Opening The Black Box

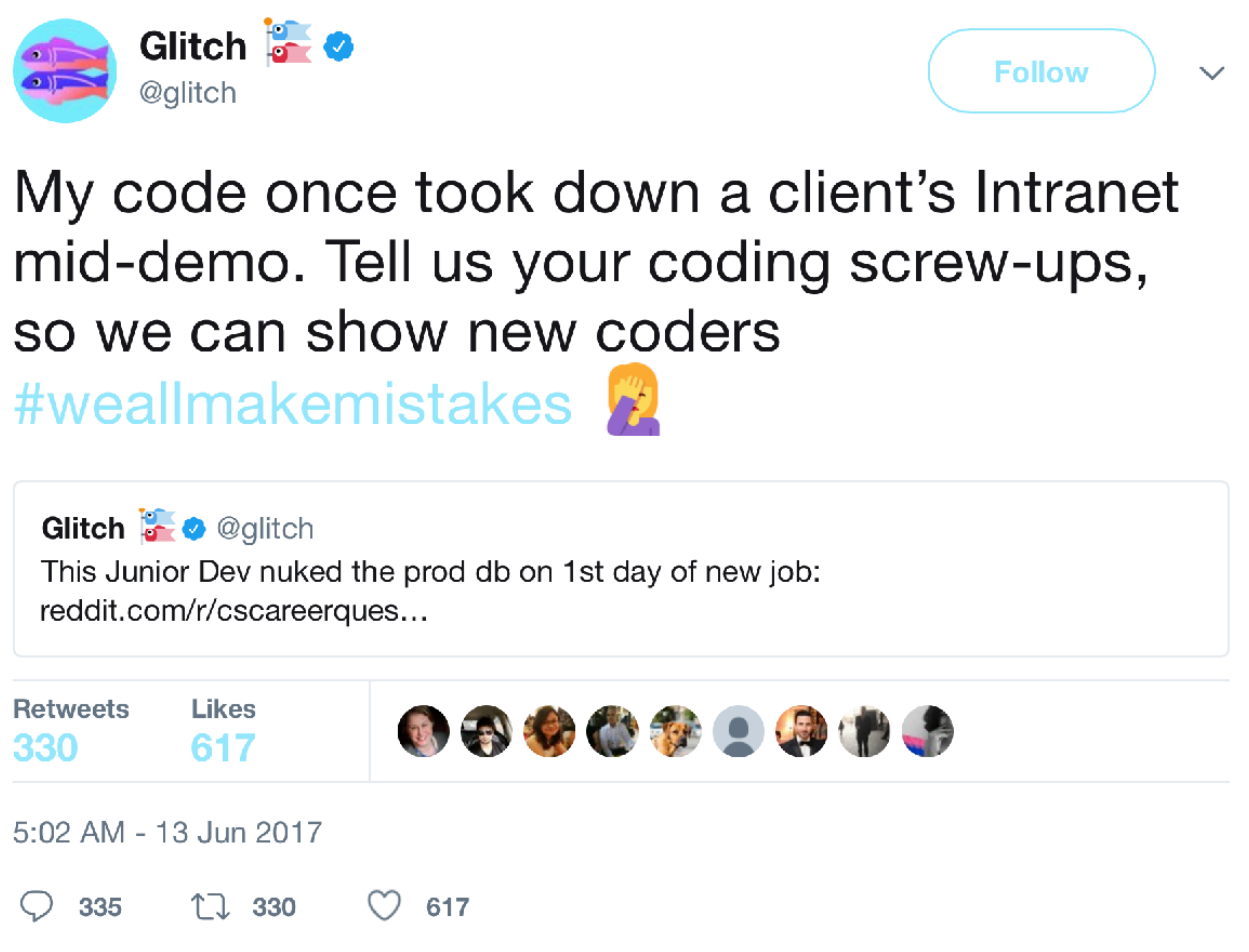
NEDCamp 11/18/2017

Howdy!
I'm an engineer at Acquia

Dustin Younse
@milsyobtaf

https://github.com/milsyobtaf/prez

What Is Debugging?



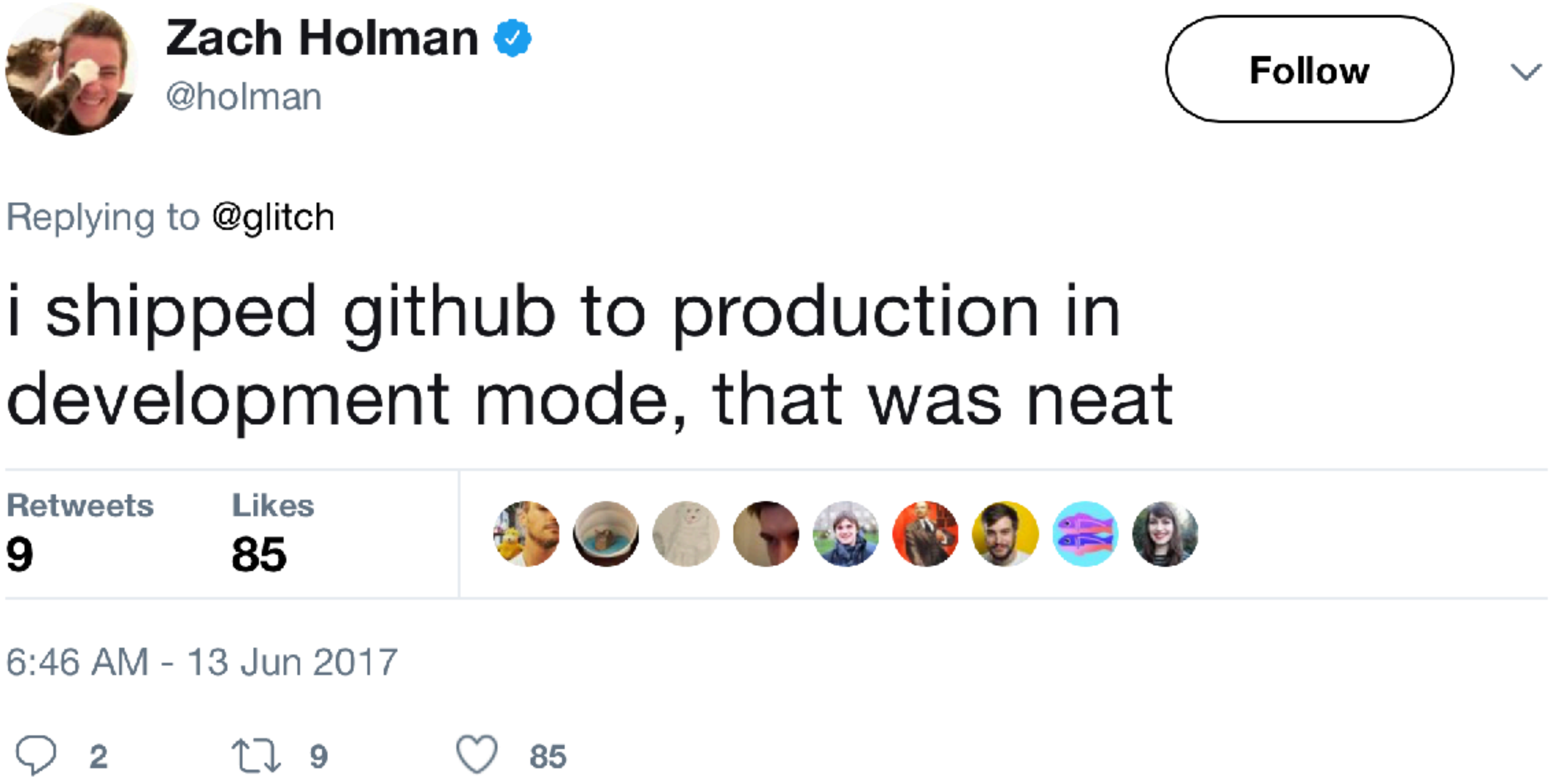
Glitch 🎏
@glitch

My code once took down a client’s Intranet mid-demo. Tell us your coding screw-ups, so we can show new coders #weallmakemistakes 🤦‍♀️

> **Glitch** 🎏 @glitch
> This Junior Dev nuked the prod db on 1st day of new job: reddit.com/r/cscareerques…

Retweets 330 Likes 617

5:02 AM - 13 Jun 2017



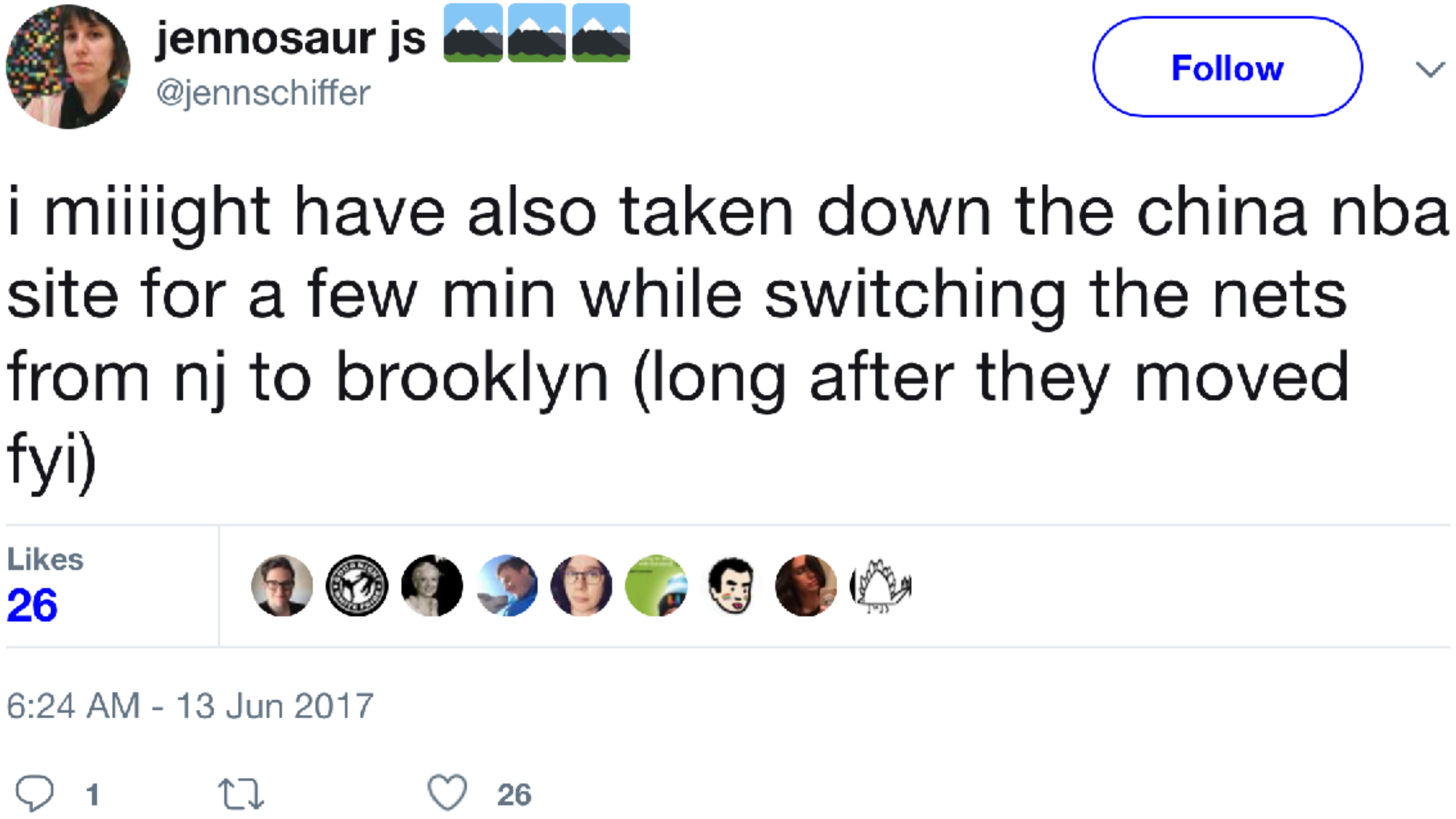
Zach Holman @holman

Replying to @glitch

i shipped github to production in development mode, that was neat

Retweets 9 Likes 85

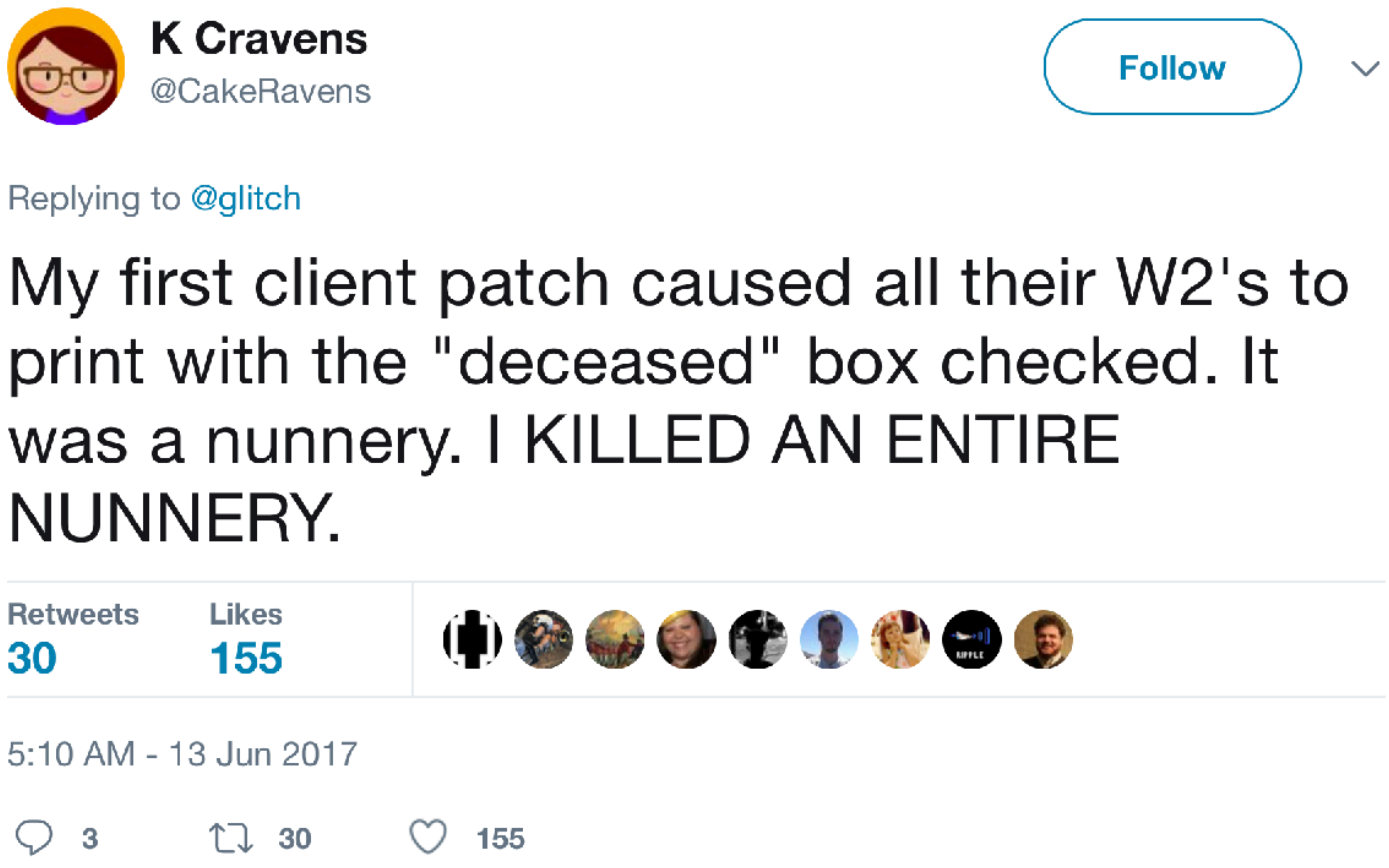
6:46 AM - 13 Jun 2017



jennosaur js 🏔🏔🏔
@jennschiffer

i miiiight have also taken down the china nba site for a few min while switching the nets from nj to brooklyn (long after they moved fyi)

6:24 AM - 13 Jun 2017



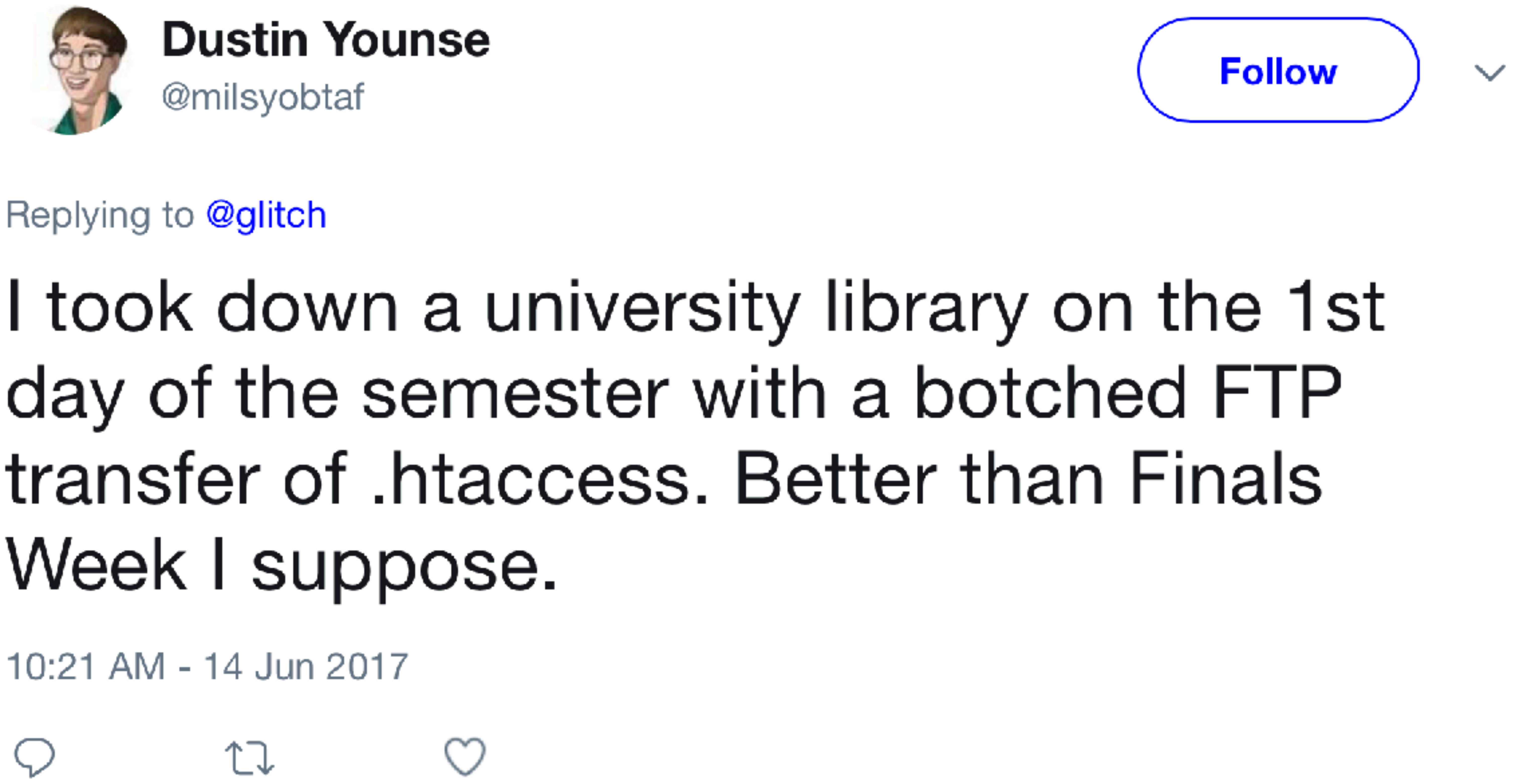
K Cravens
@CakeRavens

Replying to @glitch

My first client patch caused all their W2's to print with the "deceased" box checked. It was a nunnery. I KILLED AN ENTIRE NUNNERY.

Retweets 30 Likes 155

5:10 AM - 13 Jun 2017



Dustin Younse
@milsyobtaf

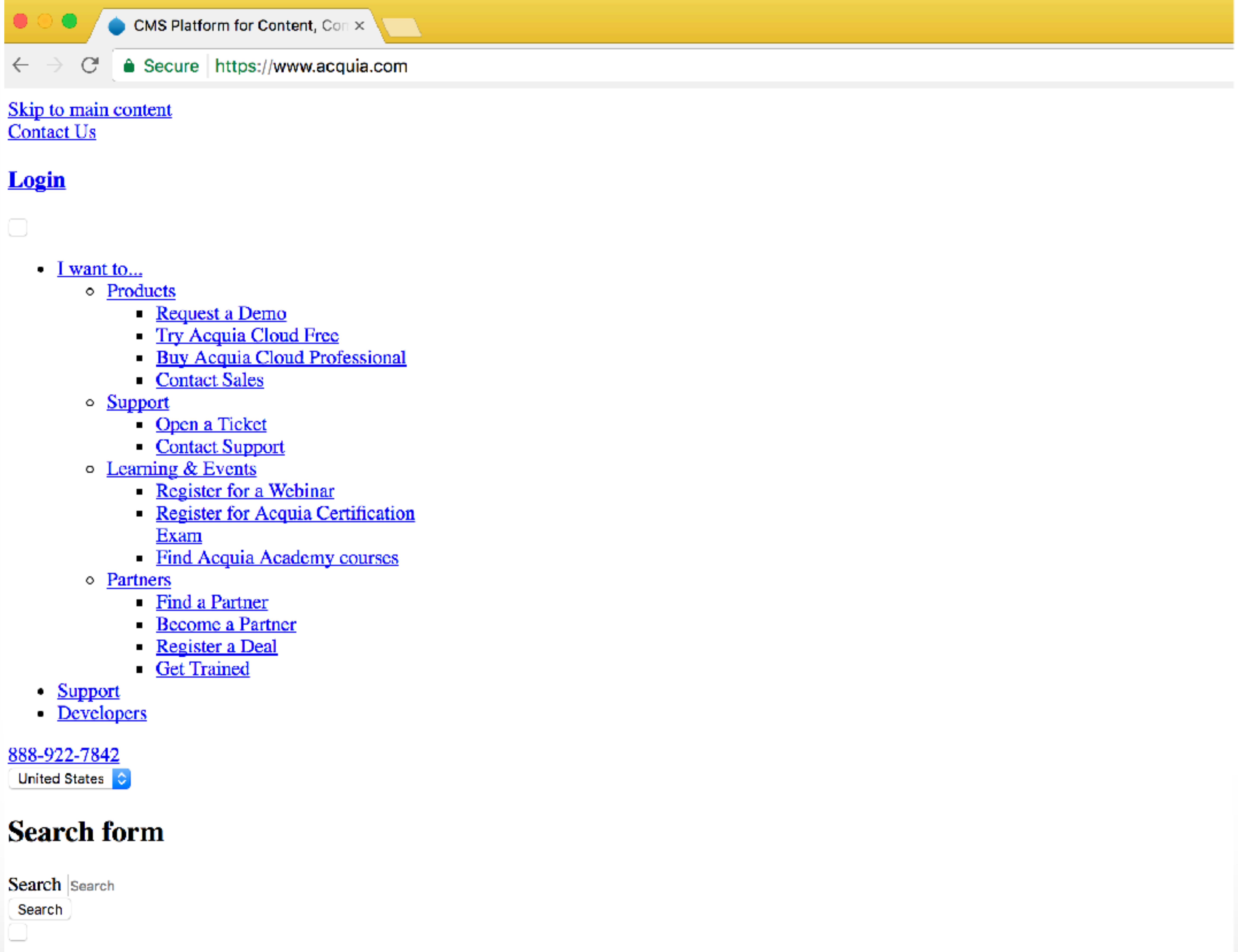
Replying to @glitch

I took down a university library on the 1st day of the semester with a botched FTP transfer of .htaccess. Better than Finals Week I suppose.

10:21 AM - 14 Jun 2017

What Is Debugging?

Nothing But Text?



[Skip to main content](#)
[Contact Us](#)

[Login](#)

□

- [I want to...](#)
  - [Products](#)
    - [Request a Demo](#)
    - [Try Acquia Cloud Free](#)
    - [Buy Acquia Cloud Professional](#)
    - [Contact Sales](#)
  - [Support](#)
    - [Open a Ticket](#)
    - [Contact Support](#)
  - [Learning & Events](#)
    - [Register for a Webinar](#)
    - [Register for Acquia Certification Exam](#)
    - [Find Acquia Academy courses](#)
  - [Partners](#)
    - [Find a Partner](#)
    - [Become a Partner](#)
    - [Register a Deal](#)
    - [Get Trained](#)
- [Support](#)
- [Developers](#)

[888-922-7842](#)
United States

Search form

Search Search
Search
□

Nothing But Text?

- CSS aggregated but not cached
- JS blocked in the browser
- File permissions in sites/default/files

Debugging vs Troubleshooting

- Mostly a semantic difference
- 5 minutes vs 5 hours?
- “Fixed” vs understood and corrected

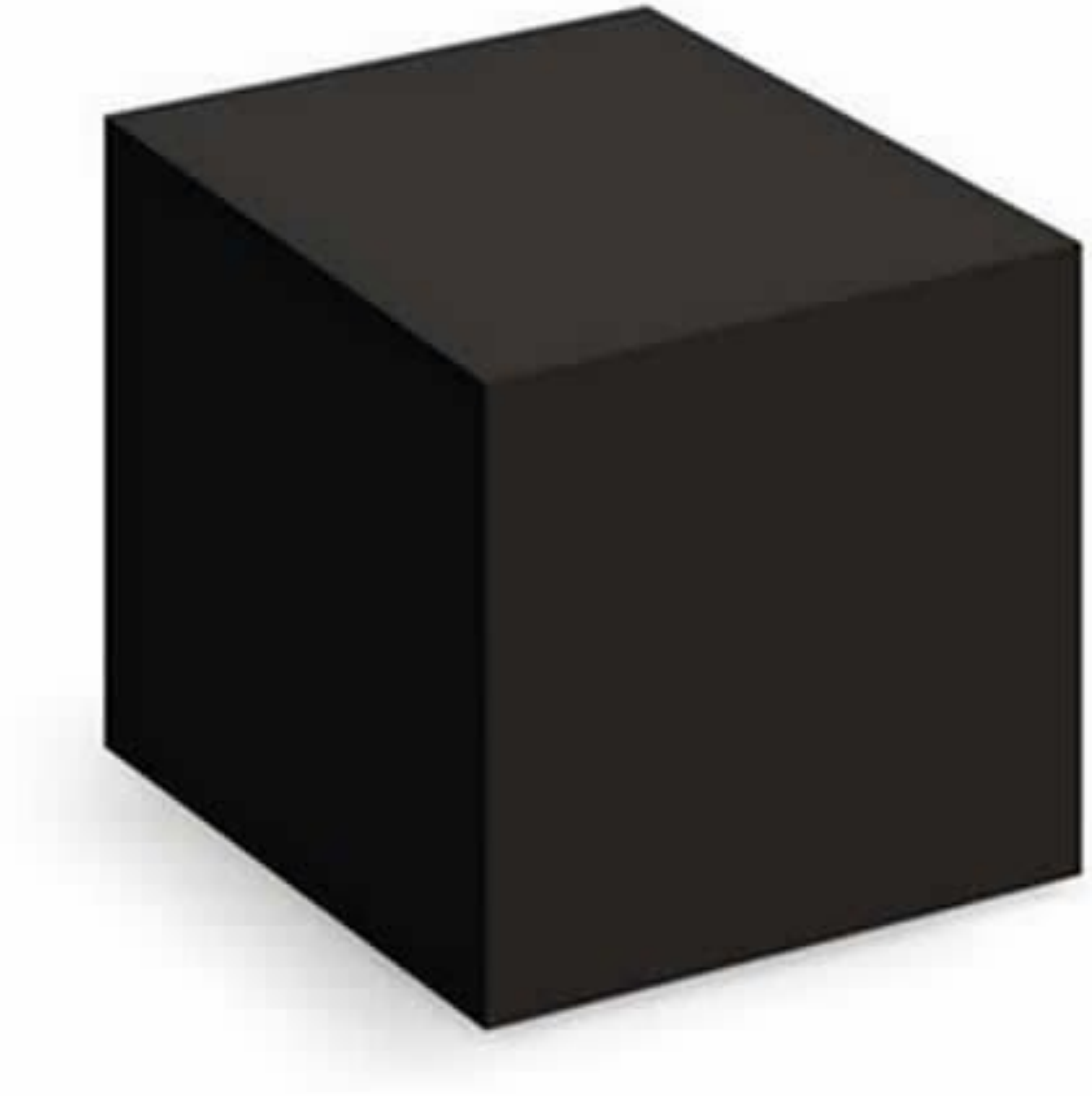
What Is The Black Box?

A Familiar Story?

- Non-technical education
- Self taught web skills
- Above average Googling abilities
- Late nights staring at error messages

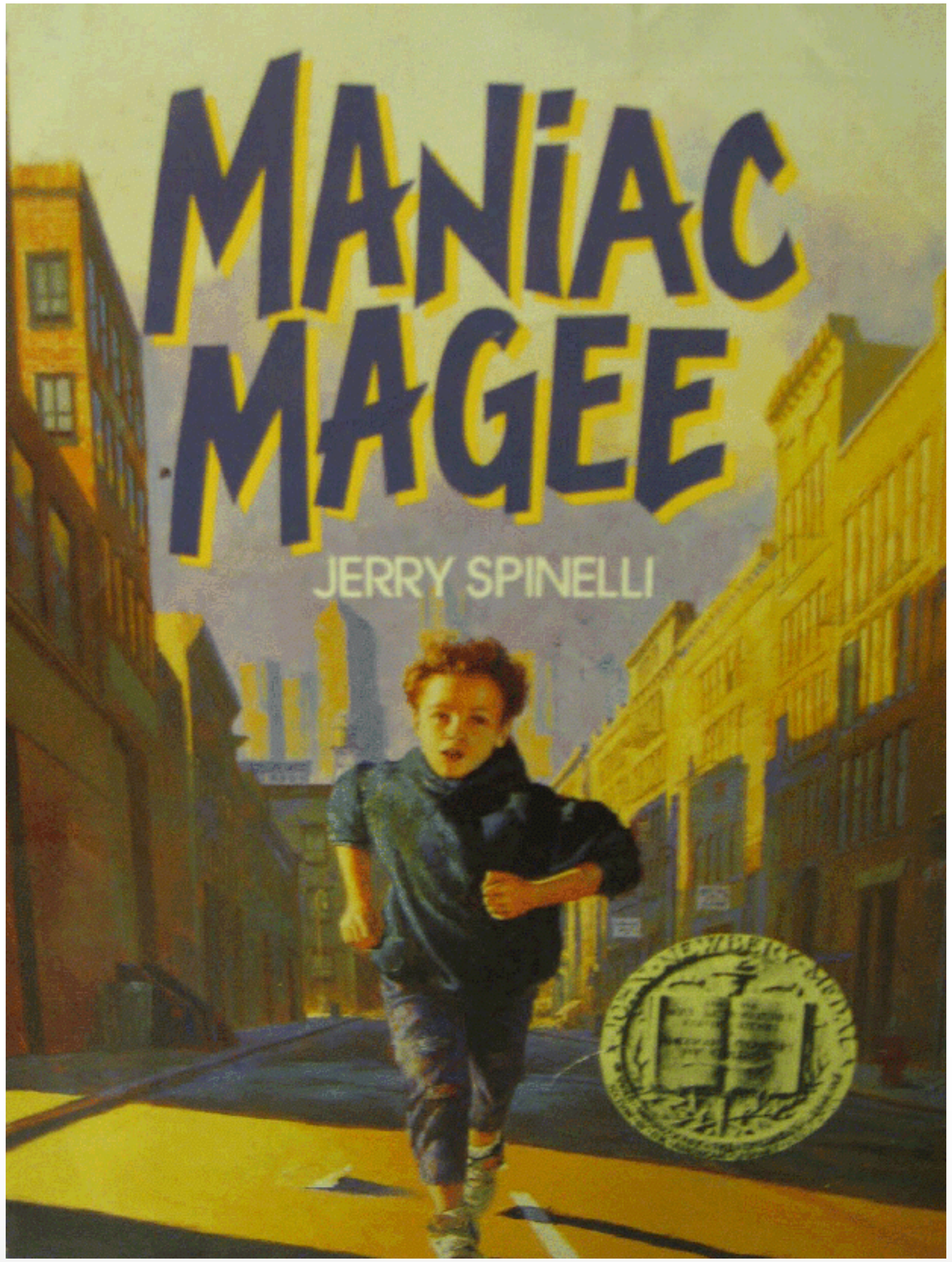
What Is Drupal?

Mental Model of Drupal



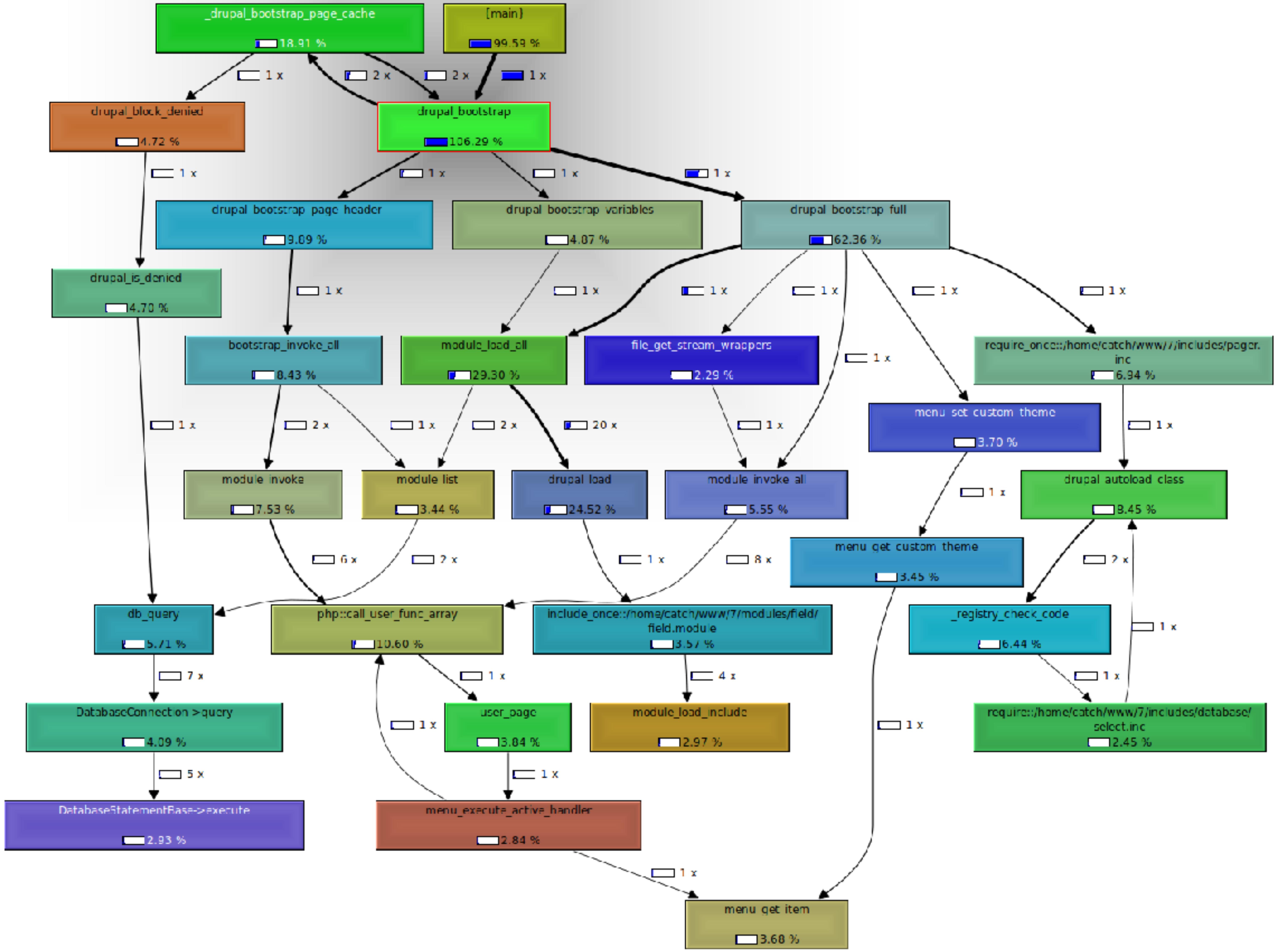
Mental Model of Drupal

Mental Model of Drupal

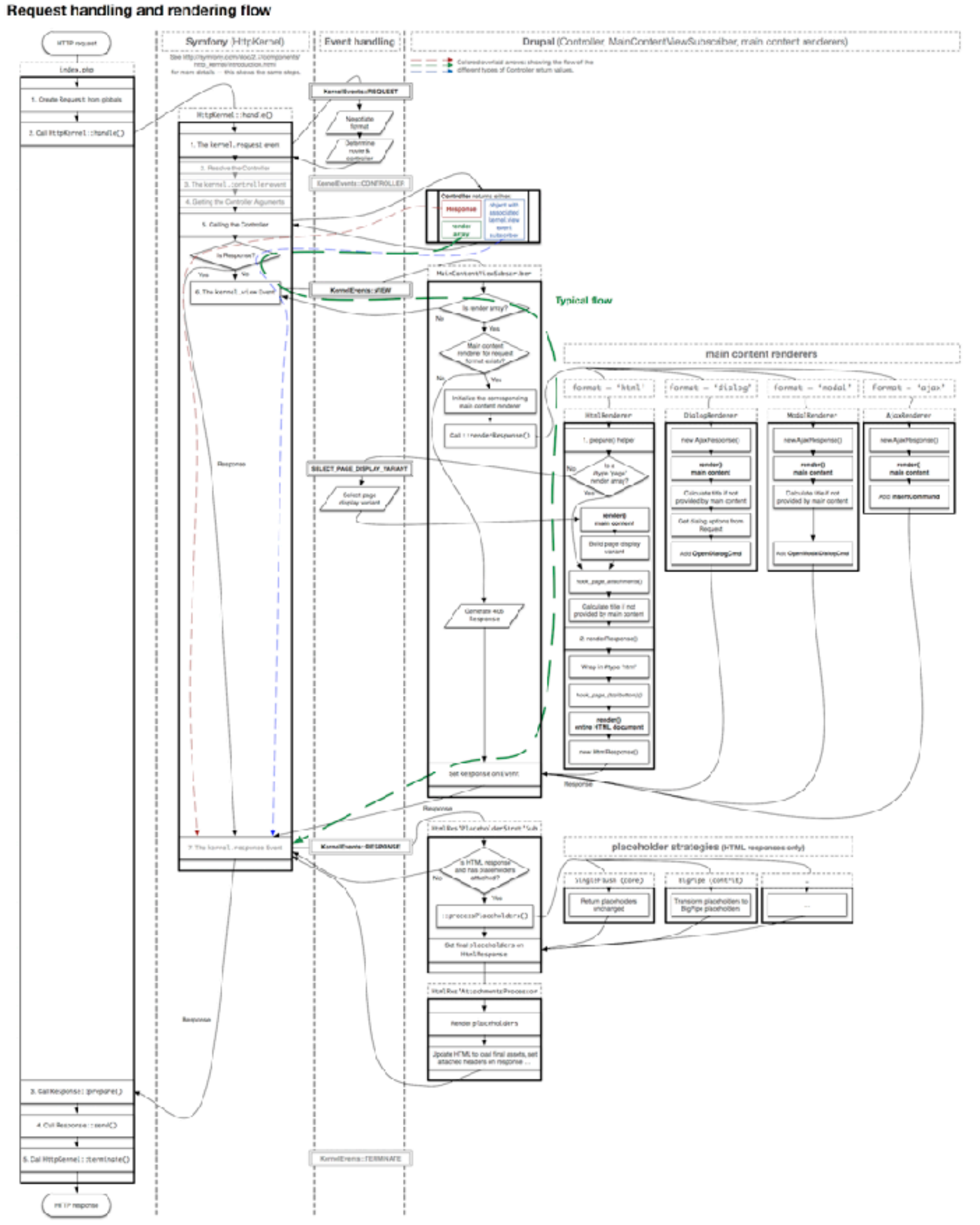


Mental Model of Drupal

Mental Model of Drupal



Mental Model of Drupal


Request handling and rendering flow
HTTP request
index.php
1. Create Request from globals
2. Call HttpKernel::handle()
3. Call Response::prepare()
4. Call Response::send()
5. Call HttpKernel::terminate()
HTTP response
Symfony (HttpKernel)
See http://symfony.com/doc/2.7/components/http_kernel/introduction.html for more details — this shows the same steps.
HttpKernel::handle()
1. The kernel.request event
2. Resolve the Controller
3. The kernel.controller event
4. Getting the Controller Arguments
5. Calling the Controller
Is Response?
Yes
No
6. The kernel.view Event
7. The kernel.response Event
Response
Event handling
KernelEvents::REQUEST
Negotiate format
Determine route & controller
KernelEvents::CONTROLLER
KernelEvents::VIEW
SELECT_PAGE_DISPLAY_VARIANT
Select page display variant
KernelEvents::RESPONSE
KernelEvents::TERMINATE
Drupal (Controller, MainContentViewSubscriber, main content renderers)
Colored/vertical arrows: showing the flow of the different types of Controller return values.
Controller returns either:
Response
render array
object with associated kernel.view event subscriber
Typical flow
MainContentViewSubscriber
Is render array?
No
Yes
Main content renderer for request format exists?
Initialize the corresponding main content renderer
Call ::renderResponse()
Generate 406 Response
Set response on event
main content renderers
format = 'html'
format = 'dialog'
format = 'modal'
format = 'ajax'
HtmlRenderer
DialogRenderer
ModalRenderer
AjaxRenderer
1. prepare() helper
Is a #type 'page' render array?
render() main content
Build page display variant
hook_page_attachments()
Calculate title if not provided by main content
2. renderResponse()
Wrap in #type 'html'
hook_page_{top,bottom}()
render() entire HTML document
new HtmlResponse()
new AjaxResponse()
render() main content
Calculate title if not provided by main content
Get dialog options from Request
Add OpenDialogCmd
Add OpenModalDialogCmd
Add insertCommand
Response
HtmlResponsePlaceholderStrategySubscriber
Is HTML response and has placeholders attached?
::processPlaceholders()
Set final placeholders on HtmlResponse
placeholder strategies (HTML responses only)
SingleFlush (core)
BigPipe (contrib)
Return placeholders unchanged
Transform placeholders to BigPipe placeholders
…
HtmlResponseAttachmentsProcessor
Render placeholders
Update HTML to load final assets, set attached headers on response …

Explanation

Some parts are greyed out because they're of lesser importance: they do happen, but are not crucial for understanding the flow.

Example: **GET /admin**.

Typically, the format will be **html**, but it could also be **ajax, dialog, modal, json, hal_json, and more.** See Symfony's Request::setFormat().

Example: **/admin path → system.admin route**
That route has the following attributes:
_content: '\Drupal\system\Controller\SystemController::systemAdminMenuBlockPage'
_title: 'Administration'

_controller is set to '\Drupal\system\Controller\SystemController::systemAdminMenuBlockPage'

\Drupal\system\Controller\SystemController::systemAdminMenuBlockPage is called, it returns a render array with no #type.

MainContentViewSubscriber only handles render arrays! If it's anything else (e.g. an object), then it's up to another VIEW event subscriber to turn it into a Response.

MainContentViewSubscriber looks at the available main content renderer services. These are **tagged with render.main_content_renderer.** Contributed modules can add additional renderers.

The flow for all four formats supported in core in their respect is displayed next, but the **explanations below only apply to the HtmlRenderer.**
If an unsupported format was negotiated, a 406 response is generated.

Example: **BlockPageVariant** selected.

The main content must already be rendered, because it might set the page title.

The result of a built page display variant is a #type 'page' render array.

hook_page_attachments() operates on #type 'page'.

Only accepts render arrays of #type 'page'!

Renders the 3 parts of the HTML body: page_top, page, page_bottom (where page corresponds to #type 'page' and hence page.html.twig). **Then** renders the entire HTML (i.e. the html.html.twig template).

Note that HtmlResponse (as well as AjaxResponse) at this point still contains attachments: asset libraries, headers, … but also placeholders.

ChainedPlaceholderStrategy looks at the available placeholder strategies. These are **tagged with placeholder_strategy.** Contributed modules can add additional renderers.

Placeholder strategies can transform standard Drupal placeholders (designed for non-deferred rendering) into other placeholders, which may be replaced with the final markup using any way possible. For example: BigPipe, ESI …

BigPipe is a contrib module during 8.0.x, will likely be in 8.1.x core. See https://www.drupal.org/project/big_pipe

Process all other HtmlResponse attachments: asset libraries, HTML <head> elements, headers …

BigPipe placeholders rendered at the end of ::send(), after most is sent.

Example response for the path **/admin**.

Start At The Beginning

When All You Have Is A Hammer

CC by 2.0 https://www.flickr.com/photos/raindrift/7095238893/

Text Editors

- “Text Editors”
- Text Editors
- IDEs

“Text Editors”

- Notepad (Windows)
- Text Edit (Mac)
- Nano (Linux / Mac)

Text Editors

- Notepad++ (Windows)
- Dreamweaver (Mac / Windows)
- Sublime Text (Mac / Linux / Windows)
- Atom (Mac / Linux / Windows)
- Microsoft VS Code (Mac / Linux / Windows)
- vim (Mac / Linux)

Integrated Development Environments

- PHPStorm (Mac / Linux / Windows)
- Eclipse (Mac / Linux / Windows)
- Komodo (Mac / Linux / Windows)

Operating Systems With Built In Text Editors

- emacs (Mac / Linux / Unix / Everything else)

Development Environments

- On your server
- On your machine
- On your machine, but it acts like the server

Development Environments

- FTP Development
  - “Cowboy coding”
- Local Environments
  - MAMP / LAMP / Vagrant
- Next generation
  - Kalabox / Lando CLI (And other Docker setups)
  - DrupalVM (And other Vagrant setups)

FTP Development

- FTP Development
  - “Cowboy coding”
  - No version control up front
  - Little debugging visibility
    - Log files (maybe), debug statements
    - Watchdog logs (maybe)

Debug Statements in Code

- print_r()
  - Raw dump of a variable or context, no formatting
  - print_r($node);
    - <pre><?php print_r($node); ?></pre>
  - dsm() / kint()
    - Formatted dump of a variable or context
    - dsm($node);

Local Development

- MAMP / Dev Desktop / Native LAMP
  - Code runs entirely locally
  - Added bonus of offline work, no internet required
- print_r(), dsm(), kint(), but also more
- xdebug

Local Development: The Next Generation

- Kalabox / Lando with Pantheon / Acquia integration
- DrupalVM and other Vagrant boxes
- print_r(), dsm(), kint(), xdebug
- Very accurate performance profiling
  - XHProf
  - Blackfire
  - Tidewater

What's xdebug?

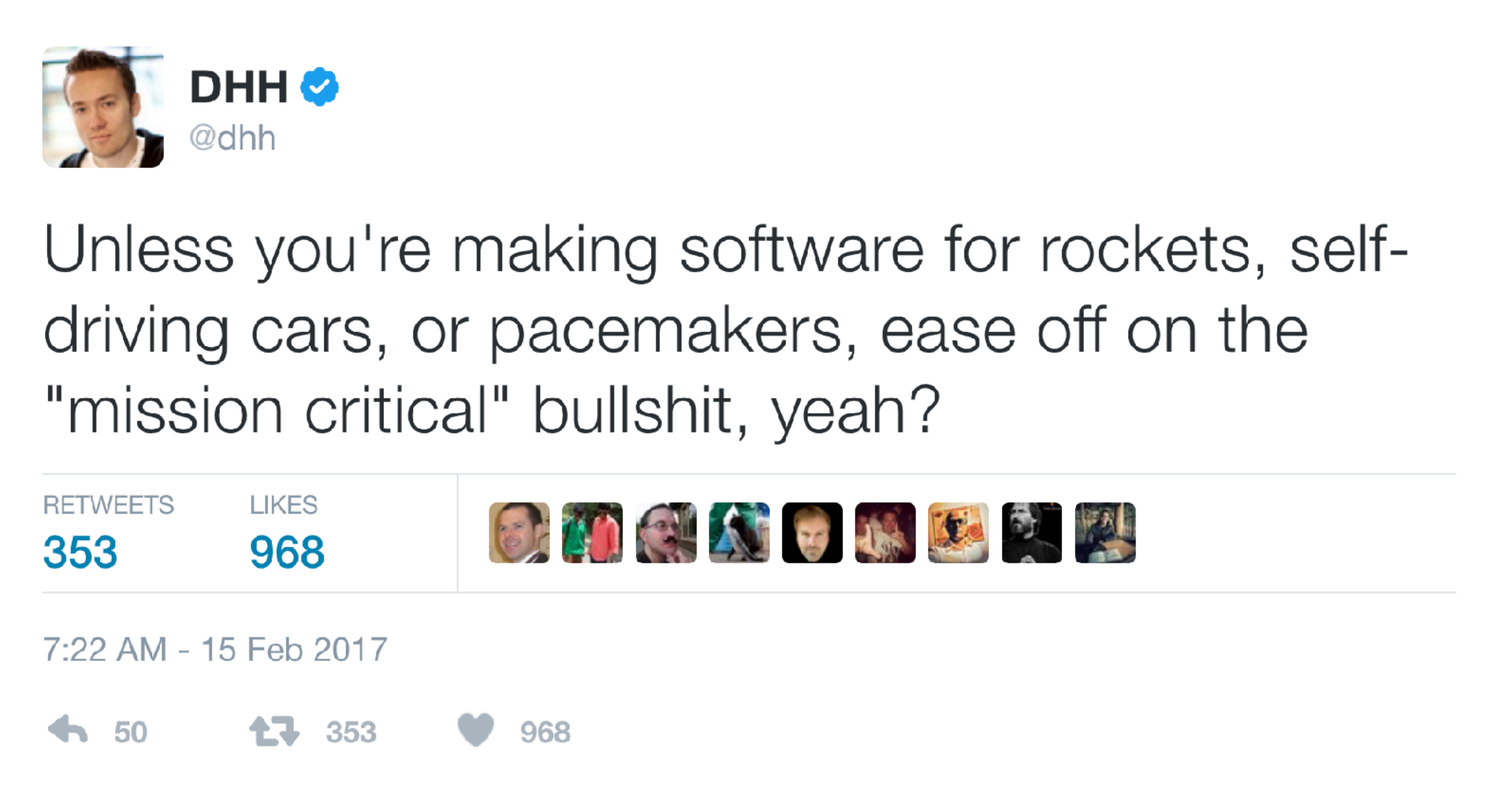
xdebug:
Step Through Debugging

- Hitting pause on code execution
- Inspecting the current state of “the stack”
- Modifying values *live*

The Ever Dangerous Live Demo

Something Is Broken!

Relax.



DHH ✓
@dhh

Unless you're making software for rockets, self-driving cars, or pacemakers, ease off on the "mission critical" bullshit, yeah?

RETWEETS
353

LIKES
968

7:22 AM - 15 Feb 2017

↩ 50 ⇄ 353 ♥ 968

Relax.

Remember Cobble's Knot

What’s Broken?

- Is something not showing up?
  - New content - is it published?
  - Old content - are permissions set properly?
- Is something showing up that shouldn’t?
  - Raw HTML and Javascript in a WYSIWYG field?

Work From The Bottom Up

- Log Files
  - You should learn how to find your log files before you need your log files
- multitail
  - Linux / Mac utility to easily view logs with more options than tail

Where Is It Broken?

- Custom Module
- template.php / .theme file
- Theme Template

The Scientific Method

Finally, Debugging

- Change ***one*** thing at a time
- Test that change
- Repeat

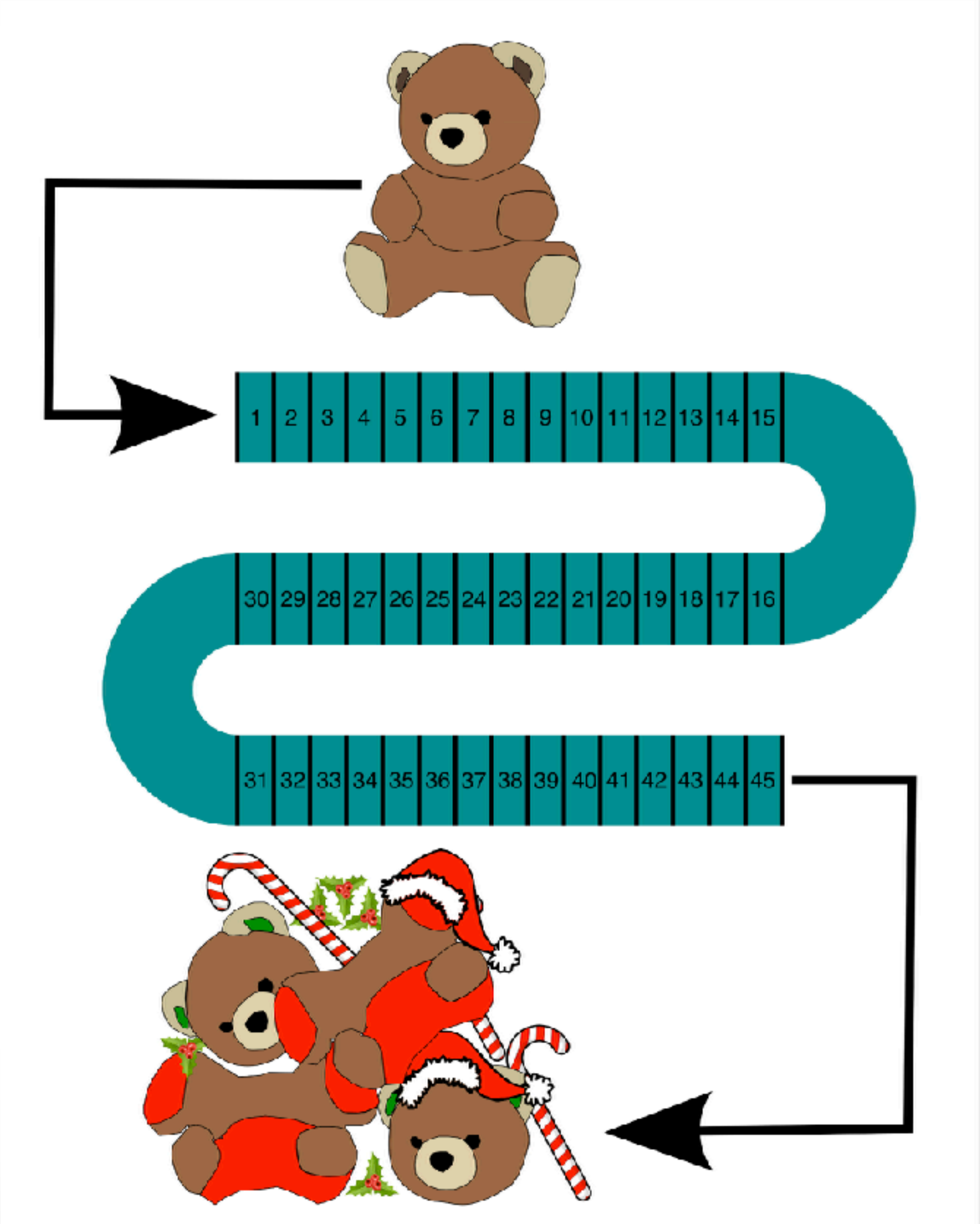
git Is Your Friend

- Save your progress as you work
  - Recreate your Features or config yml
- Makes rabbit holes manageable

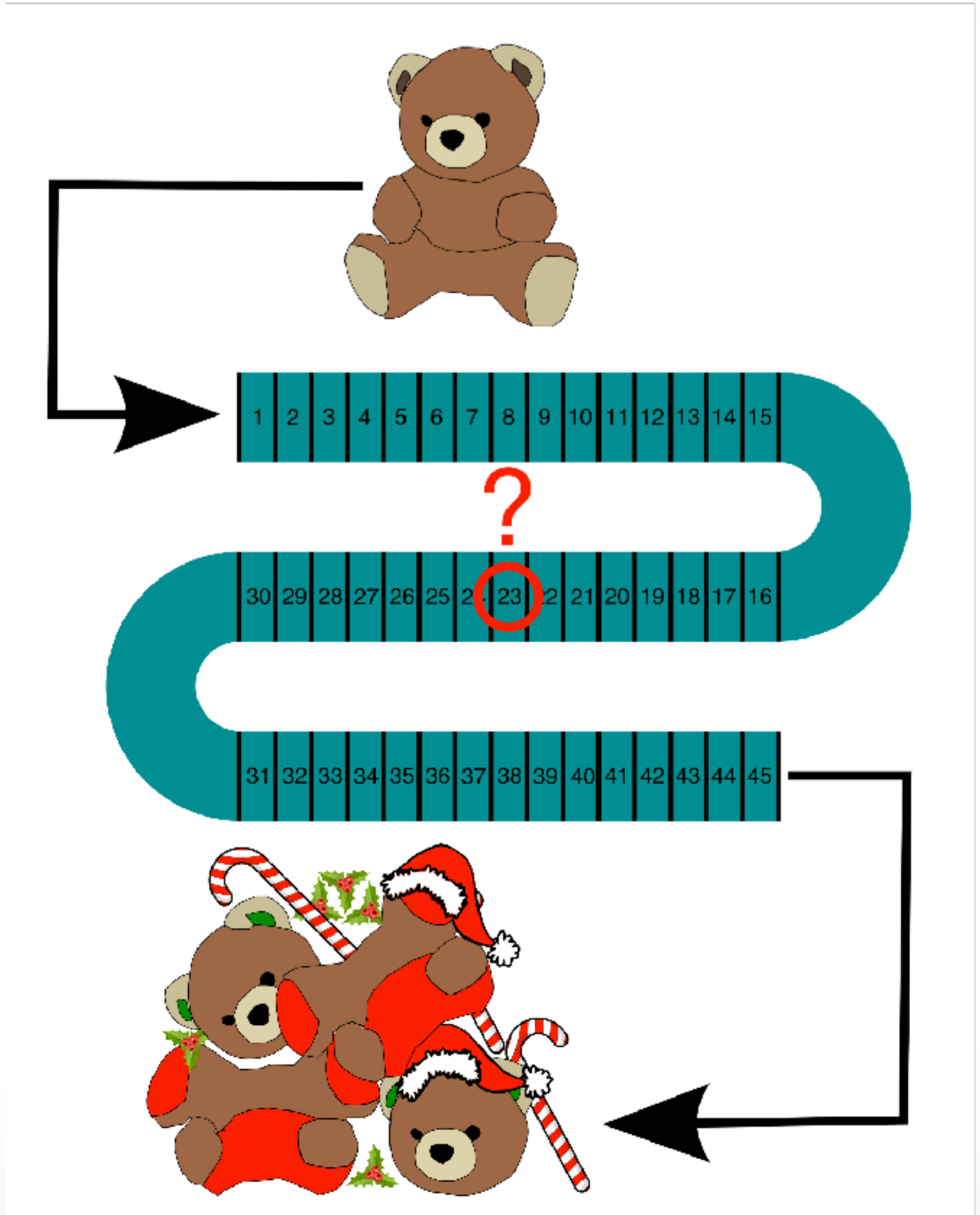
git bisect Is Your Friend

- Let’s you isolate changes between commits

git bisect Is Your Friend



git bisect Is Your Friend



git diff Is Your Friend

- Remove your debug statements
- Ensure you only changed **as much** as you needed to change
- You only commit **print_r('Butts')**; to master ***once***

git blame Finds Your Enemies

git blame ~~Finds Your Enemies~~

git ~~blame~~ Is *Your* Friend

git annotate Is *Your* Friend

- Can help you track down who wrote the offending code
  - This ***should not*** be a witch hunt
  - This ***should*** be a chance to find context for the issue

Other Debugging Methods

Debugging Techniques

- Walk away. Seriously.
- Talk to a coworker.
  - Talk to a rubber duck.
- Write it down. By hand.

Make The Future Easier

Proactive Debugging

- watchdog() (D7)
- \Drupal::logger() (D8)
- syslog Module
  - https://www.elastic.co/blog/heya-elastic-stack-and-x-pack
  - https://loggly.com
  - https://www.sumologic.com
- Write a test!
  - Simpletest / PHPUnit / Behat

Further Reading

- **The Art of Troubleshooting** by Jason Maxham
  - https://artoftroubleshooting.com/book/
- **Debugging: The 9 Indispensable Rules for Finding Even the Most Elusive Software and Hardware Problems** by David Agans
  - http://debuggingrules.com
- **Debugging During Development in Drupal 8** by Allie Jones
  - https://www.gitbook.com/book/zivtech/debug-tools-for-drupal8/
- **Using the PhpStorm Debugger** by JetBrains
  - https://confluence.jetbrains.com/x/_idqAw
- **Maniac Magee** by Jerry Spinelli
  - https://www.worldcat.org/oclc/20422223

Thanks!
Questions?
War stories?

@milsyobtaf

https://github.com/milsyobtaf/prez