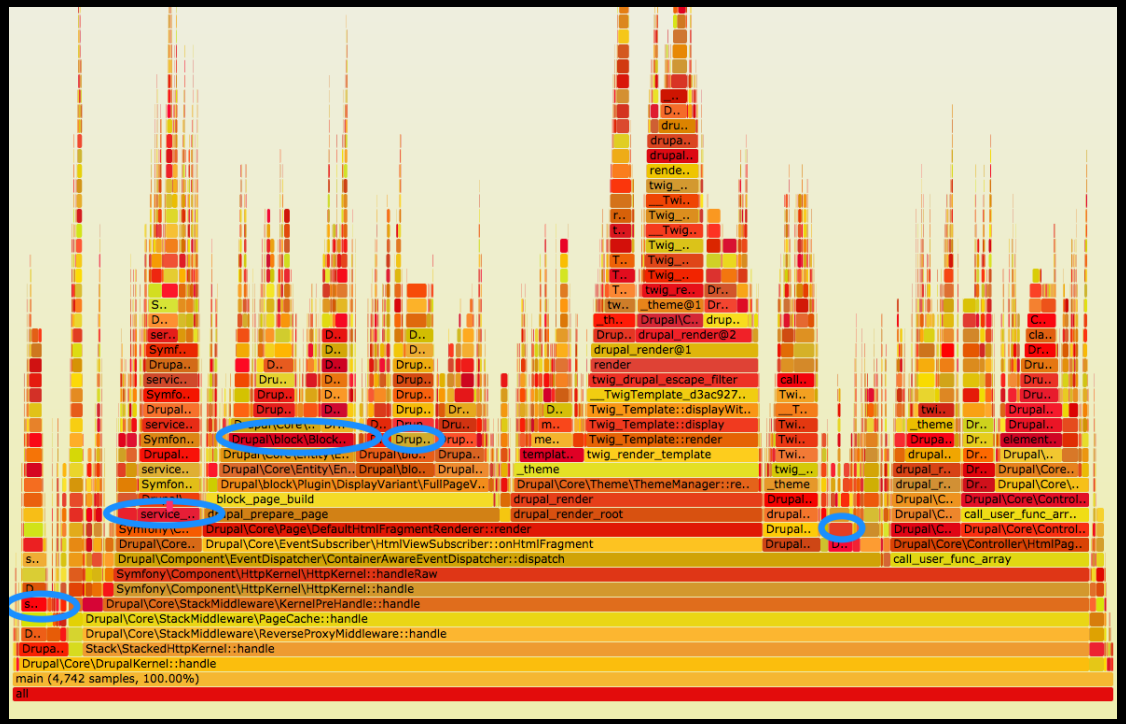
```
min/structure/block/manage/system/navigation', 'admin/structure/block/manage/system/%',
 'admin/structure/block/manage/%/%', 'admin/structure/block/%/system/navigation',
/block/manage/%/%', 'admin/structure/block/manage/system', 'admin/structure/block/manage/%',
ucture/block/%/%', 'admin/structure/%/manage/system', 'admin/structure/%/%/%', 'admin/%
age', 'admin/structure/block/%', 'admin/structure/%/manage', 'admin/%/block/manage',
, 'admin/%/block', 'admin/structure', 'admin/%', 'admin') ORDER BY fit DESC LIMIT 0, 1
```

default

type	possible_keys	key	key_len	ref	rows	Extra
index	PRIMARY	fit	4		15	Using where

https://goo.gl/L4P3MA

Xdebug/Xhprof



https://goo.gl/L4P3MA

Generating a cachegrind file

This allows you to pinpoint slow points. Once you know the bottlenecks, you can try to figure out what options are available to fix them.

xhprof is a good alternative to xdebug.

Another alternative to this is to use New Relic. It provides a lot of good information and it comes with a 14 day free trial.

https://goo.gl/L4P3MA

Questions?

Twitter: @mcarper

D7 performance wiki: https://groups.drupal.org/node/210683

Link to this presentation

https://goo.gl/L4P3MA