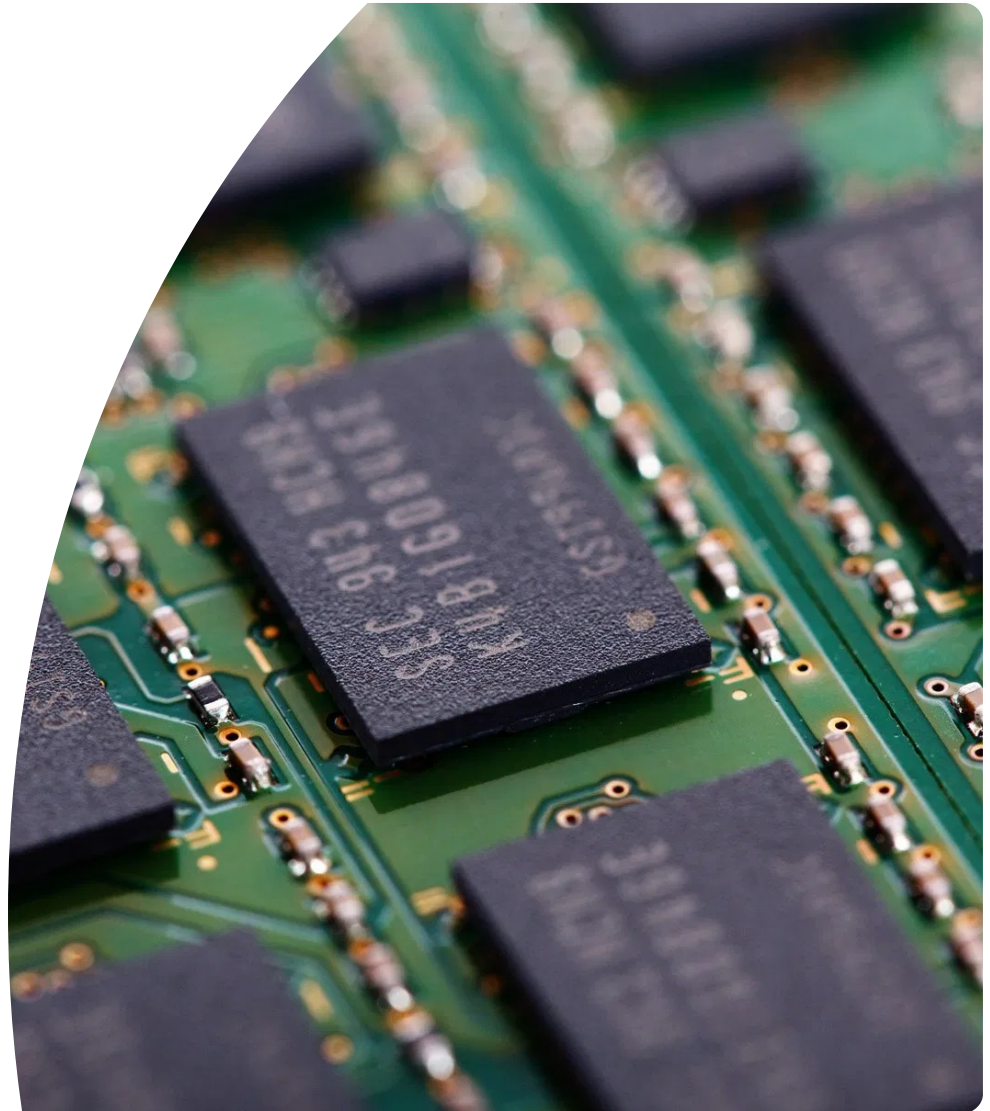


Moodle Caching at Scale

A presentation with many key values and lots in store. Content might be slightly hit and miss. Apologies for any stale content.

Brendan Heywood

catalyst 
expert open source solutions







SPEED

Is it only about speed?

- Affordability
- Service Stability / Availability
- Higher Scalability
- More predictability

https://en.wikipedia.org/wiki/List_of_system_quality_attributes

https://en.wikipedia.org/wiki/Long-tail_traffic

Case study: MDL-68481

- mod/folder/download_folder.php
- zips many files every single time
- Can we improve it using a “cache” of sorts?
- Ad hoc task to pre-build the zip?

<https://tracker.moodle.org/browse/MDL-68481>

<https://tracker.moodle.org/browse/MDL-68533>

But can we do better?

- Lots of CPU compressing already compressed files like videos and images. Just don't compress them

$$O(n \log^n) \rightarrow O(n)$$

- TTFB latency waiting for file to be written before sending to the browser. Instead stream to the browser

$$O(n) \rightarrow O(1)$$

Takeaway #1

- Caching should always be done last, and if still needed
- Eg caching large images vs resizing them (and then caching them)
- Profile with Tideways and focus on bottlenecks
- Always optimize the underlying process first
 - PHP loops into SQL
 - SQL properly indexed
- You get what you measure, so be careful what you measure
- Narrow optimization causes regressions
- The best optimizations are usually algorithm or data model changes

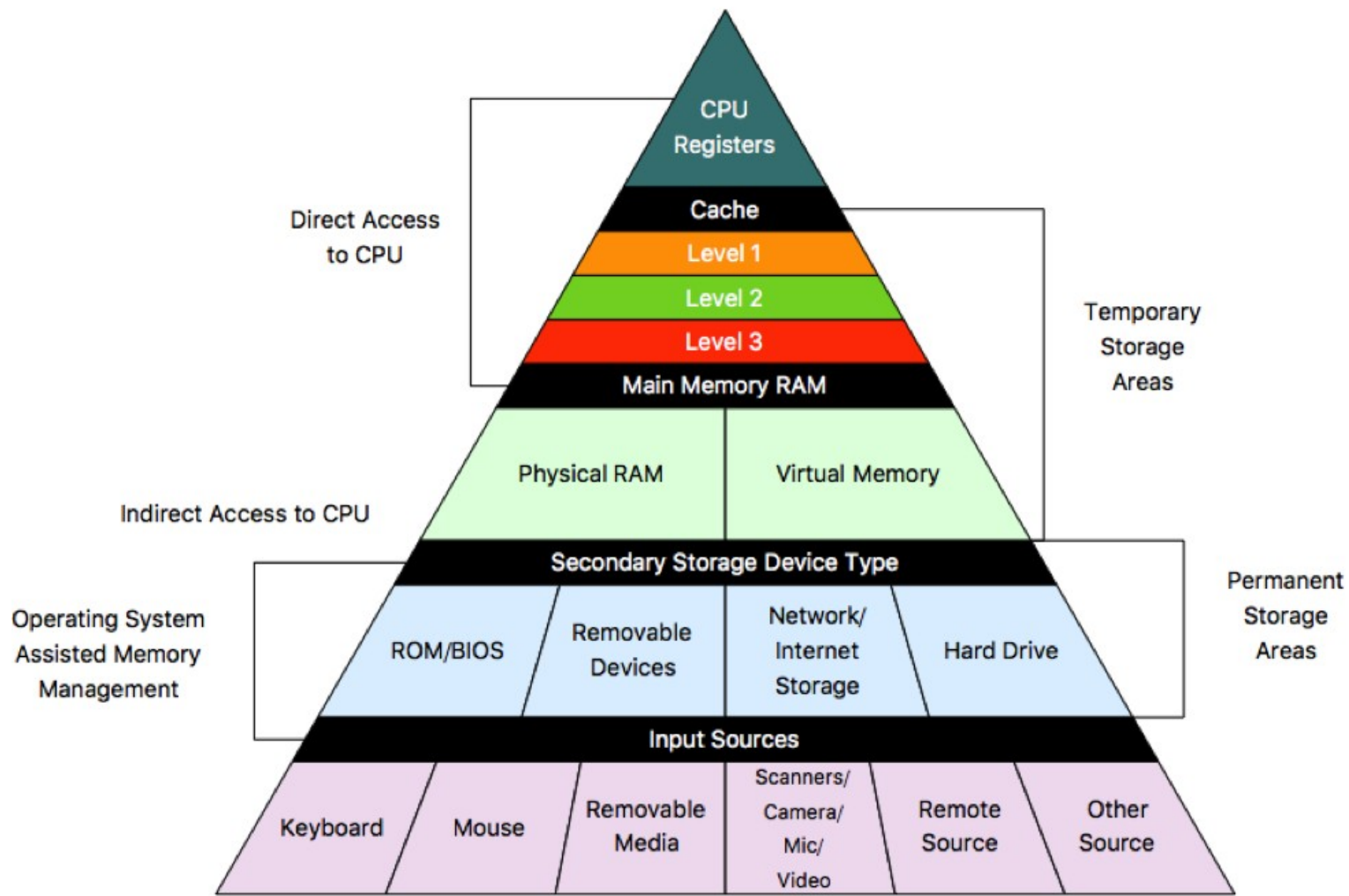
One of the biggest traps for smart engineers is optimizing a thing which shouldn't exist.

- Elon Musk



<https://www.youtube.com/watch?v=clQ36Kt7UVg> 3:00 – 5:30





https://en.wikipedia.org/wiki/Cache_hierarchy

Various caching layers

- Browser cache
- Browser local storage
- CDN / varnish
- Static cache
- memory on local APCu
- local file cache
- global cache
- shared file cache
- Database
- remote origin / API

Static caches

```
'string' => array(  
    'mode' => cache_store::MODE_APPLICATION,  
    'staticacceleration' => true,  
    'staticaccelerationsize' => 30,  
),
```

- Just like a static var but safer with deletes / purges
- In memory cache for a single request
- Use it for repeated requests, and when total memory is low
- Usually the total number of unique cache keys is low
- https://docs.moodle.org/dev/Cache_API#Cache_modifiers

Memory cache APCu

- Configured as a cache store type in MUC
- Relatively small, **never** let it fill up
- Perfect for strings, databasemeta and other caches with a very high hit rate but small items
- Only good for localizable caches

https://github.com/moodleuulm/moodle-tool_apcu

[https://docs.moodle.org/38/en/APC_user_cache_\(APCu\)](https://docs.moodle.org/38/en/APC_user_cache_(APCu))

Local file caches

```
'string' => array(  
  'mode' => cache_store::MODE_APPLICATION,  
  'canuselocalstore' => true,  
),
```

- Use for items which are not fast moving
- Ideally always stack them a second shared cache store

Configured store instances

Store name	Plugin	Ready	Store mappings	Modes	Supports
local_apcu	APC user cache (APCu)	✓	1	Application, Session	ttl, key awareness
local_file	File cache	✓	2	Application, Session	data guarantee, ttl, key awareness, searching by key
shared_file	File cache	✓	2	Application, Session	data guarantee, ttl, key awareness, searching by key
shared_redis	Redis	✓	3	Application, Session	data guarantee, locking, key awareness, searching by key

HTML Purifier - cleaned content	Application	core	htmlpurifier	local_file, shared_file	Site identifier	Yes	Edit mappings , Edit sharing , Purge
Language string cache	Application	core	string	local_apcu, shared_redis	Site identifier	Yes	Edit mappings , Edit sharing , Purge
List of available languages	Application	core	langmenu	local_apcu, shared_file	Site identifier	Yes	Edit mappings , Edit sharing , Purge
List of course contacts	Application	core	coursecontacts	shared_redis	Site identifier	No	Edit mappings , Edit sharing , Purge

Mode	Cache item	Static	H	M	Primary store	H	M	S	Final store	H	M	S
Ses	core/calendar_categories				shared_redis	1	2	3				
App	core/capabilities	** static accel. **	82	1	shared_redis	0	1	1				
Ses	core/coursecat				shared_redis	0	3	4				
Req	core/coursecatrecords				default_request	0	0	1				
App	core/coursecattree	** static accel. **	2	3	shared_redis	0	3	2				
App	core/coursemodinfo				local_file	0	2	1	shared_file	0	2	1
App	core/fontawesomeiconmapping	** static accel. **	0	1	shared_redis	0	1	1				
App	core/langmenu	** static accel. **	3	1	local_apcu	0	1	1	shared_file	0	1	1
App	core/message_processors_enabled	** static accel. **	0	1	shared_redis	0	1	1				
App	core/plugin_functions	** static accel. **	0	15	shared_redis	0	15	15				
App	core/plugin_manager				shared_redis	0	3	3				
Req	core/repositories				default_request	0	10	20				
App	core/roledefs	** static accel. **	0	1	shared_redis	0	1	1				
App	core/string	** static accel. **	955	47	local_apcu	3	44	44	shared_redis	0	44	44
App	core/yuimodules				shared_redis	0	3	3				
App	local_envbar/records				shared_redis	0	1	1				
App	tool_abconfig/experiments	** static accel. **	5	1	shared_redis	0	1	1				
App	tool_usertours/tourdata	** static accel. **	0	1	shared_redis	0	1	1				
App	tool_webanalytics/records				shared_redis	0	2	1				

Mode	Cache item	Static	H	M	Primary store	H	M	S
Ses	core/calendar_categories				shared_redis	3	0	1
App	core/capabilities	** static accel. **	77	1	shared_redis	1	0	0
Ses	core/coursecat				shared_redis	2	0	1
App	core/coursemodinfo				local_file	1	0	0
App	core/eventinvalidation	** static accel. **	0	4	shared_redis	0	4	0
App	core/fontawesomeiconmapping	** static accel. **	0	1	shared_redis	1	0	0
App	core/langmenu	** static accel. **	3	1	local_apcu	1	0	0
App	core/message_processors_enabled	** static accel. **	0	1	shared_redis	1	0	0
App	core/plugin_functions	** static accel. **	0	14	shared_redis	14	0	0
App	core/plugin_manager				shared_redis	3	0	0
App	core/string	** static accel. **	944	46	local_apcu	46	0	0
App	core/yuimodules				shared_redis	3	0	0
App	local_envbar/records				shared_redis	1	0	1
App	tool_abconfig/experiments	** static accel. **	5	1	shared_redis	1	0	0
App	tool_usertours/tourdata	** static accel. **	0	1	shared_redis	1	0	0
App	tool_webanalytics/records				shared_redis	1	1	0

<https://tracker.moodle.org/browse/MDL-68945>

Elasticity

Slashdot effect / Covid effect



**THERE ARE ONLY 2 HARD THINGS
IN COMPUTER SCIENCE:**

- 0. Cache Invalidation
- 1. Naming Things
- 7. Asynchronous Callbacks
- 2. Off-by-one errors

How to localize a cache

- You must have multiple keys for each version
- Structure the keys to be unique either based on time, or deterministic based on all the inputs
- Old unused keys eventually reclaimed via TTL, LRU, or periodic purge
- Store the current version somewhere shared: \$CFG, DB or another cache

Localised key examples

- Timestamps are ok if slow enough to not clash
'coursemodinfo' uses the `increment_revision_number()` helper
<https://github.com/moodle/moodle/blob/master/lib/modinfolib.php#L2380-L2383>
- Reuse existing version numbers
'htmlpurifier' uses moodle build version and all params as part of the key:
<https://github.com/moodle/moodle/blob/master/lib/weblib.php#L1785-L1810>
- Content hashes work well and never clash, eg like File API internals, and git commit hashes and git branch references



Stampede!

- https://en.wikipedia.org/wiki/Cache_stampede
- Mitigations and workarounds
 - Primary / secondary caches
<https://tracker.moodle.org/browse/MDL-67020>
 - Request coalescing / Locking
<https://github.com/moodle/moodle/blob/master/lib/modinfolib.php#L453-L462>
<https://tracker.moodle.org/browse/MDL-61305>
- Beware of cron
- Cache purges === DOS attack
- Preventions
 - Cache rewarming



Rewarming

- If possibly write the new value in MUC before you update the local reference
- Applies to both localized and non local caches
- For big slow moving and expensive caches, and where eventual consistency is OK then defer to an adhoc task

General dev guidelines

- Only set caches in a single place
https://en.wikipedia.org/wiki/Single-responsibility_principle
- Less bad to invalidate in many places, but better to rebuild instead
- So completely encapsulate all cache gets / sets in your own class
- The outside world should be unaware of the caching
- Cache datasources
 - A nice pattern for simple caches
https://docs.moodle.org/dev/Cache_API#Specifying_a_data_source
 - Don't have a convenient 'refresh' method
 - Difficult to use with localized caches
<https://tracker.moodle.org/browse/MDL-68456>

Automating MUC config

- Setting up MUC is a pain (worse if you have hundreds of sites)
- `datadir/muc/config.php` needs to be retired
- We want a way to completely override and automate the entire cache management
<https://tracker.moodle.org/browse/MDL-41492>
- A new plugin which handles this via Cache rulesets
https://github.com/catalyst/moodle-tool_forcedcache
 - eg 'Caching config as code'
 - It will ship with some standard rulesets ready for prod use

Caching

REFRESH TYPE	EXAMPLE SHORTCUTS	EFFECT
SOFT REFRESH	EMAIL <input type="text" value="REFRESH"/> BUTTON	REQUESTS UPDATE WITHIN JAVASCRIPT
NORMAL REFRESH	F5, CTRL-R, ⌘R	REFRESHES PAGE
HARD REFRESH	CTRL-F5, CTRL-⇧, ⌘⇧R	REFRESHES PAGE INCLUDING CACHED FILES
HARDER REFRESH	CTRL-⇧-HYPER-ESC-R-F5	REMOVELY CYCLES POWER TO DATACENTER
HARDEST REFRESH	CTRL-⌘-⌘-⇧-#-R-F5-F5-ESC-O-O-Ø-▲-SCROLL LOCK	INTERNET STARTS OVER FROM ARPANET

<https://xkcd.com/1854/>



ASK ME ANYTHING *

* I may answer with a cached response

Image credits 1/2

- Computer chips
<https://www.publicdomainpictures.net/en/view-image.php?image=18192>
- Game: Settlers of Catan
<https://www.catan.com/game/catan>
<https://www.flickr.com/photos/bods/6120445526>
- Movie: Speed
<https://www.imdb.com/title/tt0111257/>
- Everyday Astronaut interview with Elon Musk
<https://www.youtube.com/everydayastronaut>
- Movie: Shrek
<https://www.imdb.com/title/tt0126029/>

Image credits 2/2

- CPU Memory levels
<https://www.hardwaretimes.com/what-is-a-processor-register-types-of-cpu-registers-and-functions-explained/>
- Code Tee Shirt
<https://www.codetee.com/products/2-hard-things-in-computer-science>
- Movie: The Lion King
<https://www.imdb.com/title/tt0110357/>
- Reheating Pizza
<https://spoonuniversity.com/how-to/how-to-reheat-pizza-in-the-microwave>
- Reddit Ask Me Anything
<https://www.reddit.com/>