



**DRUPALCON
LONDON**

Damn Quick Drupal: How to make Drupal perform and scale like a rock star!



Presented by Michael Cooper (soyarma)

Why Me? Why This Topic?

- ❖ Writing web apps since 2001
- ❖ Built CMS that powered 1500 automotive sites
- ❖ Ran 'auto malls' with 500,000 vehicles that had to return results from DB in under 500ms
- ❖ Built over 550 websites
- ❖ Built over 150 websites in Drupal
- ❖ Built websites (in Drupal) that had to handle things like 5 POST requests/second.
- ❖ Launched Drupal sites and had them die within 90 seconds
- ❖ Learned how not to have that happen again

Molasses Drupal

Why are many Drupal websites slow and fail?



Molasses Drupal

Why are many Drupal websites slow and fail?

- ❖ Full page renders

Molasses Drupal

Why are many Drupal websites slow and fail?

- ❖ Full page renders
- ❖ Dynamic content served to anonymous users

Molasses Drupal

Why are many Drupal websites slow and fail?

- ❖ Full page renders
- ❖ Dynamic content served to anonymous users
- ❖ Excessive/slow/non-optimized database queries

Molasses Drupal

Why are many Drupal websites slow and fail?

- ❖ Full page renders
- ❖ Dynamic content served to anonymous users
- ❖ Excessive/slow/non-optimized database queries
- ❖ Naughty modules

CACHE!



Cache your cache in a cache
that is cached ... which better
be cached too



Your computer, the Internet at large,
and everything in between, is—in
many ways—just a large series of
caches.

Cache your cache in a cache
that is cached ... which better
be cached too



WHY?

**Because
stuff is
slow...**

Drupal wants cache too



im n yur walet

stealin da frootz of yur laborz

How does Drupal cache

- ❖ Drupal core and many modules instantiate caches in the DB
- ❖ Successive versions of Drupal have cached more and more and done less on the fly.
- ❖ Variables, modules, pages, blocks, oh my

How Drupal uses the DB to cache

- ❖ Cache tables store data for various modules and core
- ❖ Nearly all cache tables share the same schema and use `cache_set()` and `cache_get()` to store data and retrieve it. Look up `cache_set()` on api.drupal.org for more information.
- ❖ Caching is pluggable.
 - ❖ You can create your own `cache.inc` file (in Drupal 6) and create your own `cache_get` and `cache_set` functions.
 - ❖ Configure which cache 'bins' store in which cache technologies
- ❖ Examples of modules that plug into Drupal's cache system and change where cache data is stored are: `cache`, `cacheroouter`, `memcache`, and `apc`.

The Testbed

- ❖ CPU: Quad Opteron Rackspace Cloud Server
- ❖ RAM: 256MB
- ❖ Web: Apache 2.2
- ❖ PHP: 5.2.17
- ❖ MySQL 5.1
- ❖ Reverse Proxy: None



Your Tools

- ❖ Devel and Performance Monitoring
 - ❖ Page Load times, memory consumption and query logs...
 - ❖ XHPProf is your friend. Use it.
- ❖ New Relic
- ❖ Apache benchmark, Bombard, Siege, Jmeter, Sosta, Blitz.io
- ❖ Different tests for different things



Some terms

- ❖ **Cold cache:** This is the state of your Drupal site and/or server before any data has been placed into cache.
- ❖ **Warm cache:** This is the state of your Drupal site and/or server after data has been generated and stored in cache.
- ❖ **Key-Value store:** Servers/daemons such as Cassandra, memcached, APC, and Redis are all examples of key-value stores. They are simply what they sound like, simple systems for storing data and querying it by key. They are 'No SQL' in that they are schema-less.
- ❖ **Op-Code Caches/Compilers:** Programs/extensions such as APC, eAccelerator, ionCube, xCache, PhpExpress, Zend Optimizer+, WinCache all optimize and compile your PHP ahead of time so it doesn't have to be done on the fly.

Queries, queries, queries...

| | ms | # | where | query |
|--------------|-------|---|----------------------------|---|
| 43.88 | 0.54 | 1 | drupal_get_filename | SELECT filename FROM dev_system WHERE name = 'user' AND type = 'module' |
| | 0.59 | 1 | cache_get | SELECT data, created, headers, expire, serialized FROM dev_cache_path WHERE cid = 'path_alias_whitelist' |
| | 0.33 | 1 | path_alias_cache_lookup_pi | SELECT src FROM dev_url_alias WHERE dst = 'node' AND (language = 'en' OR language = '') ORDER BY language DESC |
| | 1.56 | 1 | module_list | SELECT name, filename, throttle FROM dev_system WHERE type = 'module' AND status = 1 ORDER BY weight ASC, filename ASC |
| | 0.4 | 1 | cache_get | SELECT data, created, headers, expire, serialized FROM dev_cache WHERE cid = 'variables' |
| | 0.38 | 1 | cache_get | SELECT data, created, headers, expire, serialized FROM dev_cache WHERE cid = 'strongarm' |
| | 0.33 | 1 | cache_get | SELECT data, created, headers, expire, serialized FROM dev_cache WHERE cid = 'context' |
| | 0.44 | 1 | cache_get | SELECT data, created, headers, expire, serialized FROM dev_cache WHERE cid = 'plugins:context:plugins' |
| | 10.79 | 1 | path_redirect_load_by_sour | SELECT rid FROM dev_path_redirect WHERE source = '<front>' AND language IN ('en', '') ORDER BY language DESC, source DESC, rid DESC |
| | 0.44 | 1 | menu_get_item | SELECT * FROM dev_menu_router WHERE path IN ('node') ORDER BY fit DESC LIMIT 0, 1 |
| | 20.74 | 1 | node_access_view_all_node | SELECT COUNT(*) FROM dev_node_access WHERE nid = 0 AND ((gid = 0 AND realm = 'all')) AND grant_view >= 1 |
| | 0.33 | 1 | pager_query | SELECT COUNT(*) FROM dev_node n WHERE n.promote = 1 AND n.status = 1 |
| | 14.03 | 1 | pager_query | SELECT n.nid, n.sticky, n.created FROM dev_node n WHERE n.promote = 1 AND n.status = 1 ORDER BY n.sticky DESC, n.created DESC LIMIT 0, 10 |
| | 0.82 | 1 | cache_get | SELECT data, created, headers, expire, serialized FROM dev_cache WHERE cid = 'schema' |

Performance

Memory usage:
Memory used at: devel_init()=1.46 MB,
devel_shutdown()=17.55 MB.

Executed 155 queries in 354.16 milliseconds.
Queries taking longer than 5 ms and queries
executed more than once, are **highlighted**.
Page execution time was 1548.21 ms.

Hide querylog

Warmed, but not toasty

Various and sundry caches are 'warmed up'. Page execution time and memory are nice and trim.

| | ms | # | where | query |
|---|------|---|----------------------------|--|
| Create content | | | | |
| Switch theme | | | | |
| My Account | | | | |
| Administer | | | | |
| Devel | | | | |
| Performance | | | | |
| Memory usage: Memory used at: devel_init()=1.46 MB, devel_shutdown()=17.55 MB. | | | | |
| Executed 155 queries in 48.88 milliseconds. Queries taking longer than 5 ms and queries executed more than once, are highlighted . Page execution time was 249.65 ms. | | | | |
| Hide querylog | | | | |
| Switch user | | | | |
| Execute PHP | | | | |
| Other tools | | | | |
| | 0.56 | 1 | module_list | SELECT name, filename, throttle FROM dev_system WHERE type = 'module' AND status = 1 AND bootstrap = 1 ORDER BY weight ASC, filename ASC |
| | 0.42 | 1 | drupal_get_filename | SELECT filename FROM dev_system WHERE name = 'user' AND type = 'module' |
| | 0.29 | 1 | cache_get | SELECT data, created, headers, expire, serialized FROM dev_cache_path WHERE cid = 'path_alias_whitelist' |
| | 0.28 | 1 | path_alias_cache_lookup_pi | SELECT src FROM dev_url_alias WHERE dst = 'node' AND (language = 'en' OR language = '') ORDER BY language DESC |
| | 1.52 | 1 | module_list | SELECT name, filename, throttle FROM dev_system WHERE type = 'module' AND status = 1 ORDER BY weight ASC, filename ASC |
| | 0.39 | 1 | cache_get | SELECT data, created, headers, expire, serialized FROM dev_cache WHERE cid = 'variables' |
| | 0.21 | 1 | cache_get | SELECT data, created, headers, expire, serialized FROM dev_cache WHERE cid = 'strongarm' |
| | 0.24 | 1 | cache_get | SELECT data, created, headers, expire, serialized FROM dev_cache WHERE cid = 'context' |
| | 0.24 | 1 | cache_get | SELECT data, created, headers, expire, serialized FROM dev_cache WHERE cid = 'plugins:context:plugins' |
| | 0.56 | 1 | path_redirect_load_by_sour | SELECT rid FROM dev_path_redirect WHERE source = '<front>' AND language IN ('en', '') ORDER BY language DESC, source rid DESC |
| | 0.32 | 1 | menu_get_item | SELECT * FROM dev_menu_router WHERE path IN ('node') ORDER BY fit DESC LIMIT 0, 1 |
| | 0.25 | 1 | node_access_view_all_node | SELECT COUNT(*) FROM dev_node_access WHERE nid = 0 AND ((gid = 0 AND realm = 'all')) AND grant_view >= 1 |
| | 0.33 | 1 | pager_query | SELECT COUNT(*) FROM dev_node n WHERE n.promote = 1 AND n.status = 1 |
| | 1.85 | 1 | pager_query | SELECT n.nid, n.sticky, n.created FROM dev_node n WHERE n.promote = 1 AND n.status = 1 ORDER BY n.sticky DESC, n.created LIMIT 0, 10 |
| | 0.52 | 1 | cache_get | SELECT data, created, headers, expire, serialized FROM dev_cache WHERE cid = 'schema' |

Enter Key-Value stores

| | ms | # | where | query |
|--|------|---|----------------------------|--|
| x Create content | | | | |
| Switch theme | 0.5 | 1 | module_list | SELECT name, filename, throttle FROM dev_system WHERE type = 'module' AND status = 1 AND bootstrap = 1 ORDER BY weight filename ASC |
| My Account | | | | |
| Administer | 0.48 | 1 | drupal_get_filename | SELECT filename FROM dev_system WHERE name = 'user' AND type = 'module' |
| Devel | 0.3 | 1 | path_alias_cache_lookup_p | SELECT src FROM dev_url_alias WHERE dst = 'node' AND (language = 'en' OR language = '') ORDER BY language DESC |
| Performance | 1.36 | 1 | module_list | SELECT name, filename, throttle FROM dev_system WHERE type = 'module' AND status = 1 ORDER BY weight ASC, filename AS |
| Memory usage: Memory used at: devel_init()=1.51 MB, devel_shutdown()=16.98 MB. | 0.47 | 1 | path_redirect_load_by_sour | SELECT rid FROM dev_path_redirect WHERE source = '<front>' AND language IN ('en', '') ORDER BY language DESC, source rid DESC |
| Executed 107 queries in 36.36 milliseconds. Queries taking longer than 5 ms and queries executed more than once, are highlighted . Page execution time was 212.5 ms. | 0.3 | 1 | menu_get_item | SELECT * FROM dev_menu_router WHERE path IN ('node') ORDER BY fit DESC LIMIT 0, 1 |
| Hide querylog | 0.23 | 1 | node_access_view_all_node | SELECT COUNT(*) FROM dev_node_access WHERE nid = 0 AND ((gid = 0 AND realm = 'all')) AND grant_view >= 1 |
| Switch user | 0.48 | 1 | pager_query | SELECT COUNT(*) FROM dev_node n WHERE n.promote = 1 AND n.status = 1 |
| Execute PHP | 2.04 | 1 | pager_query | SELECT n.nid, n.sticky, n.created FROM dev_node n WHERE n.promote = 1 AND n.status = 1 ORDER BY n.sticky DESC, n.crea LIMIT 0, 10 |
| Other tools | 0.35 | 1 | node_load | SELECT n.nid, n.type, n.language, n.uid, n.status, n.created, n.changed, n.comment, n.promote, n.moderate, n.sticky, n.translate, r.vid, r.uid AS revision_uid, r.title, r.body, r.teaser, r.log, r.timestamp AS revision_timestamp, r.for u.name, u.picture, u.data FROM dev_node n INNER JOIN dev_users u ON u.uid = n.uid INNER JOIN dev_node_revisions r ON n.vid WHERE n.nid = 494 |
| | 0.27 | 1 | _node_types_build | SELECT nt.type, nt.* FROM dev_node_type nt ORDER BY nt.type ASC |
| | 0.19 | 1 | comment_nodeapi | SELECT last_comment_timestamp, last_comment_name, comment_count FROM dev_node_comment_statistics WHERE nid = 494 |
| | 3.39 | 1 | path_alias_cache_lookup_p | SELECT src, dst FROM dev_url_alias WHERE src IN('node/494', 'node/327', 'node/178', 'node/158', 'node/29', 'node/83', 'node /469', 'node/205', 'node/331', 'node/120', 'node/36', 'node/390', 'node', 'node/add/fserver-distro', 'node/add/page', 'node /add/fserver-project', 'node/add/fserver-release', 'node/add/story', 'node/add') AND (language = 'en' OR language = '') language ASC |
| | 0.28 | 1 | taxonomy_node_get_terms | SELECT t.* FROM dev_term_node r INNER JOIN dev_term_data t ON r.tid = t.tid INNER JOIN dev_vocabulary v ON t.vid = v. WHERE r.vid = 494 ORDER BY v.weight, t.weight, t.name |

Is MySQL Caching?



× Create content

Switch theme

My Account

Administer > Reports

Devel

Performance

Memory usage:
Memory used at: devel_init()=1.39 MB,
devel_shutdown()=15.67 MB.

Executed 34 queries in 17.62 milliseconds.
Queries taking longer than 5 ms and queries
executed more than once, are **highlighted**.
Page execution time was 163.4 ms.

Show querylog

Switch user

Execute PHP

Other tools

| Variable | Value | Description |
|-------------------|-------|----------------------------------|
| Com_update | 0 | The number of UPDATE-statements. |
| Com_delete | 0 | The number of DELETE-statements. |
| Com_lock_tables | 0 | The number of table locks. |
| Com_unlock_tables | 0 | The number of table unlocks. |

Query performance

| Variable | Value | Description |
|-----------------------|-------|---|
| Select_full_join | 0 | The number of joins without an index; should be zero. |
| Select_range_check | 0 | The number of joins without keys that check for key usage after each row; should be zero. |
| Sort_scan | 0 | The number of sorts done without using an index; should be zero. |
| Table_locks_immediate | 209 | The number of times a lock could be acquired immediately. |
| Table_locks_waited | 0 | The number of times the server had to wait for a lock. |

Query cache information

The MySQL query cache can improve performance of your site by storing the result of queries. Then, if an identical query is received later, the MySQL result from the query cache rather than parsing and executing the statement again.

| Variable | Value | Description |
|-------------------------|-------|--|
| Qcache_queries_in_cache | 0 | The number of queries in the query cache. |
| Qcache_hits | 0 | The number of times MySQL found previous results in the cache. |
| Qcache_inserts | 0 | The number of times MySQL added a query to the cache (misses). |
| Qcache_lowmem_prunes | 0 | The number of times MySQL had to remove queries from the cache because it ran out of memory. Ideally : |

Ensure sane MySQL settings

Use **SHOW STATUS**; to see the MySQL server values, both current states and configuration settings.

- ❖ `key_buffer_size=12M` (key cache)
- ❖ `query_cache_size=24M`
- ❖ `query_cache_limit=2M`
- ❖ `table_cache=96`
- ❖ `sort_buffer_size=12M`
- ❖ `mysam_sort_buffer_size=12M`
- ❖ `tmp_table_size=12M`



No More Slow Queries

| ms | # | where | query |
|------|---|----------------------------|--|
| 0.15 | 1 | module_list | SELECT name, filename, throttle FROM dev_system WHERE type = 'module' ORDER BY filename ASC |
| 0.15 | 1 | drupal_get_filename | SELECT filename FROM dev_system WHERE name = 'user' AND type = 'module' |
| 0.15 | 1 | path_alias_cache_lookup_p | SELECT src FROM dev_url_alias WHERE dst = 'node' AND (language = 'en' OR language IS NULL) |
| 0.27 | 1 | module_list | SELECT name, filename, throttle FROM dev_system WHERE type = 'module' ORDER BY filename ASC |
| 0.21 | 1 | path_redirect_load_by_sour | SELECT rid FROM dev_path_redirect WHERE source = '<front>' AND language = 'en' ORDER BY rid DESC |
| 0.16 | 1 | menu_get_item | SELECT * FROM dev_menu_router WHERE path IN ('node') ORDER BY fit DESC |
| 0.12 | 1 | node_access_view_all_node | SELECT COUNT(*) FROM dev_node_access WHERE nid = 0 AND ((gid = 0 AND role_id IN (1, 2)) OR gid = 1) |
| 0.1 | 1 | pager_query | SELECT COUNT(*) FROM dev_node n WHERE n.promote = 1 AND n.status = 1 |
| 0.1 | 1 | pager_query | SELECT n.nid, n.sticky, n.created FROM dev_node n WHERE n.promote = 1 AND n.status = 1 ORDER BY n.created DESC LIMIT 0, 10 |
| 0.23 | 1 | node_load | SELECT n.nid, n.type, n.language, n.uid, n.status, n.created, n.changed, n.translate, r.vid, r.uid AS revision_uid, r.title, r.body, r.teaser, u.name, u.picture, u.data FROM dev_node n INNER JOIN dev_users u ON u.nid = n.uid INNER JOIN dev_revisions r ON r.nid = n.nid AND r.vid = n.vid WHERE n.nid = 494 |
| 0.13 | 1 | _node_types_build | SELECT nt.type, nt.* FROM dev_node_type nt ORDER BY nt.type ASC |
| 0.14 | 1 | comment_nodeapi | SELECT last_comment_timestamp, last_comment_name, comment_count FROM dev_node_comment_statistics WHERE nid = 494 |

Performance

Memory usage:
Memory used at: devel_init()=1.39 MB,
devel_shutdown()=16.78 MB.

Executed 97 queries in 17.19 milliseconds.
Queries taking longer than 5 ms and queries
executed more than once, are **highlighted**.
Page execution time was 200.26 ms.

Hide querylog

More Speed! Give us More!

APC is your friend

- ❖ Every time a file is read by PHP it is compiled (checked for syntax errors, optimized, compiled into byte-code).
- ❖ APC (advanced PHP cache) will do this once and then cache the results.
- ❖ APC will check the file every time it is accessed to determine if it is still the same. If that happens rarely, set `apc.stat` to 0 and you will save that check.
- ❖ Drupal modules such as `cacheroouter` will integrate APC with Drupal's standard cache clearing functions.

APC Stats

[Refresh Data](#)
[View Host Stats](#)
[System Cache Entries](#)
[User Cache Entries](#)
[Version Check](#)
[Clear opcode Cache](#)

General Cache Information

| | |
|---------------------|---|
| APC Version | 3.1.6 |
| PHP Version | 5.2.16 |
| APC Host | dev.nosquaresoftware.com (Drupal-Linux) (184.106.225.163) |
| Server Software | Apache/2.2.3 (CentOS) |
| Shared Memory | 1 Segment(s) with 96.0 MBytes (mmap memory, pthread mutex locking) |
| Start Time | 2011/08/07 04:02:07 |
| Uptime | 4 days, 13 hours and 2 minutes |
| File Upload Support | 1 |

File Cache Information

| | |
|-----------------------------|----------------------------|
| Cached Files | 292 (34.1 MBytes) |
| Hits | 68477 |
| Misses | 292 |
| Request Rate (hits, misses) | 0.18 cache requests/second |
| Hit Rate | 0.17 cache requests/second |
| Miss Rate | 0.00 cache requests/second |
| Insert Rate | 0.00 cache requests/second |
| Cache full count | 0 |

User Cache Information

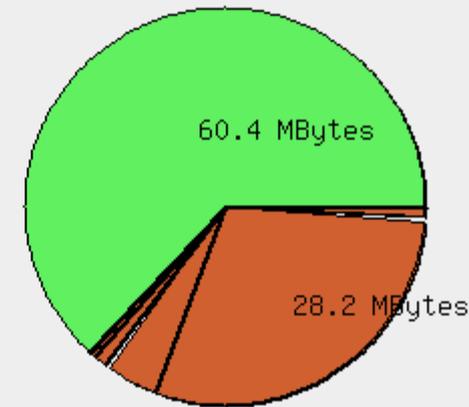
| | |
|-----------------------------|----------------------------|
| Cached Variables | 117 (1.1 MBytes) |
| Cached Variables | 117 (1.1 MBytes) |
| Hits | 6994 |
| Misses | 384 |
| Request Rate (hits, misses) | 0.02 cache requests/second |
| Hit Rate | 0.02 cache requests/second |
| Miss Rate | 0.00 cache requests/second |
| Insert Rate | 0.00 cache requests/second |
| Cache full count | 0 |

Runtime Settings

| | |
|----------------------|---|
| apc.cache_by_default | 1 |
| apc.canonicalize | 1 |
| apc.coredump_unmap | 0 |
| apc.enable_cli | 0 |
| apc.enabled | 1 |
| apc.file_md5 | 0 |

Host Status Diagrams

Memory Usage
(multiple slices indicate fragments)

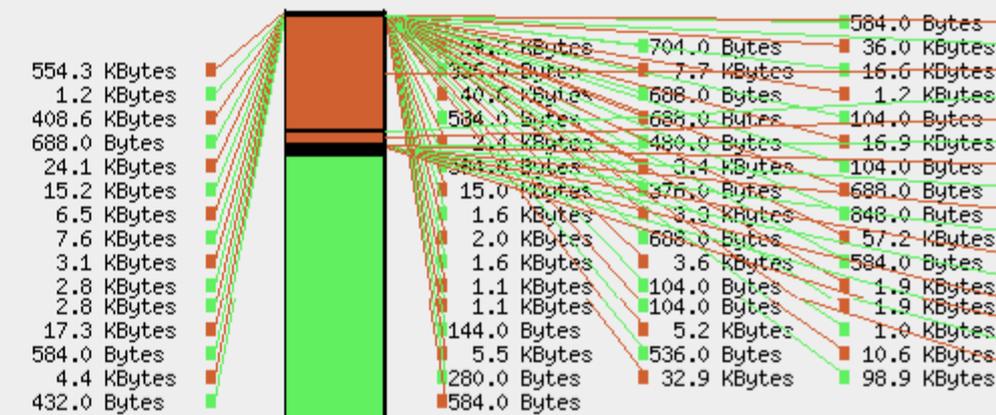


Hits & Misses

99.6%



Detailed Memory Usage and Fragmentation



Fragmentation: 0.46% (282.7 KBytes out of 60.6 MBytes in 52 fragments)

The Result:

✕ Create content

Switch theme

My Account

Administer

Devel

Performance

Memory usage:
Memory used at: devel_init()=**1.21** MB,
devel_shutdown()=**15.89** MB.

Executed 97 queries in 14.81 milliseconds.
Queries taking longer than 5 ms and queries
executed more than once, are **highlighted**.
Page execution time was 180.81 ms.

Show querylog

Switch user

Execute PHP

Other tools



DrupalCon London Benc

Home > Elit Refero Wisi Commoveo Paratus Amet

Consequat Meus Iustum Saluto Tation

node (page) - Paulatim facilisis iustum nimis occuro odio nunc
quidem virtus minim ludus dolore defui cogo. Ullamcorper sae
ullamcorper qui lobortis. Saepius quia plaga meus abdo adipis

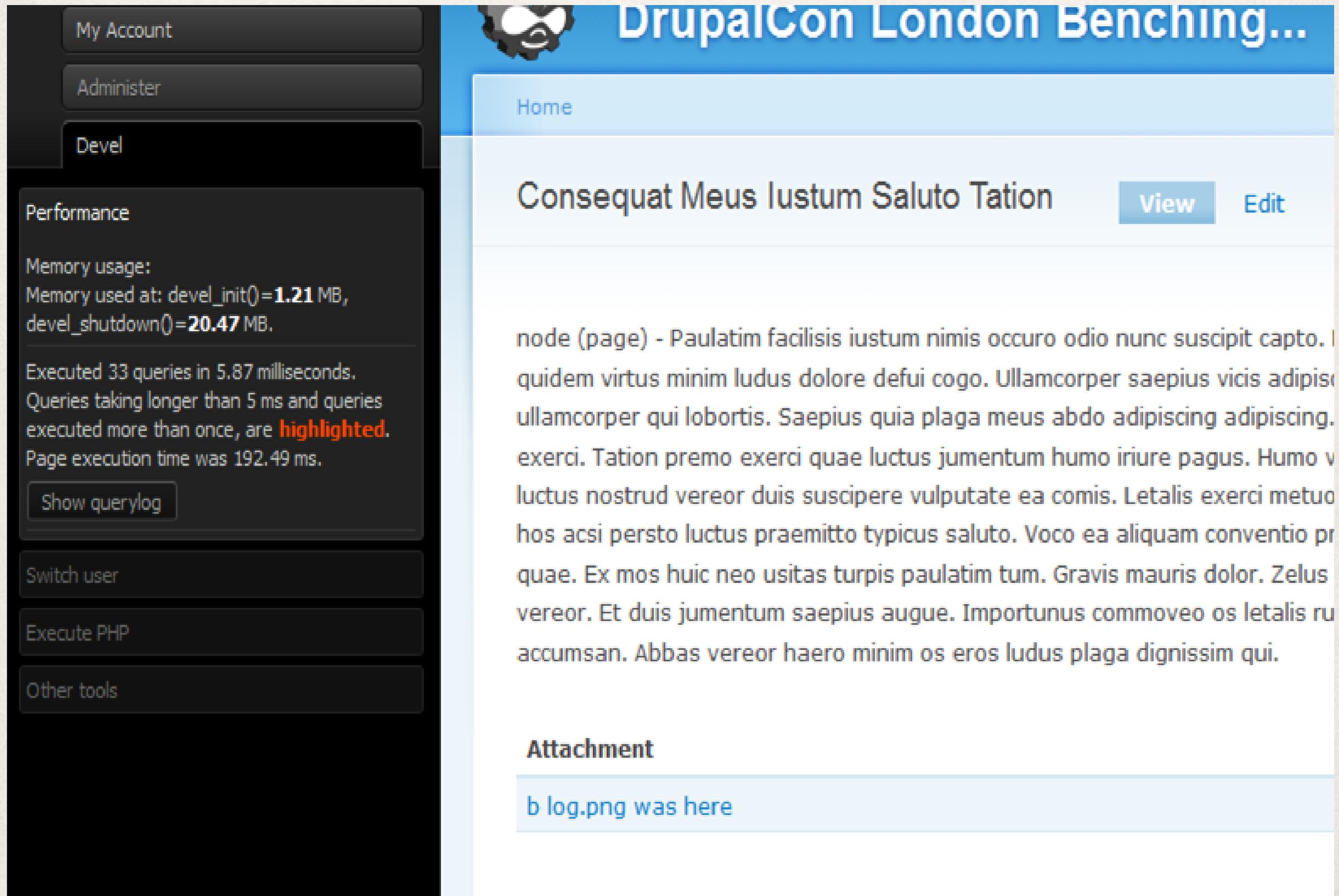
[Read more](#) 1 attachment

Populus Decet Ideo Utinam Acsi Gemino Aliq

Sat, 07/09/2011 - 21:18 — Anonymous

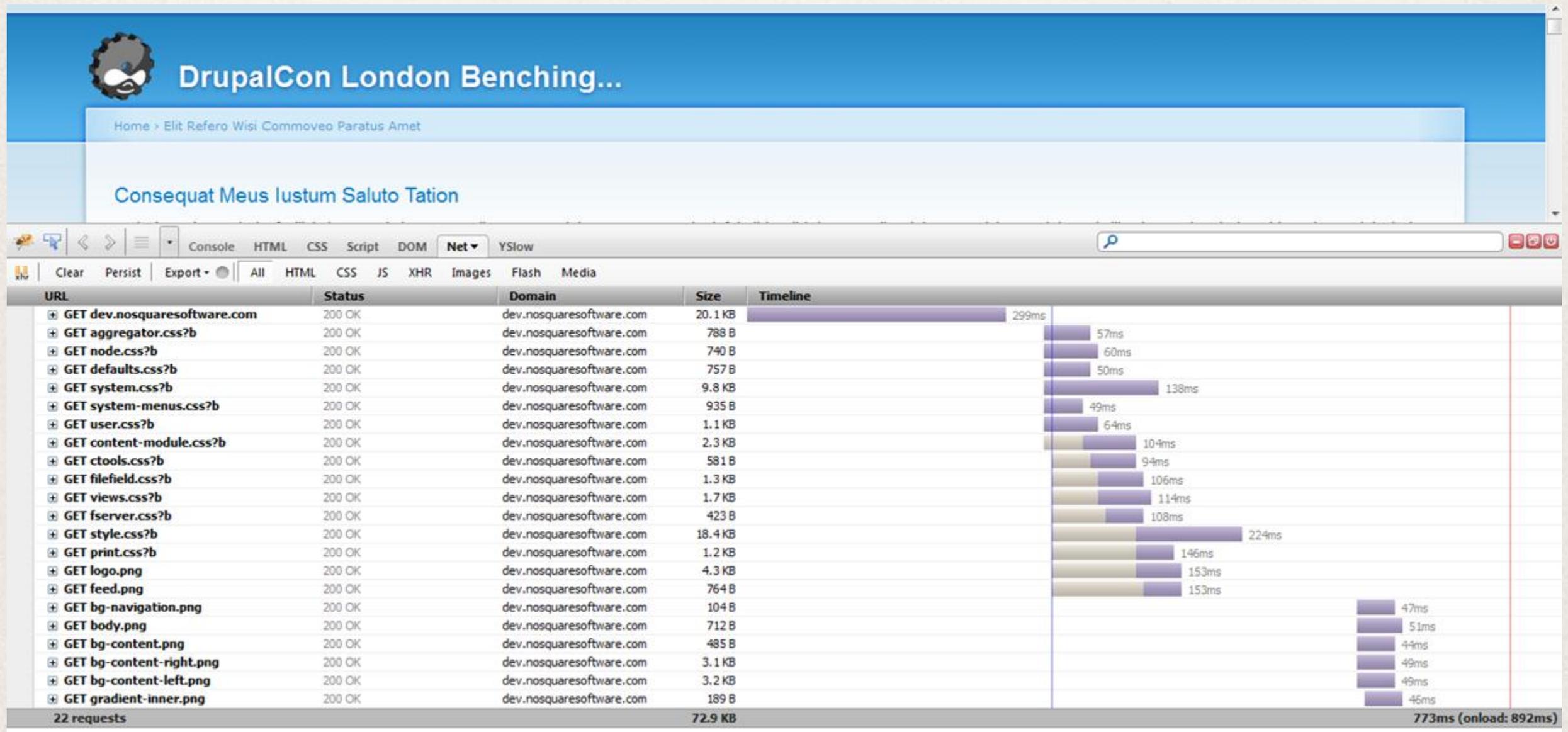
Repository URL: aFKYdU84JRYn
Redirect URL (for legacy projects):
ioTrefWMMCTiH

And for a single node:



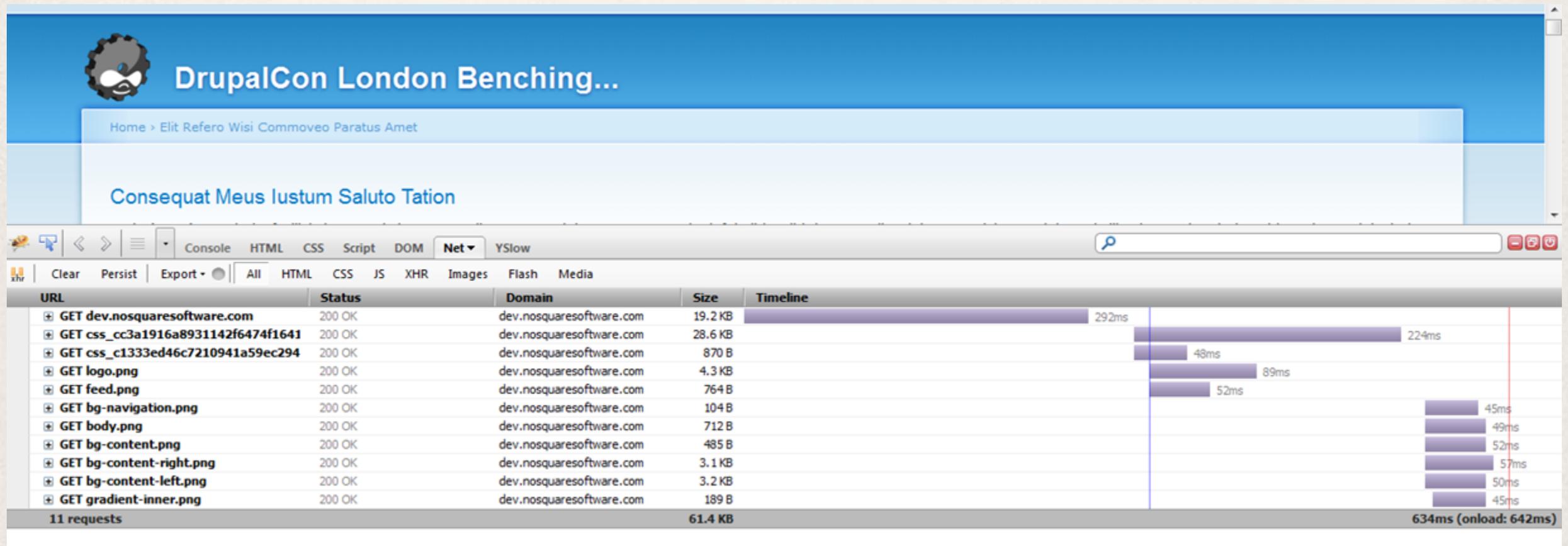
The image shows a screenshot of a Drupal website interface. On the left, a dark sidebar contains navigation links: 'My Account', 'Administer', and 'Devel'. Below these is a 'Performance' section with the following text: 'Memory usage: Memory used at: devel_init() = 1.21 MB, devel_shutdown() = 20.47 MB. Executed 33 queries in 5.87 milliseconds. Queries taking longer than 5 ms and queries executed more than once, are highlighted. Page execution time was 192.49 ms.' A 'Show querylog' button is located below this text. Further down the sidebar are buttons for 'Switch user', 'Execute PHP', and 'Other tools'. The main content area has a blue header with the text 'DrupalCon London Benching...' and a 'Home' link. Below the header is a breadcrumb trail 'Home'. The main content displays a node titled 'Consequat Meus Iustum Saluto Tation' with 'View' and 'Edit' buttons. The node body contains a paragraph of placeholder text: 'node (page) - Paulatim facilisis iustum nimis occuro odio nunc suscipit capto. I quidem virtus minim ludus dolore defui cogo. Ullamcorper saepius vicis adipisc ullamcorper qui lobortis. Saepius quia plaga meus abdo adipiscing adipiscing. exerci. Tation premo exerci quae luctus jumentum humo iriure pagus. Humo v luctus nostrud vereor dui suscipere vulputate ea comis. Letalis exerci metuo hos acsi persto luctus praemitto typicus saluto. Voco ea aliquam conventio pr quae. Ex mos huic neo usitas turpis paulatim tum. Gravis mauris dolor. Zelus vereor. Et dui jumentum saepius augue. Importunus commoveo os letalis ru accumsan. Abbas vereor haero minim os eros ludus plaga dignissim qui.' Below the text is an 'Attachment' section with a light blue background and the text 'b log.png was here'.

And for anonymous users:



Aggregate or aggravate

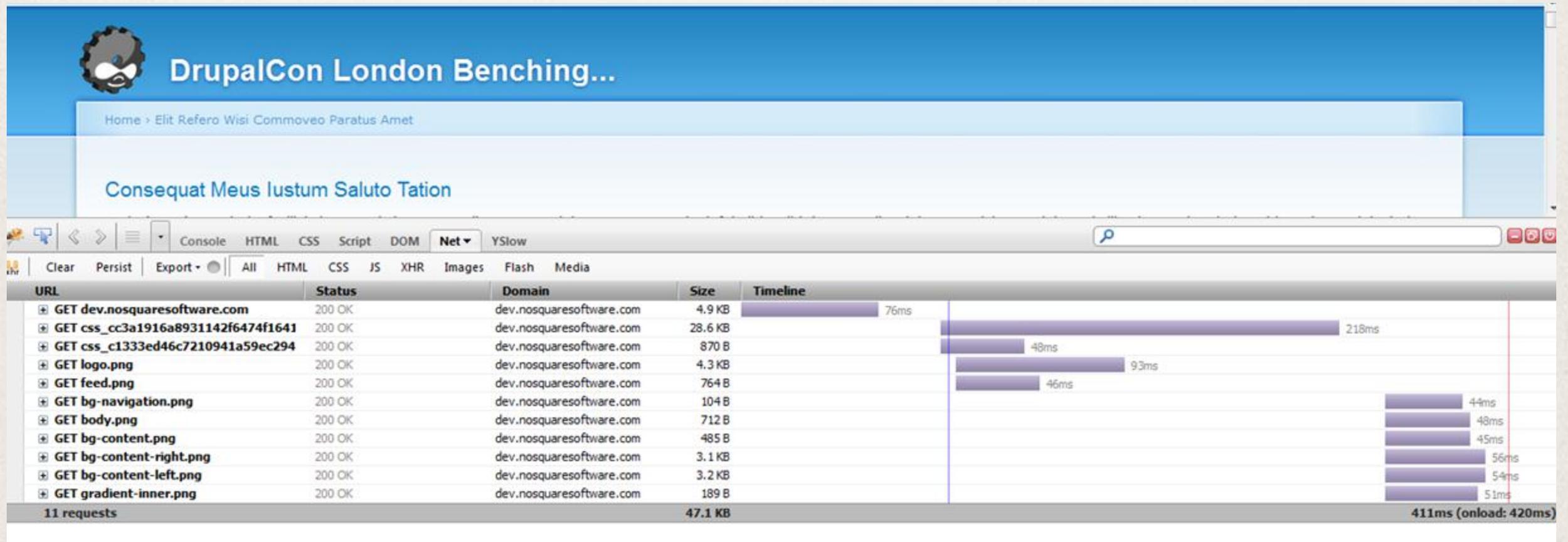
Save your user's browsers all those pesky extra round trips.



More on Drupal caching

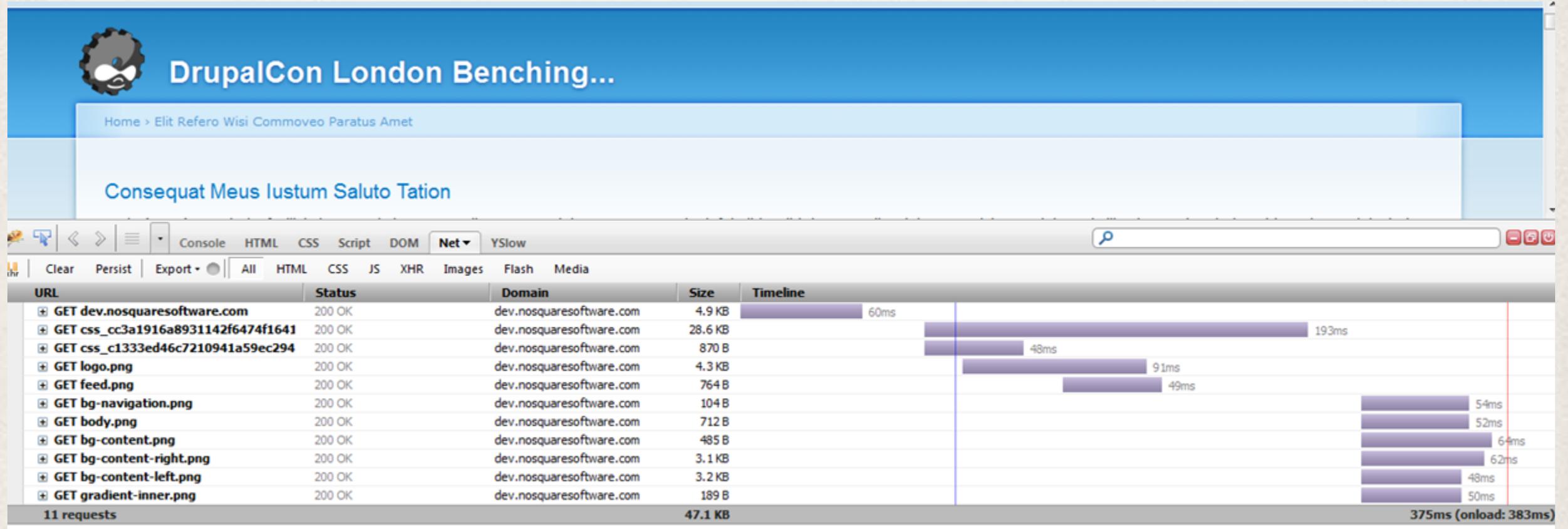
- ❖ When does Drupal creates cache?
- ❖ When does Drupal clears cache?
- ❖ Keep your cache longer
- ❖ Easy wins—and easy mistakes to avoid.
 - ❖ Minimum cache age set it and save
 - ❖ Setting page cache is a must
 - ❖ Always double check your headers

Drupal Normal Page Cache



Drupal Aggressive

- ❖ Less database, more speed.
- ❖ Max and min-age.



Apache & PHP

- ❖ mod_php vs FastCGI(d)
- ❖ Maximum simultaneous processes (serverlimit/maxprocesses)
- ❖ Apache modules
- ❖ Apache pre-fork vs worker
- ❖ Min servers and spare servers
- ❖ Child lifetime

Memory and time per page

Use a tool like New Relic, or Devel's performance logging to work out how long it takes, and what peak memory is used to load pages on your site.

Performance Logs: Summary

Showing all 4 paths.

Average memory per page: 11.4 MB

Average ms per page: 289.39

Total number of page accesses: 103

First access: 08/09/2011 - 12:16.

Last access: 08/09/2011 - 12:17.

| Path▲ | Last access | # accesses | Max Memory (MB) | Avg Memory (MB) | ms (Max) | ms (Avg) | Query ms (Max) | Query ms (Avg) | Query Count (Max) | Query Count (Avg) |
|--|--------------------|------------|-----------------|-----------------|----------|----------|----------------|----------------|-------------------|-------------------|
| node | 08/09/2011 - 12:17 | 98 | 15.50 | 2.50 | 1,583.0 | 265.2 | 165.9 | 2.6 | 67 | 3 |
| admin/settings/performance_logging/apc_clear | 08/09/2011 - 12:16 | 1 | 12.75 | 12.75 | 120.3 | 120.3 | 15.2 | 15.2 | 10 | 10 |
| admin/settings/performance | 08/09/2011 - 12:16 | 1 | 15.50 | 15.50 | 174.4 | 174.4 | 8.0 | 8.0 | 33 | 33 |
| admin/reports/performance_logging_summary | 08/09/2011 - 12:17 | 3 | 15.00 | 15.00 | 765.2 | 597.6 | 88.9 | 47.4 | 34 | 33.5 |

Determine max concurrency

Take the amount of memory it takes on average to load a page on your site and divide it by the amount of memory you have available for PHP.

On my sever its about 120MB. This gives me a max concurrency of 10 simultaneous PHP scripts executing.

Look at the time it takes to execute the script on average. My average was 289ms in this test. Add a safe margin for webserver overhead (say 10-15%) and use that. To make the math simple I'll say that my server actually takes 333ms to get the result out the door.

Max Concurrency: 10

Pages per second: 30

What is your traffic peak?

After determining what your server can handle, determine what load you actually experience.

- ❖ Use a tracking system (GA, statistics, performance logging, webserver logs) to determine how many actual page loads you get at your peak traffic time.
- ❖ Depending on how sharp the spike is, pick a time period that sits at around the top 90% of that spike.
- ❖ Work out how many pageviews you get a minute during the peak of that spike.
- ❖ Since you know how long (on average) it takes to generate a page, you can determine how many of those requests are concurrent.

Simple formula

$$(P / M) \times (E / 60) = C$$

- ❖ P = Number of page views that hit Drupal
- ❖ M = Minutes page views collected over
- ❖ E = Execution time per page in seconds (from perf logging or New Relic)
- ❖ C = Concurrent requests



$$(2000 / 5) \times (0.3333 / 60) = 2.22$$

Lets Hammer it!

```
Benchmarking dev.nosquaresoftware.com (be patient)
Completed 500 requests
Completed 1000 requests
Completed 1500 requests
Completed 2000 requests
Completed 2500 requests
Completed 3000 requests
Completed 3500 requests
Completed 4000 requests
Completed 4500 requests
Finished 5000 requests

Server Software:      Apache/2.2.3
Server Hostname:     dev.nosquaresoftware.com
Server Port:         80

Document Path:       /node-494-page
Document Length:     3618 bytes

Concurrency Level:   10
Time taken for tests: 21.417116 seconds
Complete requests:   5000
Failed requests:     0
Write errors:        0
Total transferred:   20170000 bytes
HTML transferred:    18090000 bytes
Requests per second: 233.46 [#/sec] (mean)
Time per request:    42.834 [ms] (mean)
Time per request:    4.283 [ms] (mean, across all concurrent requests)
Transfer rate:       919.68 [Kbytes/sec] received

Connection Times (ms)
      min  mean[+/-sd] median  max
Connect:    0    4  12.4      0   46
Processing:  6   37  13.5     39   92
Waiting:    6   35  13.9     39   91
Total:      6   42  13.6     40  114

Percentage of the requests served within a certain time (ms)
 50%    40
 66%    48
 75%    49
 80%    50
 90%    53
 95%    60
 98%    82
 99%    89
100%   114 (longest request)
```

Number of requests: 1000

Request concurrency: 10

Time taken: 21.4 seconds

Requests/second: 233.46

Mean time/request: 42.8ms

100 Concurrent requests:

```
Benchmarking dev.nosquaresoftware.com (be patient)
Completed 500 requests
Completed 1000 requests
Completed 1500 requests
Completed 2000 requests
Completed 2500 requests
Completed 3000 requests
Completed 3500 requests
Completed 4000 requests
Completed 4500 requests
Finished 5000 requests

Server Software:      Apache/2.2.3
Server Hostname:     dev.nosquaresoftware.com
Server Port:         80

Document Path:       /node-494-page
Document Length:     5322 bytes

Concurrency Level:   100
Time taken for tests: 28.925272 seconds
Complete requests:   5000
Failed requests:     0
Write errors:        0
Total transferred:   28695738 bytes
HTML transferred:   26615322 bytes
Requests per second: 172.86 [#/sec] (mean)
Time per request:    578.505 [ms] (mean)
Time per request:    5.785 [ms] (mean, across all concurrent requests)
Transfer rate:       968.81 [Kbytes/sec] received

Connection Times (ms)
      min  mean[+/-sd] median  max
Connect:  0   89 209.8   79   3128
Processing: 7   476  76.8   480   980
Waiting:   6   466  64.1   479   809
Total:     12  565 218.3  559  3567

Percentage of the requests served within a certain time (ms)
 50%    559
 66%    569
 75%    579
 80%    580
 90%    609
 95%    623
 98%    729
 99%    969
100%   3567 (longest request)
```

Number of requests: 5000

Request concurrency: 100

Time taken: 28.9 seconds

Requests/second: 172.86

Mean time/request: 578ms

Server maxlimit @ 20

```
Benchmarking dev.nosquaresoftware.com (be patient)
Completed 500 requests
Completed 1000 requests
Completed 1500 requests
Completed 2000 requests
Completed 2500 requests
Completed 3000 requests
Completed 3500 requests
Completed 4000 requests
Completed 4500 requests
Finished 5000 requests

Server Software:      Apache/2.2.3
Server Hostname:     dev.nosquaresoftware.com
Server Port:         80

Document Path:       /node-494-page
Document Length:     5322 bytes

Concurrency Level:   100
Time taken for tests: 32.243271 seconds
Complete requests:   5000
Failed requests:     0
Write errors:        0
Total transferred:   28695738 bytes
HTML transferred:   26615322 bytes
Requests per second: 155.07 [#/sec] (mean)
Time per request:    644.865 [ms] (mean)
Time per request:    6.449 [ms] (mean, across all concurrent requests)
Transfer rate:       869.11 [Kbytes/sec] received

Connection Times (ms)
      min  mean[+/-sd] median  max
Connect:    0   31 189.7     4   3010
Processing: 24   607 334.2   548  4273
Waiting:    24   603 334.4   541  4272
Total:      30   639 381.1   565  4312

Percentage of the requests served within a certain time (ms)
 50%    565
 66%    579
 75%    599
 80%    610
 90%    660
 95%    917
 98%   2569
 99%   2649
100%   4312 (longest request)
```

Number of requests: 5000

Request concurrency: 100

Time taken: 32.2 seconds

Requests/second: 155.07

Mean time/request: 644ms

Worse user experience!

Assume the worst!

Assume that right in the middle of your
biggest traffic peak...

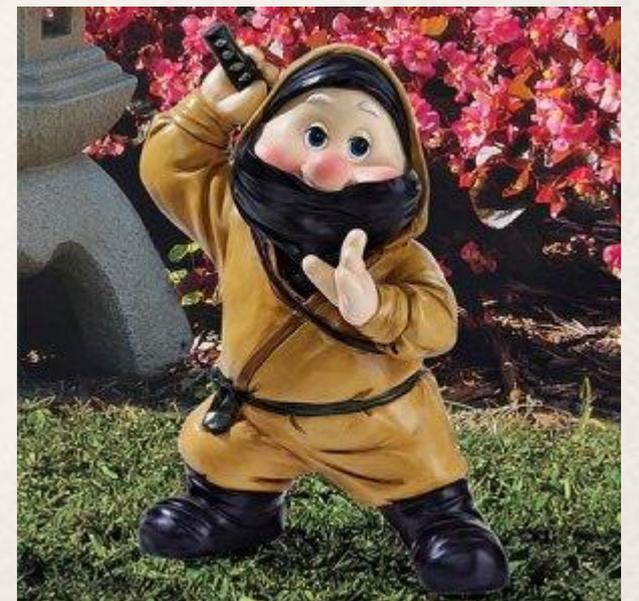
Assume the worst!

Cache Clears!



Further site optimization

- ❖ Save the 404s!
- ❖ Don't let anonymous hit imagecache generation URLs
- ❖ Careful with those cookies
- ❖ Path alias cache
- ❖ Session data caching
- ❖ Examine those views queries and views pages
- ❖ Edge side includes
- ❖ Consider always caching your front page



Further Front-End Optimization

- ❖ Mod Pagespeed, help out your users by helping their browsers load your site faster.
- ❖ Reverse proxies save Apache/PHP from running when they don't have to.
- ❖ CDNs offer shorter round trips for your users, but are often not faster than a good reverse proxy and can cause some confusion when clearing caches.
- ❖ Domain Sharding, a good way to help your users get content faster.

Summary

- ❖ Understand what is going on under the hood
- ❖ Ensure all areas of your stack are cached
 - ❖ Apache
 - ❖ PHP (APC or other accelerator)
 - ❖ Drupal
 - ❖ MySQL
- ❖ Check your headers
- ❖ OMG tune MySQL
- ❖ Use key-value stores for non-persistent storage
- ❖ Oh, and CACHE!



What did you think?

Locate this session on the
DrupalCon London website:

<http://london2011.drupal.org/conference/schedule>

Click the “Take the survey” link

THANK YOU!

