Creating & Consuming APIs

Beau Lebens / beau@automattic.com / @beaulebens

AUTOMATTIC

I. API Design
II. Basic Tools
III. Requesterlude
IV. Consuming APIs
V. Customizing WP APIs
VI. Building with APIs

CHAPTER I

API Design

API
JSON
“Programmatic access to
content/services in a universal format
via simple HTTP requests”
REST

SOAP

XML-RPC

REST

Request / Response
Client / Server

HTTP Methods

http://the.internet

GET	Request one or many resources.
PUT	Modify an existing resource.
POST	Create a new resource.
DELETE	Delete an existing resource.

Idempotency

“

clients can make that same call repeatedly while producing the same result

http://www.restapitutorial.com/lessons/idempotency.html

”

GET	Idempotent
PUT	Idempotent
POST	NOT idempotent
DELETE	Idempotent...ish

Data Formats

JSON

Standards where possible

Use strict formats

Content Negotiation

VS

File Extensions

Accept: application/json

vs

https://.../something.json

Naming

example.com/api

VS

api.example.com

/collection/instance[...]

/users/

GET, POST

/sites/123

GET, PUT, DELETE

/sites/123/comments/123

Pagination / Filtering / Ordering

?offset=100&number=10

?filter[page]=10&filter[posts_per_page]=10

?page=10&posts_per_page=10

?search=

?filter[search]=

?s=

?order_by=title&order=ASC

?filter[orderby]=title&filter[order]=ASC

?orderby=title&order=ASC

HTTP Status Codes

200 OK

201 Created

204 No Content

304 Not Modified

400 Bad Request

401 Unauthorized

403 Forbidden

404 Not Found

429 Too Many Requests

500 Internal Server Error

Authentication

HTTP Basic

OAuth

Token/Secret

Send it as a header,
not in the querystring

HTTPS Everywhere

Versioning

/v2.1/posts/123

vs

/posts/123?version=2.1

vs

Accept: application/json; version=2.1

Testing

Documentation

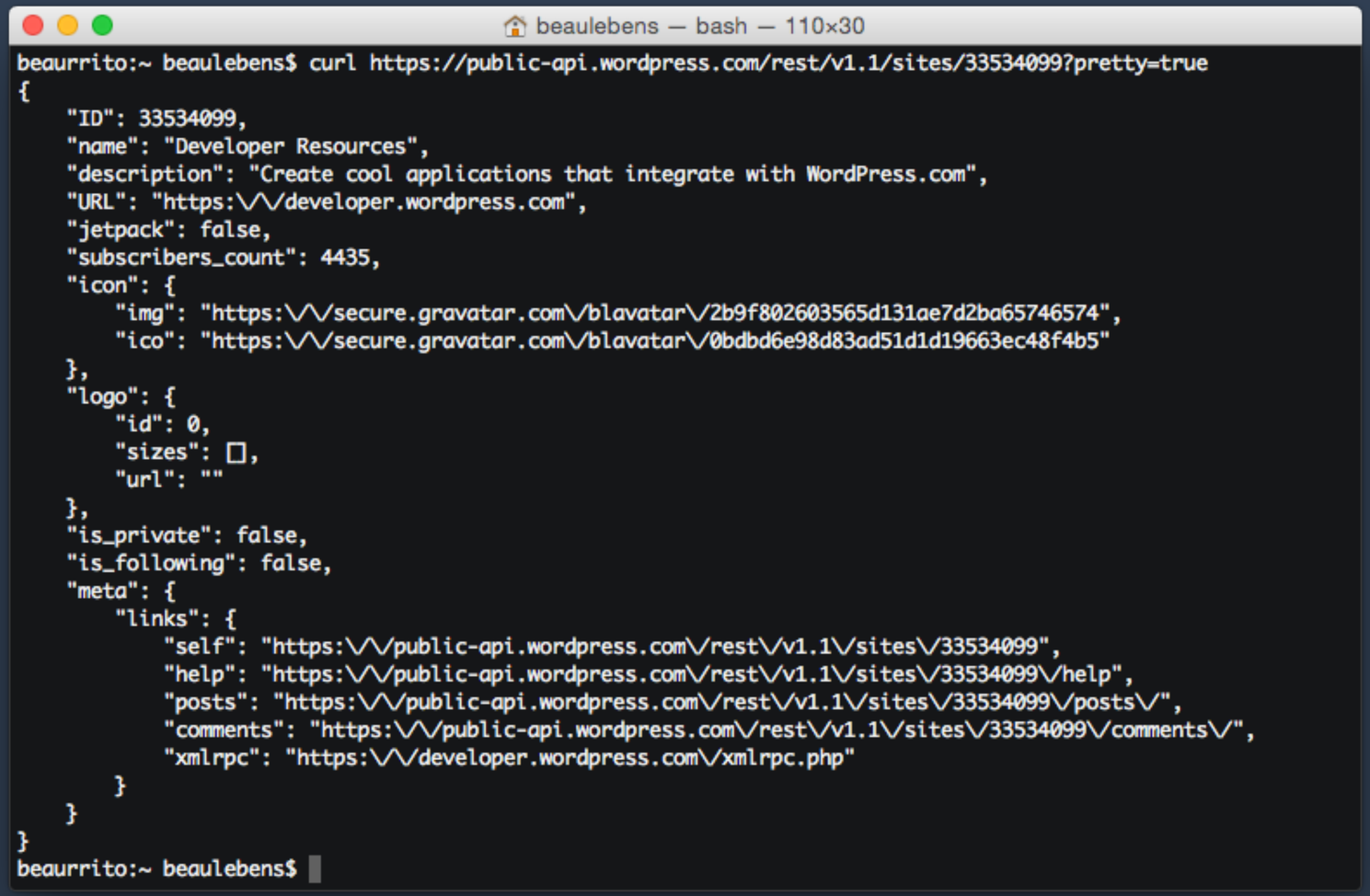
Example Requests/Responses

Standardized
Consistent
Guessable

CHAPTER II

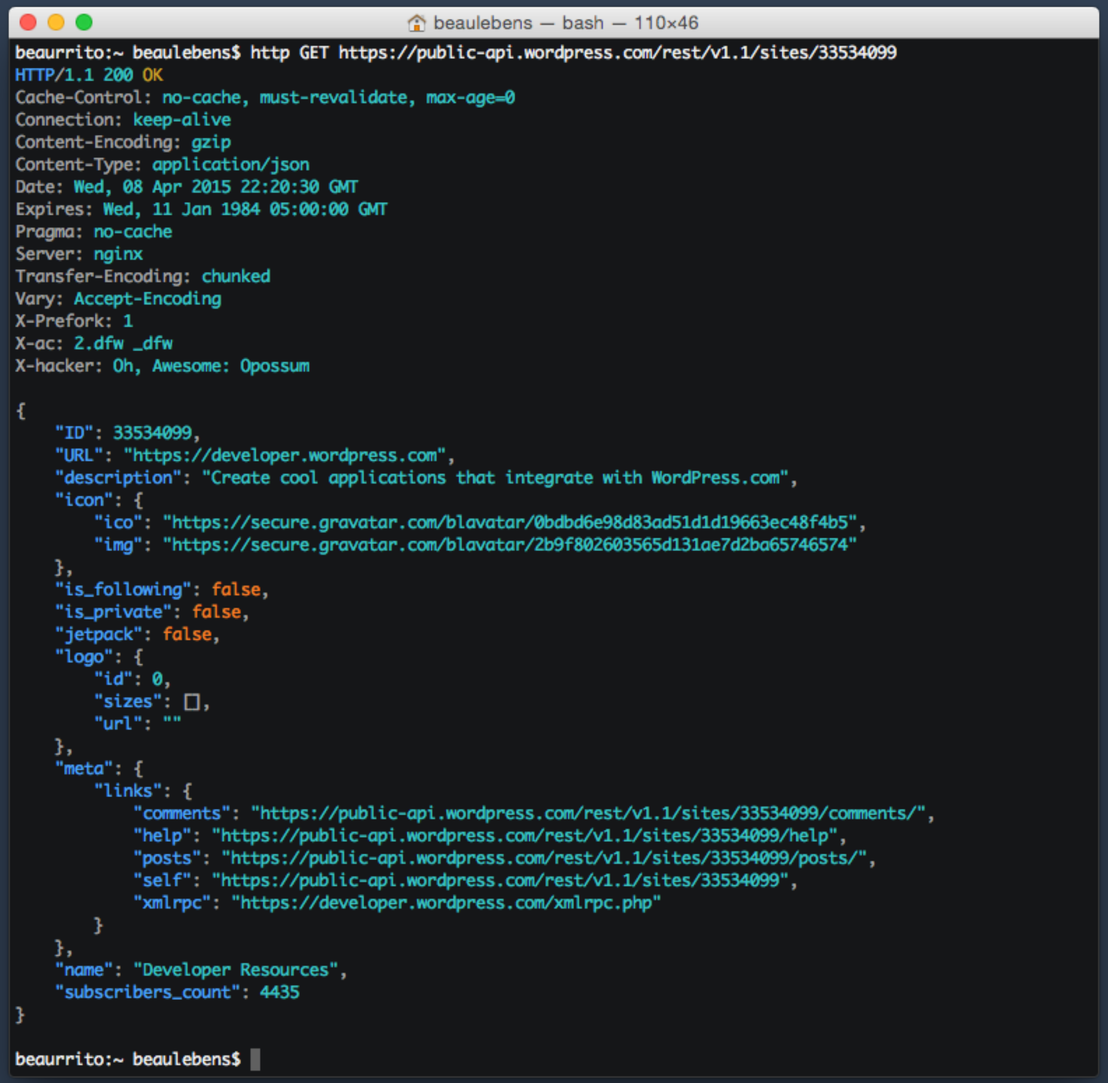
Basic Tools

curl



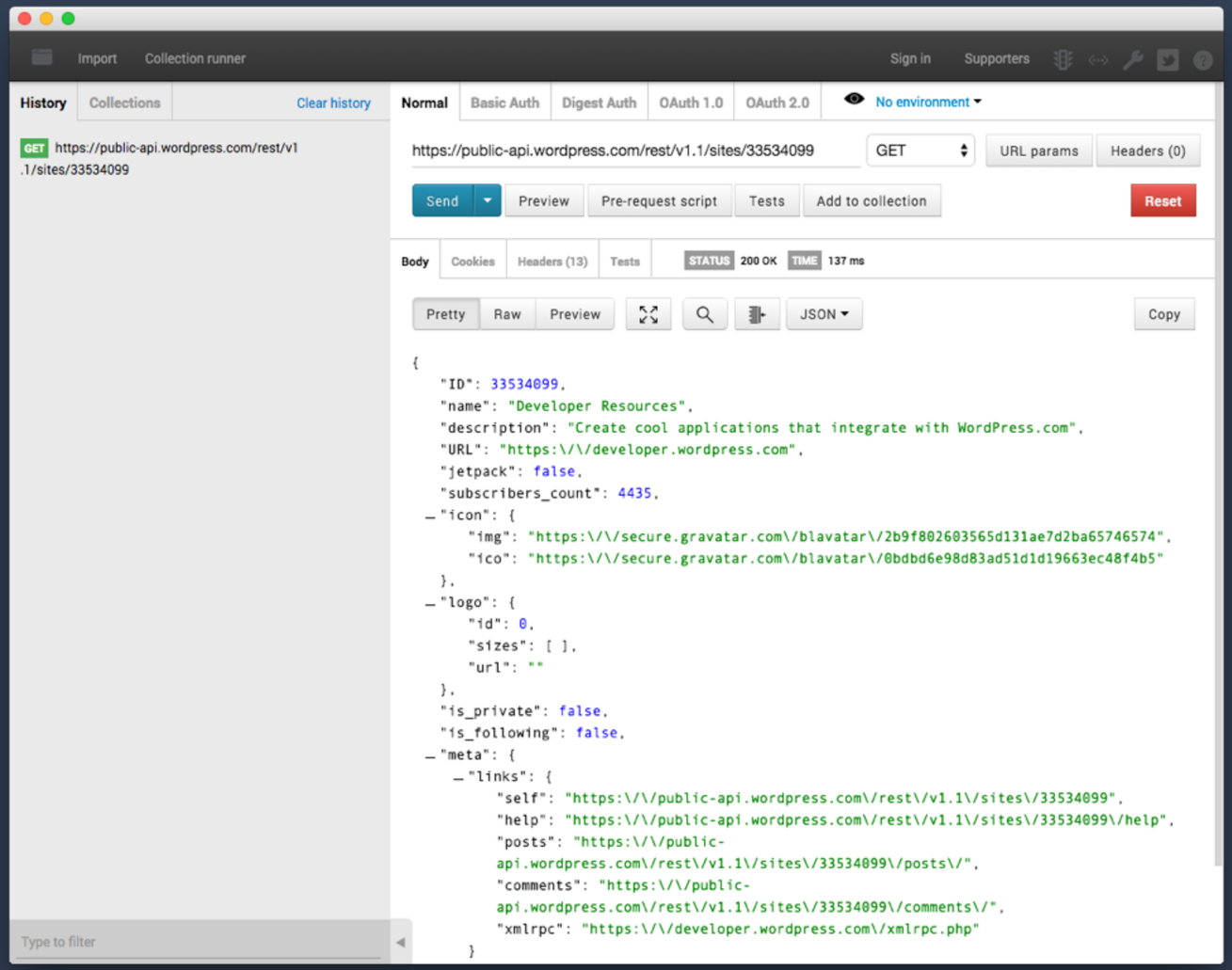
/usr/bin/curl

HTTPie



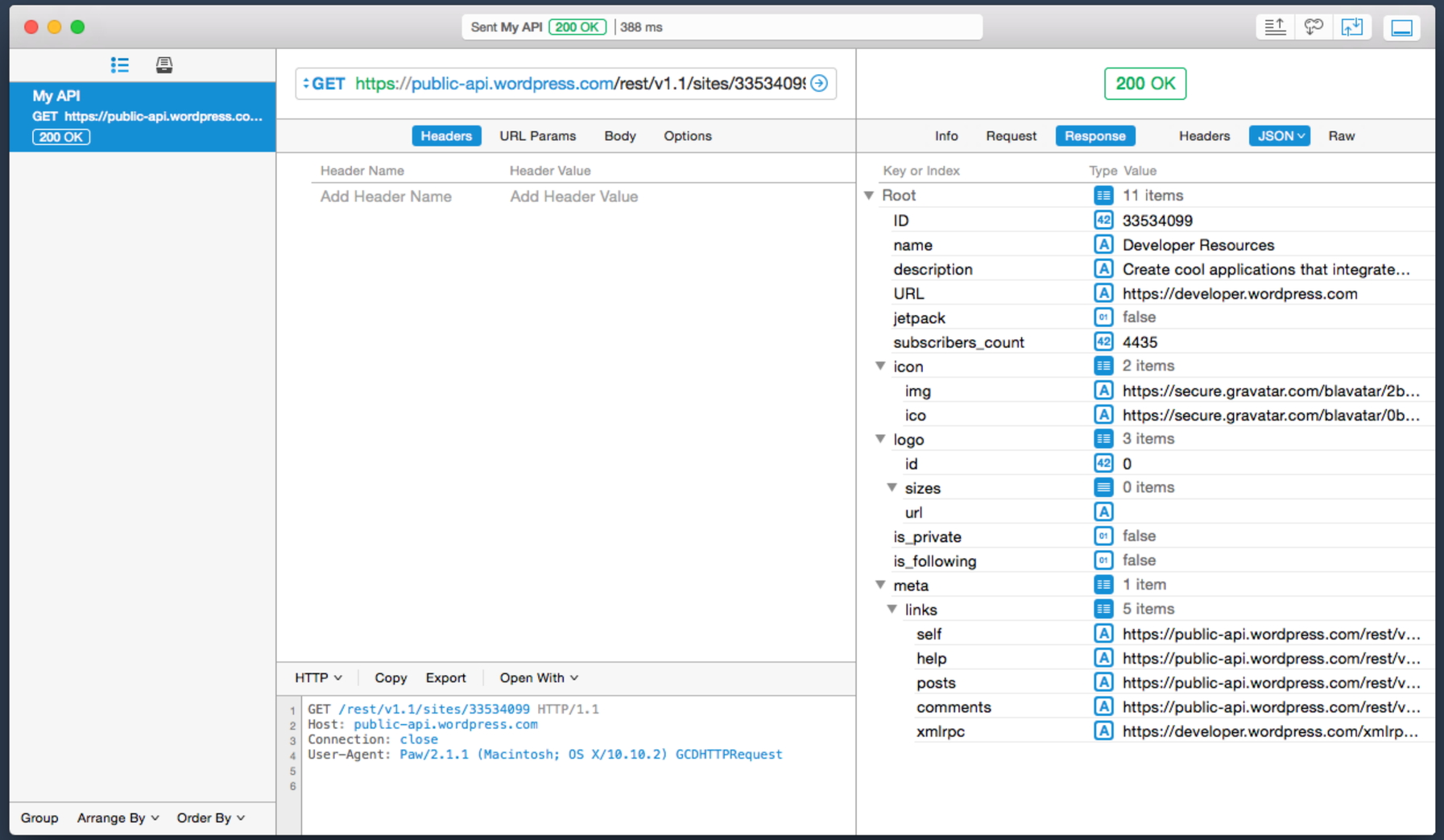
http://httpie.org

Postman



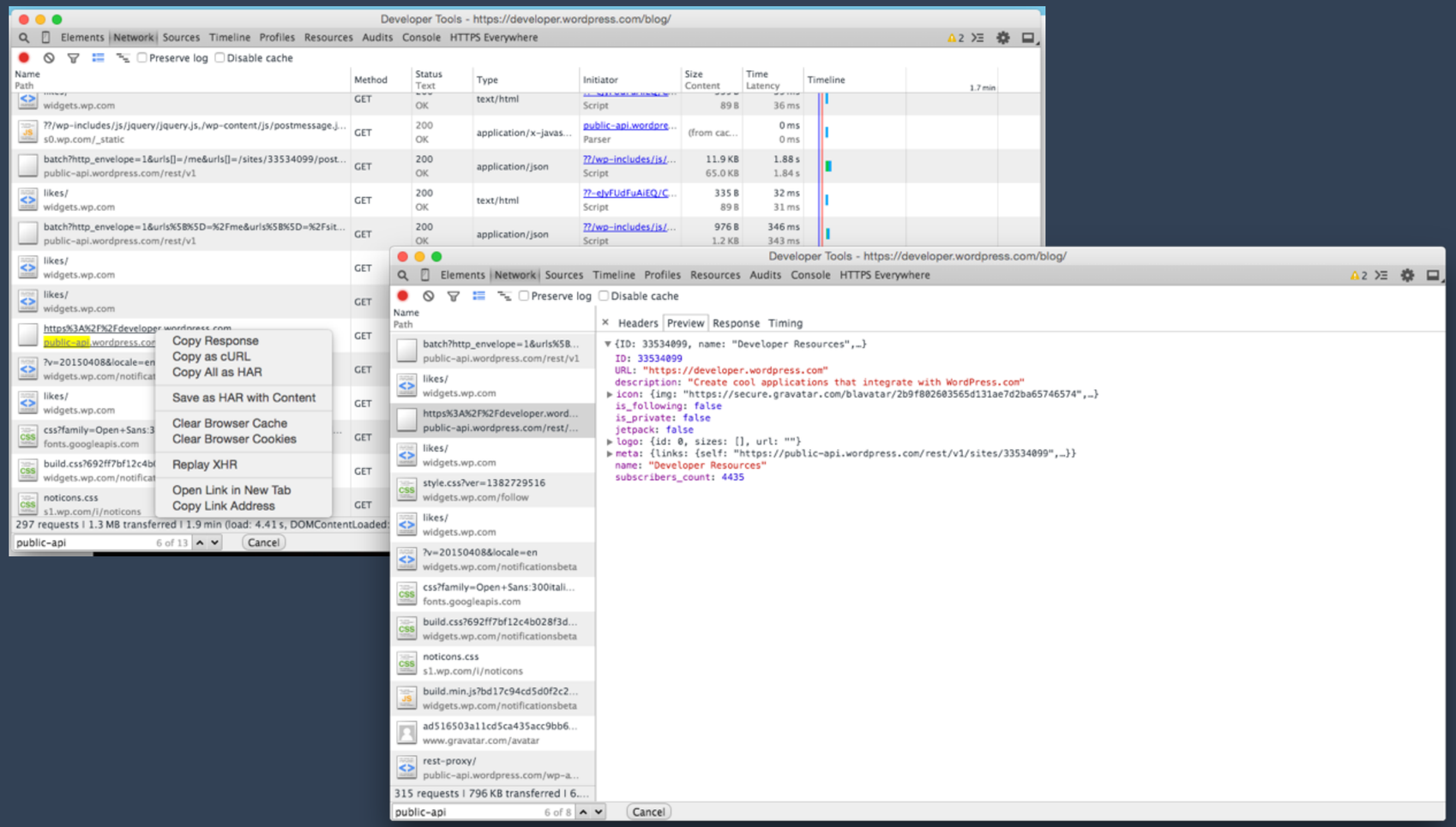
https://www.getpostman.com/

Paw



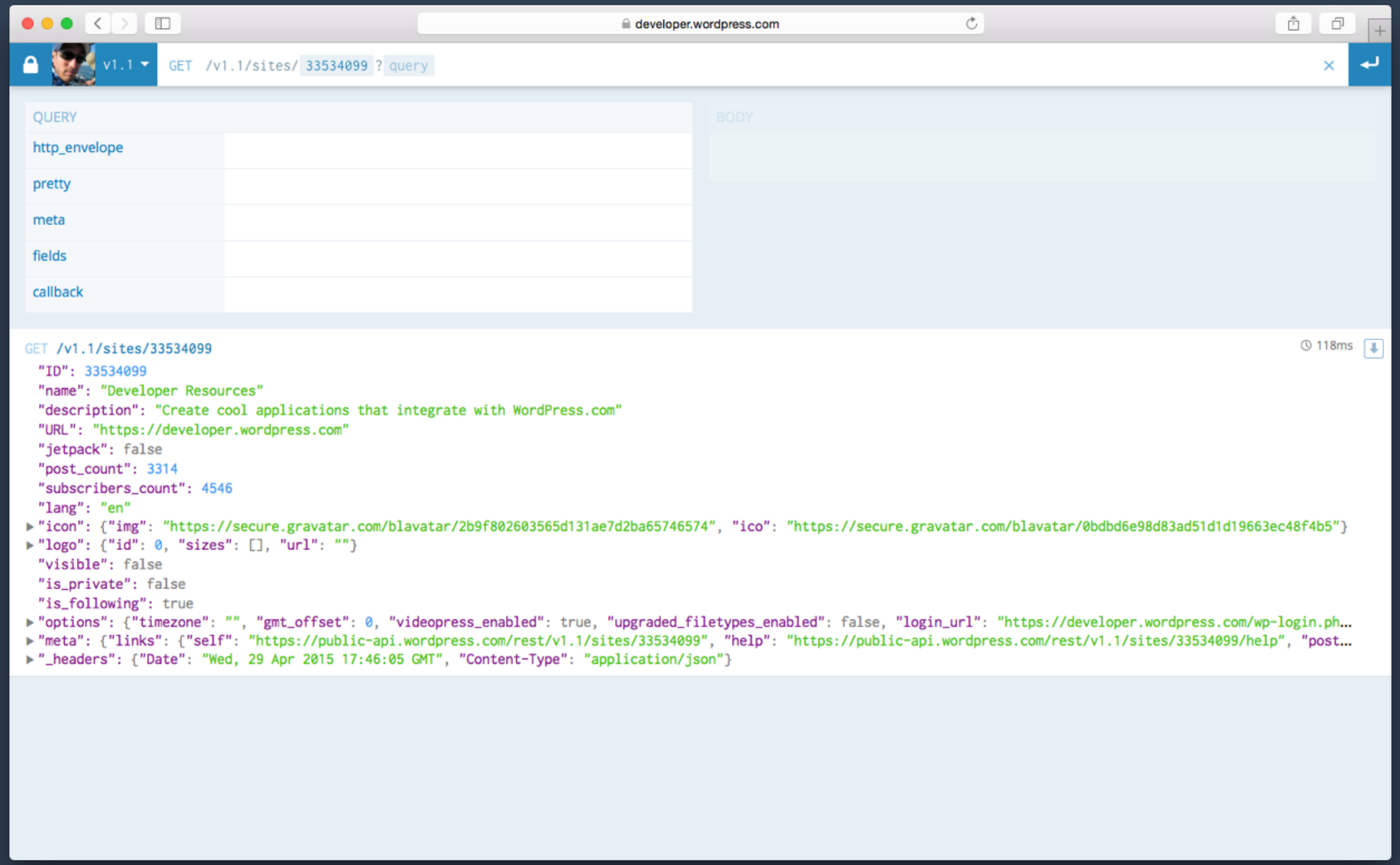
https://luckymarmot.com/paw

Chrome DevTools



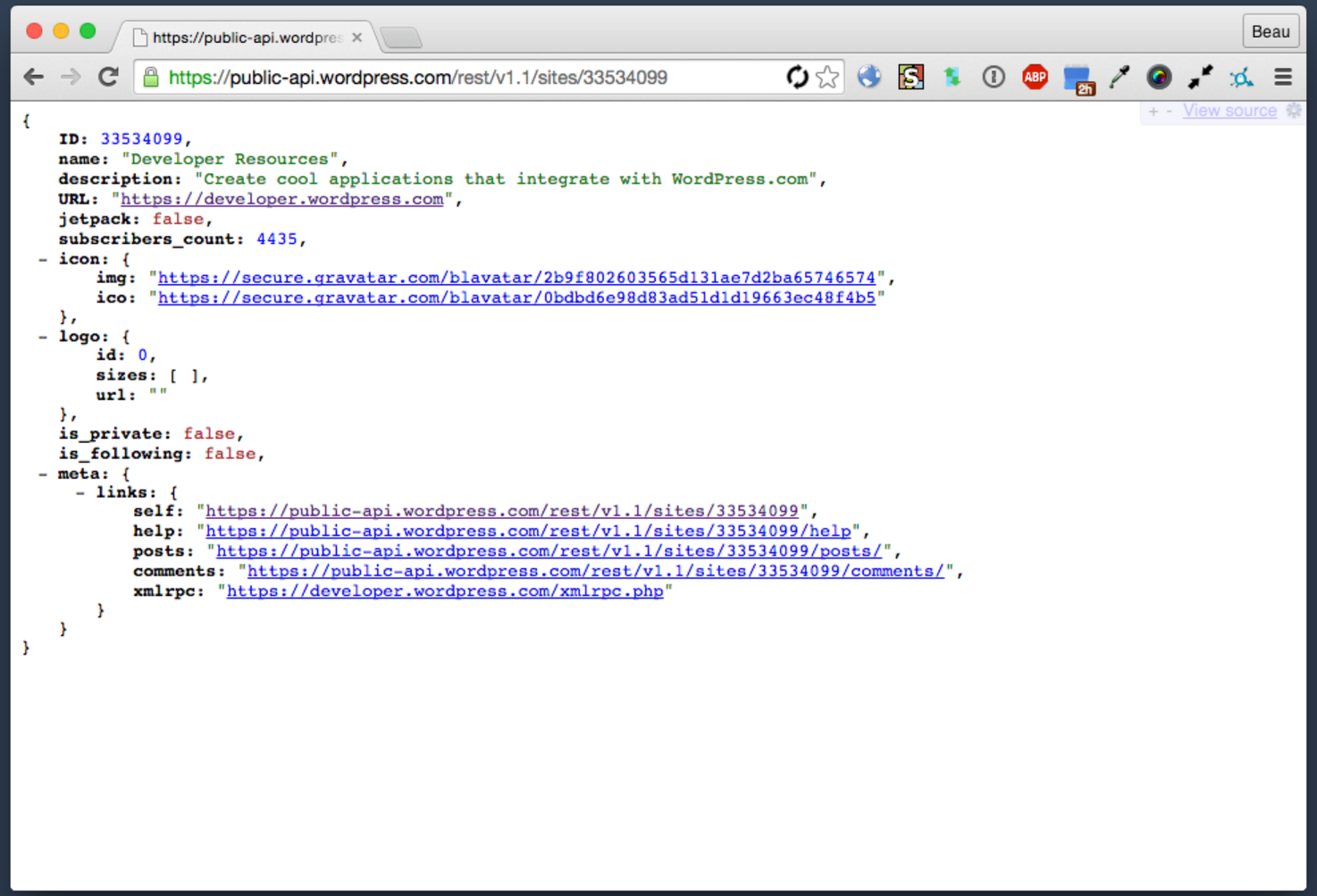
https://google.com/chrome

WP.com Developer Console



https://developer.wordpress.com/docs/api/console/

JSONView



http://jsonview.com/

CHAPTER III

Requesterlude

Use the tool of your choice

https://developer.wordpress.com

http://wp-api.org

What is the “description” for TechCrunch?

https://developer.wordpress.com — http://wp-api.org

How many comments does the second most-commented post on The WordPress.com Developer Blog have?

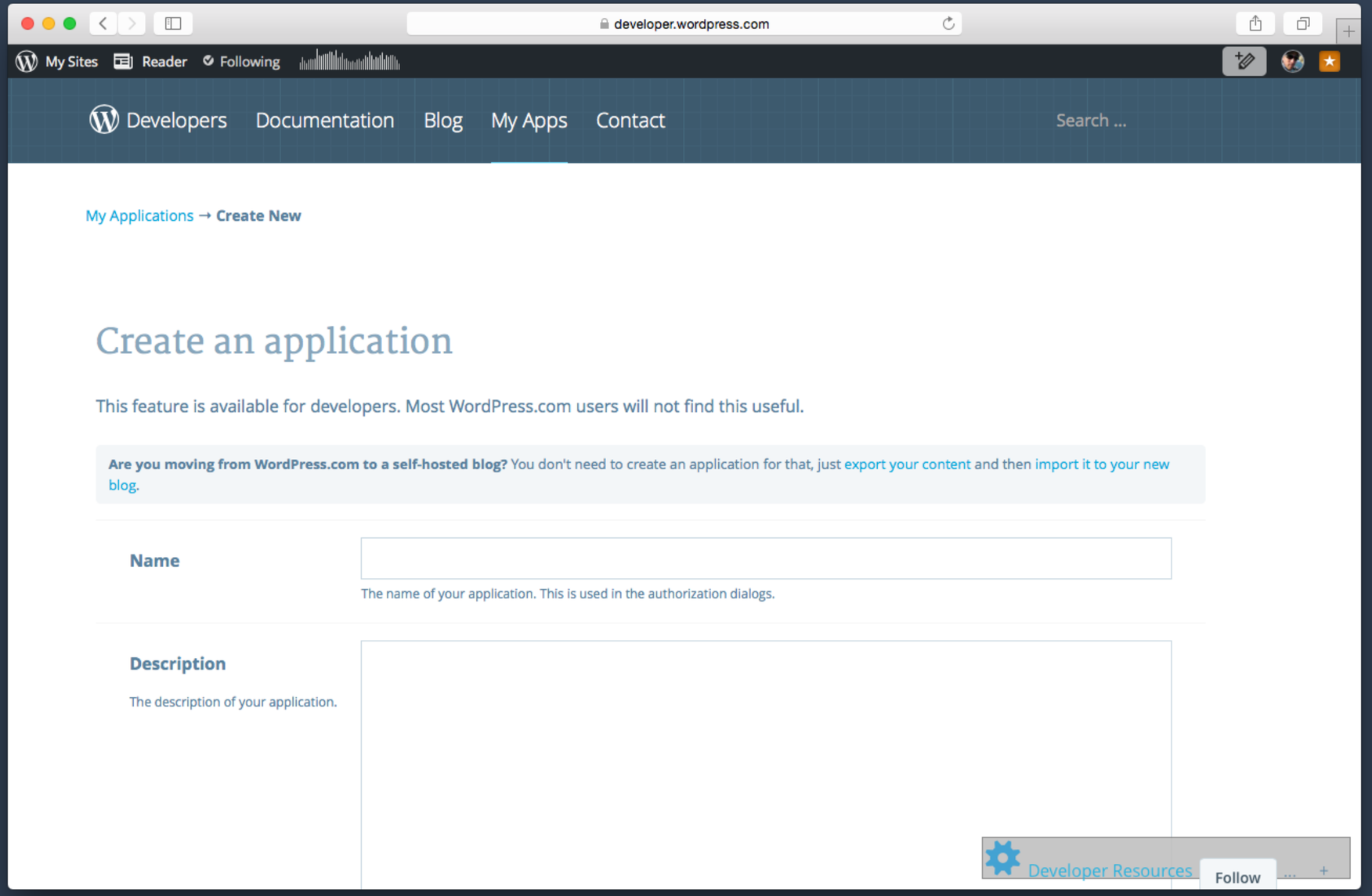
Who is the author of the first comment returned for post ID 44 on the “fishing” WP install on your VM?

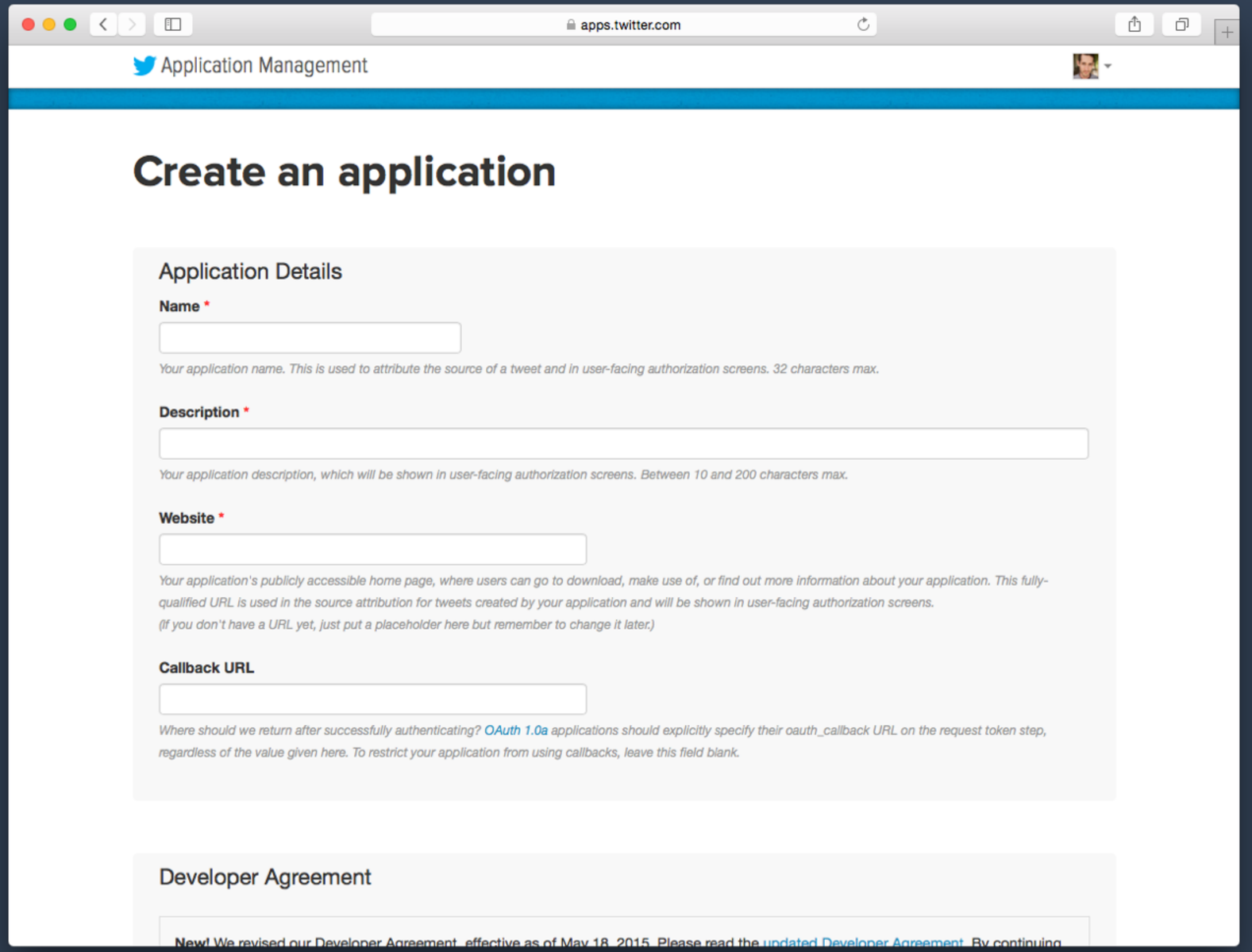
http://vip-workshop.dev/rest-api/fishing/

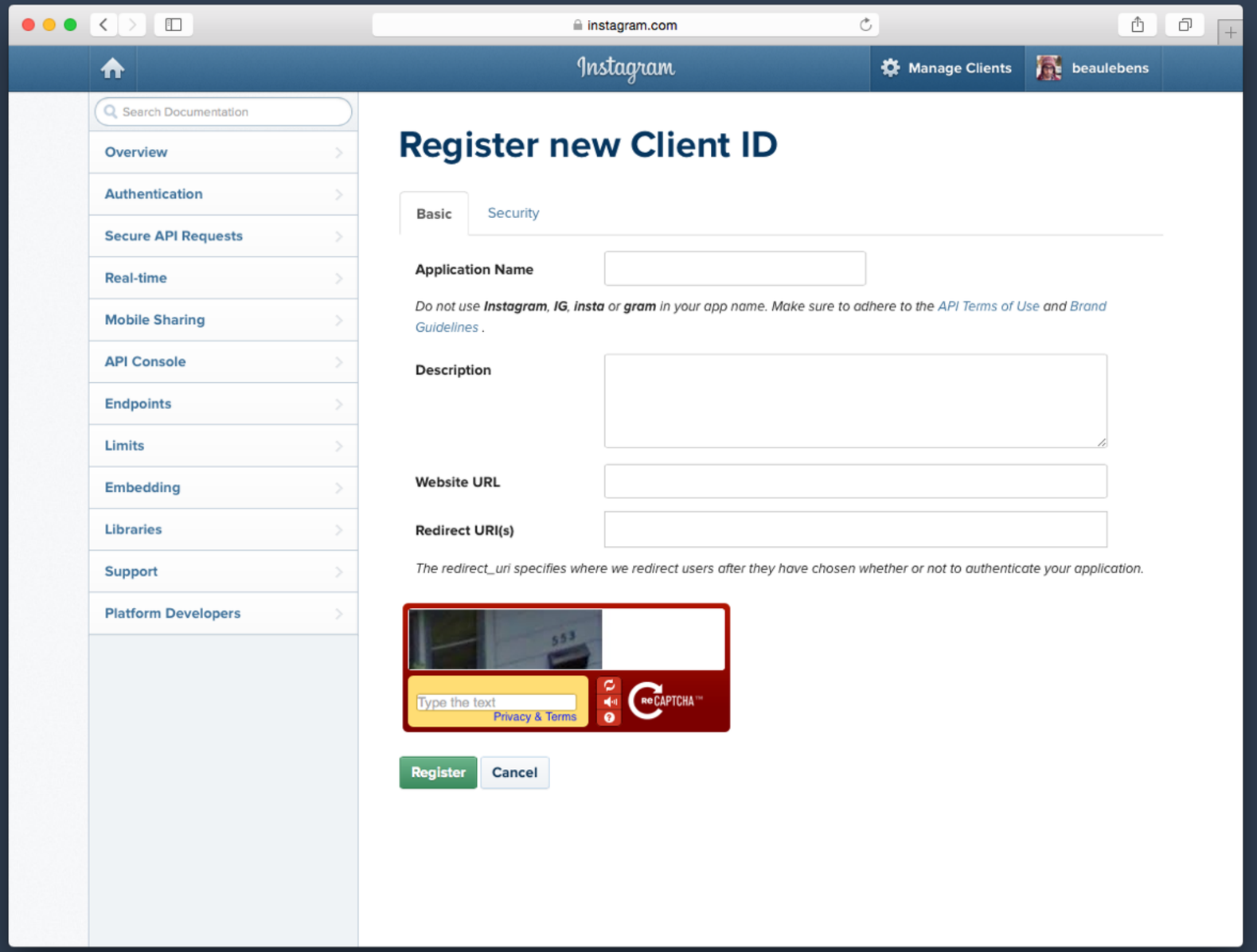
CHAPTER IV

Consuming APIs

Application Registration


developer.wordpress.com
My Sites
Reader
Following
Developers
Documentation
Blog
My Apps
Contact
Search ...
My Applications → Create New
Create an application
This feature is available for developers. Most WordPress.com users will not find this useful.
Are you moving from WordPress.com to a self-hosted blog? You don't need to create an application for that, just export your content and then import it to your new blog.
Name
The name of your application. This is used in the authorization dialogs.
Description
The description of your application.
Developer Resources
Follow


apps.twitter.com
Application Management
Create an application
Application Details
Name *
Your application name. This is used to attribute the source of a tweet and in user-facing authorization screens. 32 characters max.
Description *
Your application description, which will be shown in user-facing authorization screens. Between 10 and 200 characters max.
Website *
Your application's publicly accessible home page, where users can go to download, make use of, or find out more information about your application. This fully-qualified URL is used in the source attribution for tweets created by your application and will be shown in user-facing authorization screens.
(If you don't have a URL yet, just put a placeholder here but remember to change it later.)
Callback URL
Where should we return after successfully authenticating? OAuth 1.0a applications should explicitly specify their oauth_callback URL on the request token step, regardless of the value given here. To restrict your application from using callbacks, leave this field blank.
Developer Agreement
New! We revised our Developer Agreement, effective as of May 18, 2015. Please read the updated Developer Agreement. By continuing


instagram.com
Instagram
Manage Clients
beaulebens
Search Documentation
Overview
Authentication
Secure API Requests
Real-time
Mobile Sharing
API Console
Endpoints
Limits
Embedding
Libraries
Support
Platform Developers
Register new Client ID
Basic
Security
Application Name
Do not use Instagram, IG, insta or gram in your app name. Make sure to adhere to the API Terms of Use and Brand Guidelines .
Description
Website URL
Redirect URI(s)
The redirect_uri specifies where we redirect users after they have chosen whether or not to authenticate your application.
Type the text
Privacy & Terms
reCAPTCHA
Register
Cancel

wp_remote_(get|post|request)

Parallel / Asynchronous

Conditional Requests + Caching

ETag / If-None-Match

localStorage / memcache / DB / FS

Client Libraries

Authentication-only

~~Native PHP (OAuth 1.0a)~~

https://github.com/adoy/PHP-OAuth2

wordpress.org

WordPress.org

Search WordPress.org

Showcase Themes Plugins Mobile Support Get Involved About Blog Hosting

Download WordPress

Plugin Directory

Username Password Log in (forgot?) or Register

Developer Center

Search Plugins

Search

Popular Tags More »

widget (5,152)
Post (3,225)
plugin (3,156)
admin (2,683)
posts (2,467)
sidebar (1,979)
shortcode (1,858)
google (1,818)
twitter (1,763)
images (1,743)
page (1,705)
comments (1,699)
image (1,589)

Keyring

An authentication framework that handles authorization with external web services.

Download Version 1.6.1

Description Installation FAQ Changelog Stats Support Reviews Developers

See the Keyring Developer's Guide for more details.

Keyring provides a very hookable, completely customizable framework for connecting your WordPress to an external service. It takes care of all the heavy lifting when making authenticated requests, so all you need to do is implement cool features and not worry about these tricky bits.

Out of the box, Keyring currently comes with base Service definitions for webservices

Requires: 3.3 or higher
Compatible up to: 4.1.4
Last Updated: 2015-1-6
Active Installs: 1,000+

Ratings

passportjs.org

Passport

Home Guide

Passport

Simple, unobtrusive authentication for Node.js.

About Passport

Passport is authentication middleware for Node.js. Extremely flexible and modular, Passport can be unobtrusively dropped in to any Express-based web application. A comprehensive set of strategies support authentication using a username and password, Facebook, Twitter, and more.

Read the Guide »

Features

- 140+ authentication strategies
- Single sign-on with OpenID and OAuth
- Easily handle success and failure
- Supports persistent sessions
- Dynamic scope and permissions
- Pick and choose required strategies
- Implement custom strategies
- Does not mount routes in application
- Lightweight code base

```
$ npm install passport
```

Community

- GitHub
- Twitter
- Google Groups
- Stack Overflow
- IRC: #passport.js

Also

Locomotive
Powerful MVC web framework.

© 2011-2013 Jared Hanson
Logo by Stuart P. Bentley, CC BY-SA 3.0

API-specific

CHAPTER V

Customizing WP APIs

http://vip-workshop.dev/rest-api/mma/

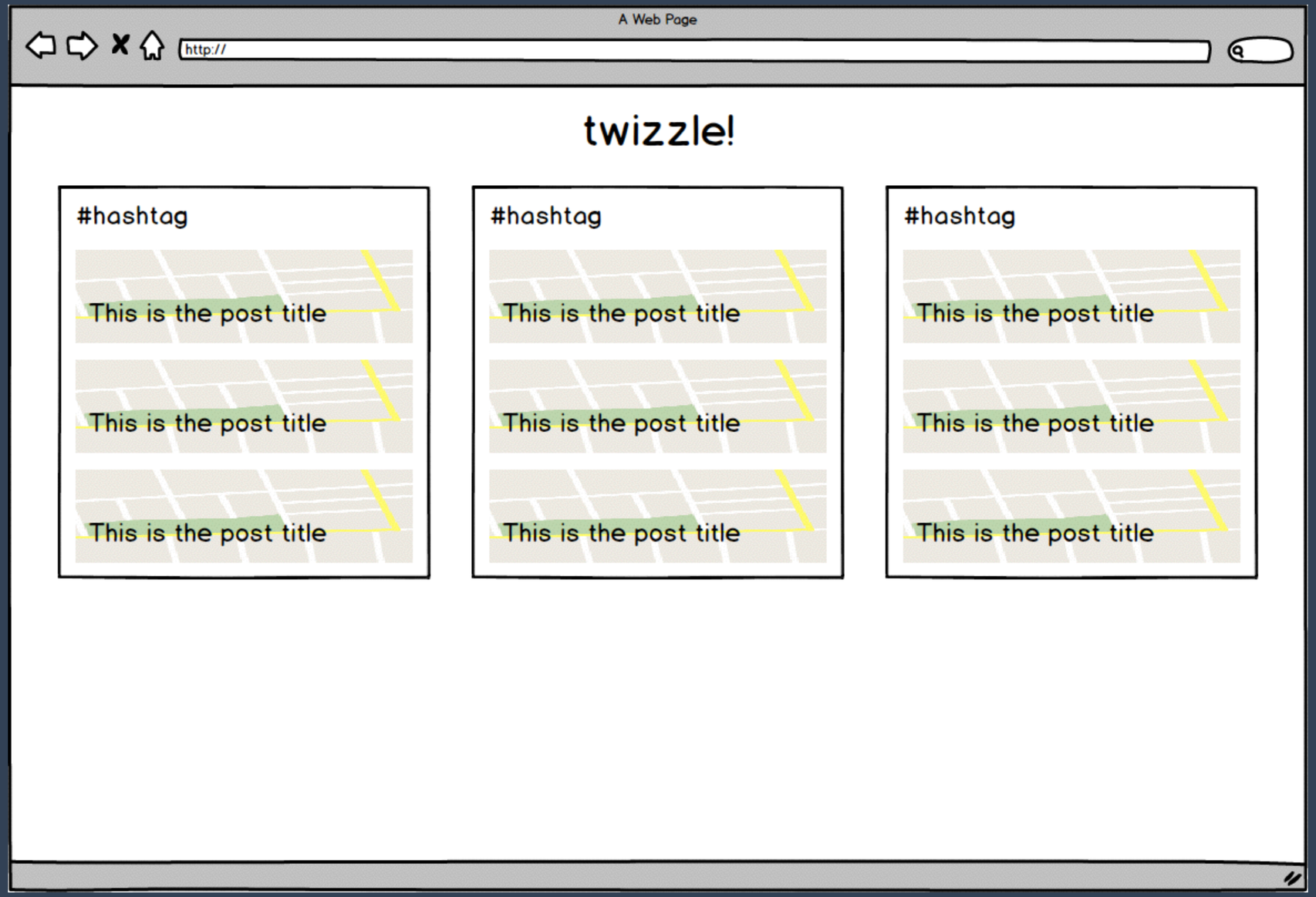
Custom Post Type
Custom Taxonomy
User Properties

CHAPTER VI

Building with APIs

twizzle

A card-based, visual UI for viewing posts according to their tags.


A Web Page
http://
twizzle!
#hashtag
This is the post title
This is the post title
This is the post title
#hashtag
This is the post title
This is the post title
This is the post title
#hashtag
This is the post title
This is the post title
This is the post title

```
git checkout tags/1.0
```

(from the host machine, in www/rest-api/twizzle/)

http://vip-workshop.dev:3030/

Request 10 posts from
http://vip-workshop.dev/rest-api/fishing/
and console.log()

```
git checkout tags/1.1
```

Get a list of unique tag slugs from the retrieved posts and console.log()

```
git checkout tags/1.2
```

Sort posts according to their tags and console.log()

```
git checkout tags/1.3
```

Render a card per tag,
with #tag as the title

```
git checkout tags/1.4
```

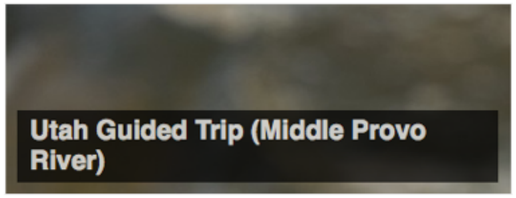
Render posts into each tag card

```
git checkout tags/1.5
```

localhost

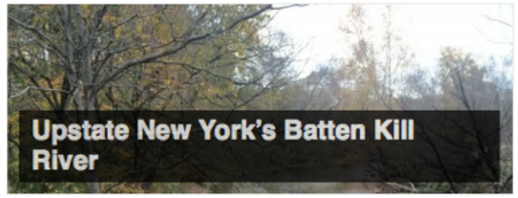
twizzle!

#catch

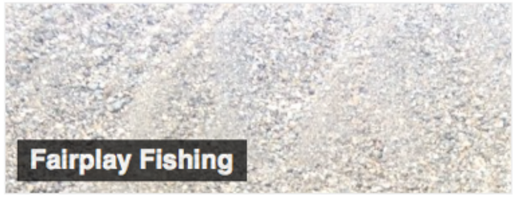


#japan

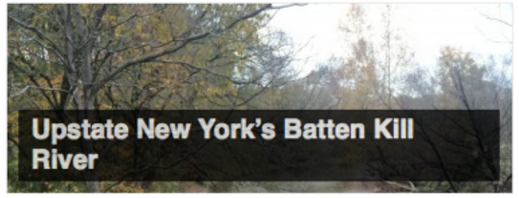
#upstate-new-york



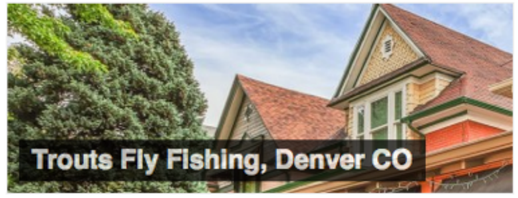
#fairplay



#hornbeck



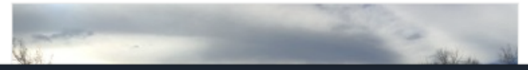
#shop



#store

#middle-boulder-creek

#south-platte



Data Layer?
localStorage Caching?
Query WP.com instead?
Get author/Gravatar of posts?
Parallel requests?

Modify tags via drag/drop?
Create post within card?

Beau Lebens

beau@automattic.com / @beaulebens

AUTOMATTIC