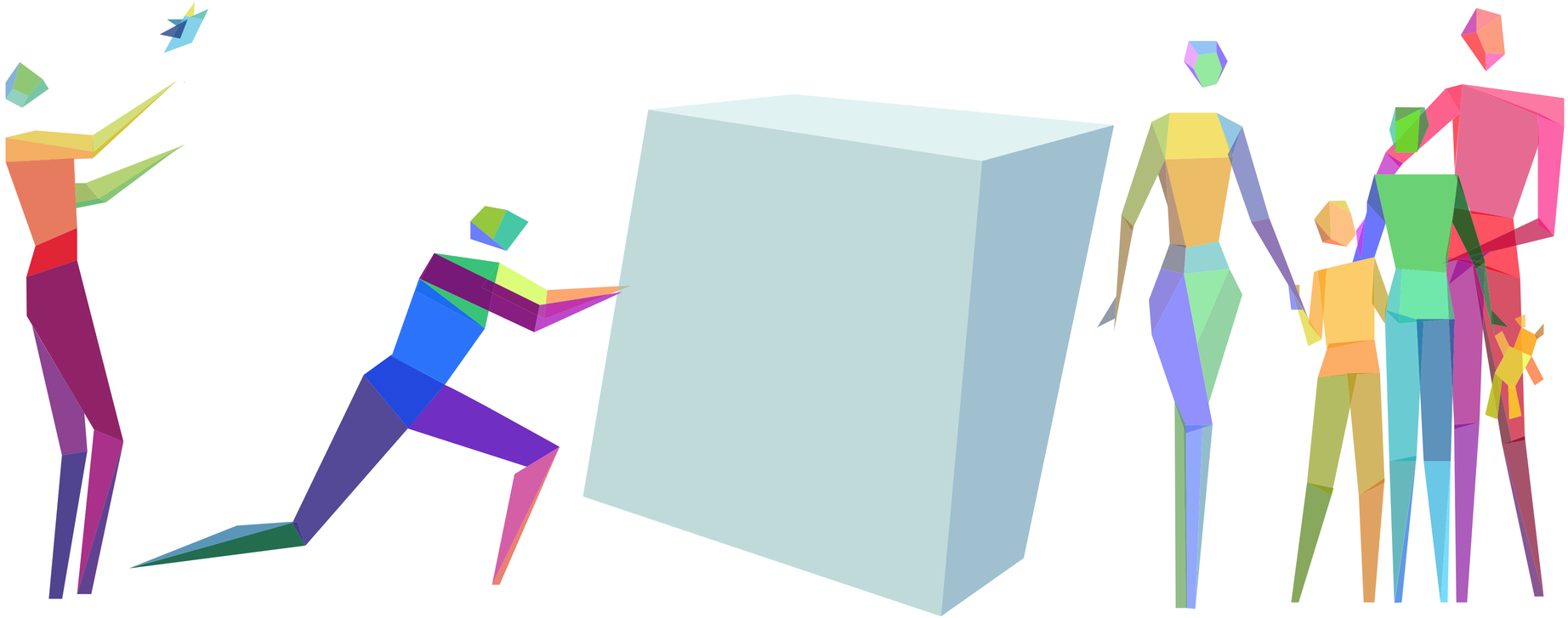
DrupalCon Barcelona

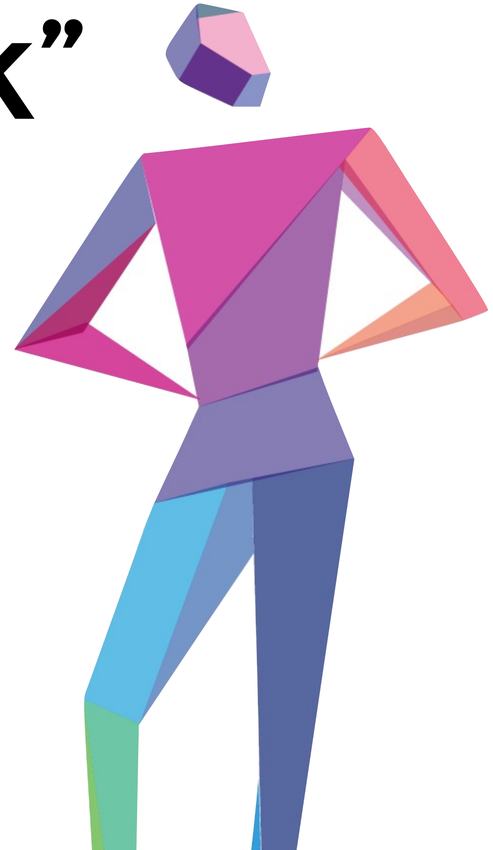
dries@buytaert.net // @Dries

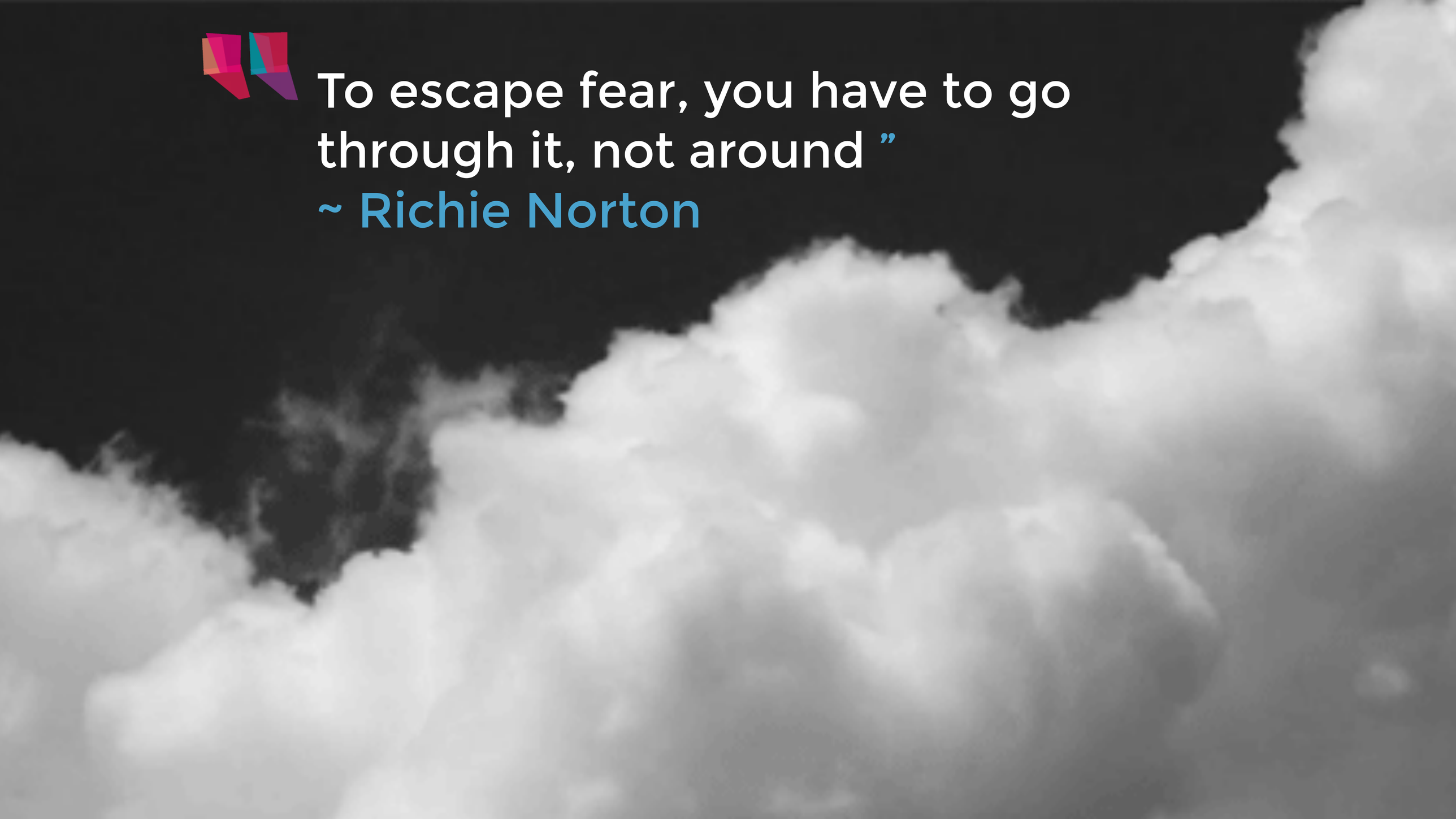
#Driesnote

Work-life balance



“We need to talk”

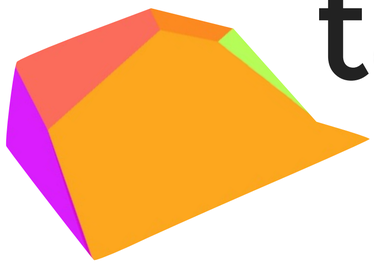



To escape fear, you have to go through it, not around ”
~ Richie Norton

Uncomfortable
questions

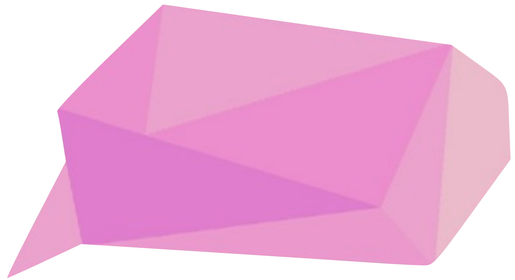
Drupal companies
are losing business

Is Drupal losing momentum?

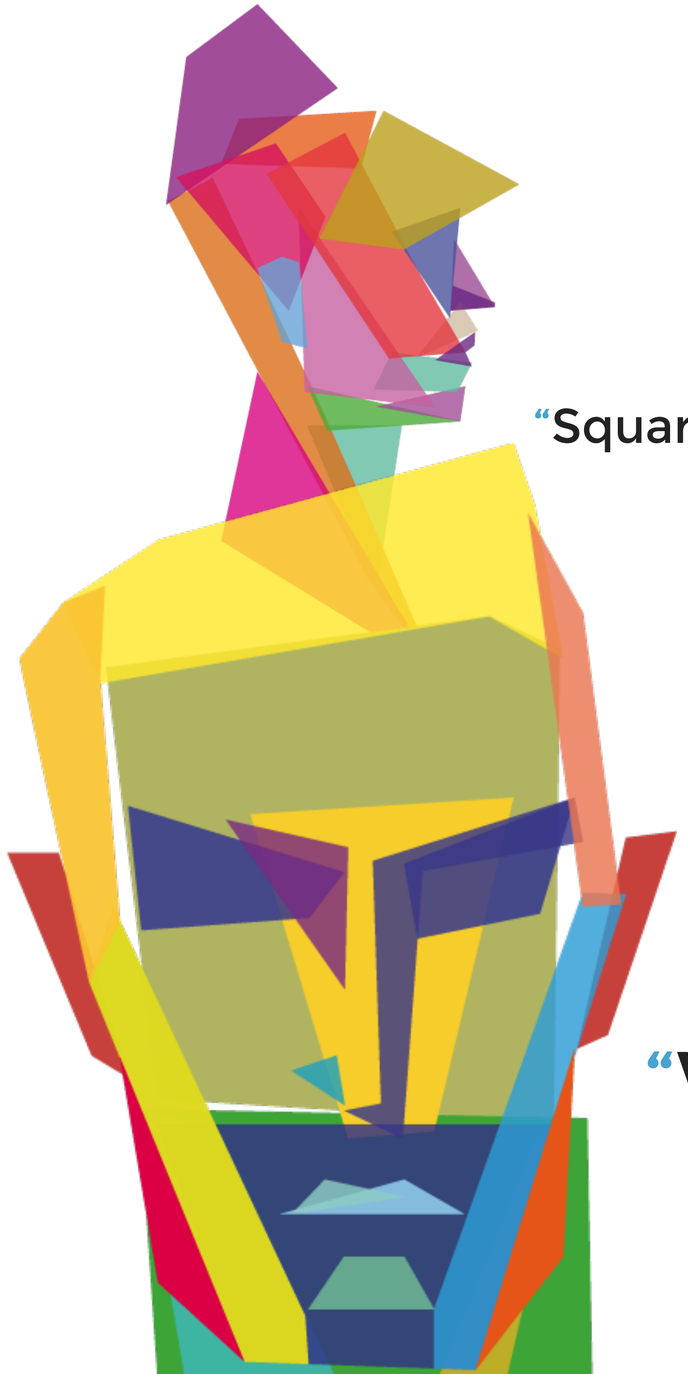


We need to talk about that

Why can’t we
release on time?



We need to talk about that

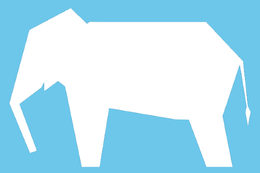


“Wordpress is getting even bigger”

“Squarespace has a great developer experience”

“Proprietary CMSes are becoming full-featured product suites”

“What about content APIs?”



Can we compete?

We need to talk about that

Can we get user experience right?

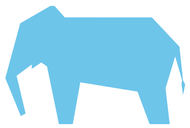
“Where is the “new post” button?”

“Why can’t I just drag it in?”

“I give up, this is just too hard”

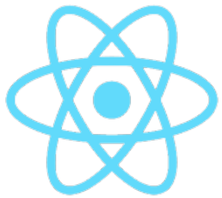
“What is a ‘view’?”

We need to talk about that



Client side apps

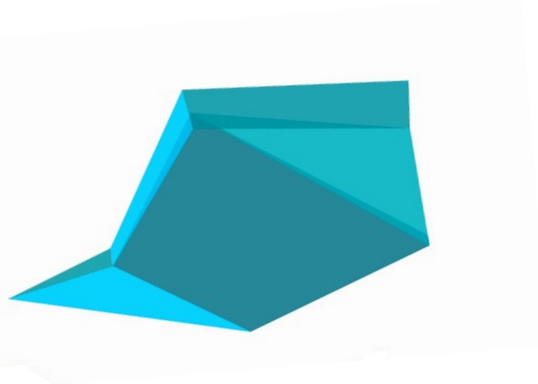
are a big trend and
they're not going away

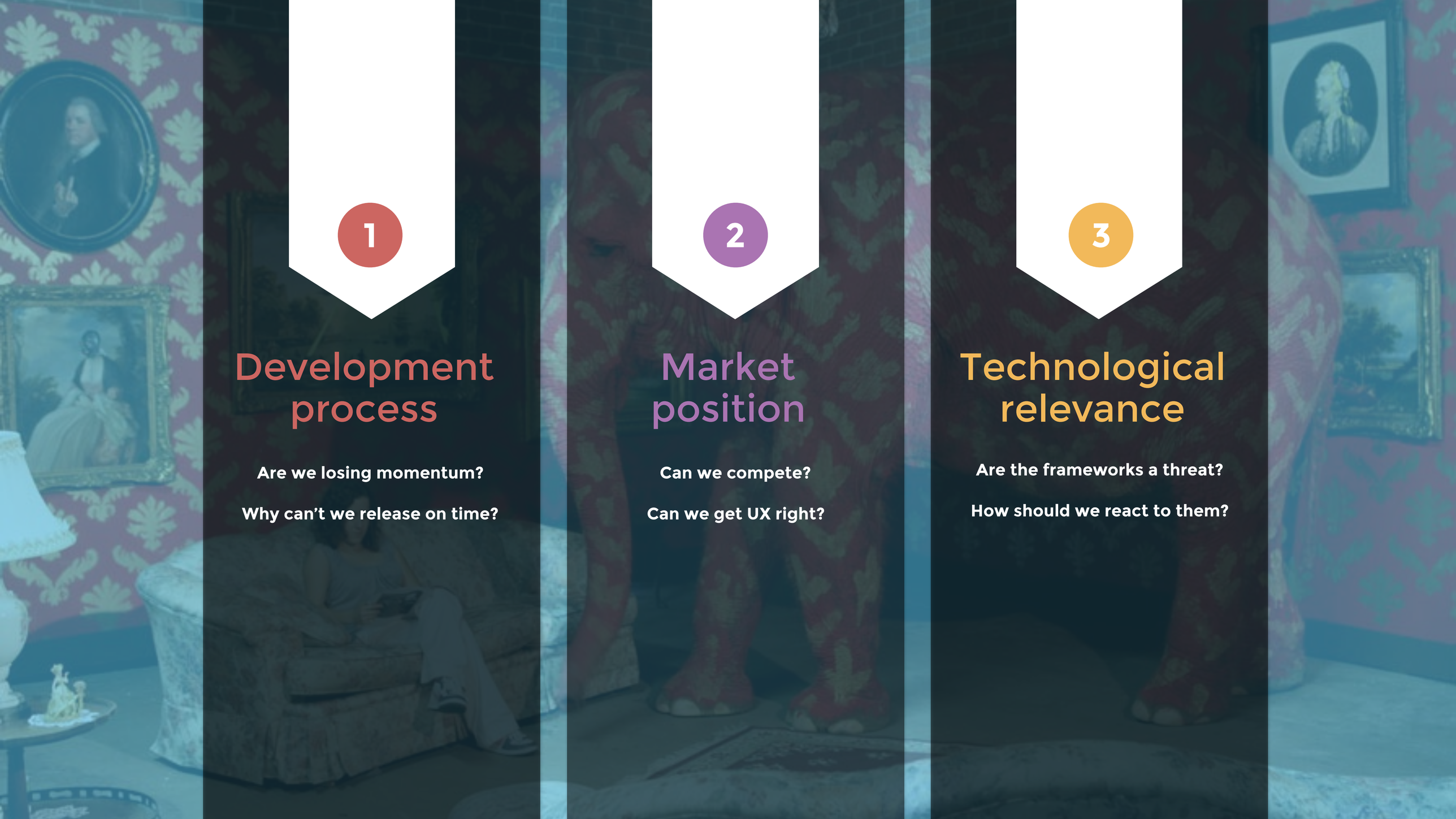


React

Are they a threat?
Going headless looks fun
My friends are using Angular
Will my PHP skills be in demand?
I don't want to be 'out-of-touch'
I won't have to fight the markup
Customers are asking for 'web apps'

We need to
talk about that





1

Development process

Are we losing momentum?

Why can’t we release on time?

2

Market position

Can we compete?

Can we get UX right?

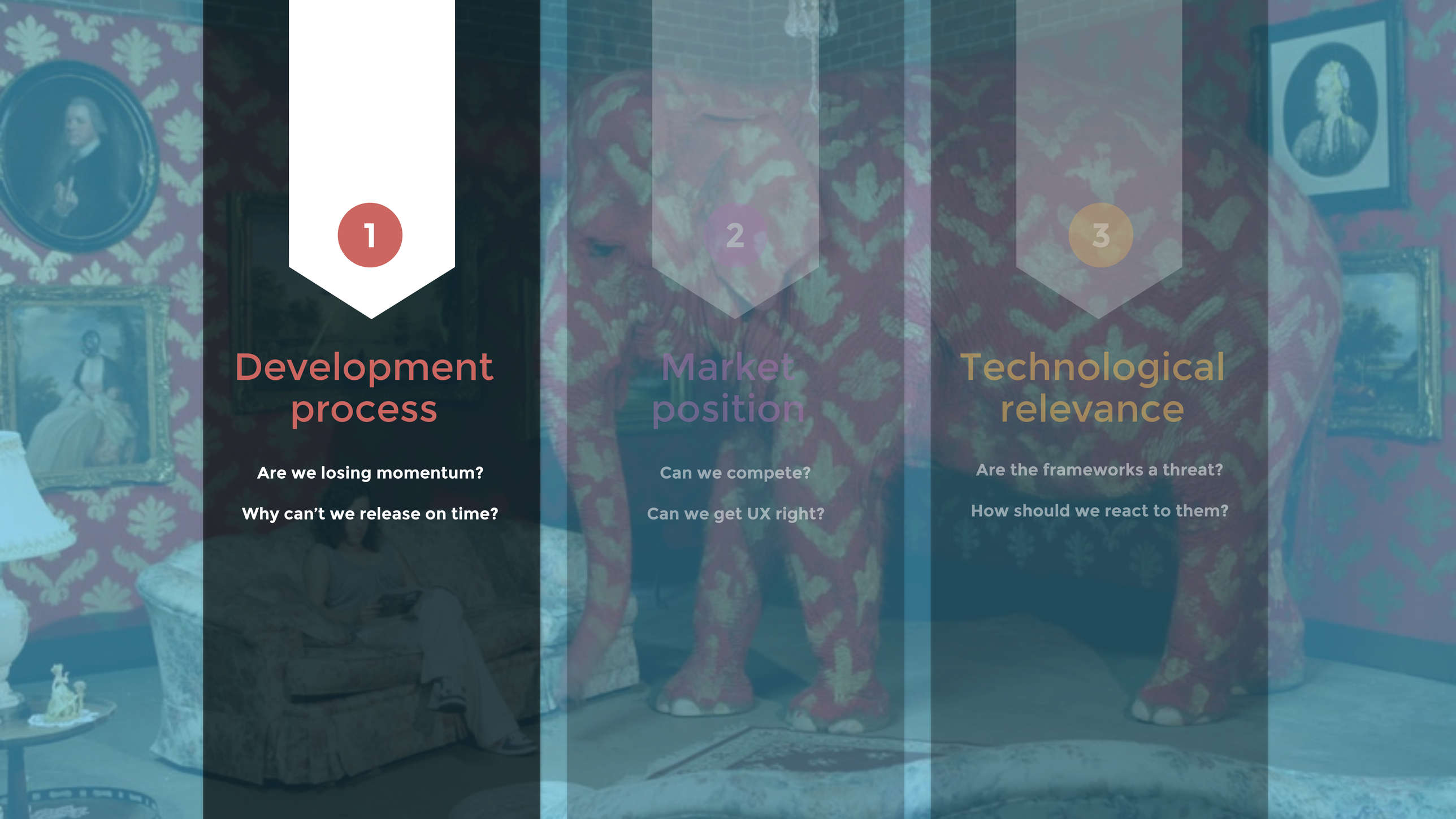
3

Technological relevance

Are the frameworks a threat?

How should we react to them?

So let’s talk



1

Development process

Are we losing momentum?

Why can't we release on time?

2

Market position

Can we compete?

Can we get UX right?

3

Technological relevance

Are the frameworks a threat?

How should we react to them?

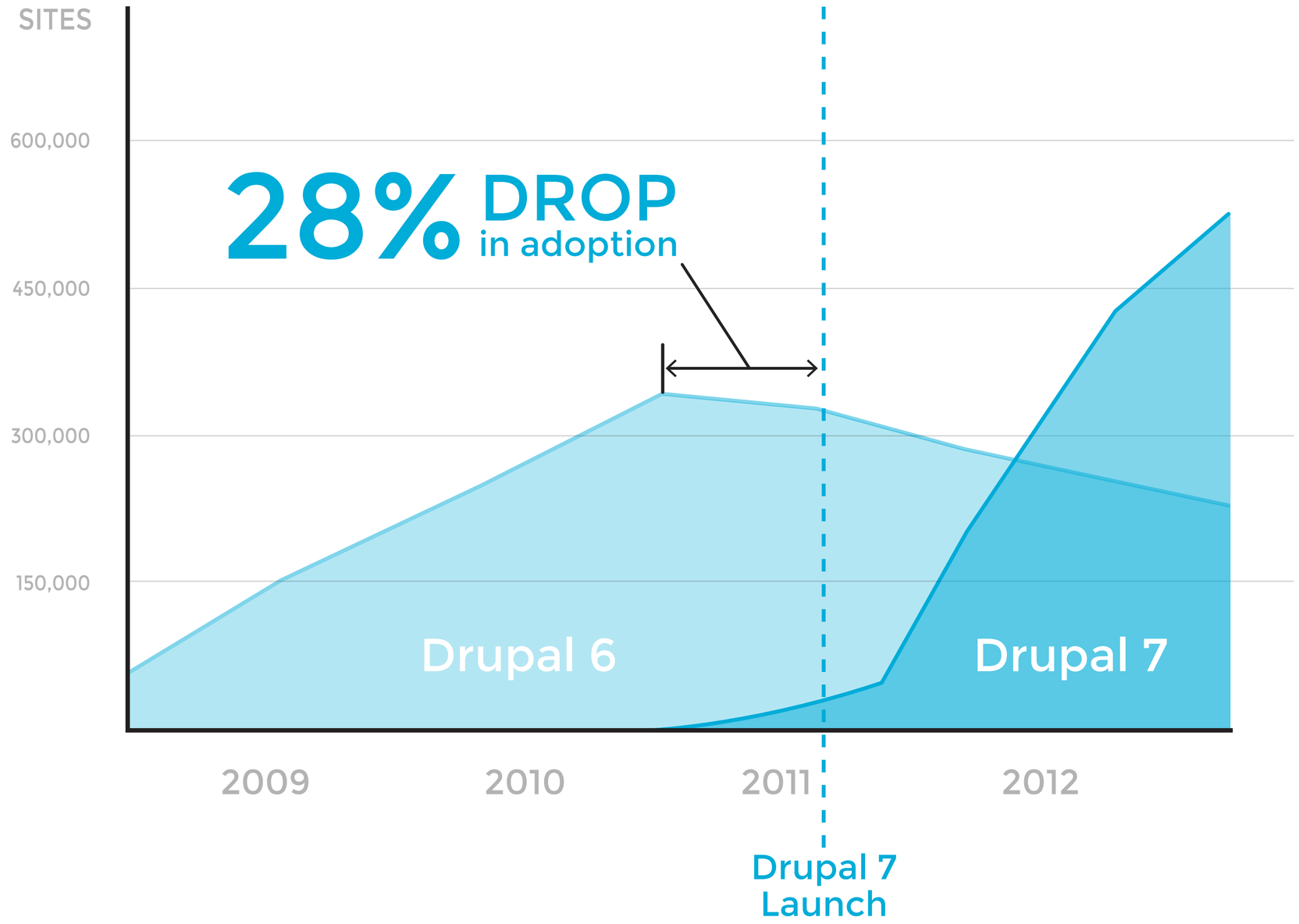
1
Development process
Is Drupal losing momentum?

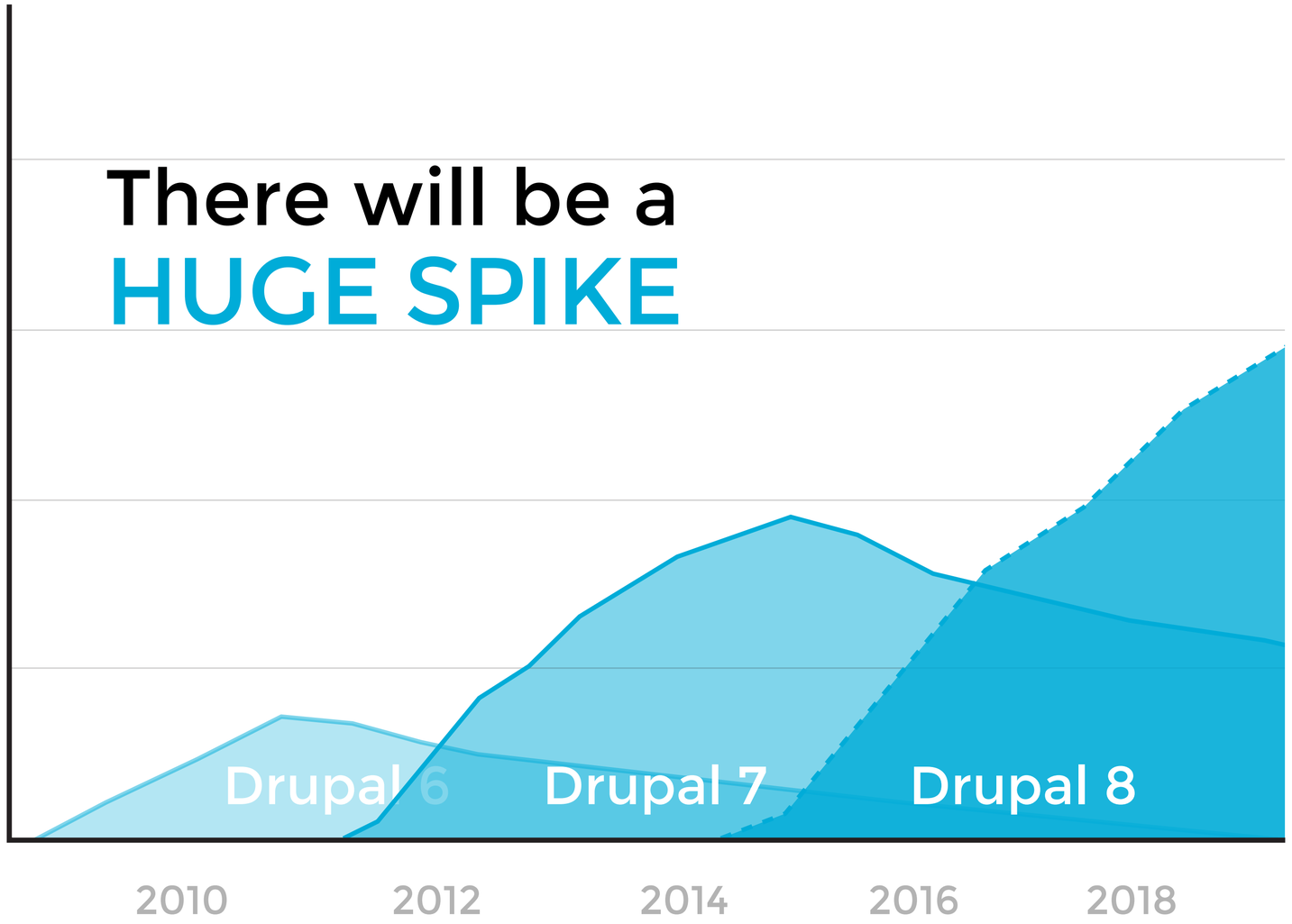
Yes.

This is not unusual

The Osbourne Effect

Announcement of a new release slowed adoption of the current version


SITES
600,000
450,000
300,000
150,000
28% DROP
in adoption
Drupal 6
Drupal 7
2009
2010
2011
2012
Drupal 7
Launch


There will be a
HUGE SPIKE
Drupal 6
Drupal 7
Drupal 8
2010
2012
2014
2016
2018

We need to get
Drupal 8 released

Attend sprints
Donate money to fix bugs
Start developing on Drupal 8 today
Upgrade contrib modules
Donate time to fix bugs

3,000+ contributors

15,000+ committed patches

1,300+ fixed criticals

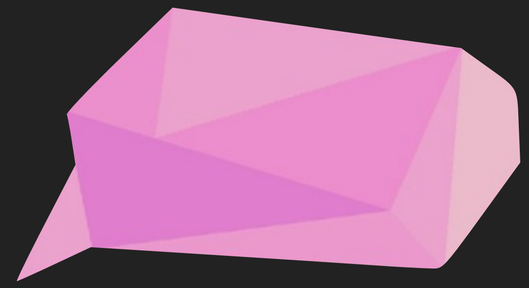
1 month without a surprise

1 big problem left

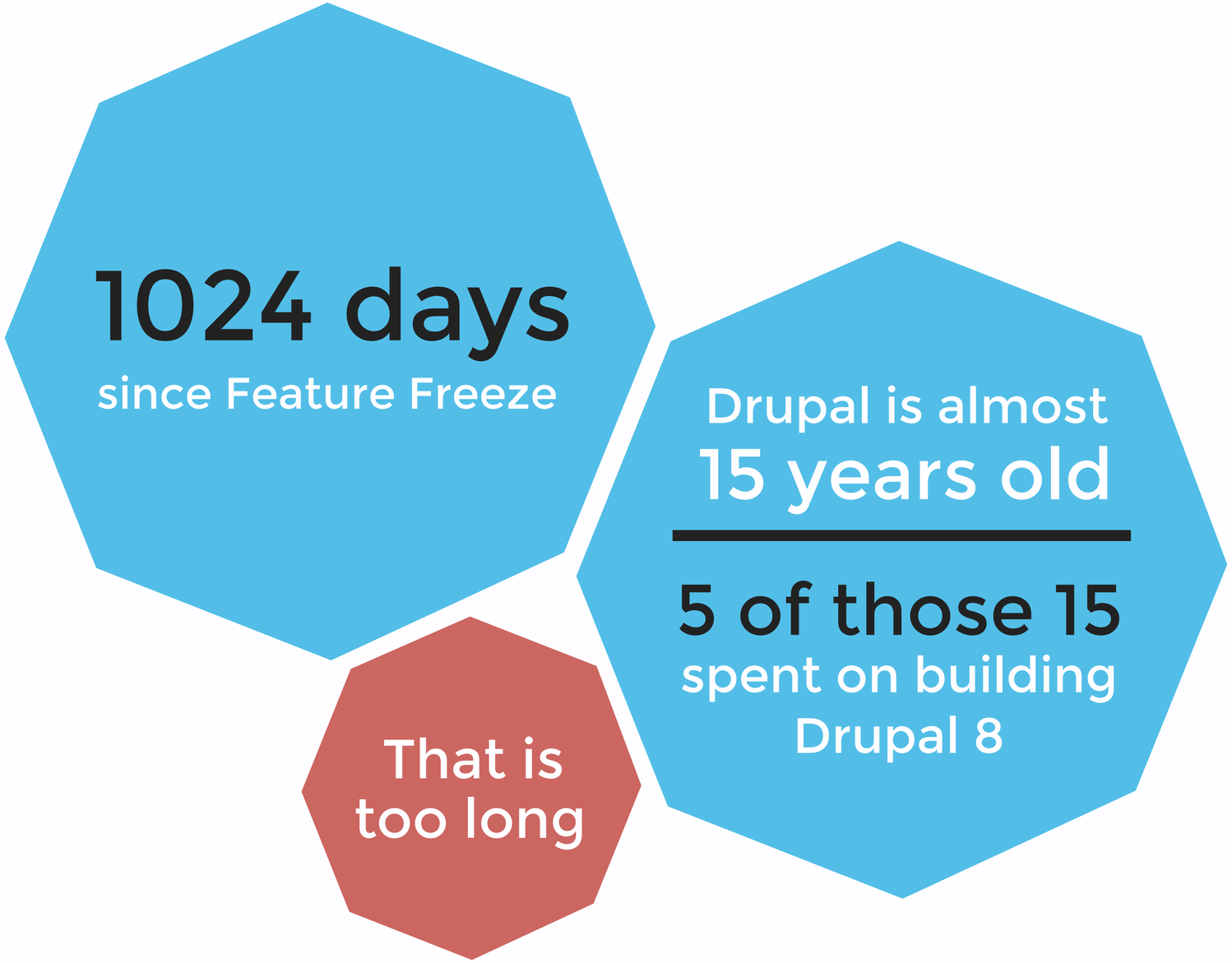
RC1

October 7th, 2015 (*)

(*) Assuming no security or data loss issues



However, there are some
things we need to change


1024 days
since Feature Freeze
Drupal is almost
15 years old
5 of those 15
spent on building
Drupal 8
That is
too long

1

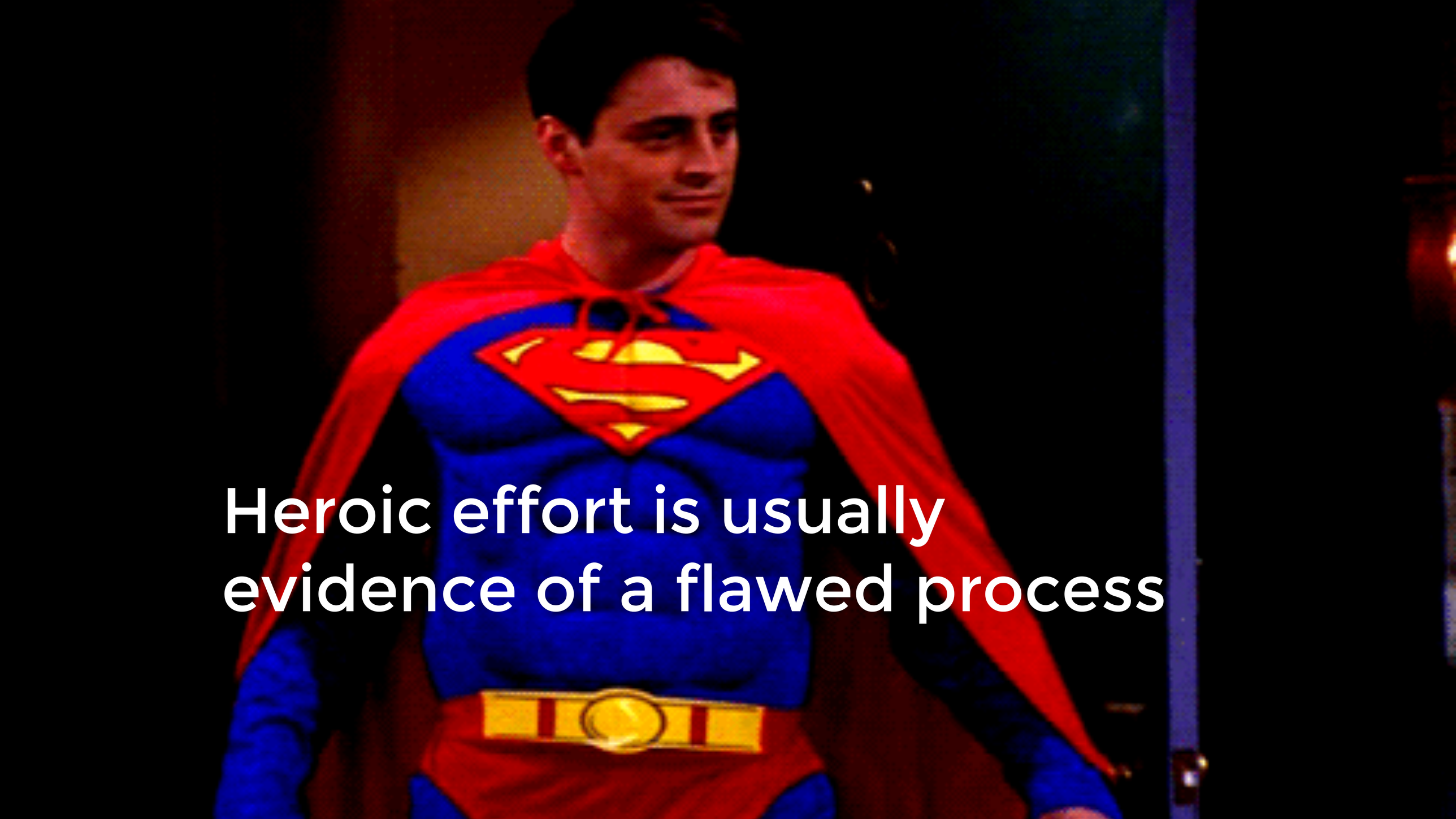
Development process

Why can’t we release on time?

No one is to blame for this

No one is to blame for this

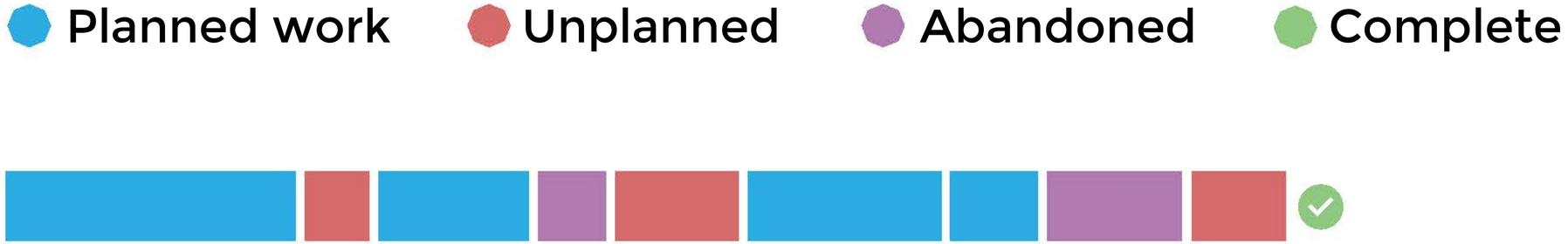
Except me

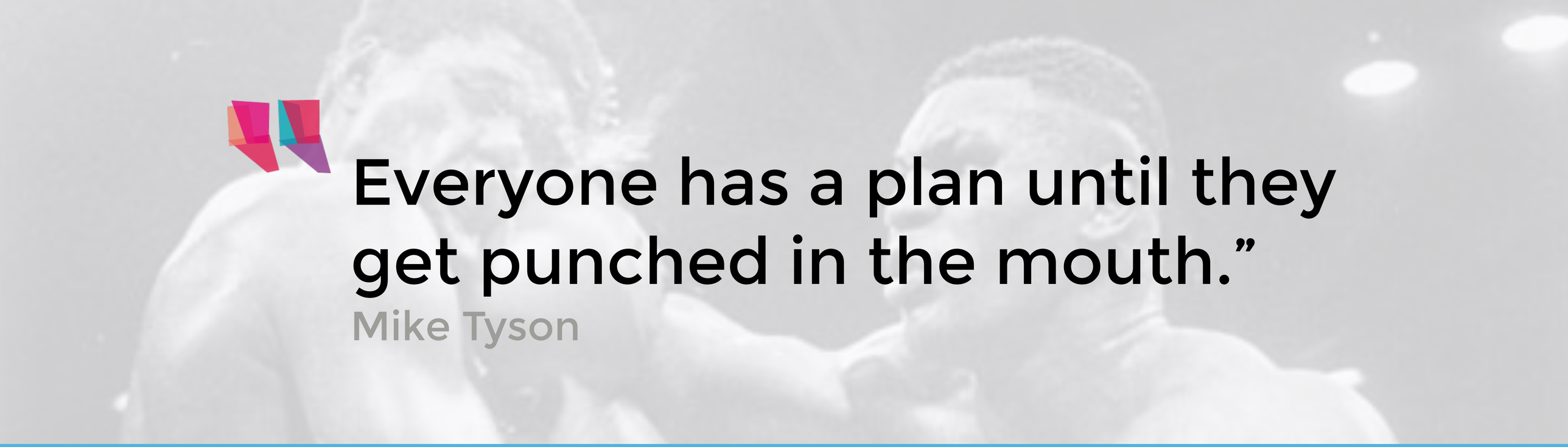

Heroic effort is usually
evidence of a flawed process

Expected life of a feature

Planned work Unplanned Abandoned Complete

Expected life of a feature





Everyone has a plan until they get punched in the mouth.”

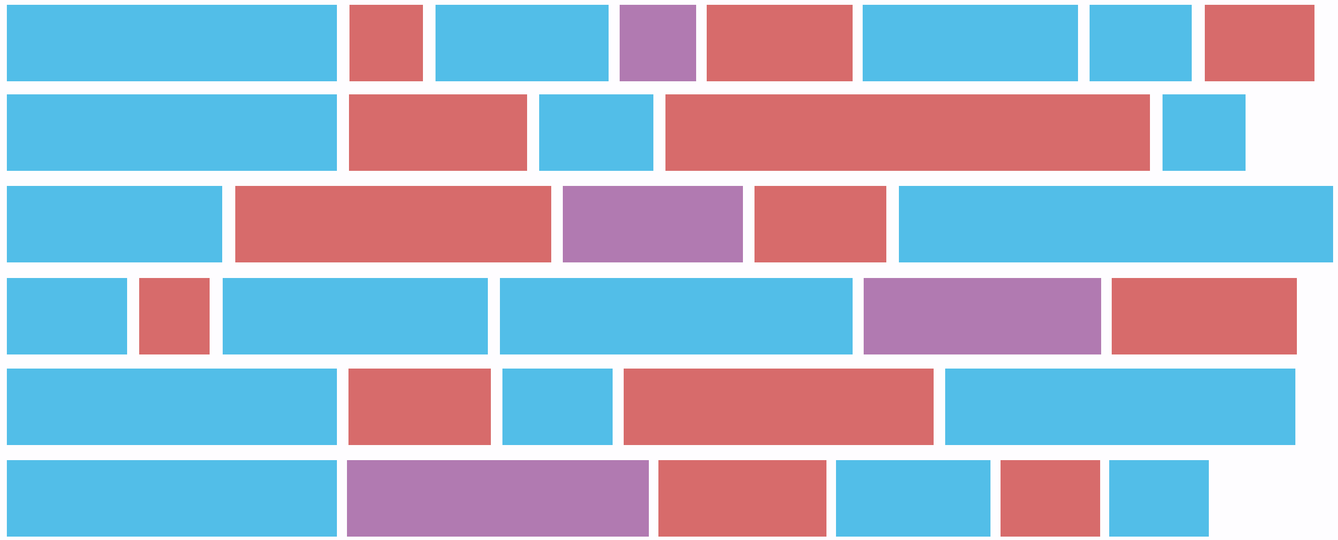
Mike Tyson

The Planning Fallacy

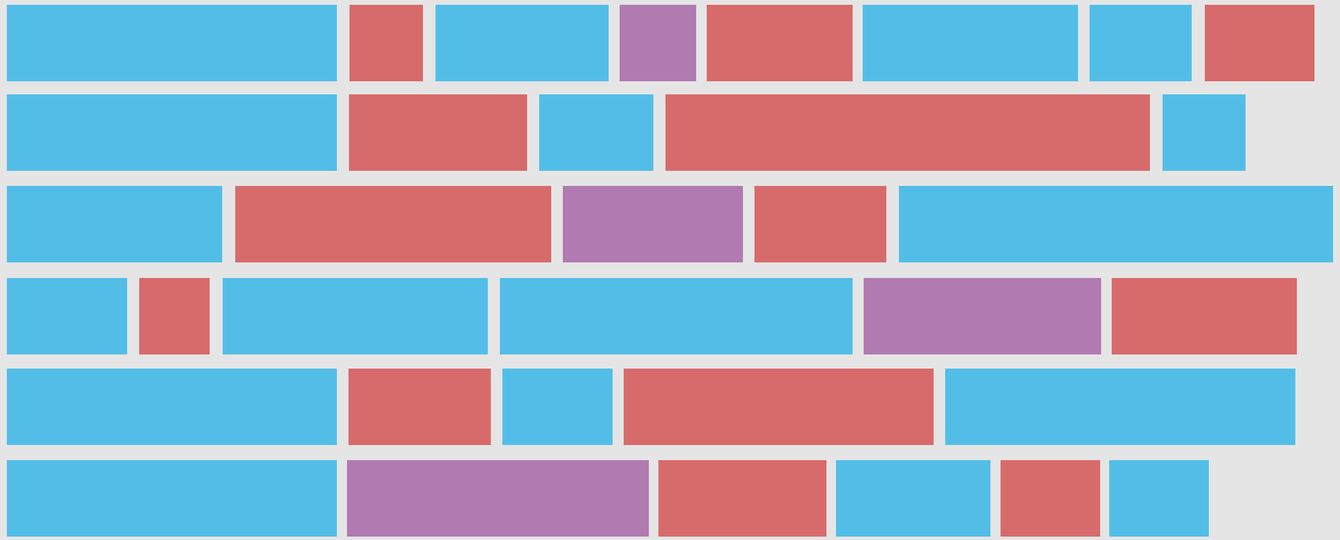
Daniel Kahneman and Amos Tversky, 1979

“Regardless of skill or prior knowledge humans tend to underestimate project timelines by up to 75%”

Multiply times many features…

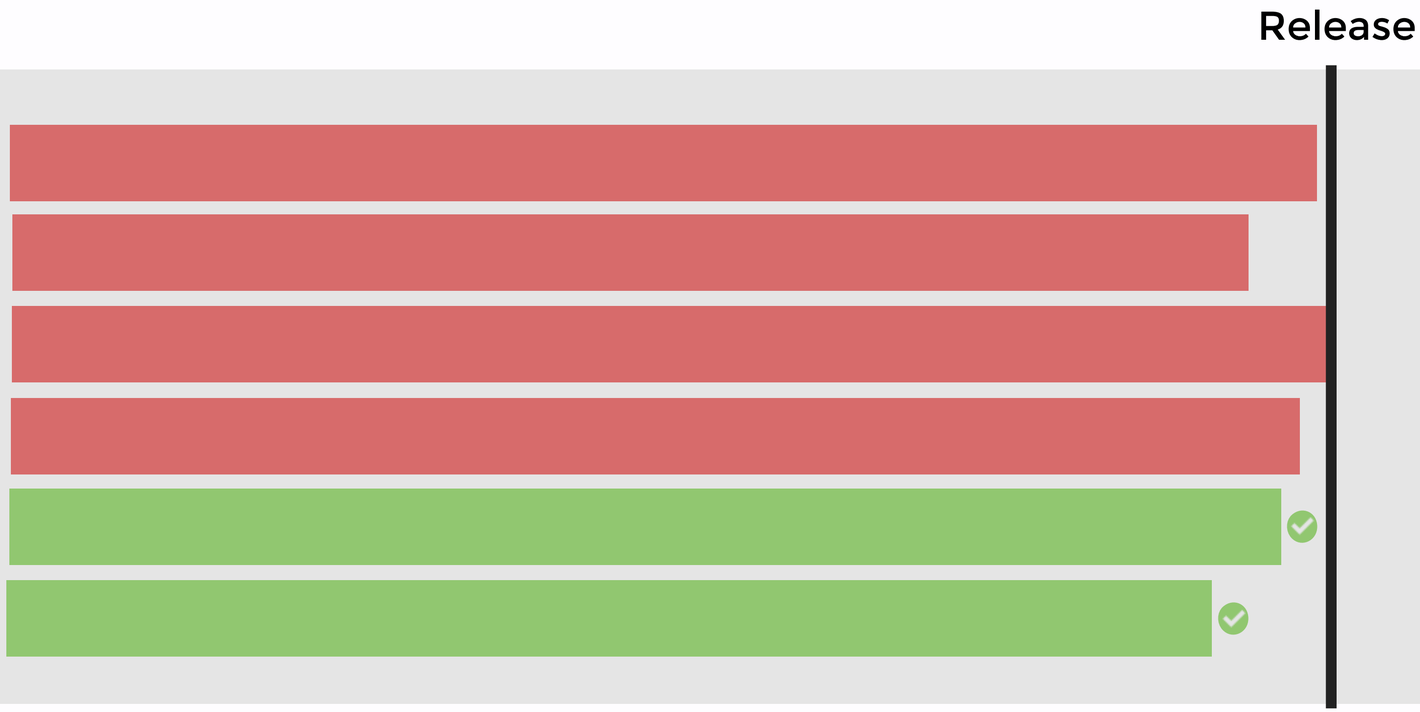


… and all in one branch



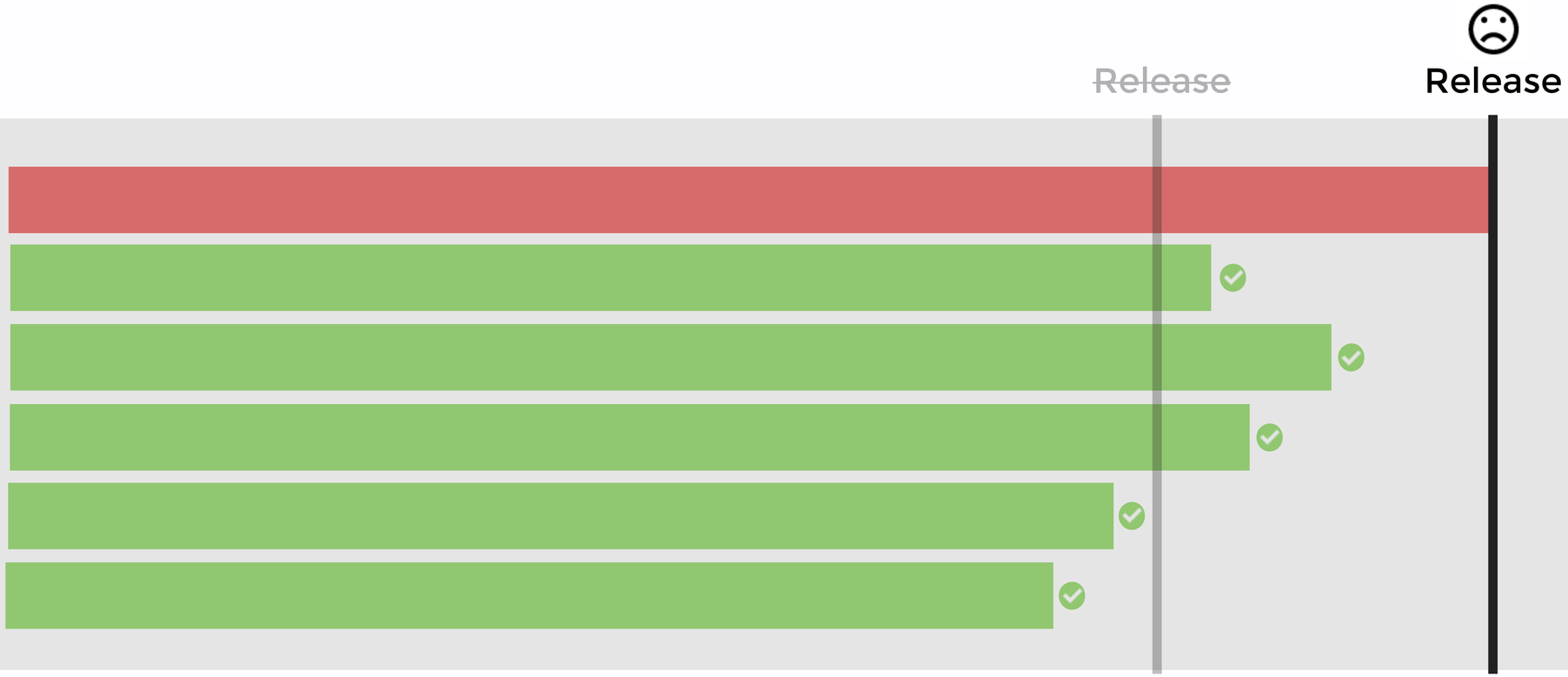
Main branch

… and all in one branch



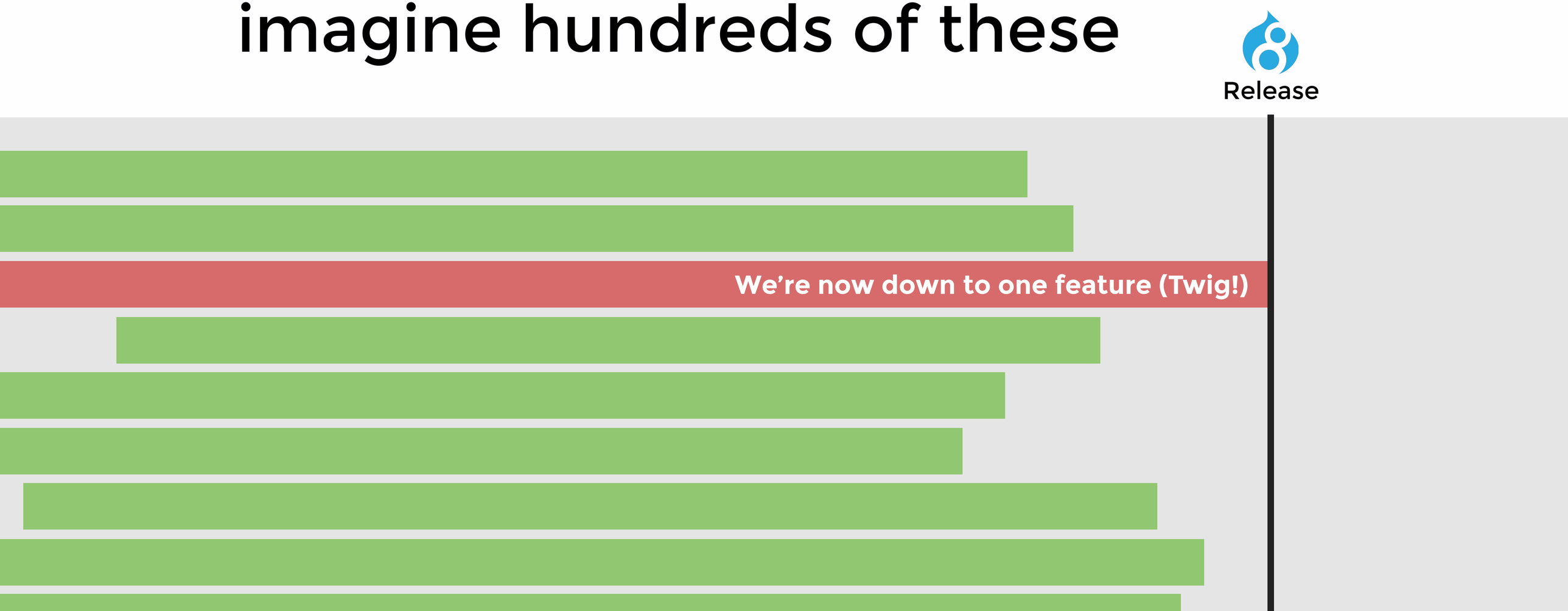
Main branch

Release pushed back



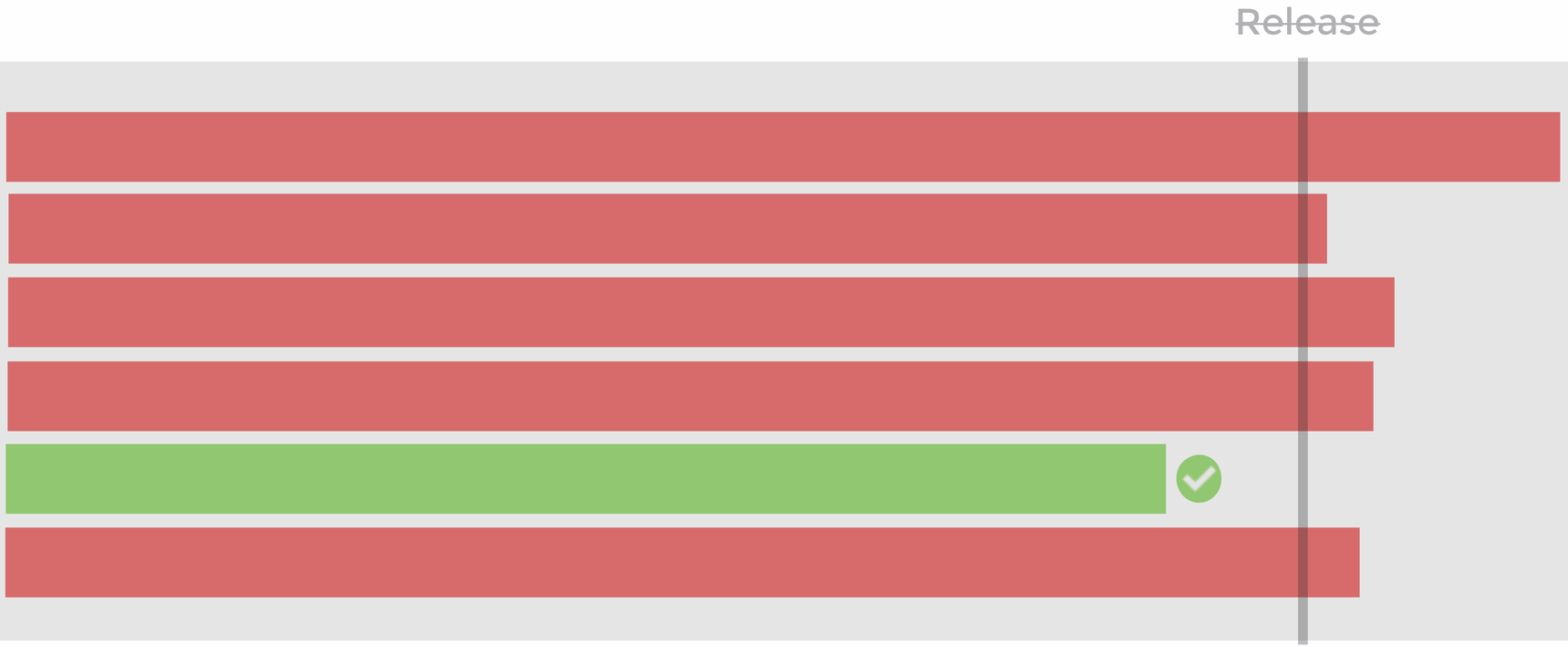
We can only go as fast as
our slowest feature

We added hundreds of features to Drupal 8 so imagine hundreds of these



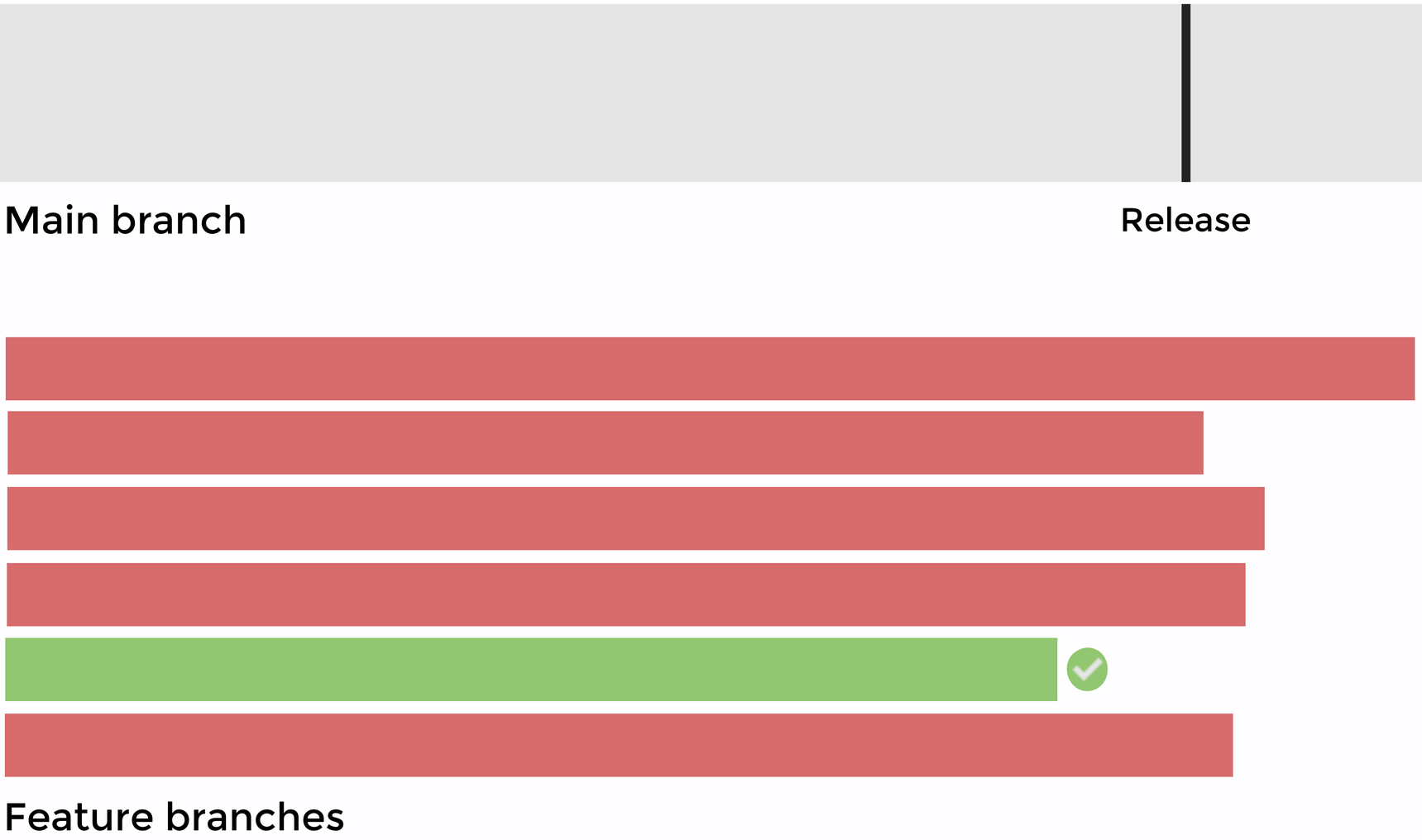
There must be a better way.

We need to stop this or we will not survive.

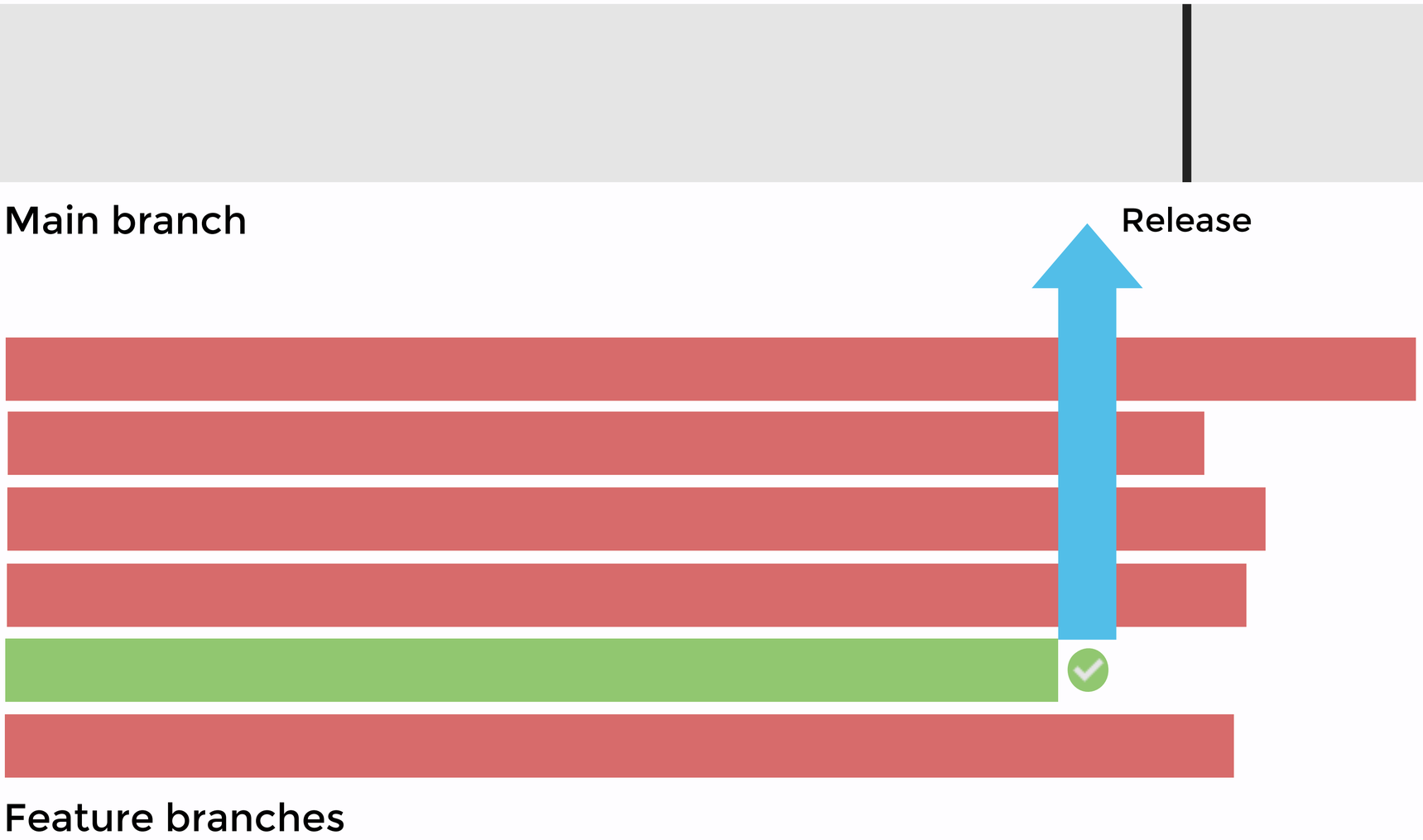


Main branch

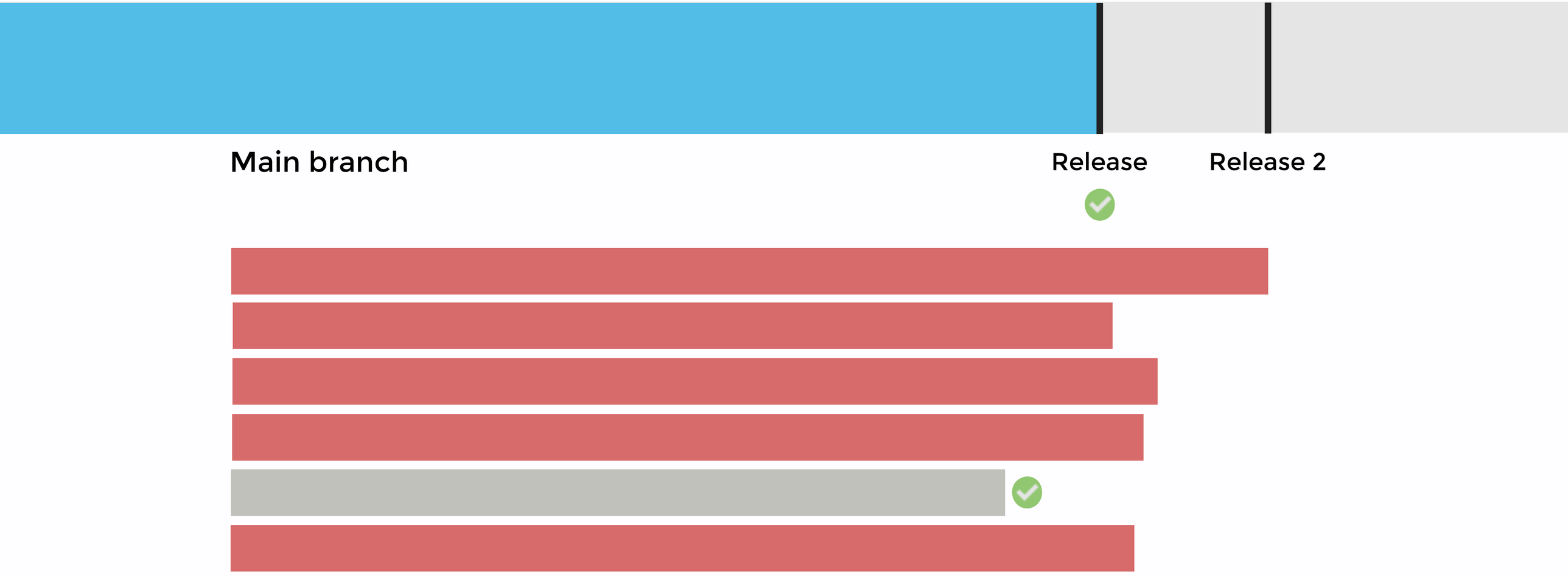
#1: Create feature branches



#2: Only merge when shippable



#3: Date-based releases



But what about merges?

#1: Break up features

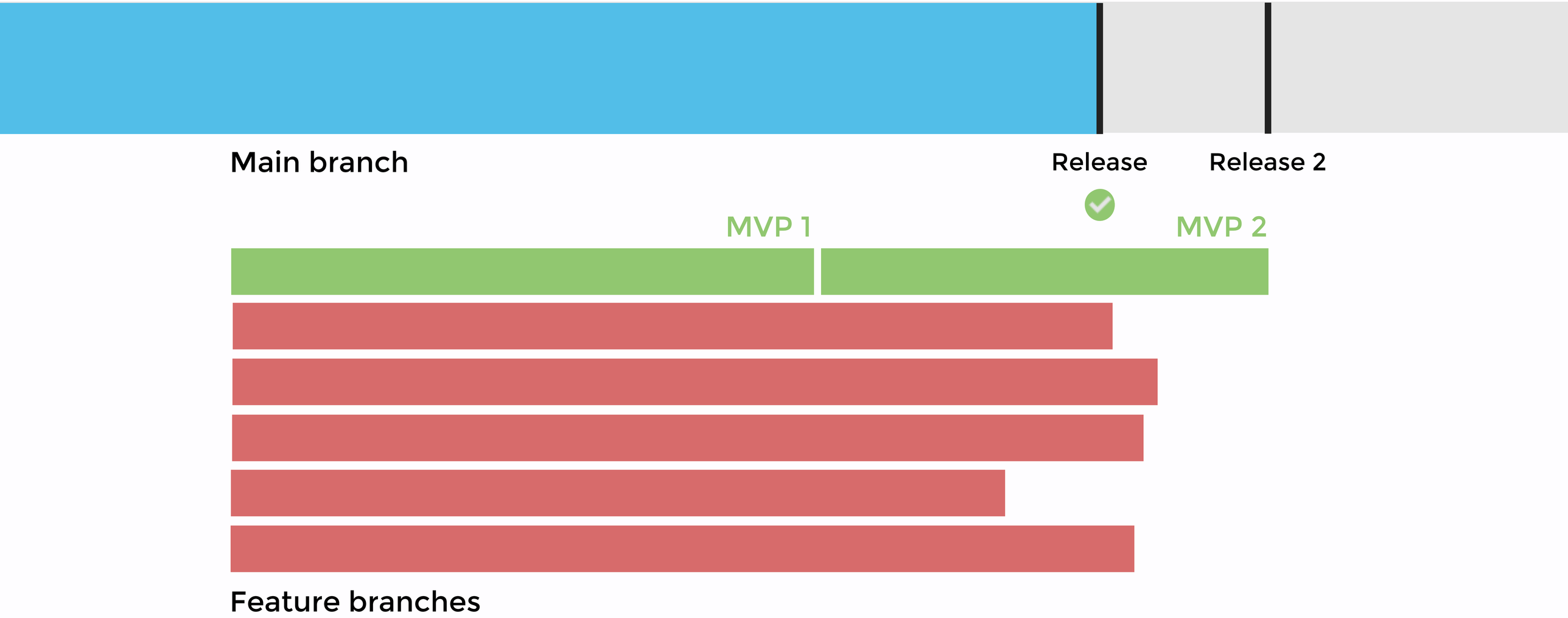
Main branch

Release

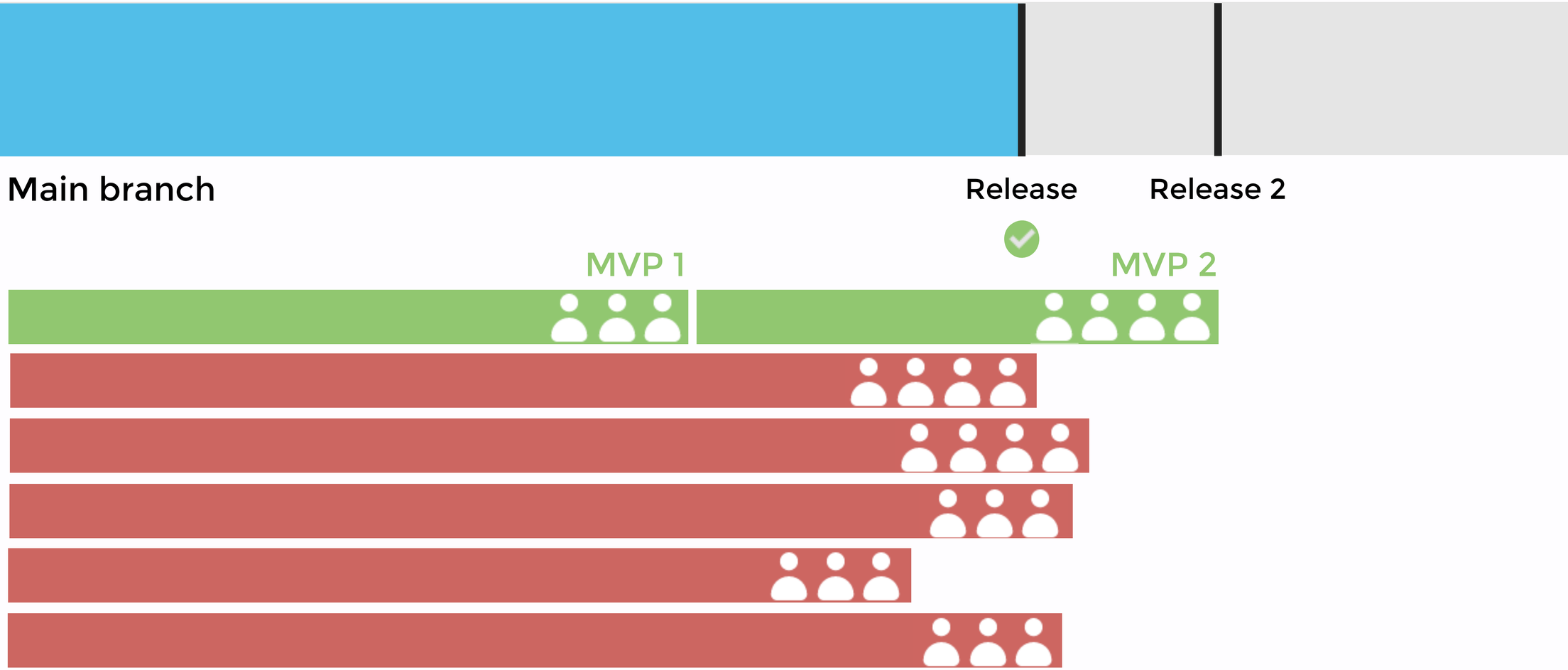
Release 2

Feature branches

#1: Break up features



#2: Cross functional teams

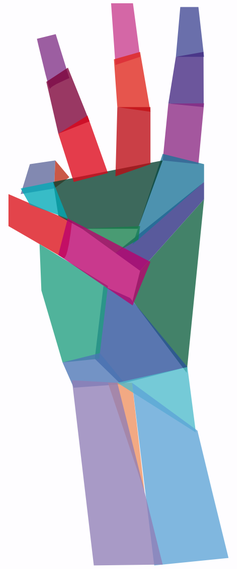


#3: Orchestrate merges

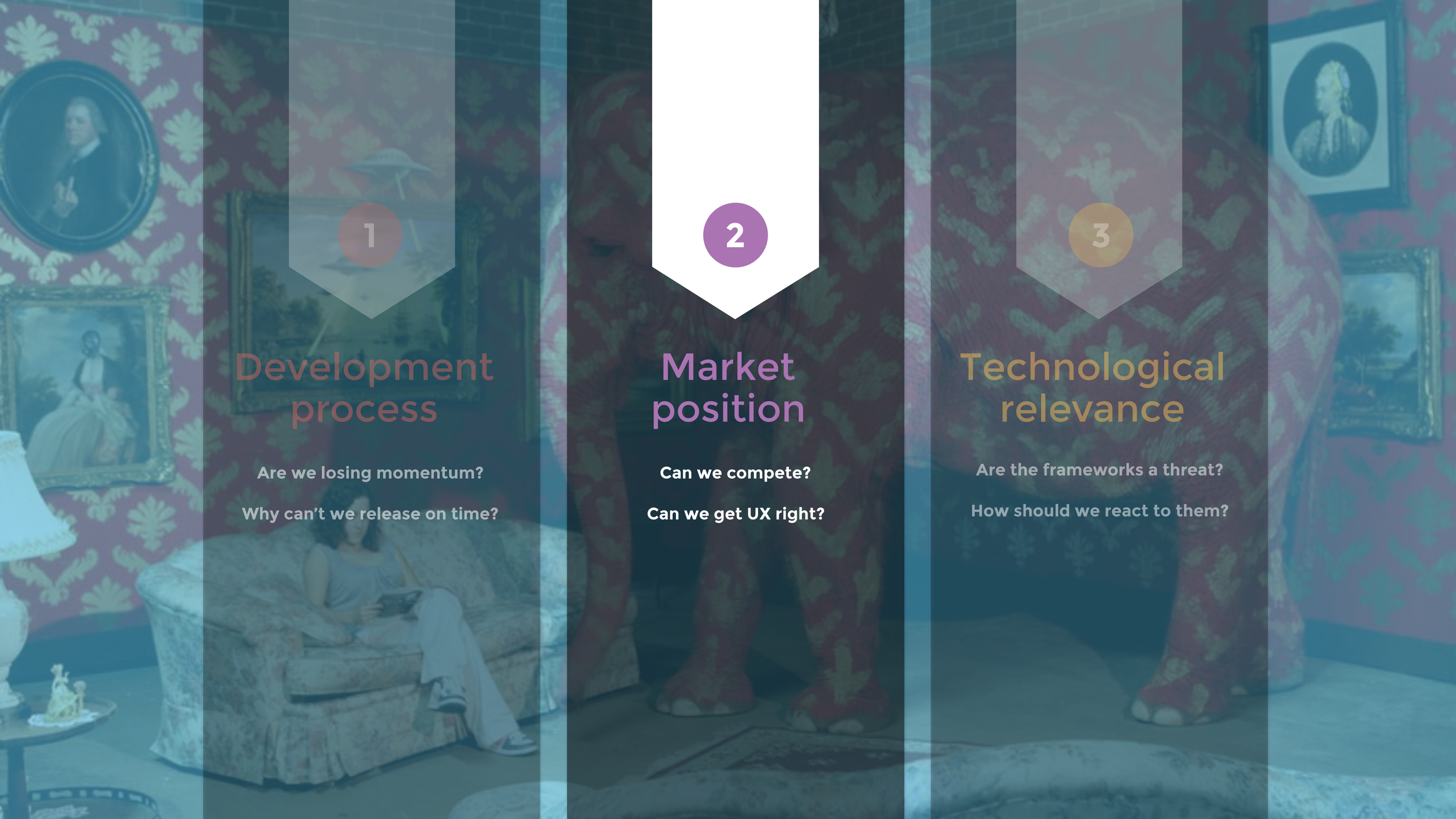
To sum up:

We didn’t release on time

This wasn’t your fault



We will fix the way we work



1

Development process

Are we losing momentum?

Why can't we release on time?

2

Market position

Can we compete?

Can we get UX right?

3

Technological relevance

Are the frameworks a threat?

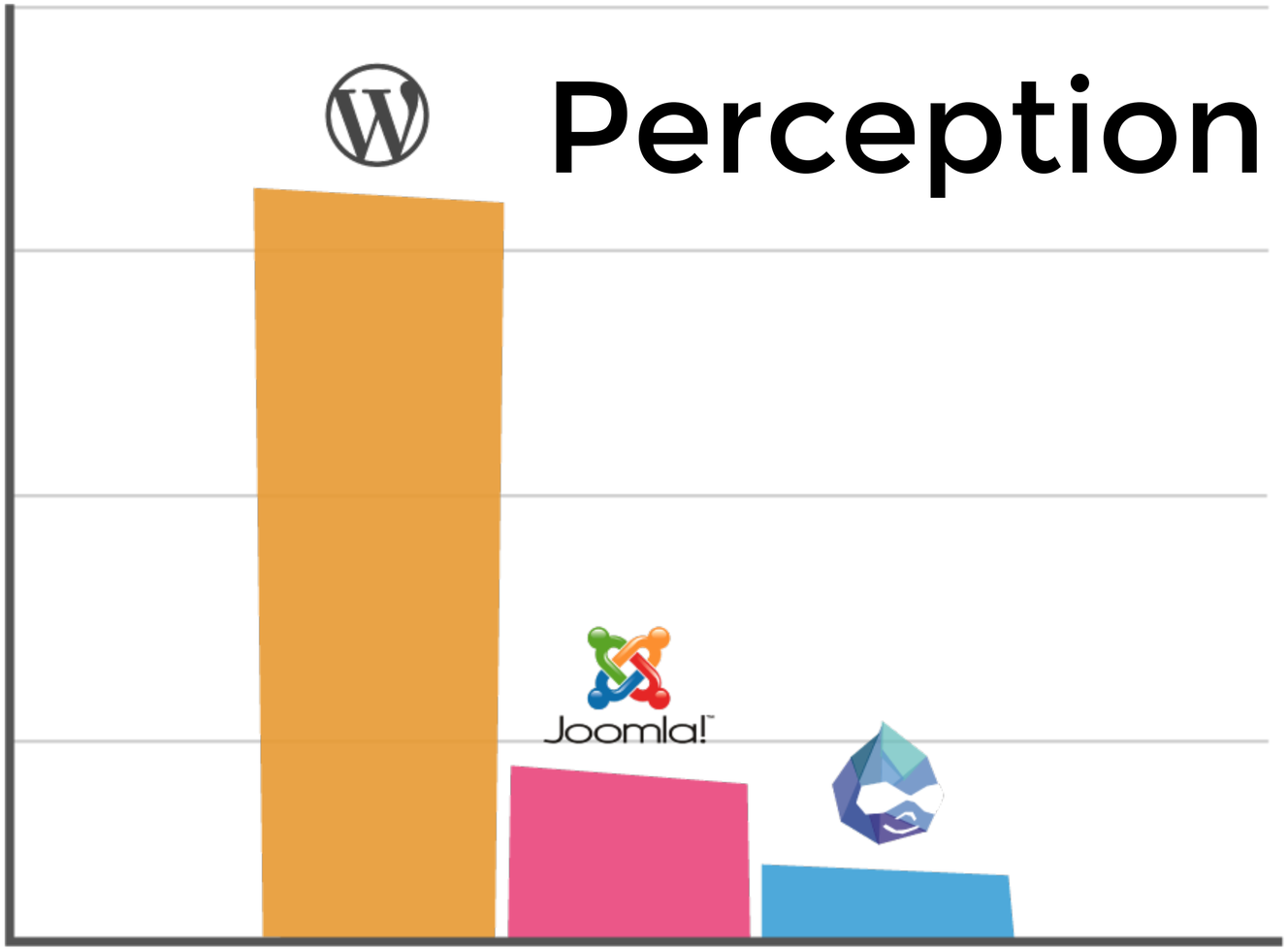
How should we react to them?

2

Market position

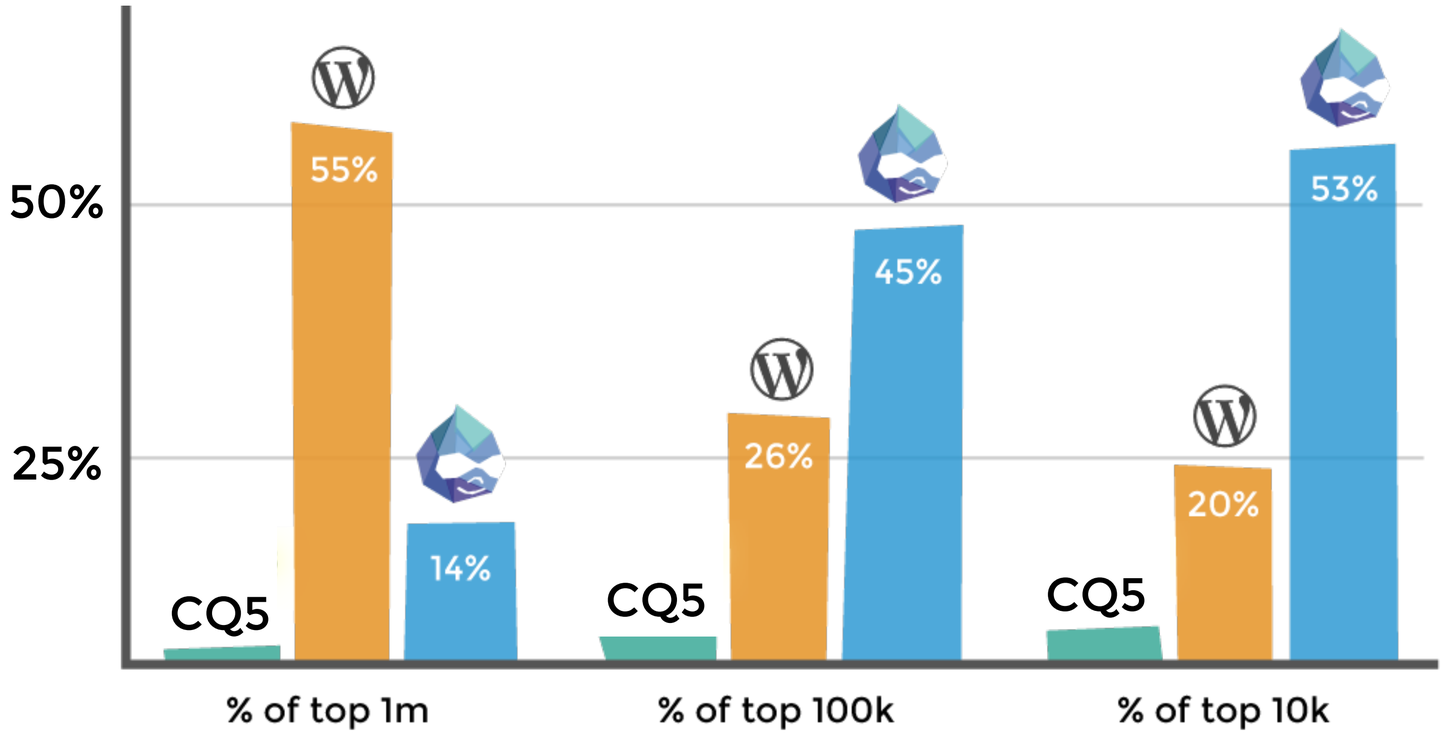
Can we compete?

Total sites



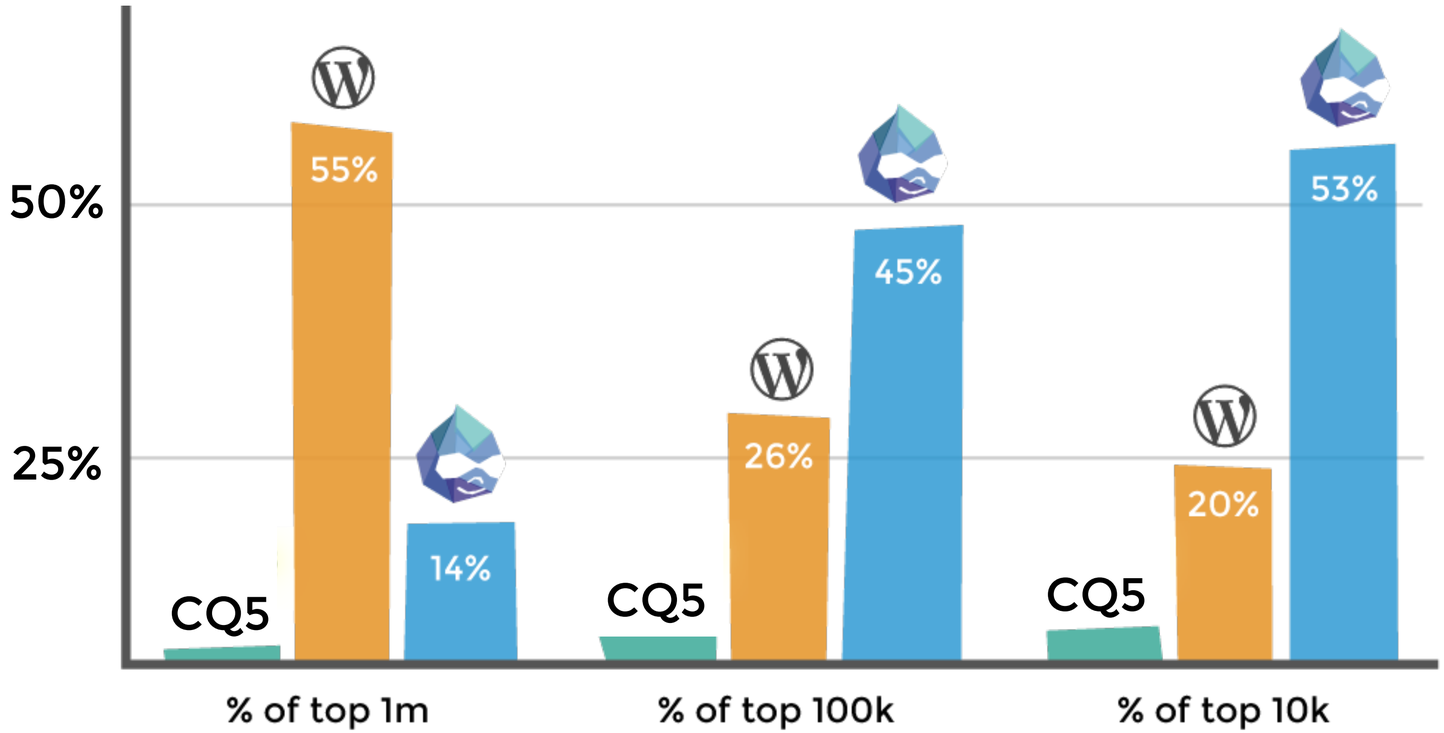
Alexa ranks, including subdomains

Reality



Alexa ranks, including subdomains

How did they do this?



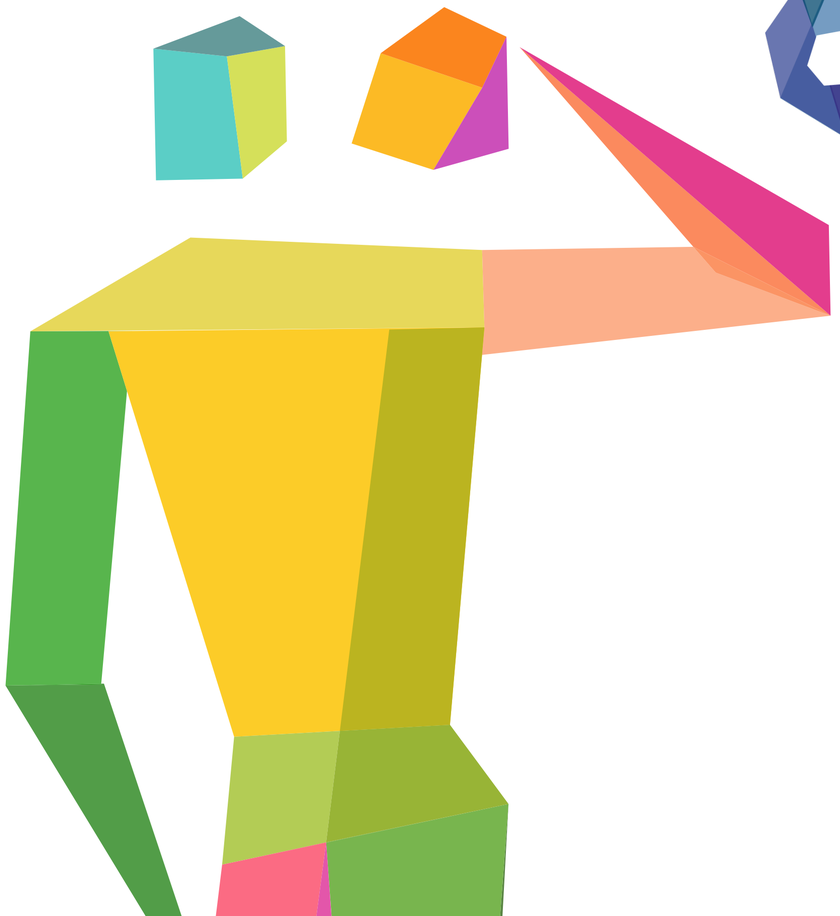
It’s a shipload of apples to a truckload of really big oranges

Site builder experience

~~Developer experience~~

~~Site builder experience~~

Developer experience



Déjà vu?

DrupalCon Barcelona 2007

(8 years ago!)

Slide from keynote in Barcelona 2007

Drupal 7 killer release

1. Better media handling
2. Custom content types in core
3. WYSIWYG editor
4. Better performance
5. Better tools to structure/organize content
6. Basic Views like module
7. Automatic upgrade functionality

70

UX 70%

8. Better performance for authenticated users
9. Views query builder
10. Improve node access system

30

DX 30%

Drupal 7 killer release

1. Better media handling
2. Custom content types in core
3. WYSIWYG editor
4. Better performance
5. Better tools to structure/organize content
6. Basic Views like module
7. ~~Automatic upgrade functionality~~

Done

8. Better performance for authenticated users
9. Views query builder
10. Improve node access system

Done

1. Better media handling
2. Custom content types in core
3. WYSIWYG editor
4. Better performance
5. Better tools to structure/organize content
6. Basic Views like module
7. ~~Automatic upgrade functionality~~

70

Done

UX
30%

8. Better performance for authenticated users
9. Views query builder
10. Improve node access system

30

11. Symfony framework
12. Configuration management
13. Web services
14. Performance and caching
15. Plugin system
16. Twig
17. Full views in core

Also done

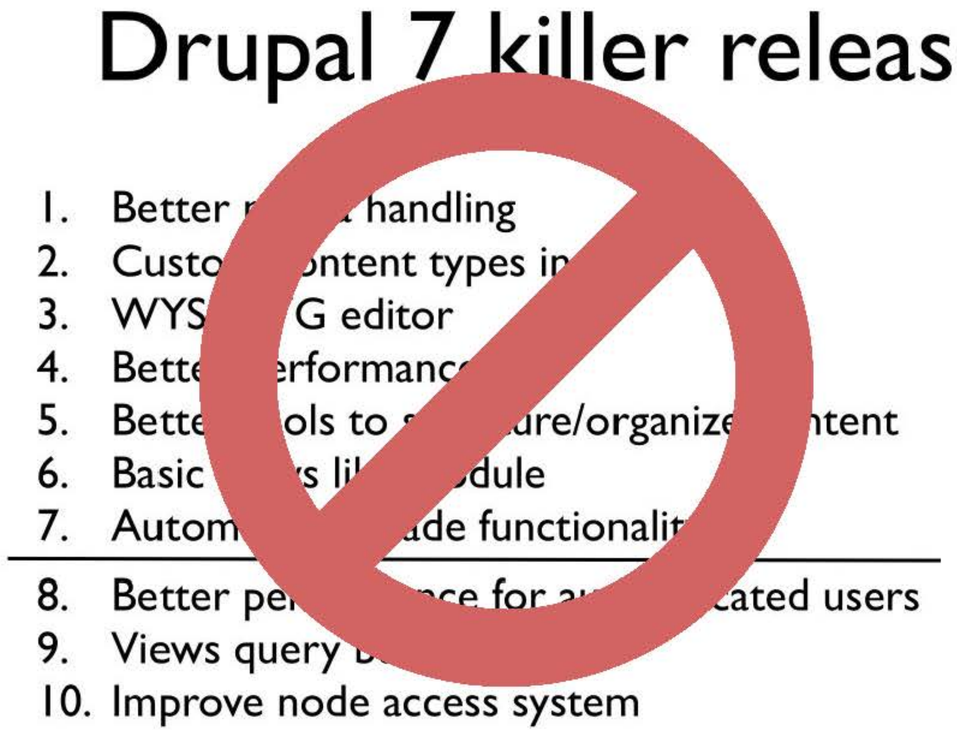
DX
70%

Can we get user experience right?

We need to focus more on

the non-coders

Experience is not just a list of features



It’s getting user effort

as close to zero as possible

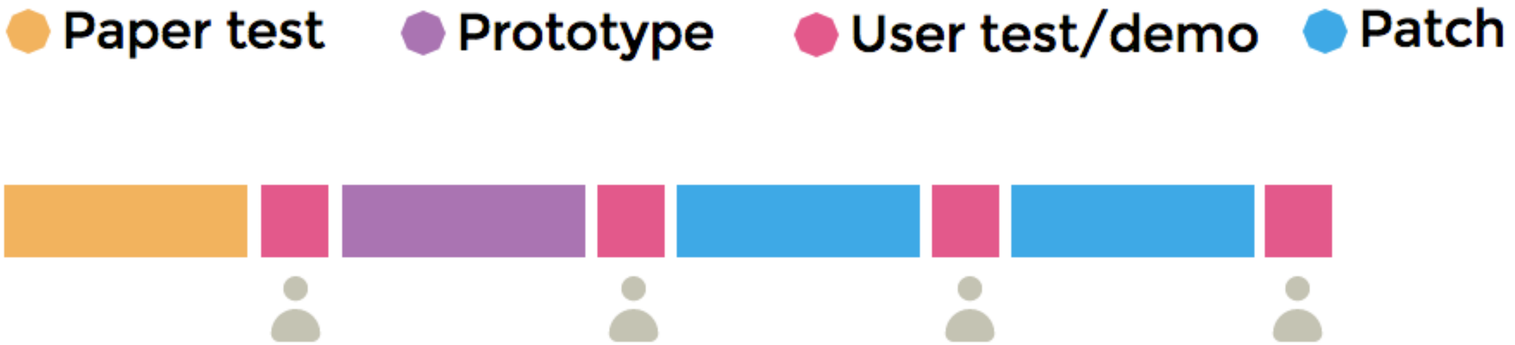
How do you know?

Let them try it!

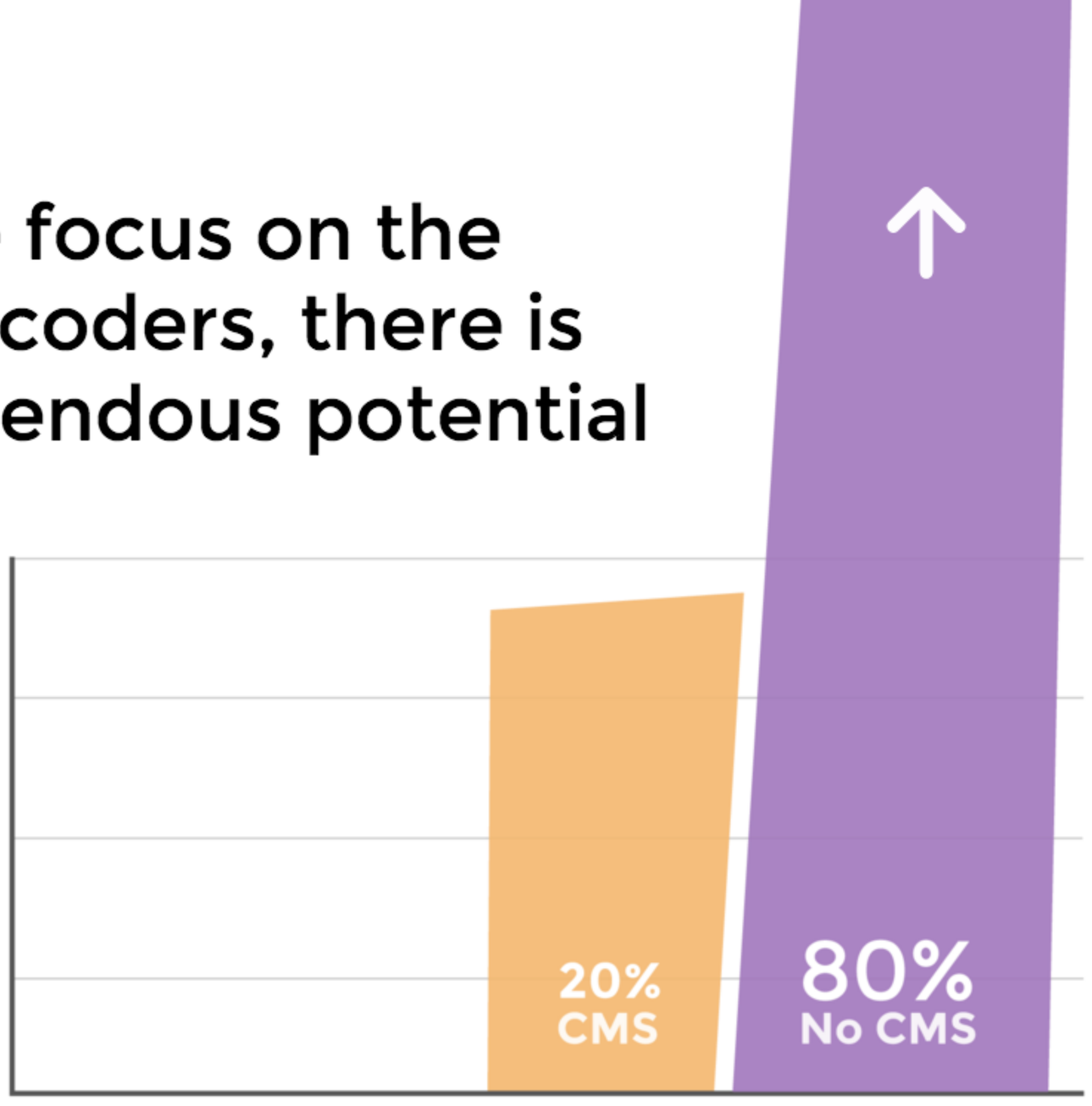
Expected life of a feature

Planned work Complete & shippable

Build, measure, learn



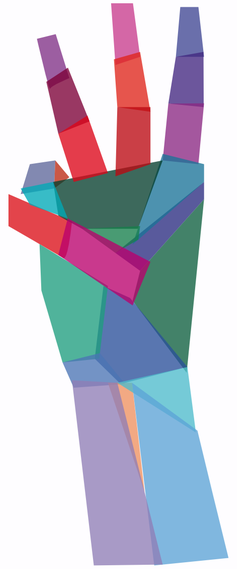
If we focus on the non-coders, there is tremendous potential



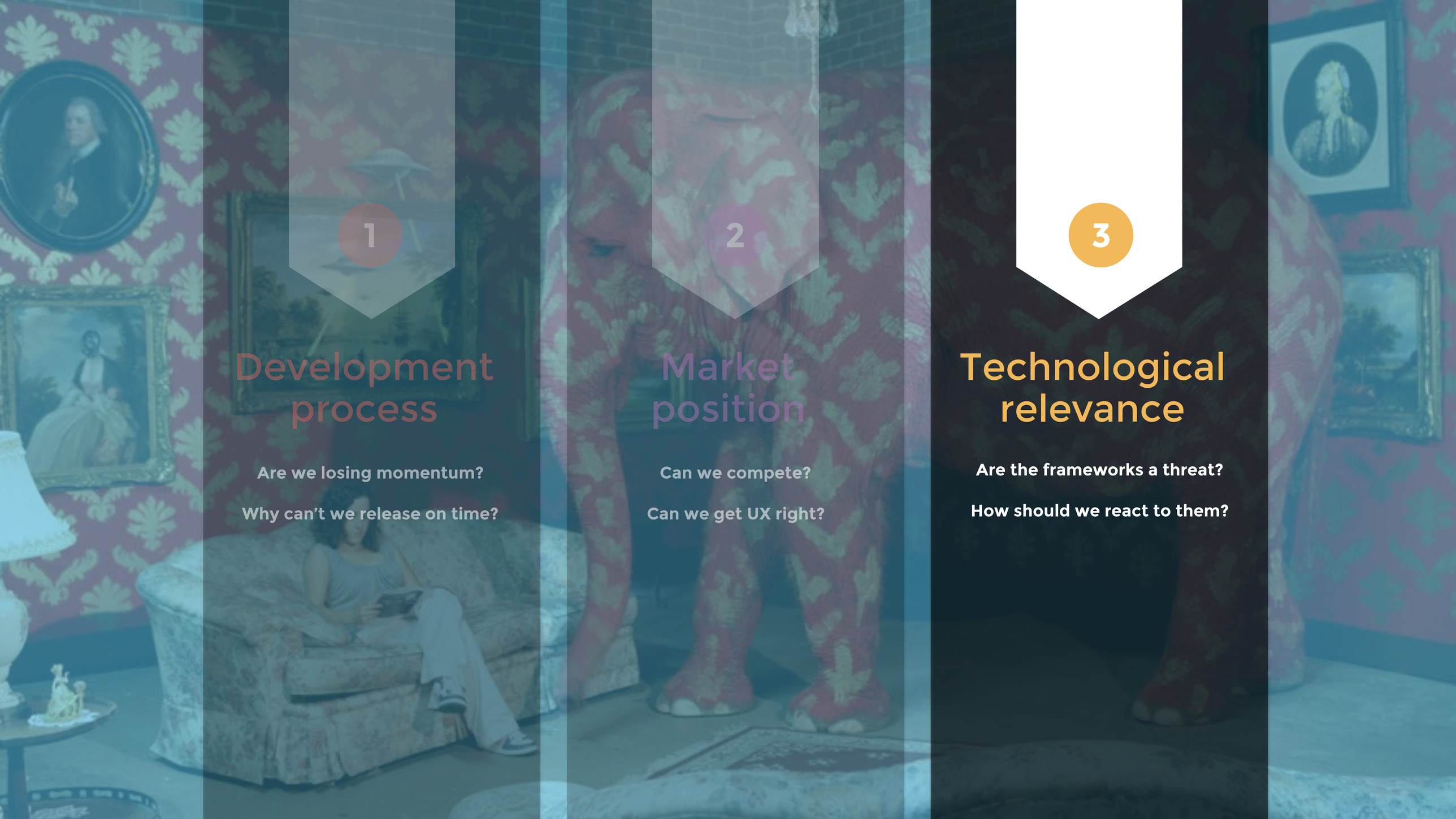
To sum up:

We're in a good position

But we still need to get better at UX



If we do, there's a big upside



1

Development process

Are we losing momentum?

Why can't we release on time?

2

Market position

Can we compete?

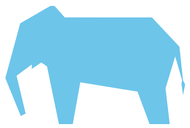
Can we get UX right?

3

Technological relevance

Are the frameworks a threat?

How should we react to them?



Client side apps

are a big trend and
they're not going away

3

Technological relevance

BACKBONE.JS

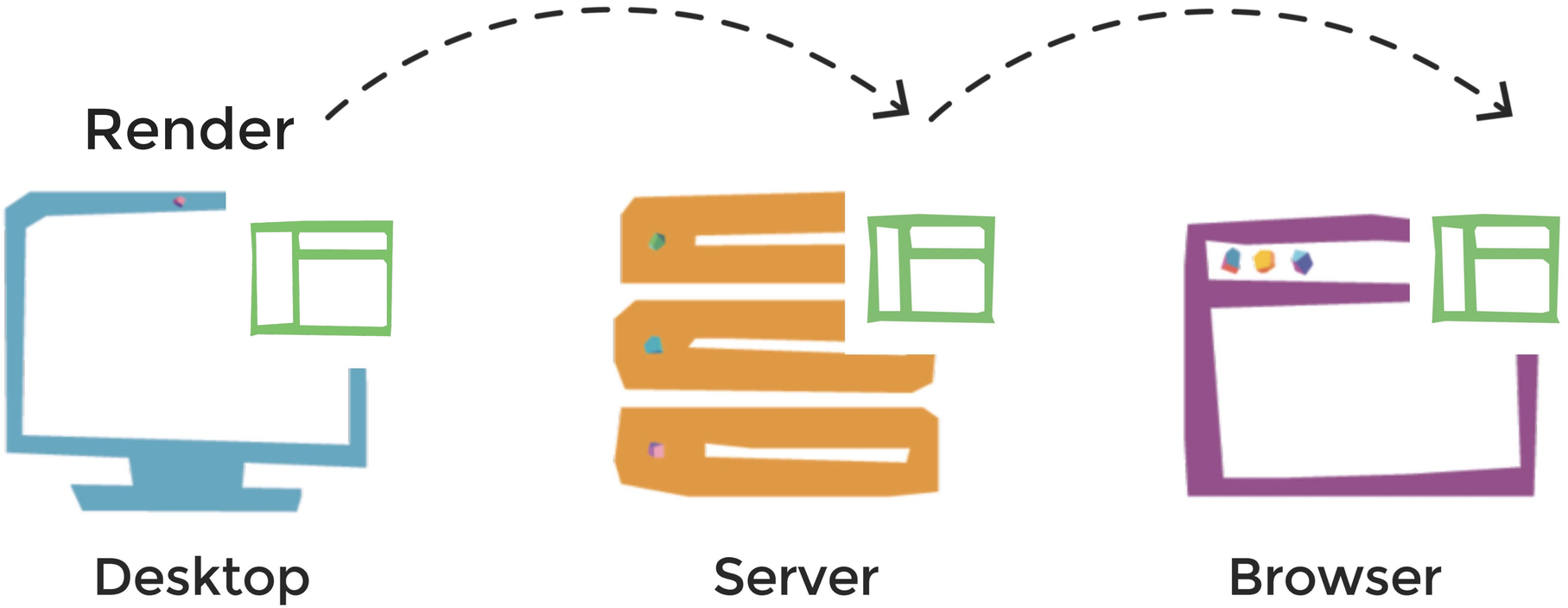
ANGULARJS by Google

React

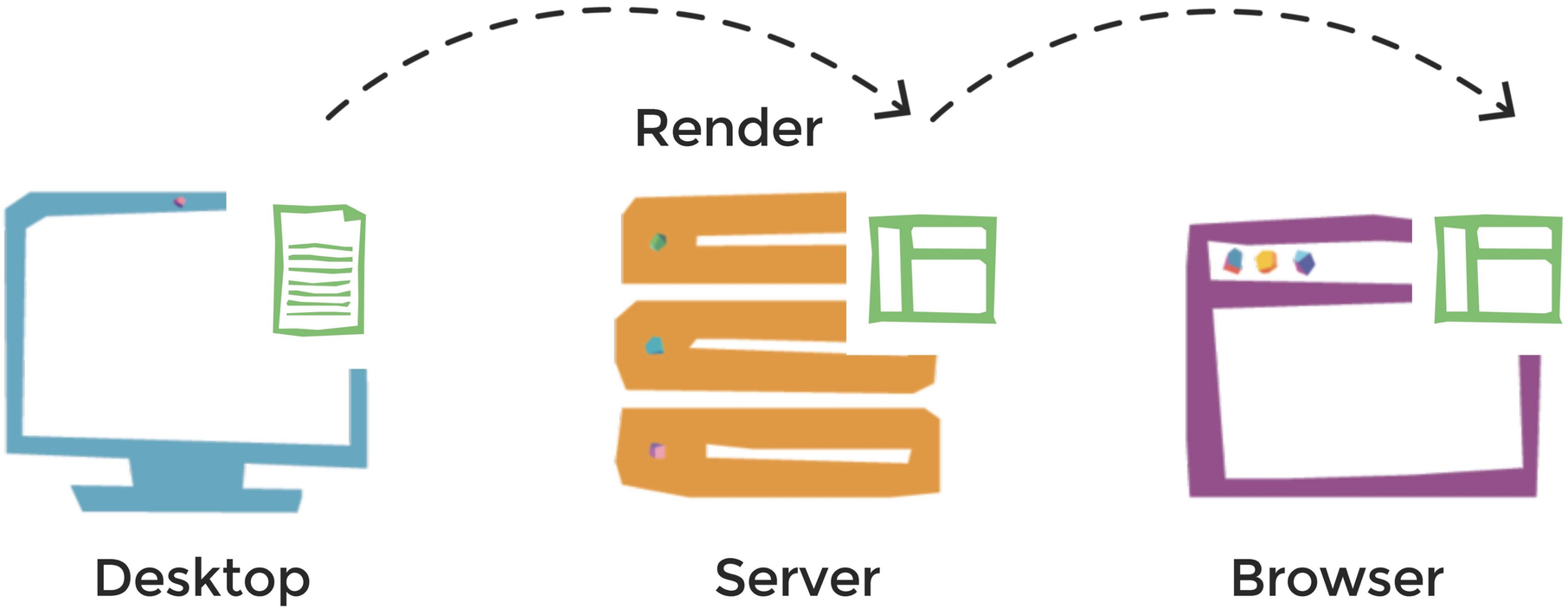
ember

A little web history

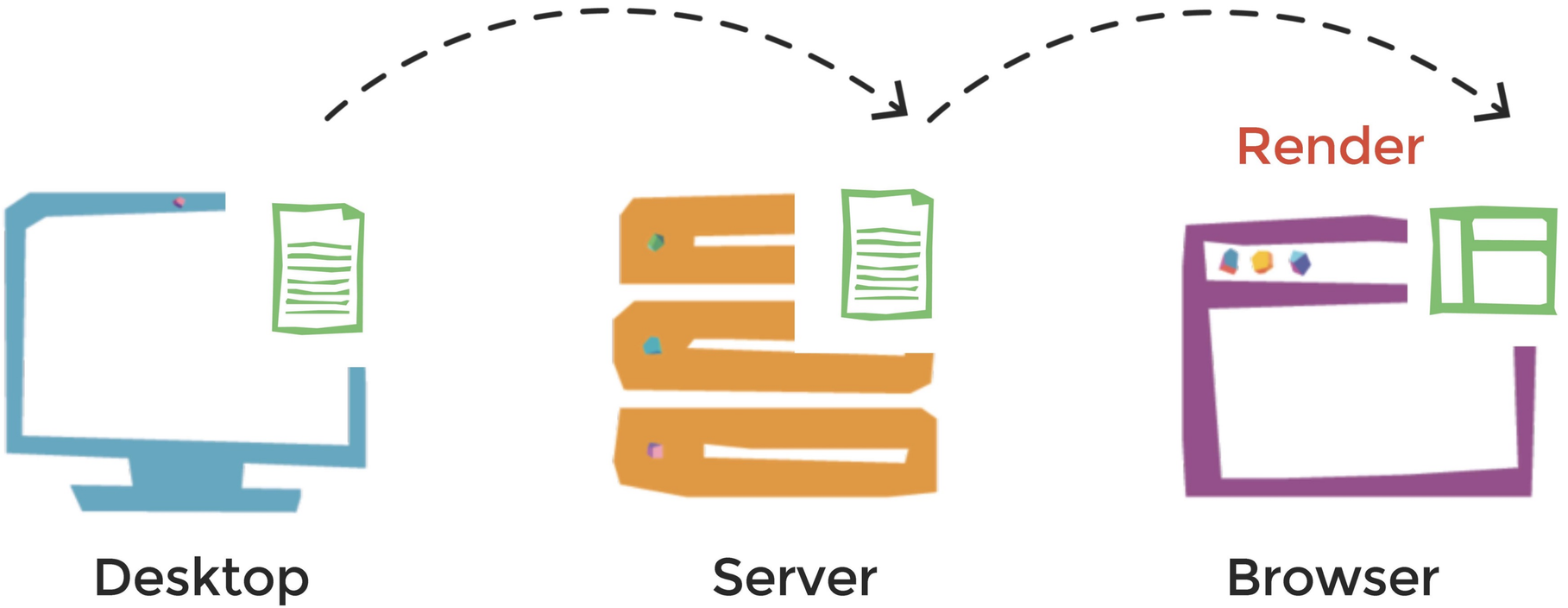
Early Web



Traditional CMS

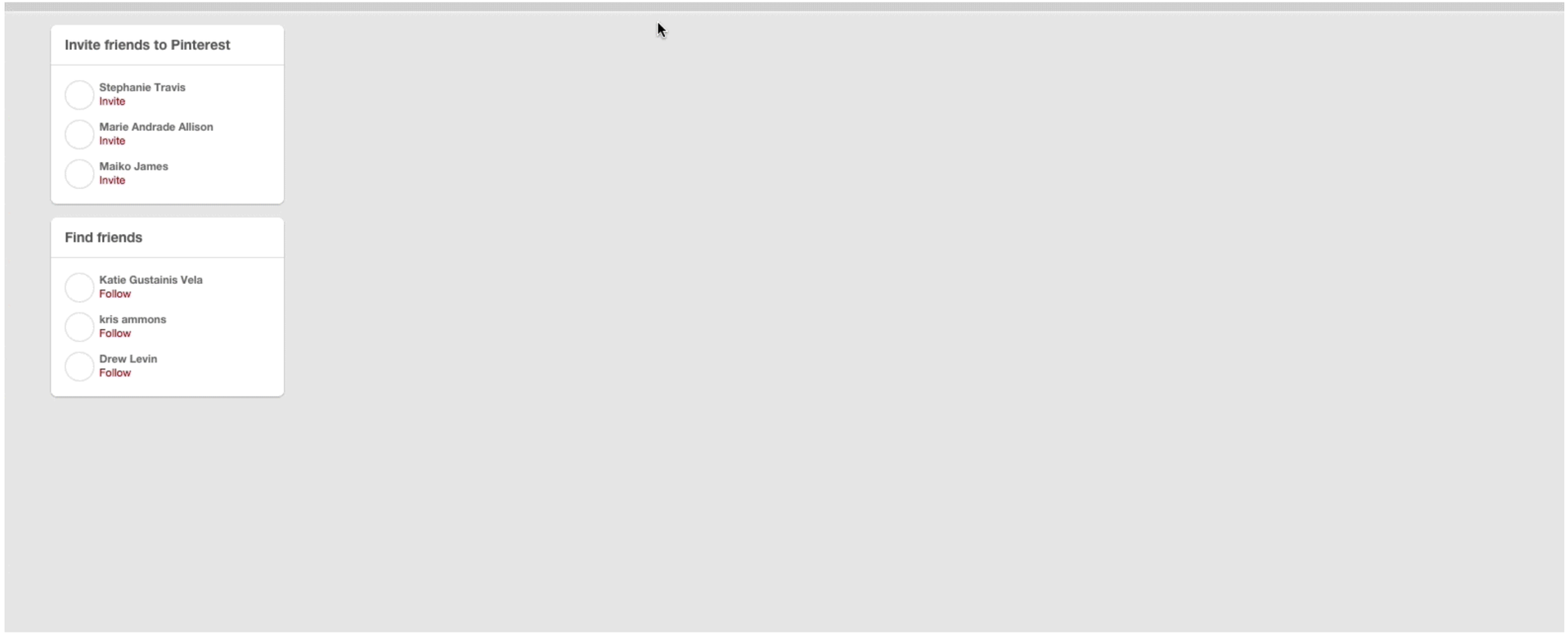


Client side apps



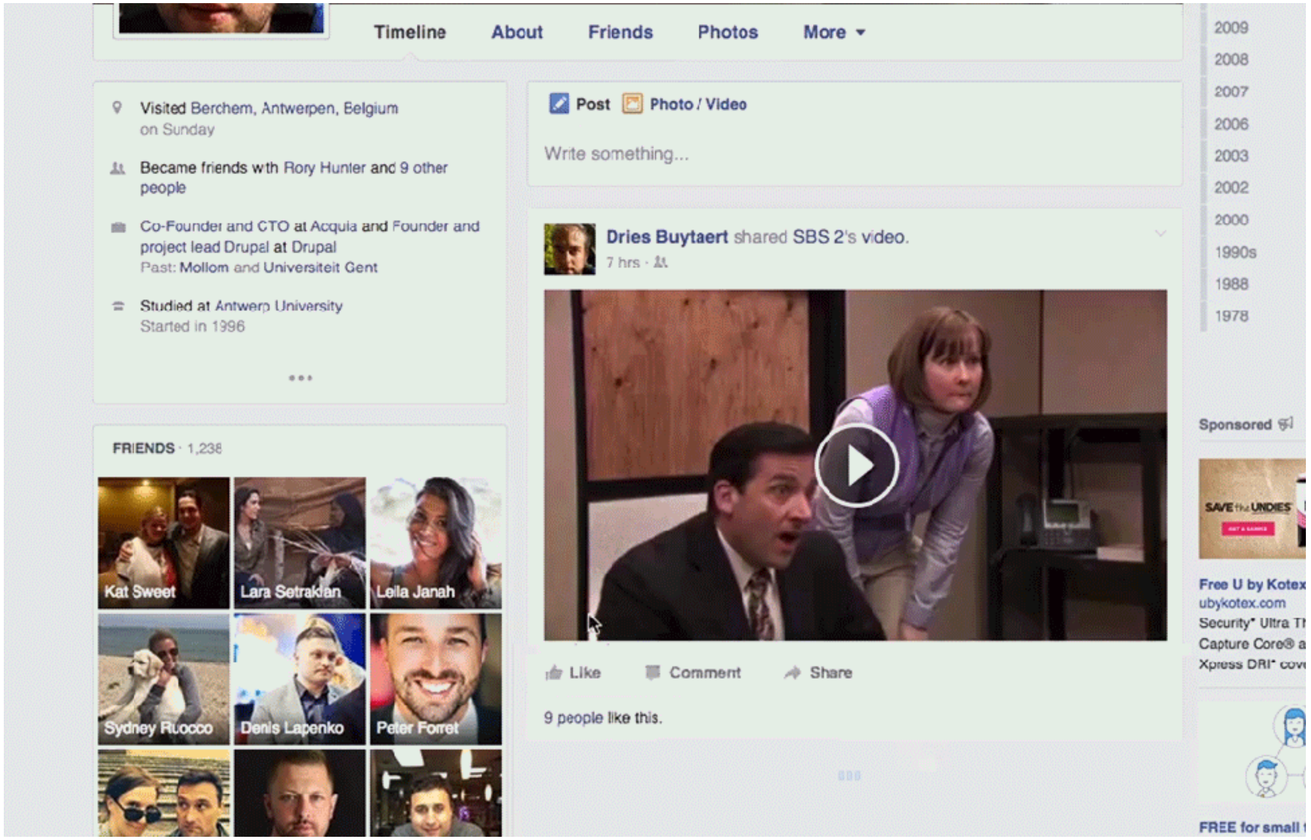
What are the benefits of client side apps?

Optimistic
Feedback



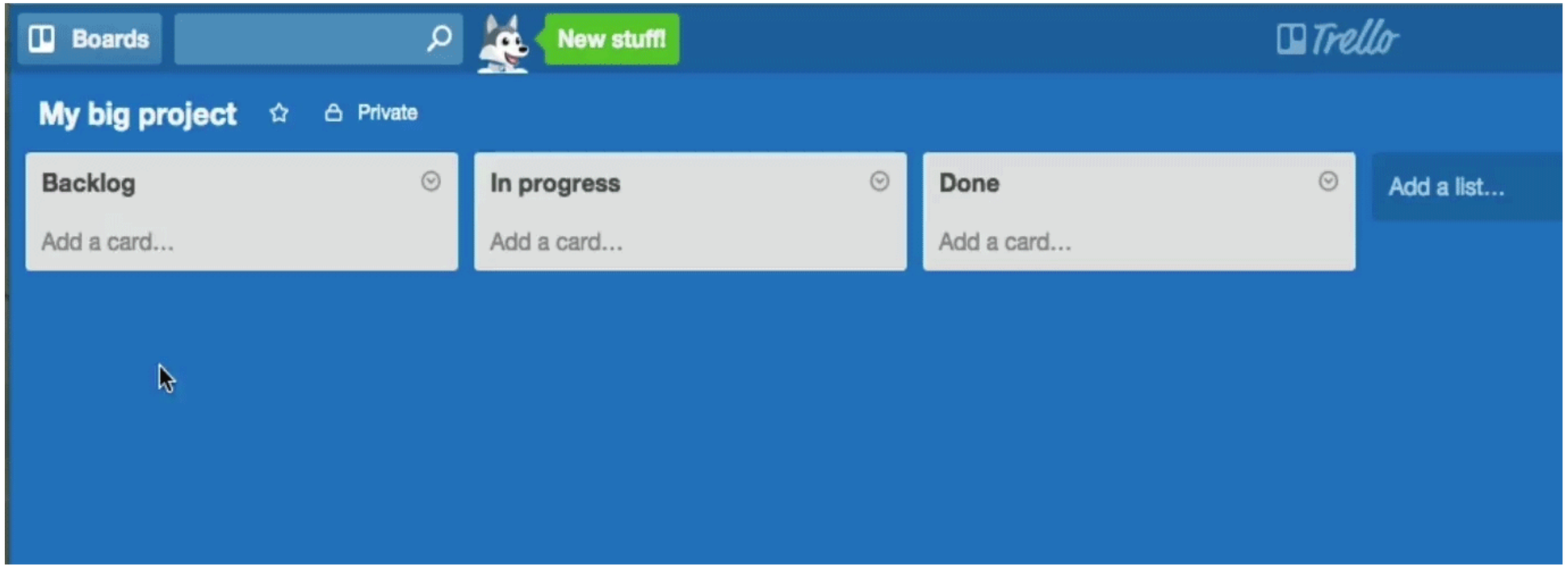
What are the benefits of client side apps?

Optimistic Feedback | Non-blocking user interfaces



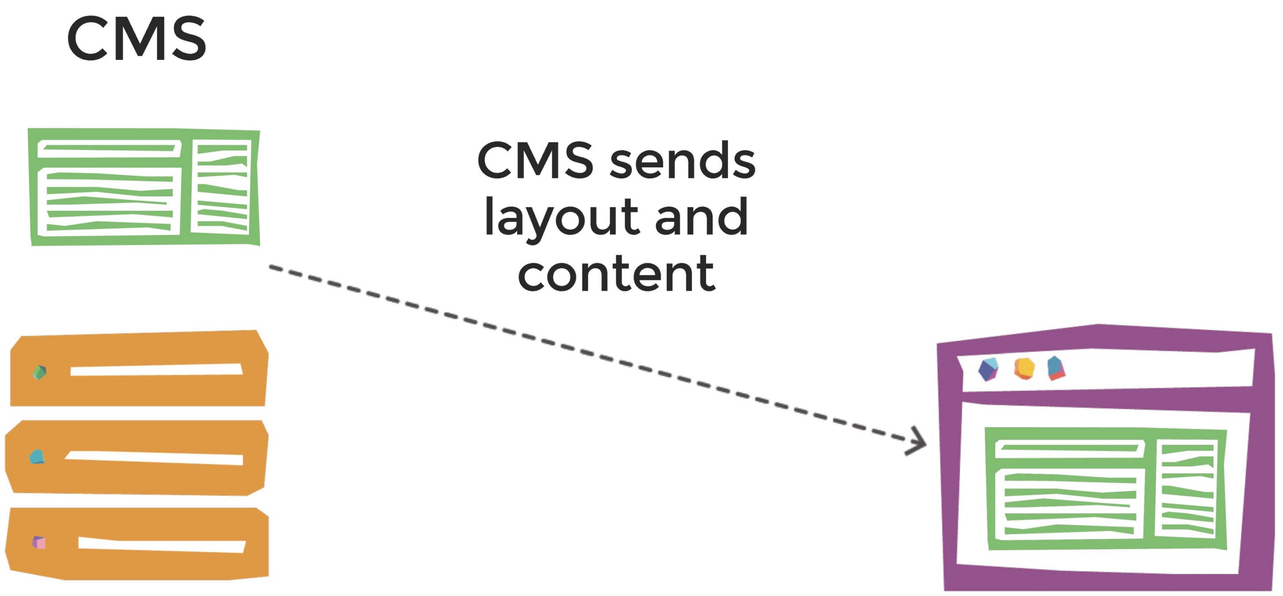
What are the benefits of client side apps?

Optimistic Feedback | Non-blocking user interfaces | Application Experience



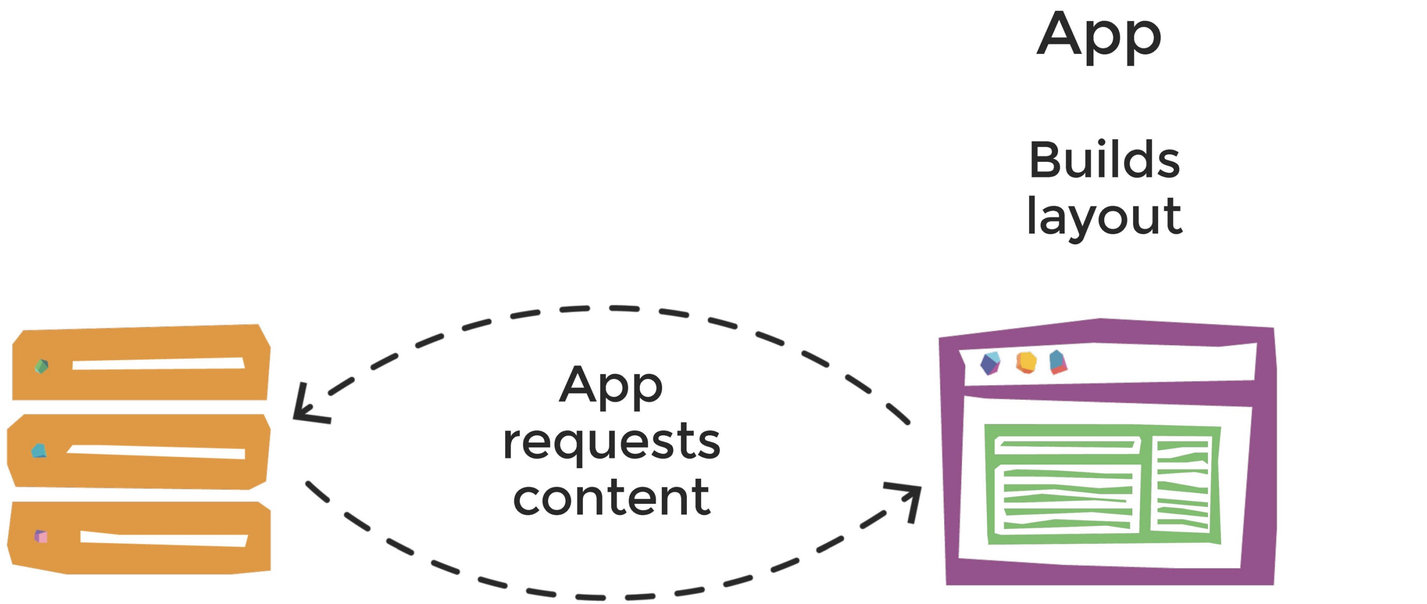
Architecture:

Traditional



Architecture:

Client side apps



BACKBONE.JS

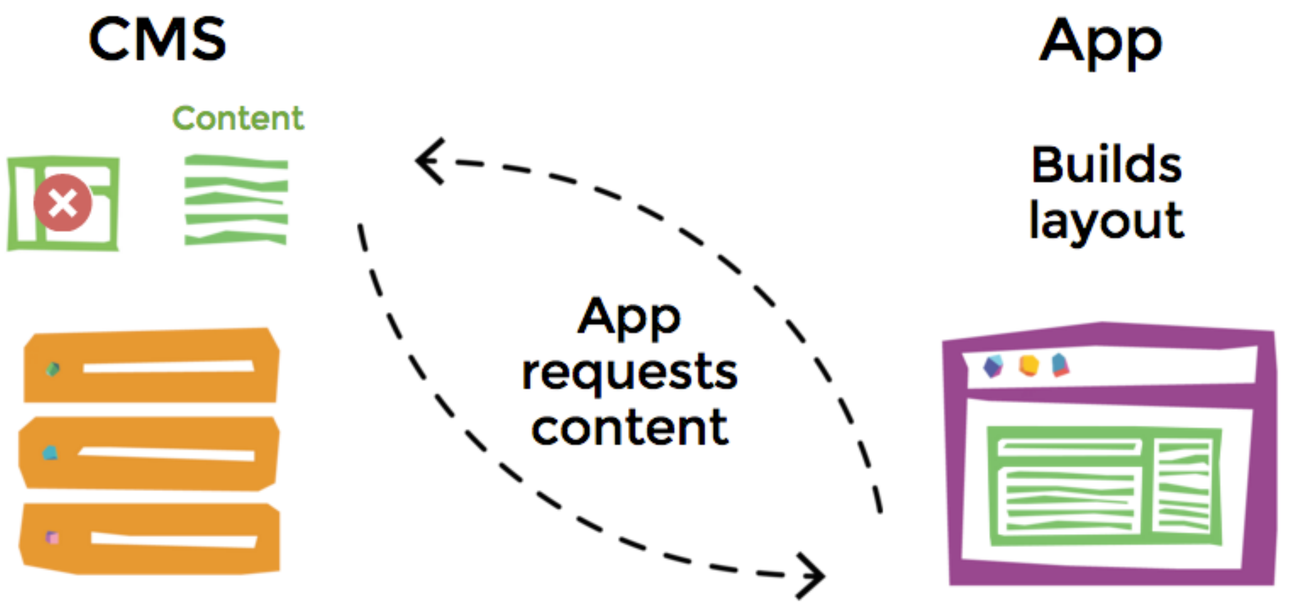
ANGULARJS by Google

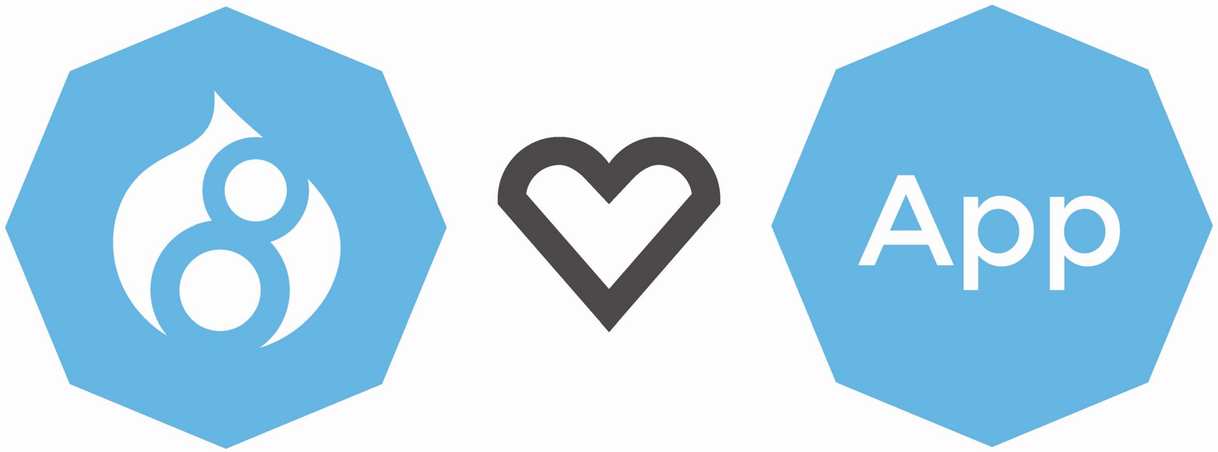
React

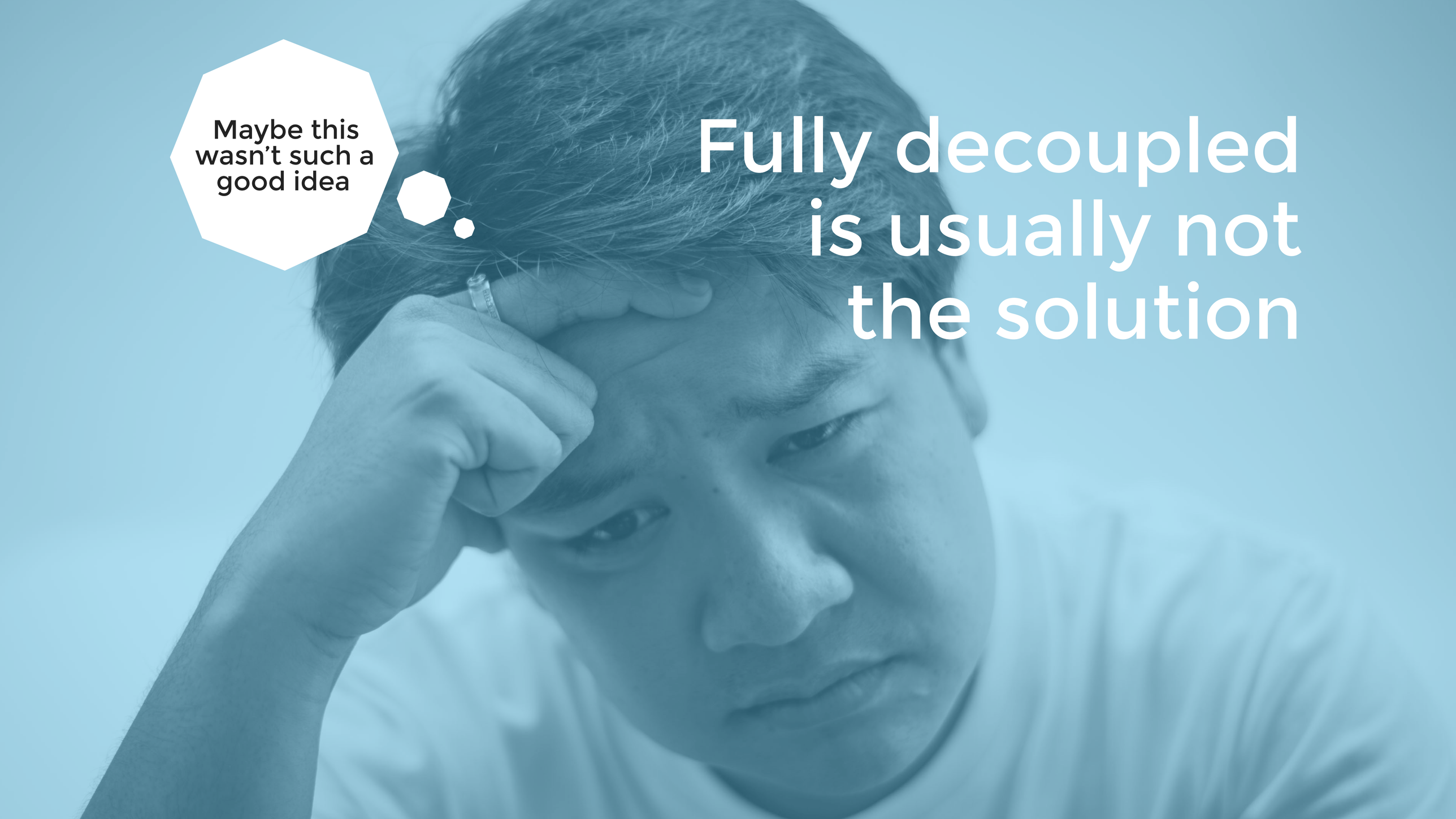
ember

Architecture:

Decoupled (CMS + Client side app)



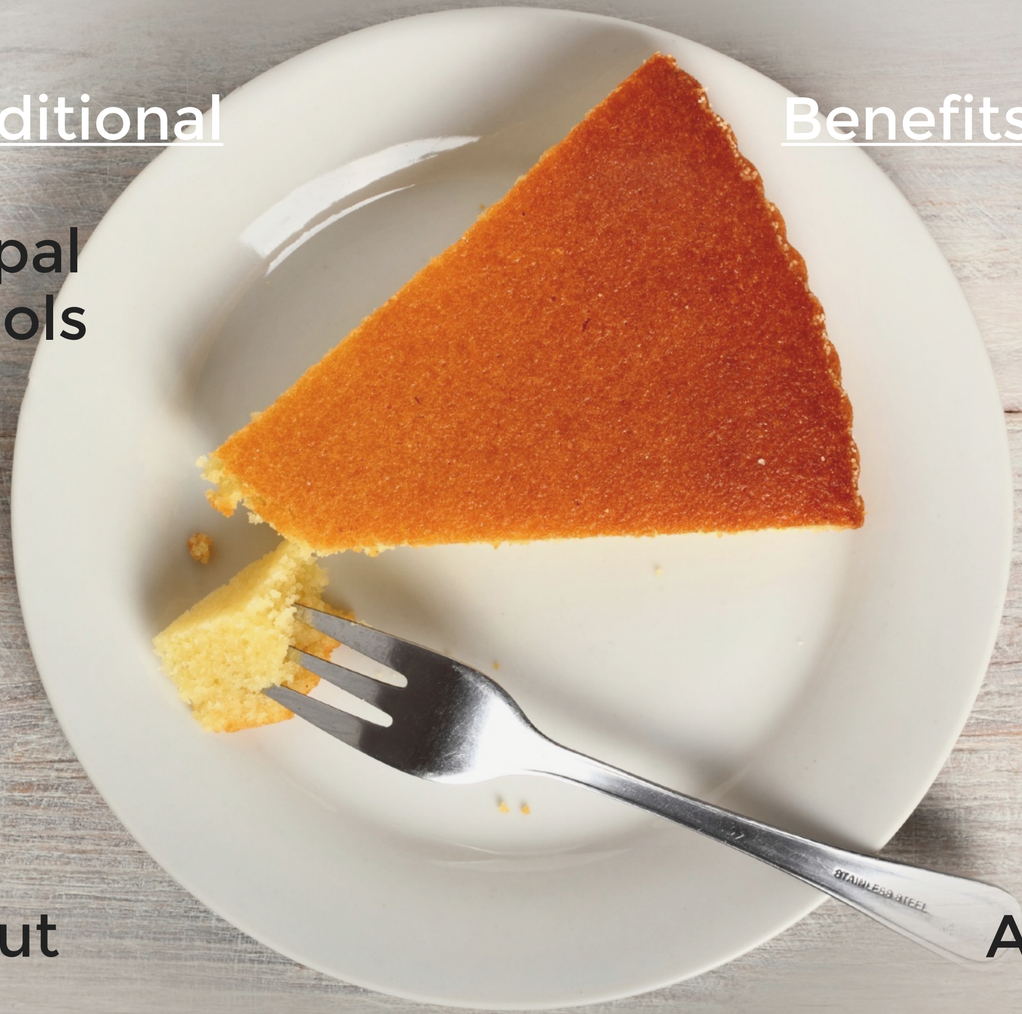

App



Fully decoupled is usually not the solution

You lose a lot

Can we have our cake and eat it?



Benefits of Traditional

All of your Drupal site building tools

Faster performance with BigPipe

All of your layout tools

Benefits of Decoupled

Optimistic feedback

Non-blocking interfaces

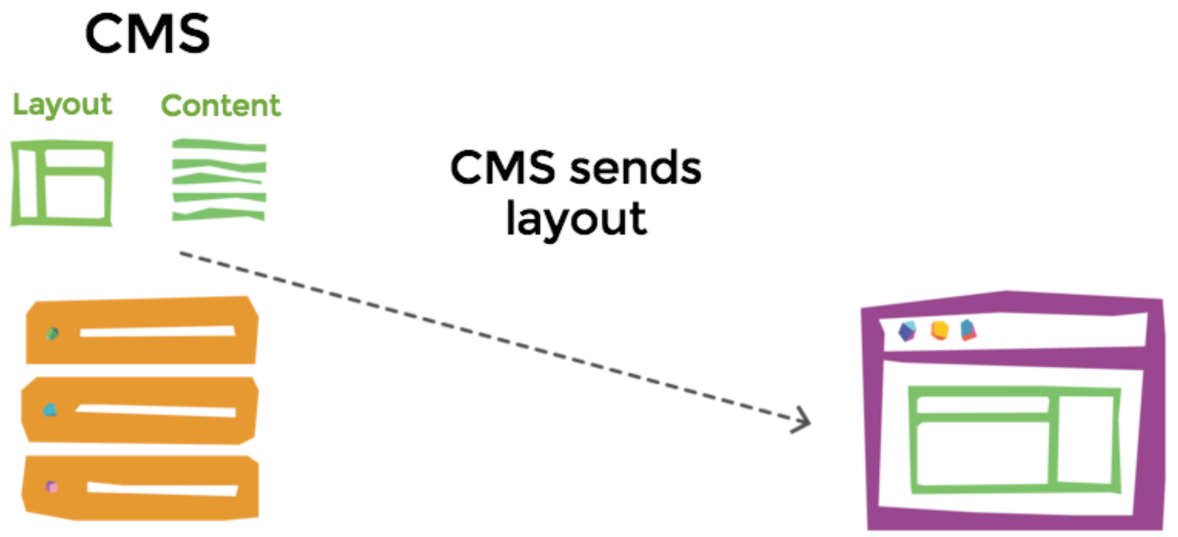
Application-like experience

presents:

Progressive Decoupling

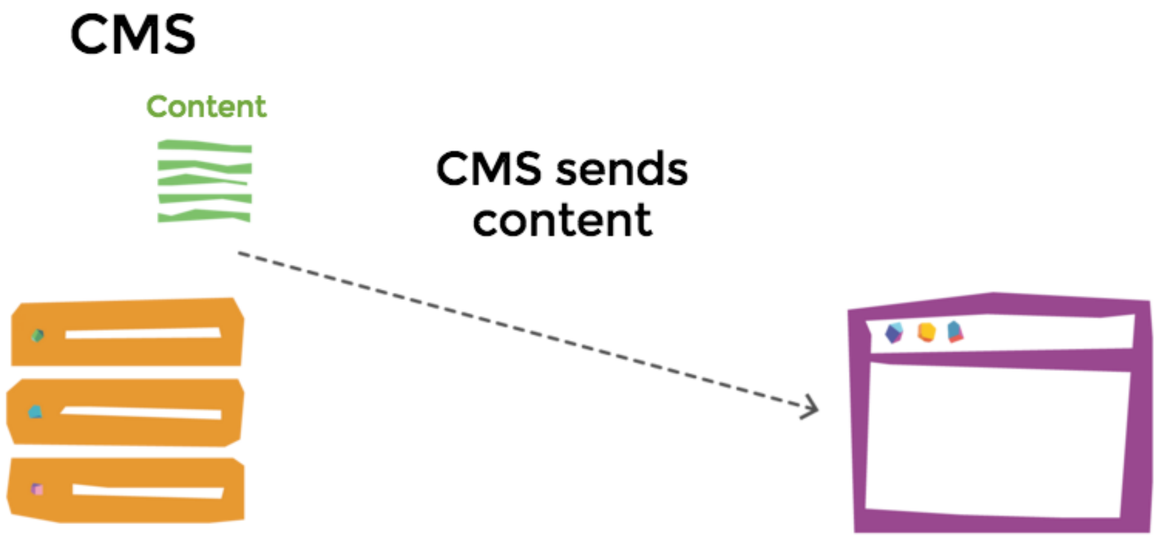
Architecture:

Progressive decoupling



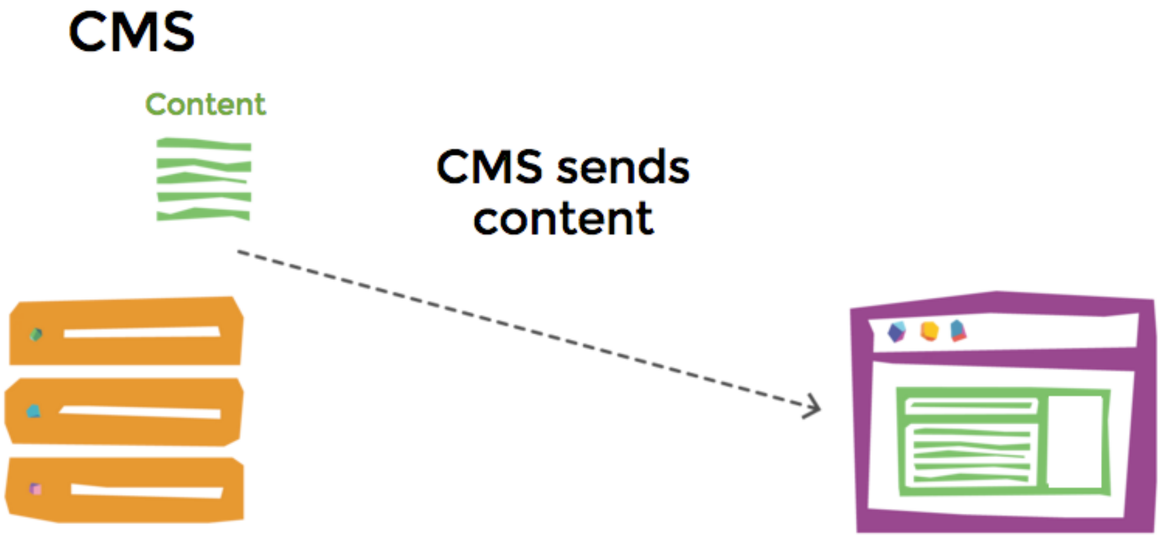
Architecture:

Progressive decoupling



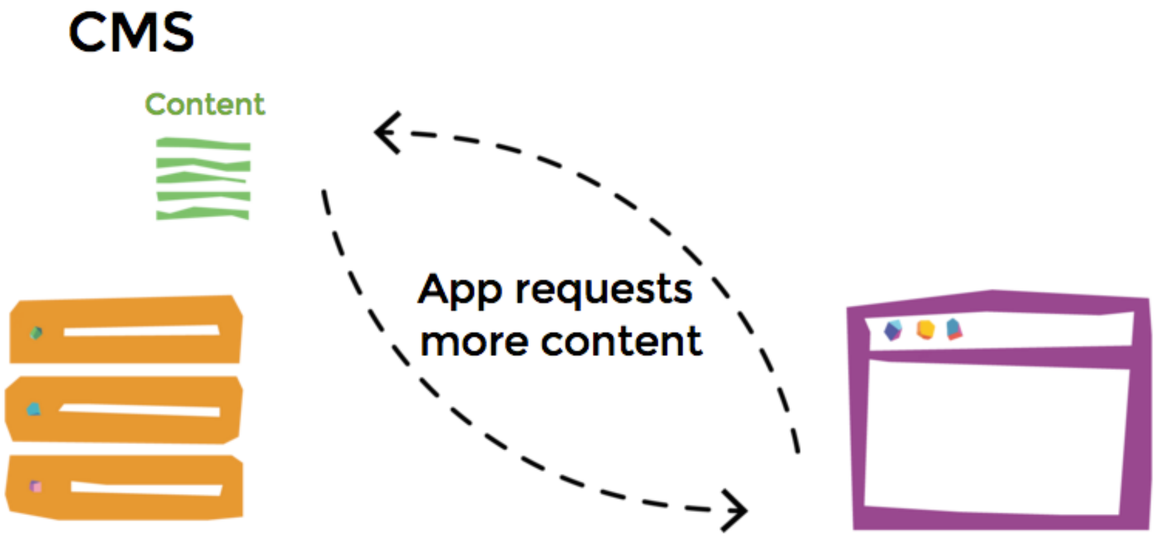
Architecture:

Progressive decoupling



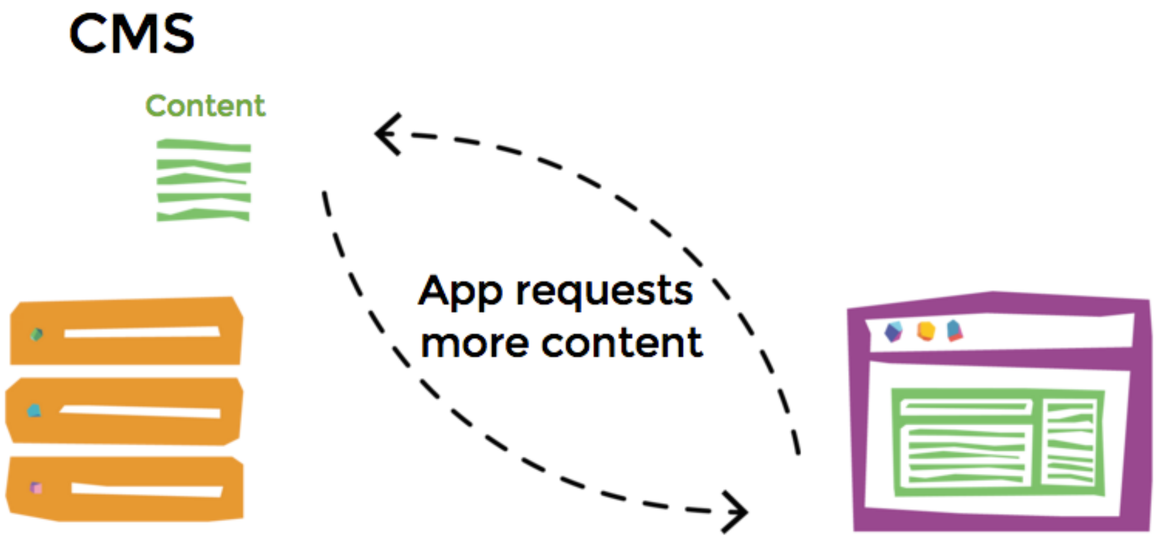
Architecture:

Progressive decoupling



Architecture:

Progressive decoupling



We have our options

open

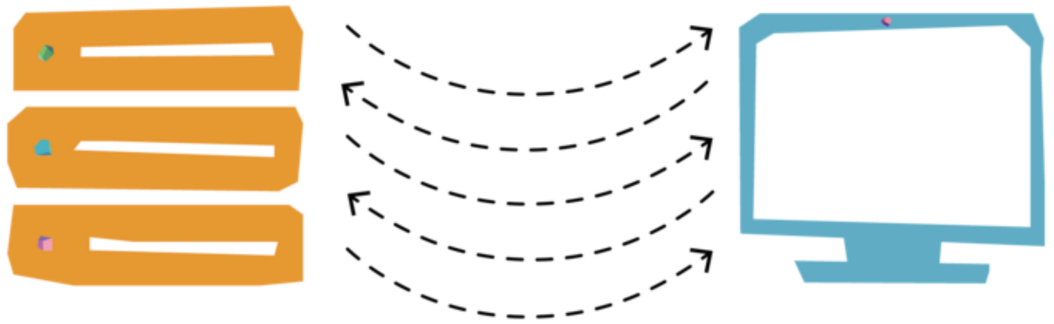
Build better front-ends with Backbone and Twig

Build progressively decoupled sites with page building tools

Feed your Drupal data to client side apps (native)

REST in Drupal 8

Great, but not perfect



Multiple round trips | Too much, or too brittle data | Poor dev experience

What we need:

One API call
for everything
you need and
nothing you
don't

Better dev
experience
for query
building

We have these!

Sebastian Siemssen 'fubhy'

Content | Site-Install

graphql.zensations.at/admin/content

Back to site | Manage | Shortcuts | admin | Edit

Content | Structure | Appearance | Extend | Configuration | People | Reports | Help

Content | Comments | Files

Home » Administration

+ Add content

Published status | Type | Title | Language | Region (field_region)

– Any – | – Any – | | – Any – | – Any –

Filter

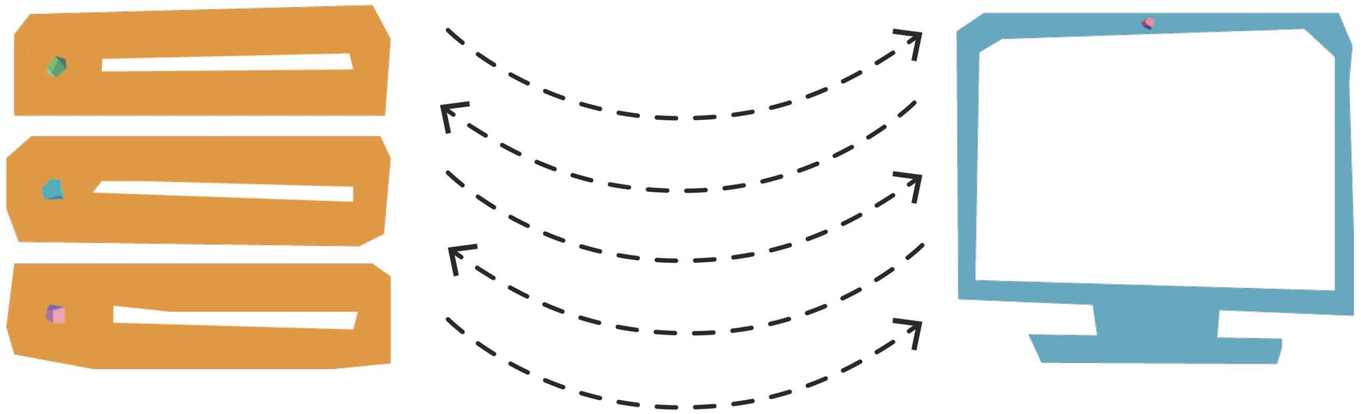
With selection

Delete content

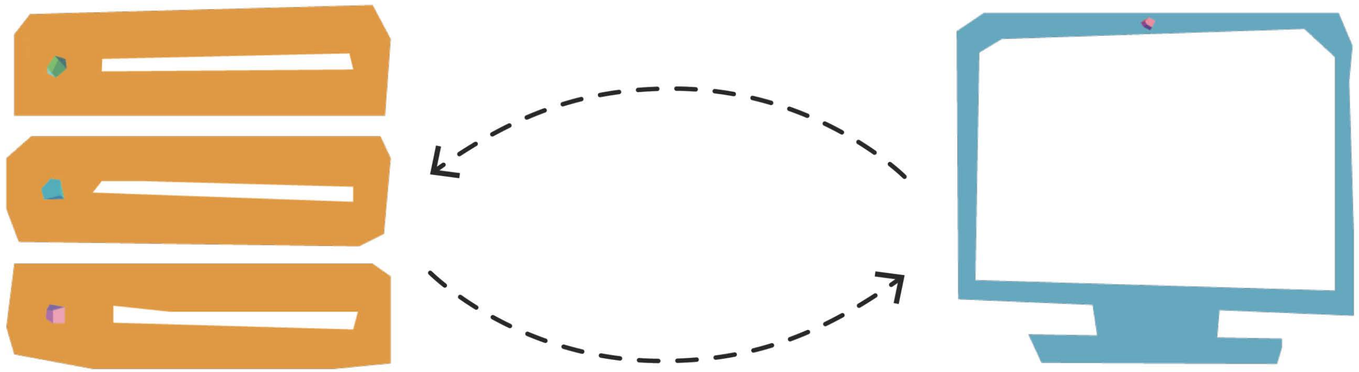
Apply

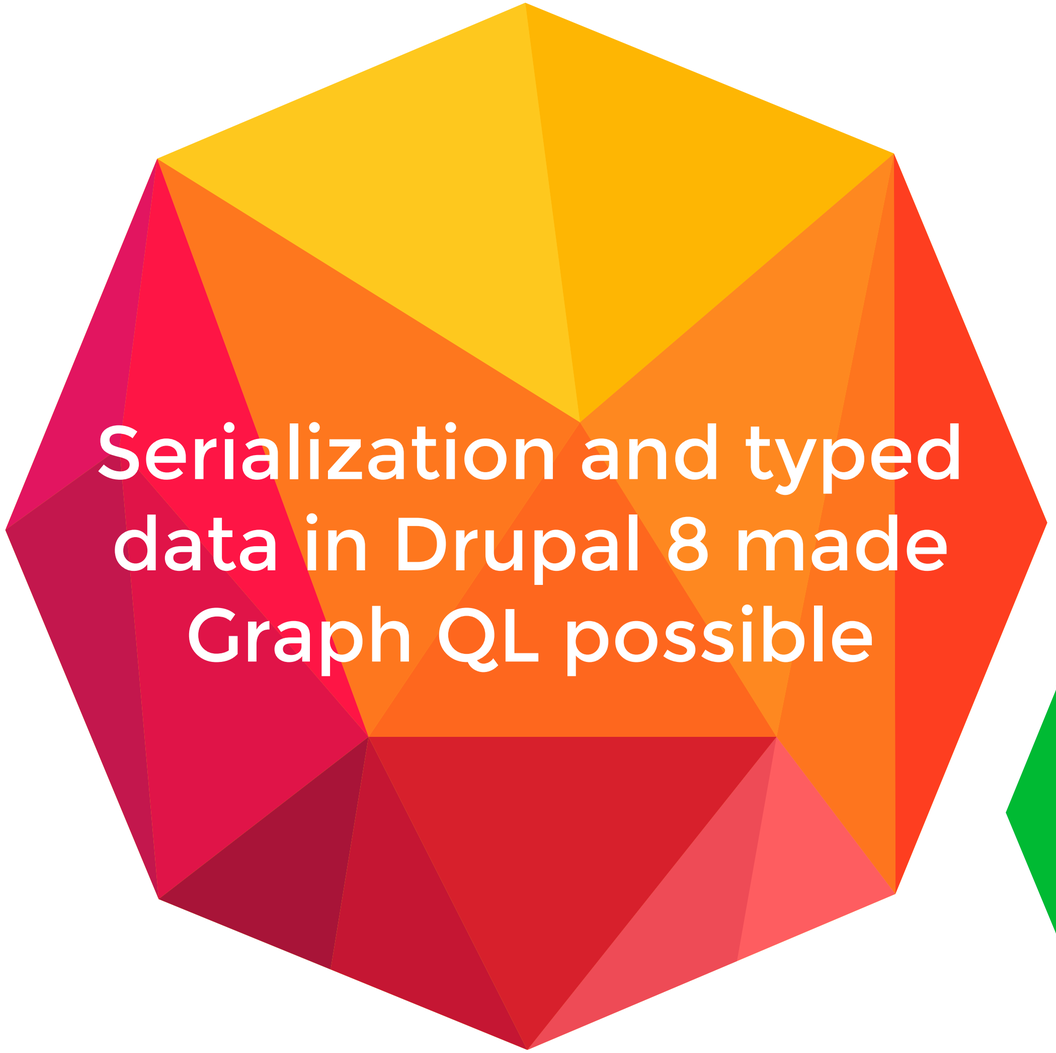
	TITLE	CONTENT TYPE	AUTHOR	STATUS	UPDATED	OPERATIONS
☐	Drupalcon	Conference	admin	Published	09/08/2015 – 15:10	EditDelete
☐	Drupalcon	Conference	admin	Published	09/08/2015 – 15:10	EditDelete

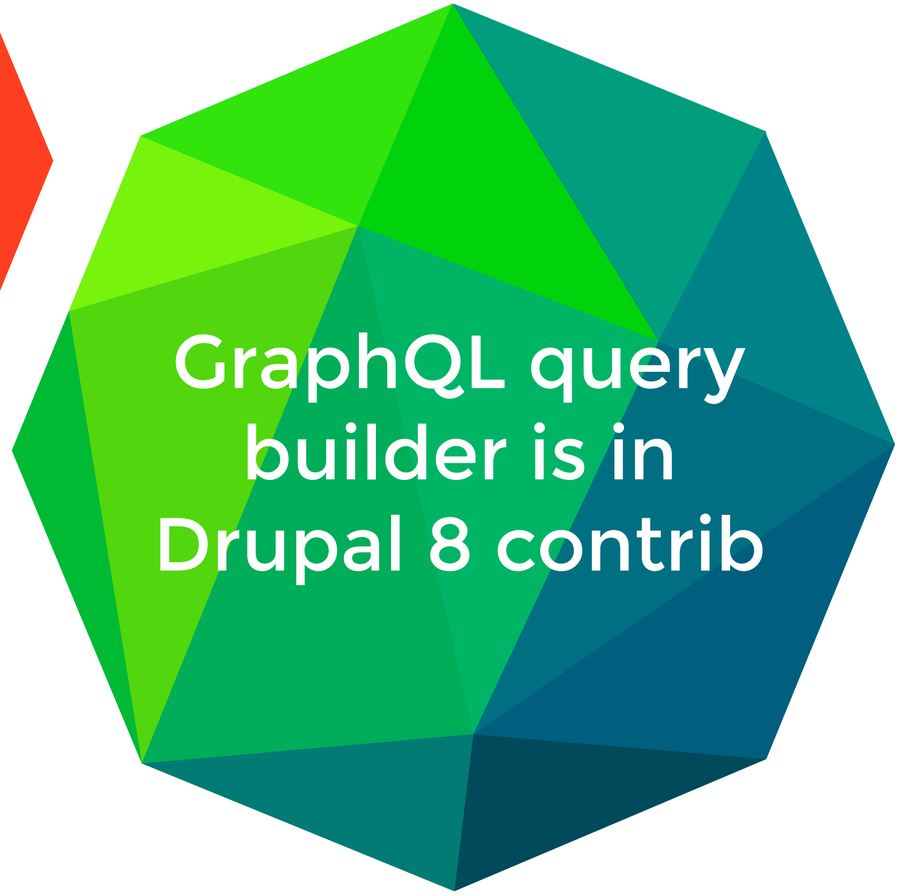
REST in Drupal 8



GraphQL in Drupal




Serialization and typed
data in Drupal 8 made
Graph QL possible


GraphQL query
builder is in
Drupal 8 contrib

Progressive decoupling and GraphQL puts us ahead of the competitors

Drupal 8

tecore

TYPO3

SQUARESPACE

Adobe Experience Manager

WordPress

Competitors

ghost

CLOUD CMS

contentful
future. friendly. content.

prismic.io

Innovators

E.JS

React

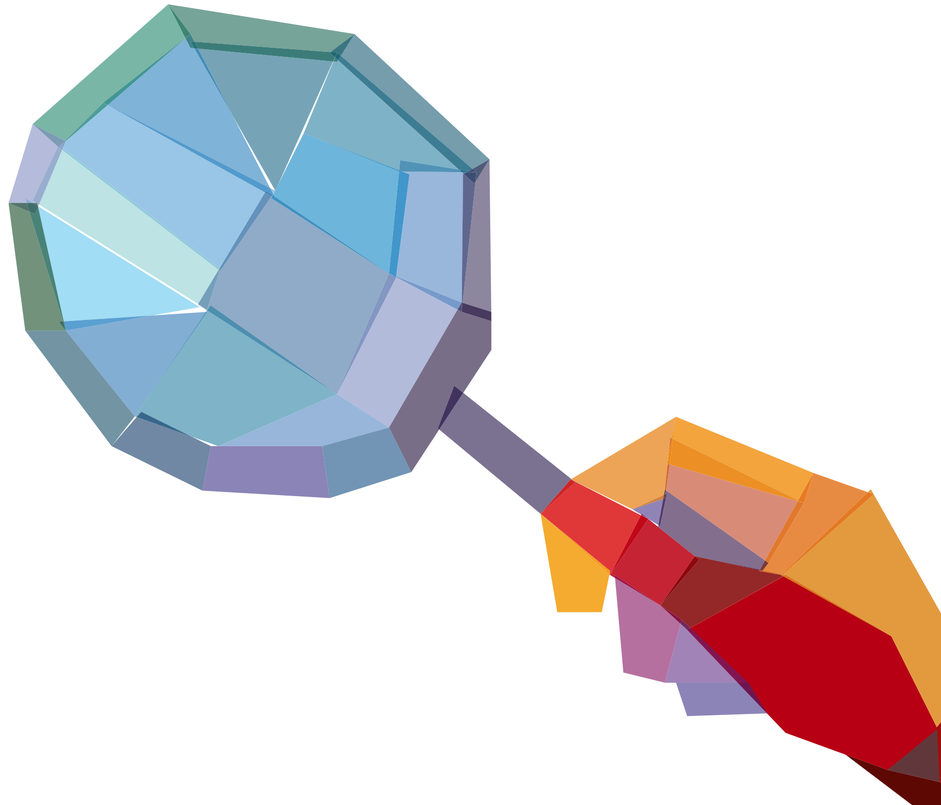
ember

ANGULARJS
by Google

Javascript MVCs

This is not a little shift, it's a big shift

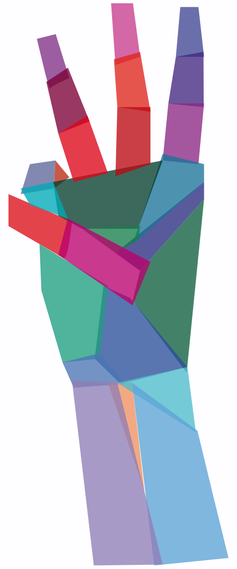
We need to keep exploring decoupled architectures



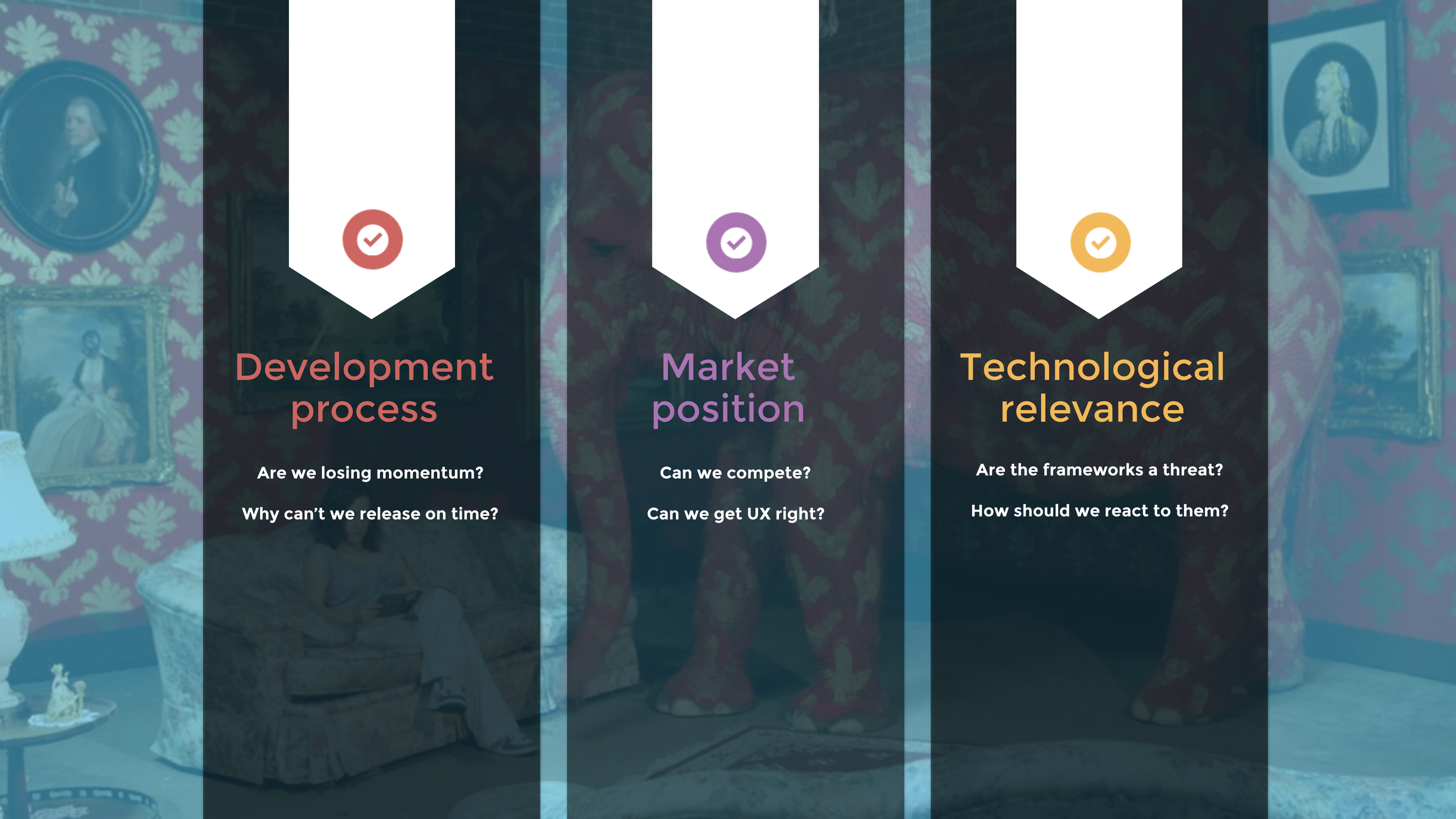
To sum up:

Fully decoupled is not always the best solution

Progressive decoupling is the best of both worlds



Drupal 8 + GraphQL is ideal for apps and sites



Development process

Are we losing momentum?

Why can't we release on time?

Market position

Can we compete?

Can we get UX right?

Technological relevance

Are the frameworks a threat?

How should we react to them?

The take-aways

1. Regain Drupal 8 and momentum will come

The take-aways

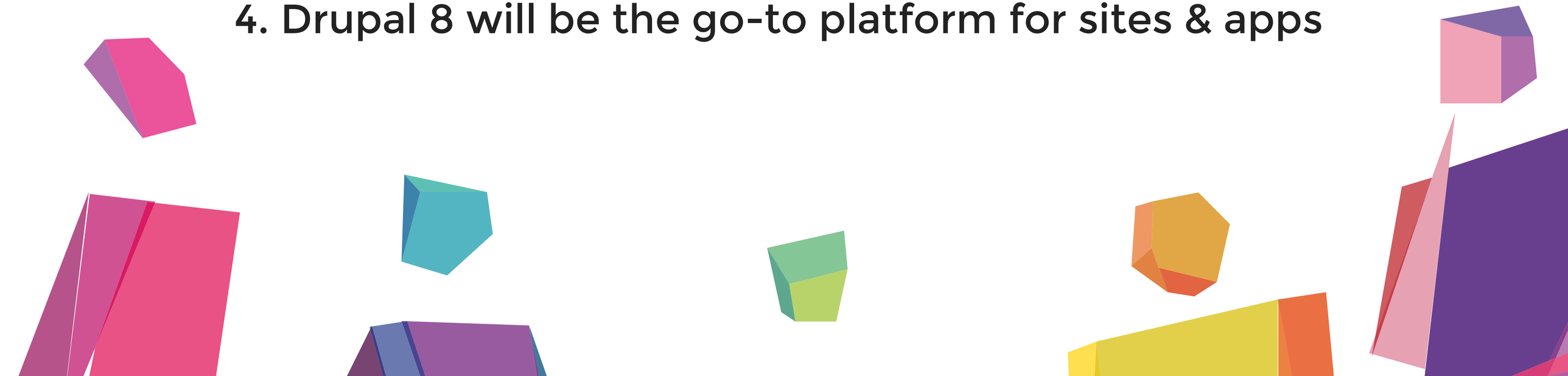
1. Regain Drupal 8 and momentum will come
2. Move to a more sustainable release process

The take-aways

1. Regain Drupal 8 and momentum will come
2. Move to a more sustainable release process
3. Put non-coders first to increase our impact

The take-aways

1. Regain Drupal 8 and momentum will come
2. Move to a more sustainable release process
3. Put non-coders first to increase our impact
4. Drupal 8 will be the go-to platform for sites & apps

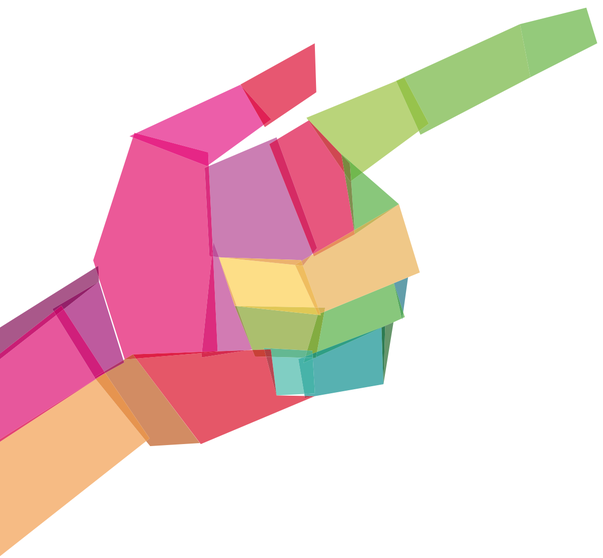


One final (Dries) note:

3.1 billion are online

With 1 in 40 sites
3.1 billion people online
Drupal touches
almost all of them

By 2018 there will be
5 BILLION



Want more details?

http://buytaert.net

Thanks!

dries@buytaert.net // @Dries
#Driesnote