

# Lessons Learned from a Re(write|design)

Diving into the world of feature-flipping, performance optimization and removing of technical debt.

# About me

**Hi, I'm Adam McKerlie.**

I'm a python developer at G Adventures. I like good beer, optimizing SQL queries and making stuff run.

You can contact me at

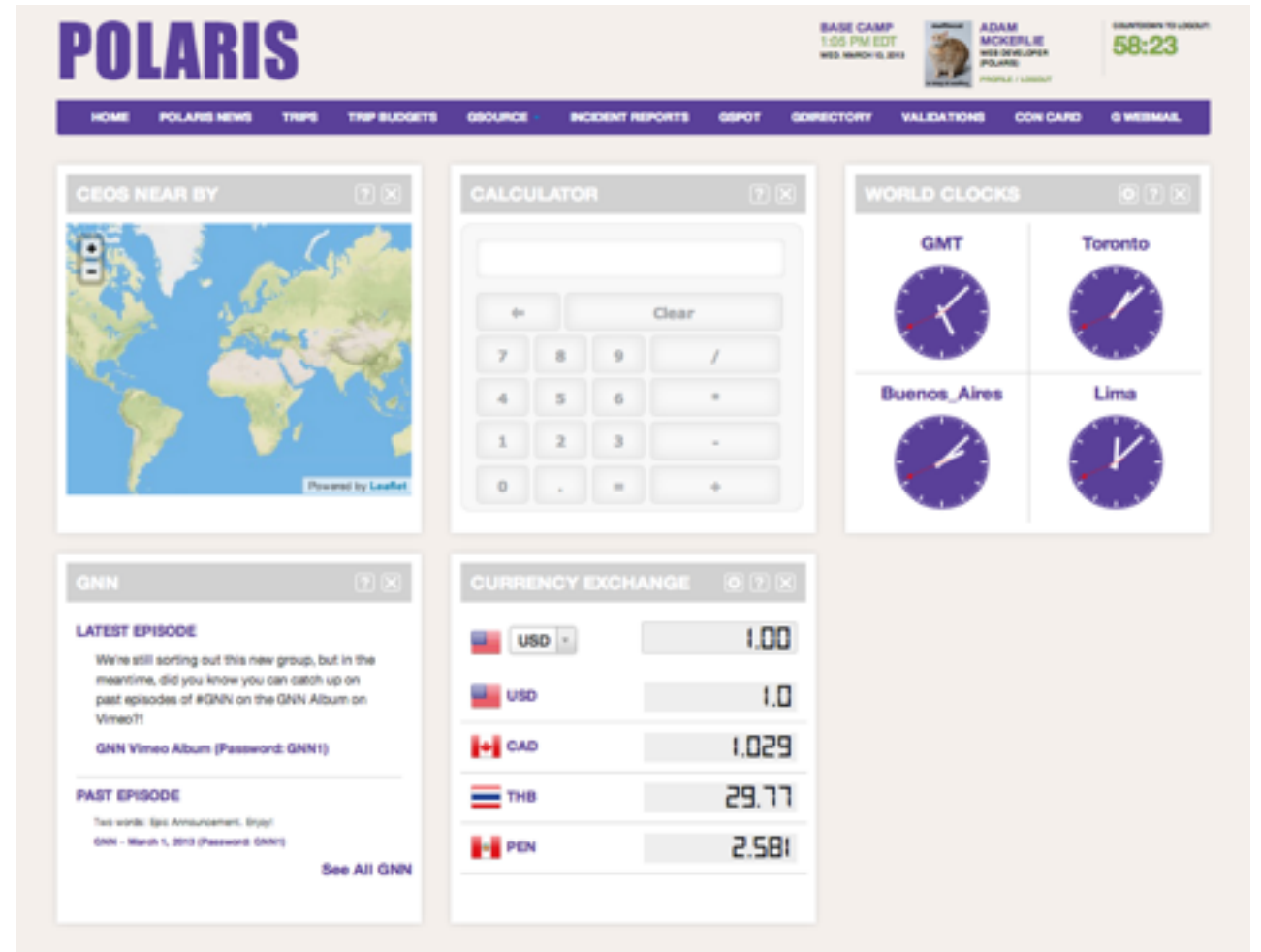
**Twitter** [@adammckerlie](#)

**IRC** [silent1mezzo](#)

**Web** [procrastinatingdev.com](#)



# Then vs Now



Adam McKerlie

3

Wednesday, March 20, 13

- Old PHP site, needed to be refreshed
- Didn't want to continue using PHP

# Why Rewrite a Working Application?

- Other teams were using Python/Django
- Codebase was getting unmaintainable
- Easier to hire

# How Did We Do It?

# **Two Applications**

# **One Login**

- Authenticated on the PHP side
- PHP sent a request to Django with the username, password hash and secret
- Authenticated in Django when the user visited the site

# One App at a Time



- Built what we thought users would need the most
- Specifications from Business
- Launched side-by-side with old site
- Turned off old site once we confirmed everything was working

# Feature Flipping

- Django Waffle (<https://github.com/jsocol/django-waffle>)
- Views

```
if waffle.flag_is_active(request, 'NewDesign'):
    # Do Stuff Here
```

- Templates

```
{% load waffle_tags %}
{% switch NewDesign %}
    OMG CONTENT
{% endswitch %}
```

# Contest Rollout

Wednesday, March 20, 13

- Small core group of users
- We gave out invites to those users when they suggested ideas
- Those invites were then sent out to other users
- Eventually we flipped the flag for everyone

- We knew the core group of users would give good feedback
- By giving it to a few users we could test a totally new concept live without affecting the entire company
- Giving invites to these users gave the impression that they had a wanted commodity, giving more incentive to provide feedback
- Once everything was tested we felt confident releasing it to the entire company

# Template Structure

- New design moved to new/

```
templates/  
  base.html  
  homepage/  
    home.html  
    ...  
  notifications/  
    ...  
  new/  
    base.html  
    homepage/  
      home.html  
      ...  
    notifications/  
      ...|
```

- Same for Static

- Allows us to easily tell what templates still need to be moved over
- Allows us to easily select which template to use. If they have the flag, just add new/ to the template\_name
- Once everything is done we can just move everything back one level

# Realization #1

Specifications Change Quickly



- Don't get too attached
- Need to know the business to anticipate changes
- Learn to say no

# Realization #2

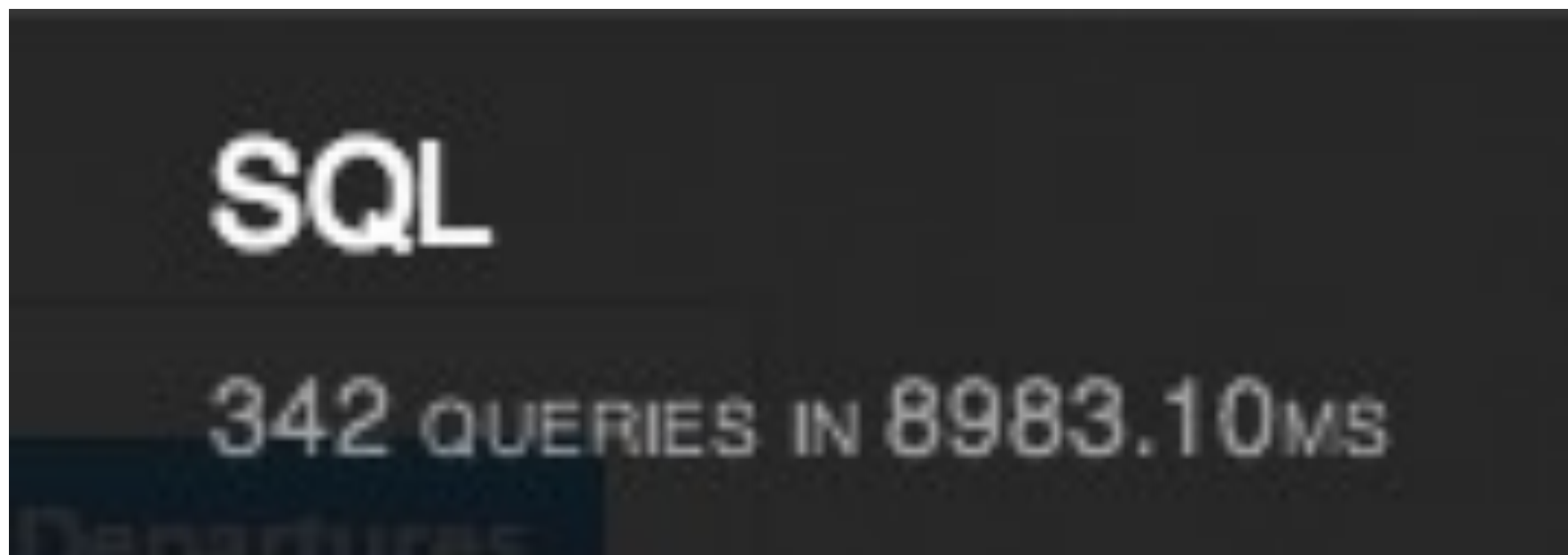
Your Users aren't Always Right

- They only know from what they've seen
- They don't know the technology
- You can tell them what they want
- Still be ready to learn from them

# Realization #2

The ORM is Great...Until it's Not

# Bad

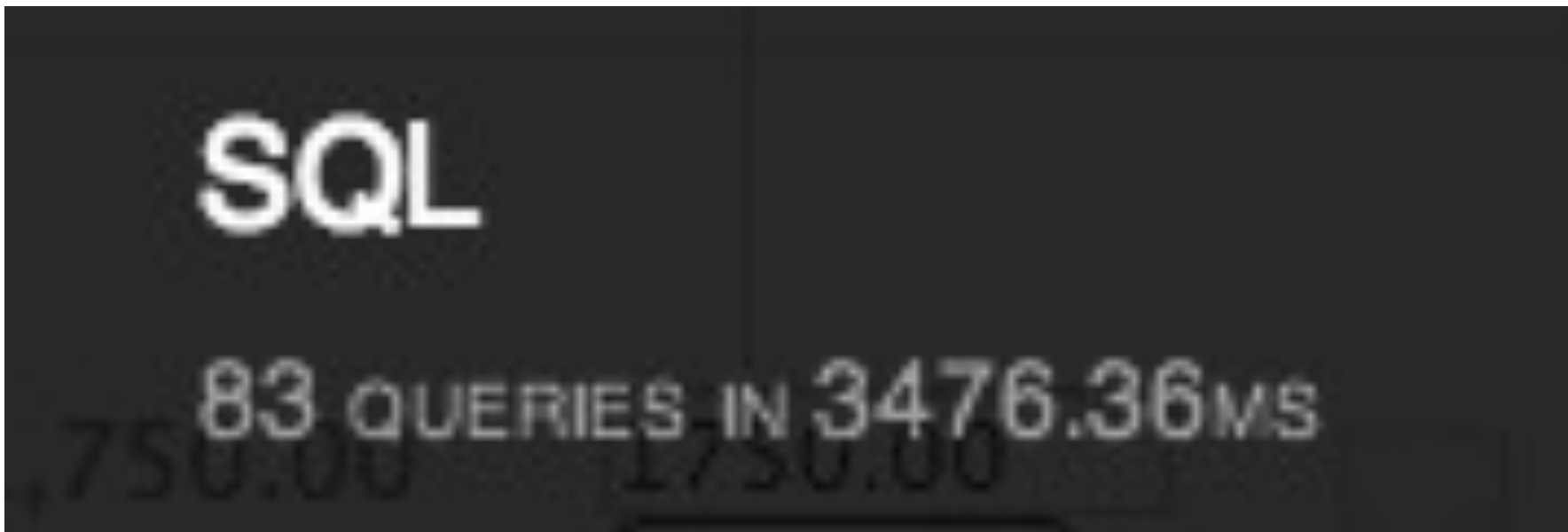


# Better

SQL

20 QUERIES IN 2489.86MS

# Wat?



# Best

SQL

27 QUERIES IN 293.40MS



- Can produce some very weird (slow) joins
- Hundreds of slow queries kill a page

# **Realization #4**

People with Slow Internet  
Connections Don't Notice  
Performance Gains


- When they're used to waiting a few minutes per page load, 45 second page decrease doesn't make much of a difference
- Youtube

- (New design, super quick, noticing that people were having the page load in 20-30 seconds. These people weren't being captured before because of how long it took)

# Realization #5

Simple Features get the Most Praise

- Lots of feedback and praise

 Showing **4 changed files** with **35 additions** and **13 deletions**.

- \*Crickets\*

 Showing **17 changed files** with **2,088 additions** and **97 deletions**.

# **Realization #6**

People Don't Use Your  
Application in Ways You Expect

- Saving after every change instead of a final save
- Importance of seeing what your users do
- Mixpanel, Clicktale, In Person

# Thanks!

Questions?