The Search for Better Glue

Taylor Barnett (@taylor_atx)

LonghornPHP 2021

The Search for Better Glue

Taylor Barnett (@taylor_atx)

LonghornPHP 2021

Hypothesis:

People dislike glue work because of bad, past experiences

What is Glue?

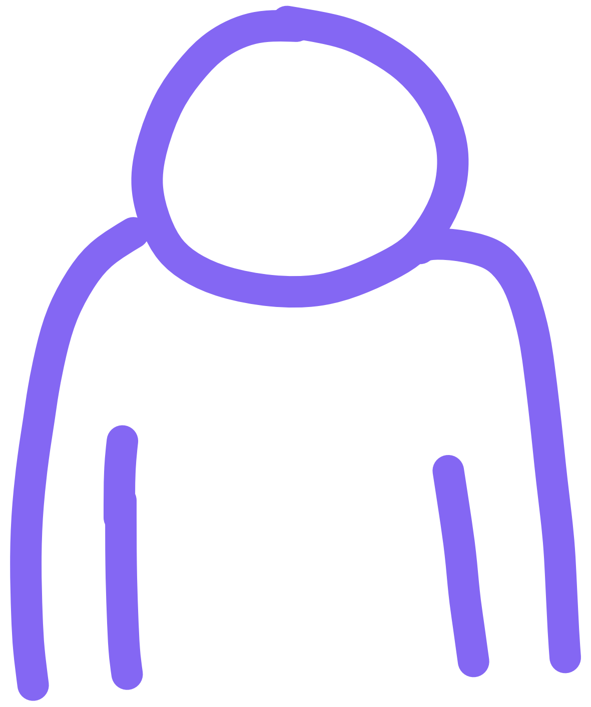
Glue (verb): To integrate different parts of a system together that would otherwise be incompatible

(also a noun)

"...the less glamorous - and often less-promotable - work that needs to happen to make a team successful."

-Tanya Reilly

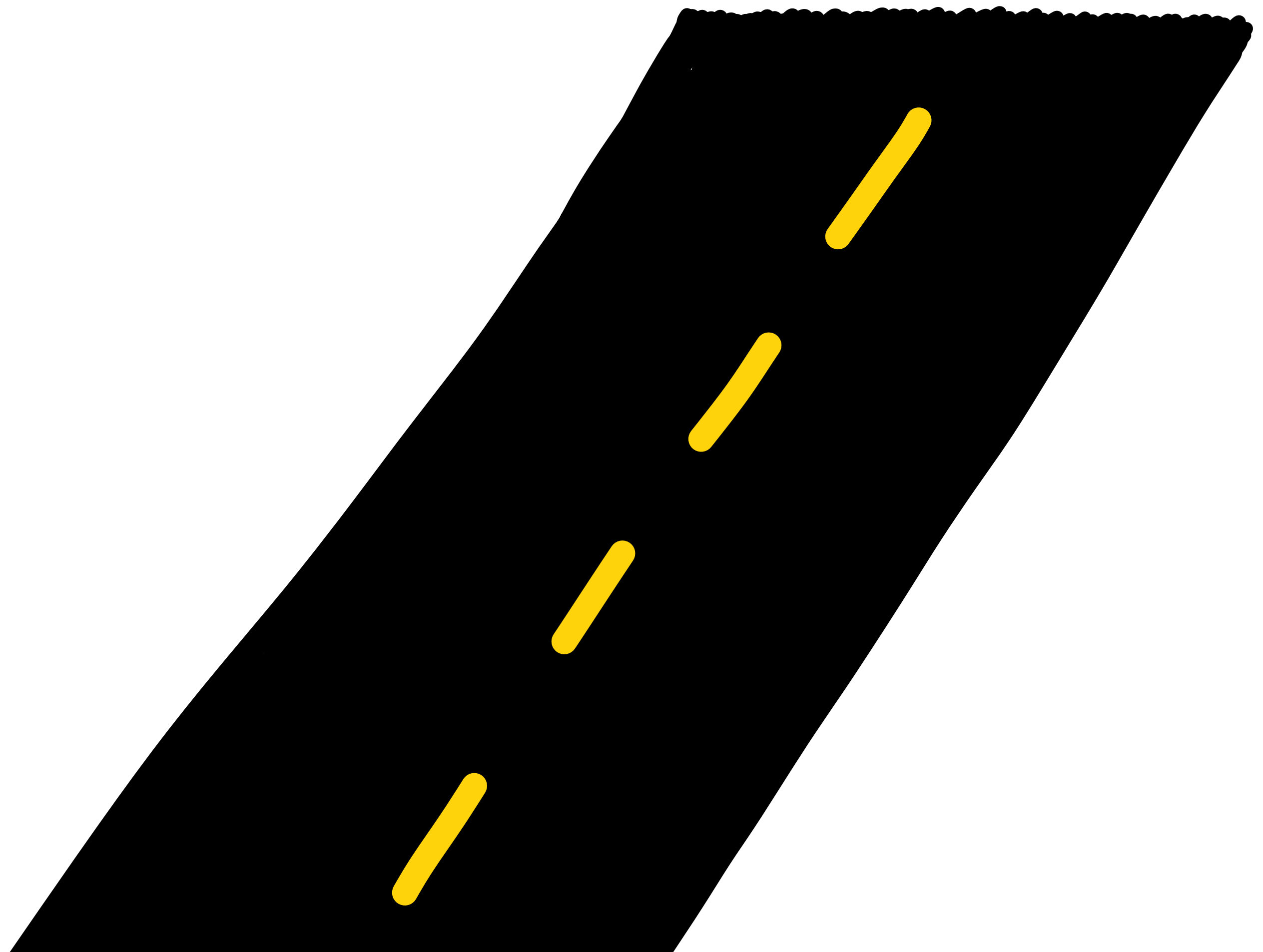
Human



Design

Code

The road to improving glue work has been narrow, slow, and with many road blocks



Patchwork of tools and services

Glue is complex

We often have to solve for glue ourselves

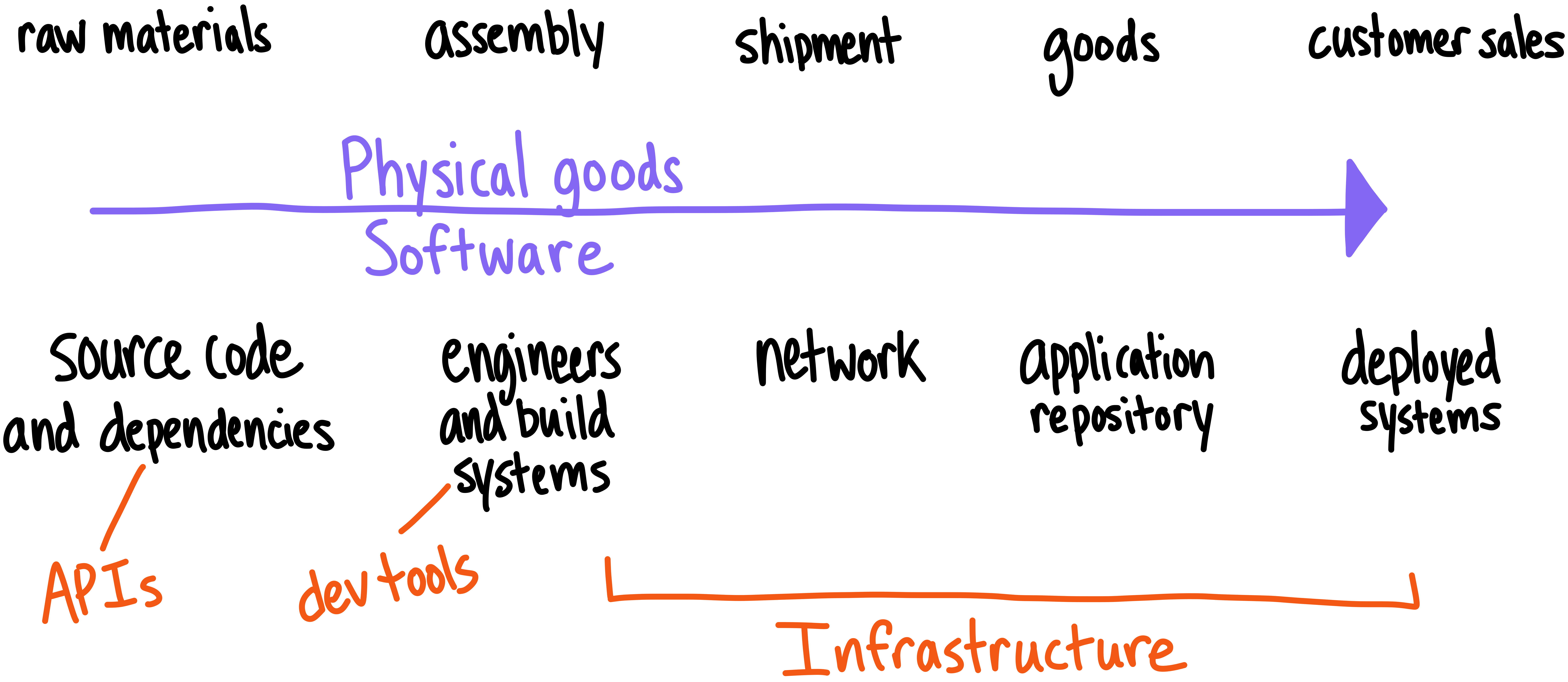
“And you may ask yourself,
“Well... how did [we] get here?”

-Talking Heads

First there was AWS...

First there was AWS...

Software supply chain



raw materials

assembly

shipment

goods

customer sales

Physical goods

Software

Source code and dependencies

engineers and build systems

network

application repository

deployed systems

APIs

dev tools

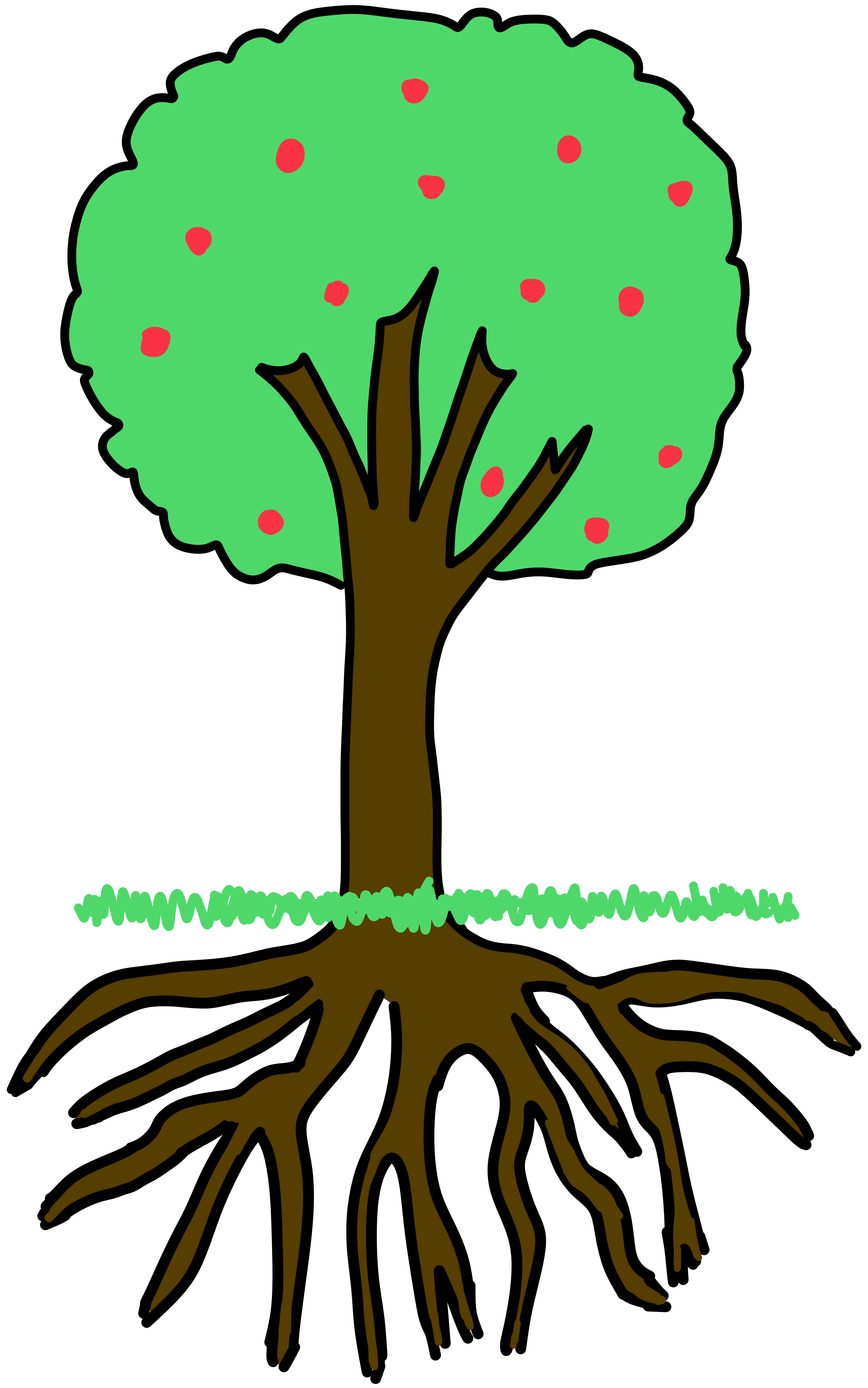
Infrastructure

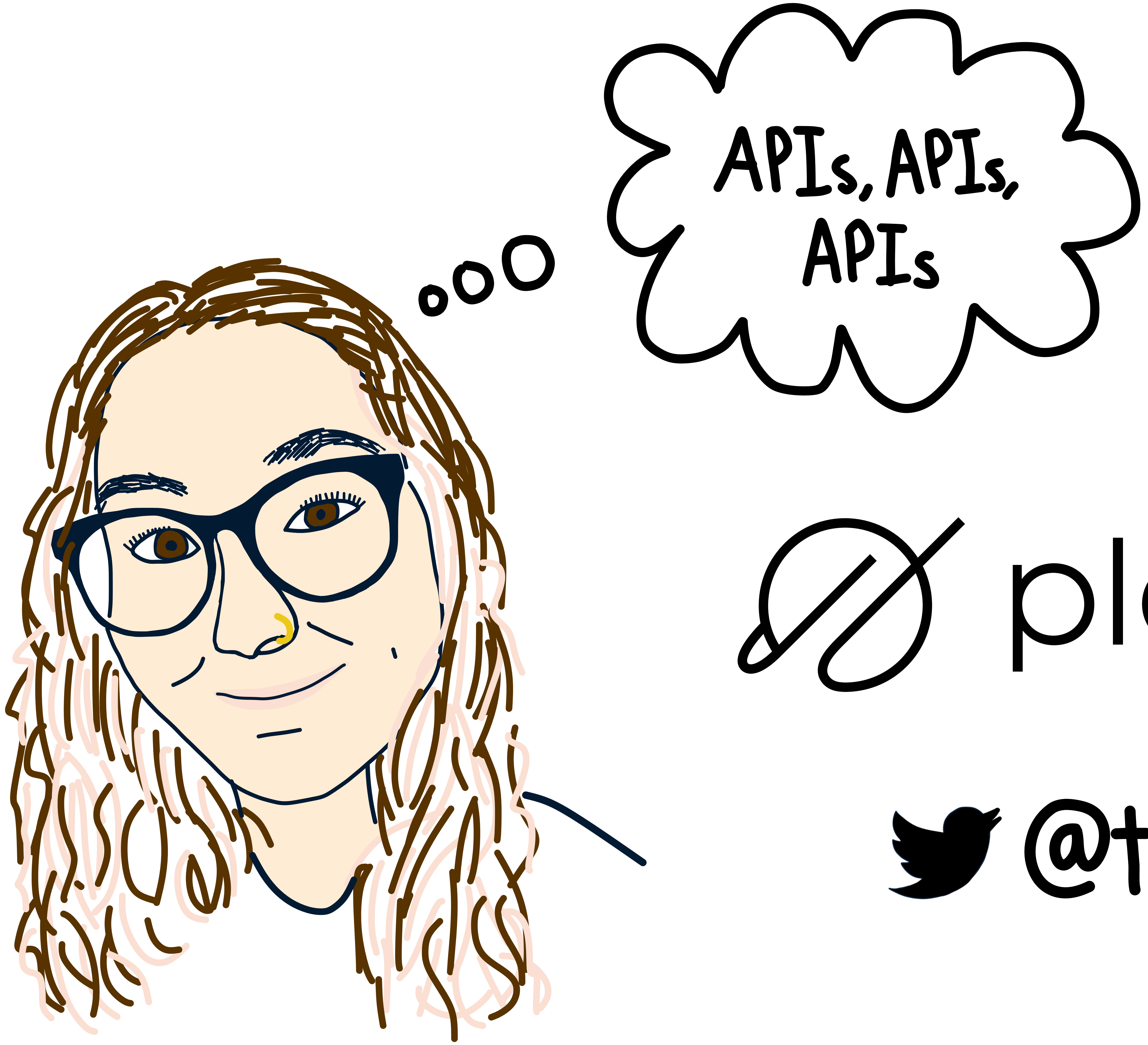
@planetscaledata

@taylor_atx

Core competencies

(Fancy business term)

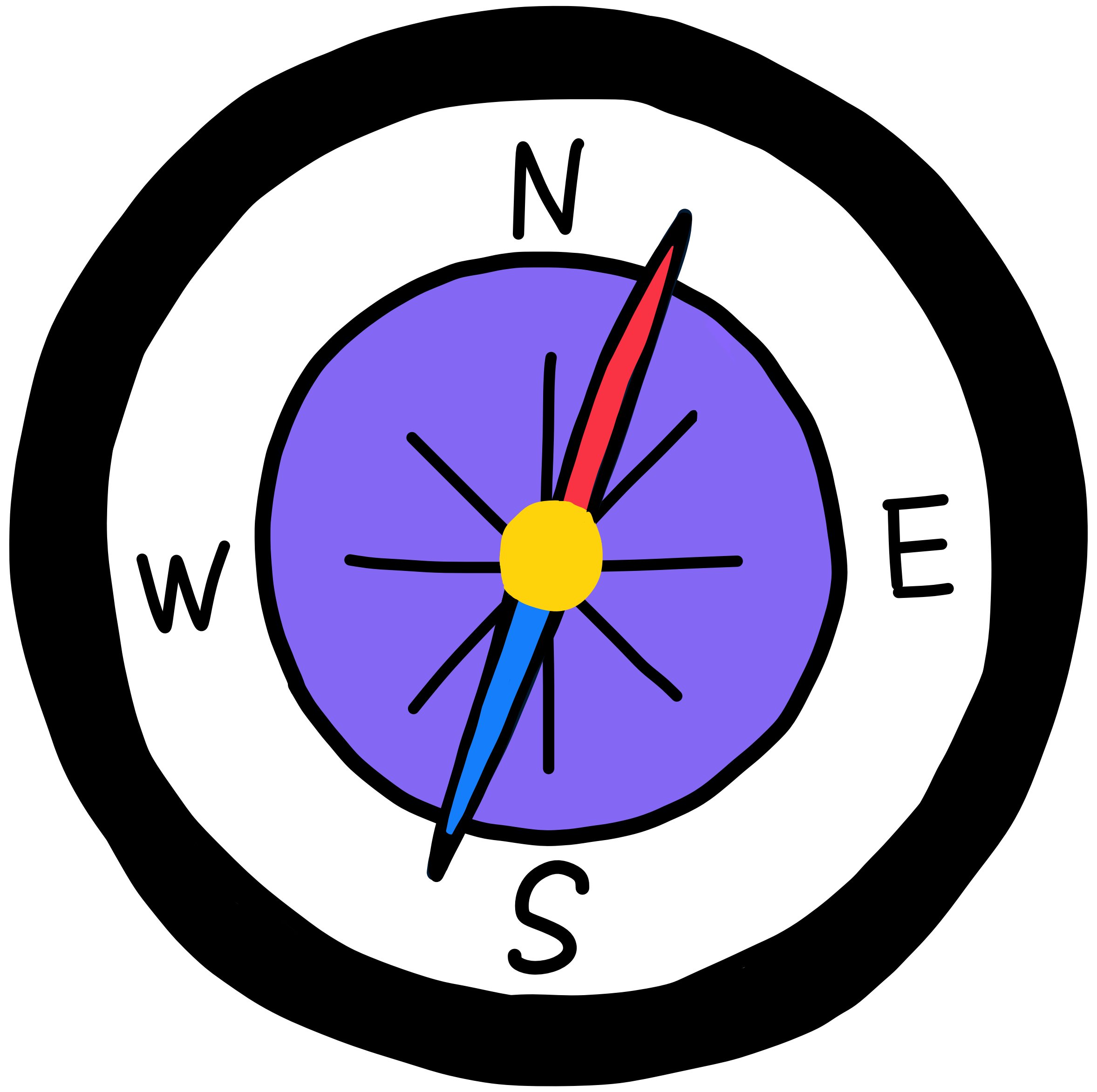




planetscale

@taylor_atx

We've solved old problems, but discovered new problems



Where do we go from here?

1 Learn how to "outsource" well

The new role of the developer

Vendor Engineering

Buy vs. Build

What problem are we solving for?

What opportunity are we going for?

Perception that third-party is unreliable, BUT some developers:

- Don't write docs
- Don't do periodic security updates
- Don't do regular maintenance
- Don't have time to fix any bugs

@coilysiren

Example:

Self-hosted vs. hosted WordPress

Example:

A marketing blog

Example:

Database as a Service

Ask good questions to gauge compatibility and fit

What friction you can deal with and what is a dealbreaker?

Vendor future roadmaps

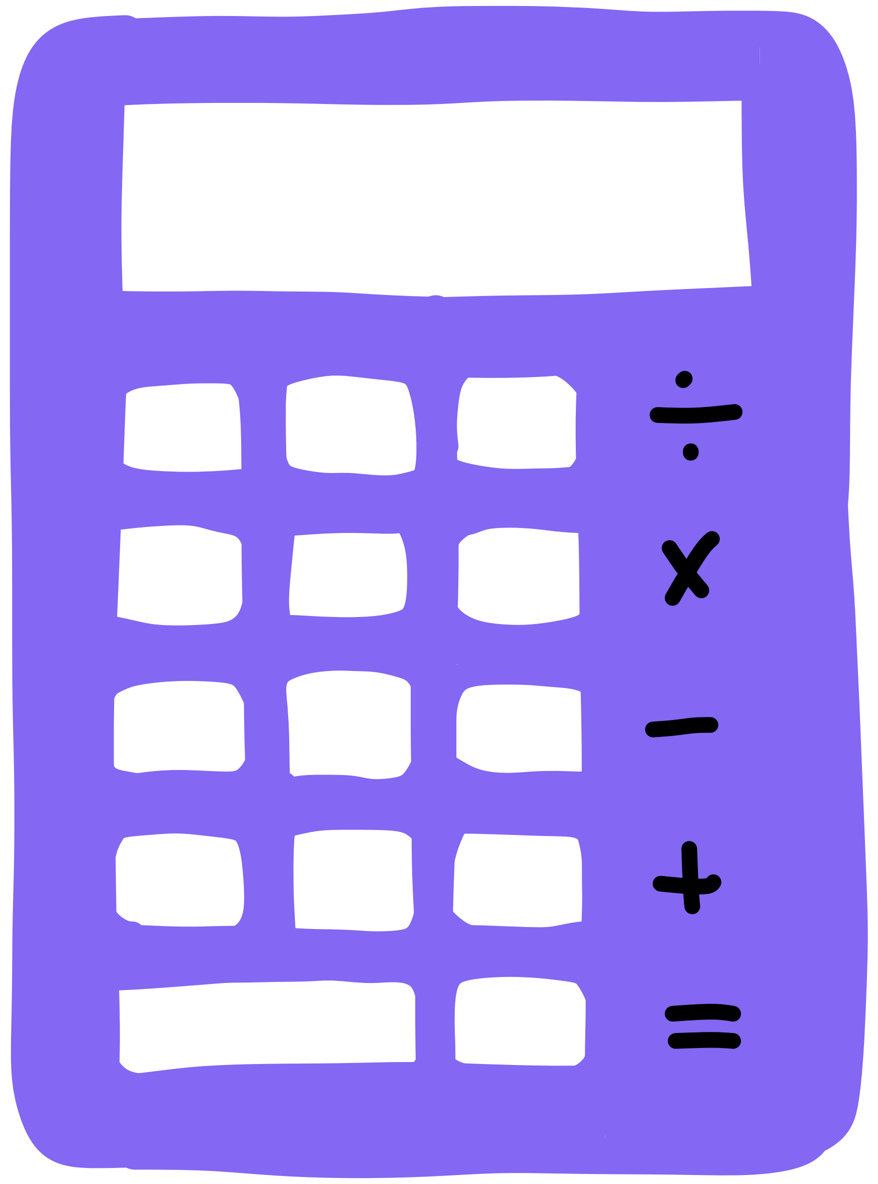
How do they take input?

How flexible are they to work with?

Calculate and qualify the cost to develop and operate

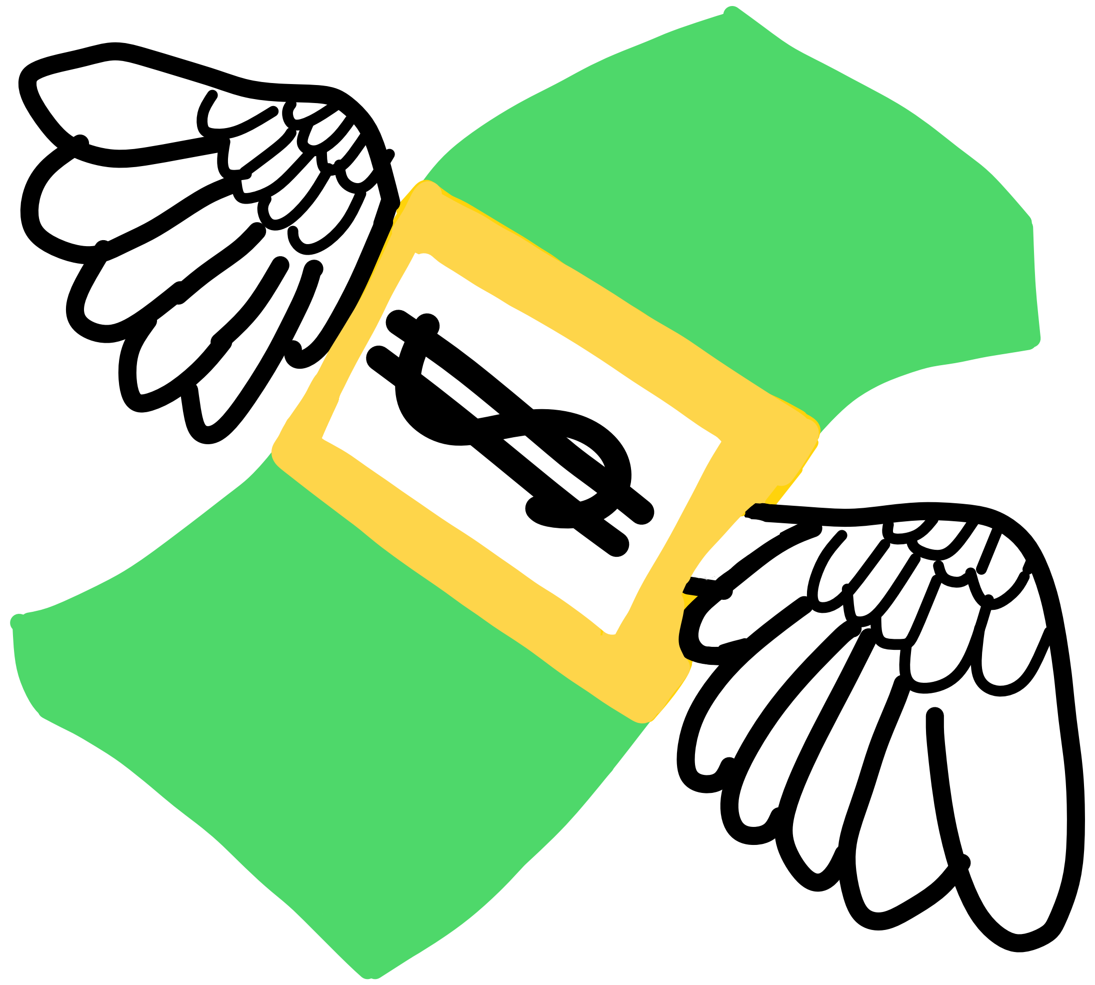
Get to end-to-end prototype as quickly as possible to find hidden costs

Remove as much as labor as possible

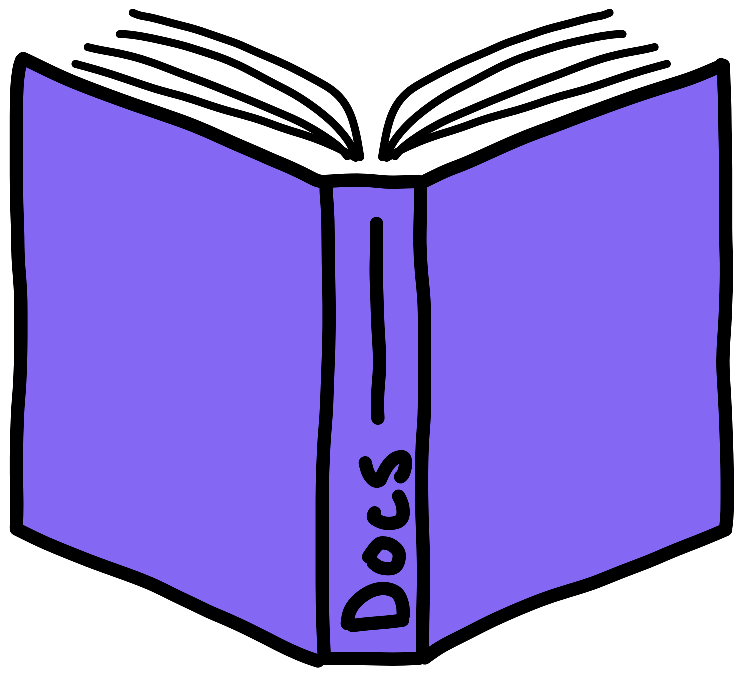


Consider the true cost of ownership and advocate internally

Manage up to executive and finance teams



Don't forget the documentation!



@planetscaledata

@taylor_atx

Reward team members for doing this work well

@planetscaledata

@taylor_atx

⚠ 503 Service Unavailable

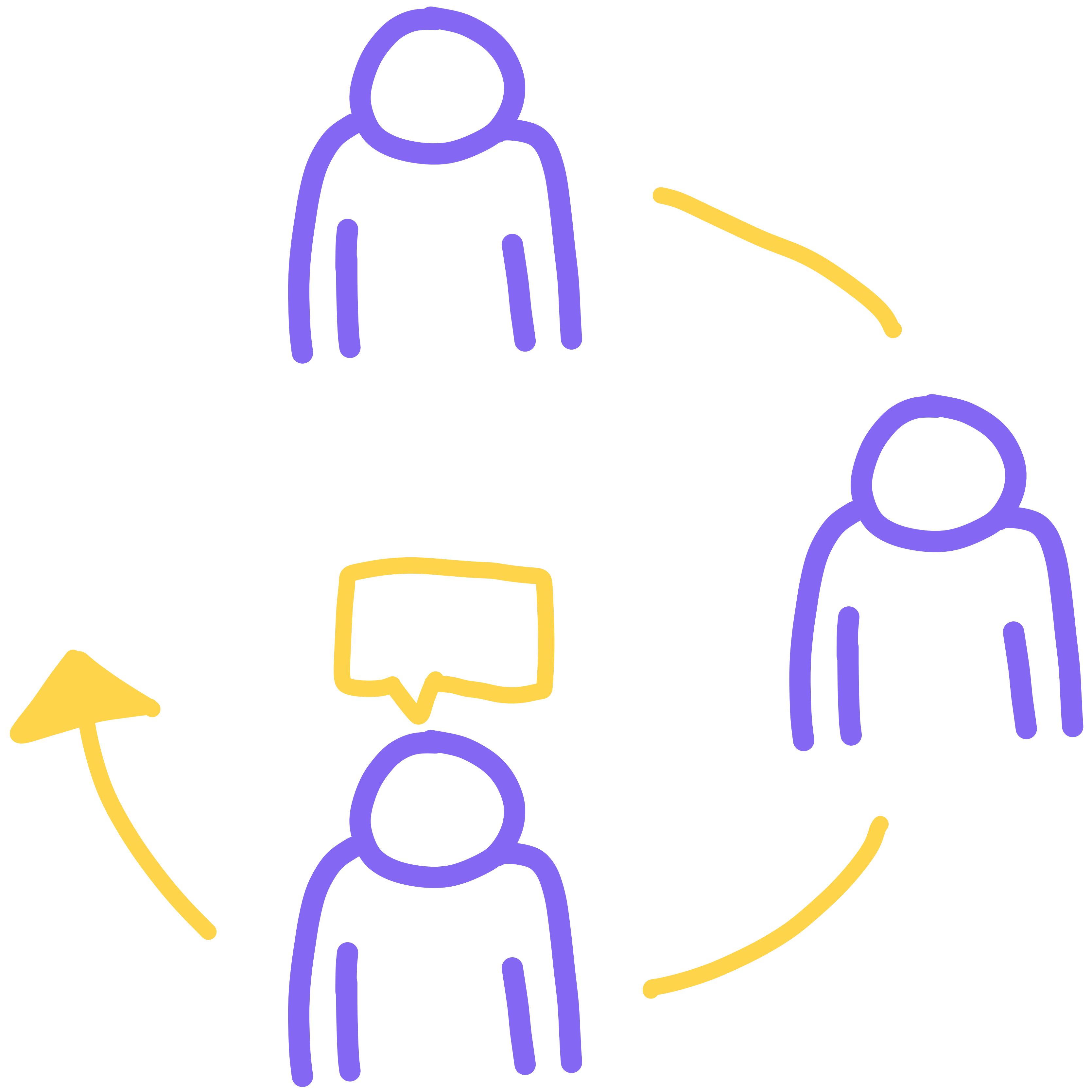
S3 Outage in 2017

When they break, we *could* break

2 Failure will happen, learn to handle it gracefully

Things will go down, but how do we handle it and grow?

Retrospectives



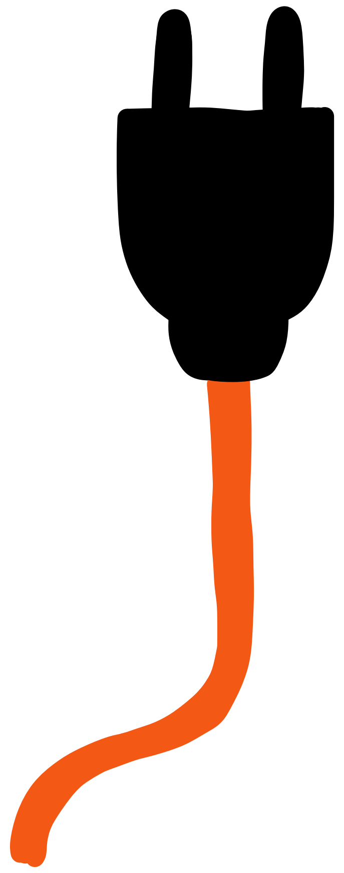
Handle it gracefully

Problems like:

- Connections
    - Rate limiting
    - Internet connectivity
    - Retry mechanisms
- Error codes and messages
- SDK quality
- Authentication
- Progressive data presentation
- Versioning
- Caching
- Validation
- Service outages
- Testing

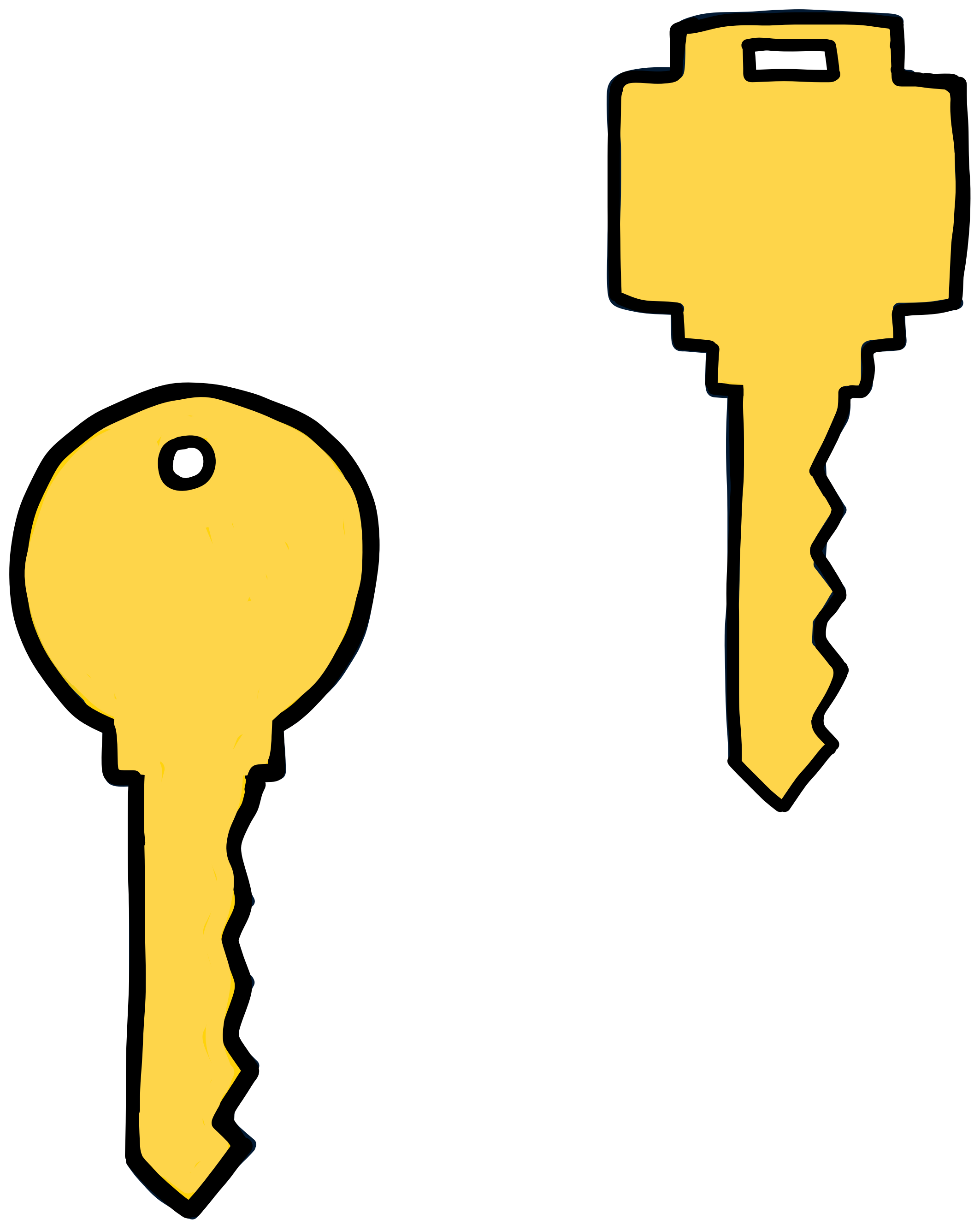
Some services or endpoints are offline:

- Don't disable everything!
- Frontend circuit breakers
- Non-critical services should fail silently, disable others intelligently
- Don't lead users down a broken path, disable what you can
- Set expectations



Authentication:

- 6+ different common authentication types
- How are they all being handled safely?



Connections:

- What if someone has a slower data connection?
- How do things load or fail?
- Progressive data presentation

Rate limiting:

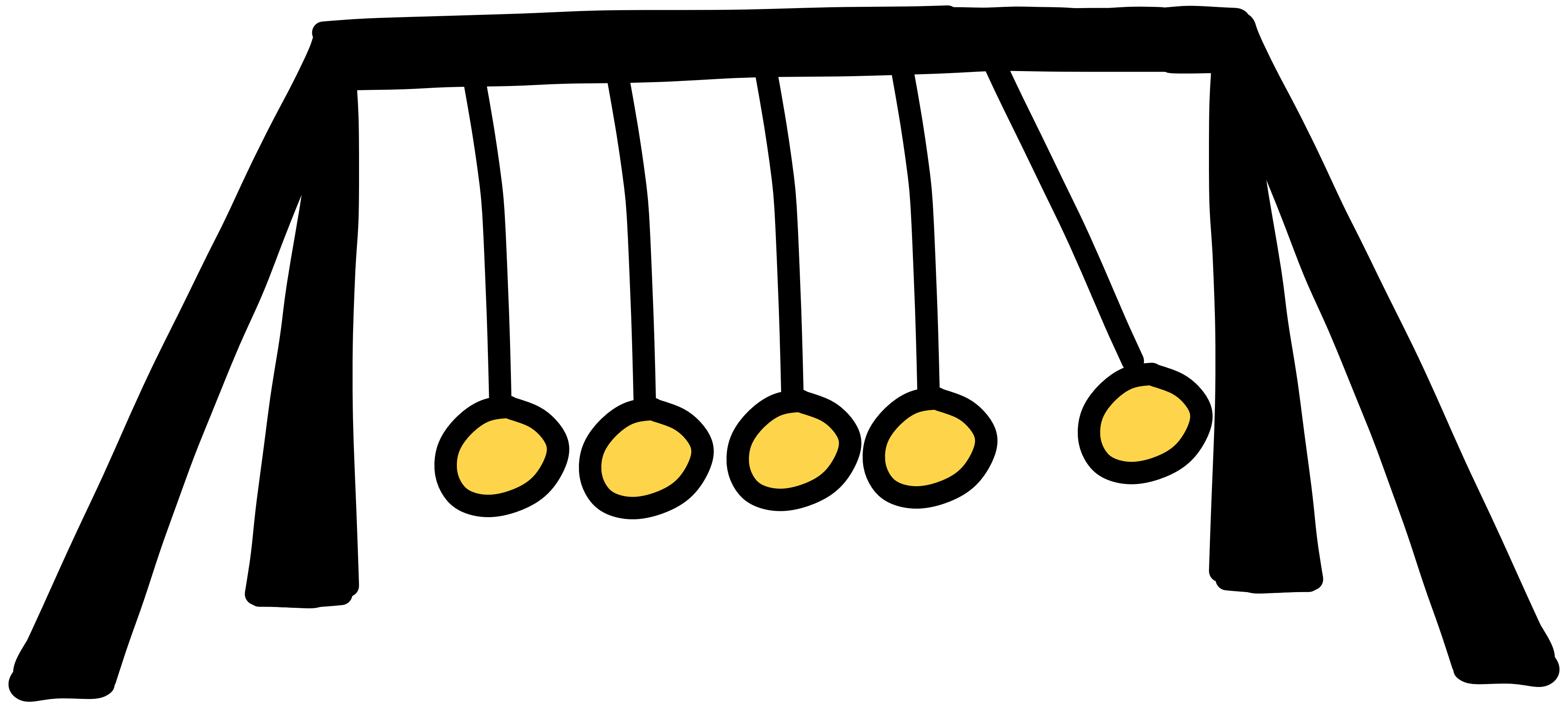
- API retry issues
- Requires some restraint and strategy
- When do you tell the user?

⚠ 429 Too Many Requests

Decoupling:

- Decoupling services from vendors
- Important that your glue code isn't so tightly glued
- API calls directly to an API without a thin wrapper make it hard to switch vendors

3 Find ways to abstract the glue code



Complexity is neither created nor destroyed - just shuffled around

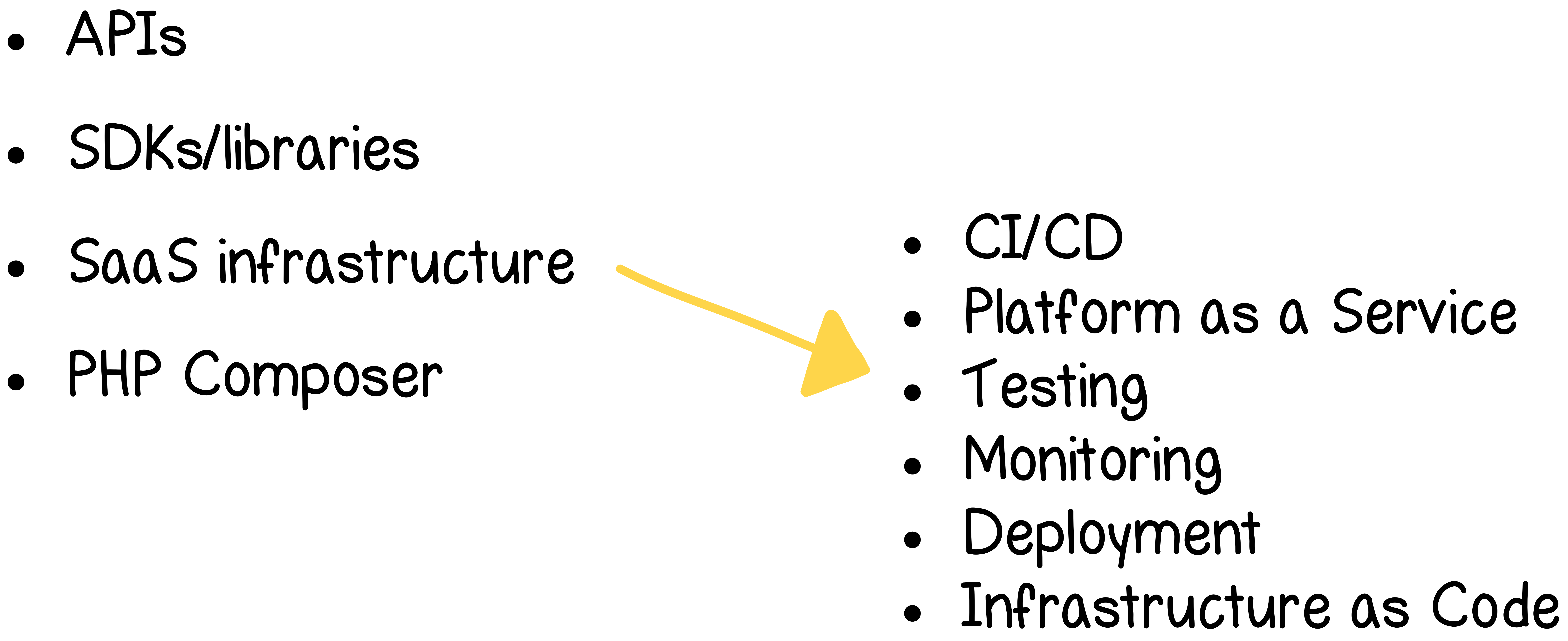
@planetscaledata

@taylor_atx

Abstraction tools:

- APIs
- SDKs/libraries
- SaaS infrastructure
- PHP Composer

Abstraction tools:



- APIs
- SDKs/libraries
- SaaS infrastructure
- PHP Composer

- CI/CD
- Platform as a Service
- Testing
- Monitoring
- Deployment
- Infrastructure as Code

Complexity-revealing tools:

- Debugger
- Performance profiler
- Observability tools

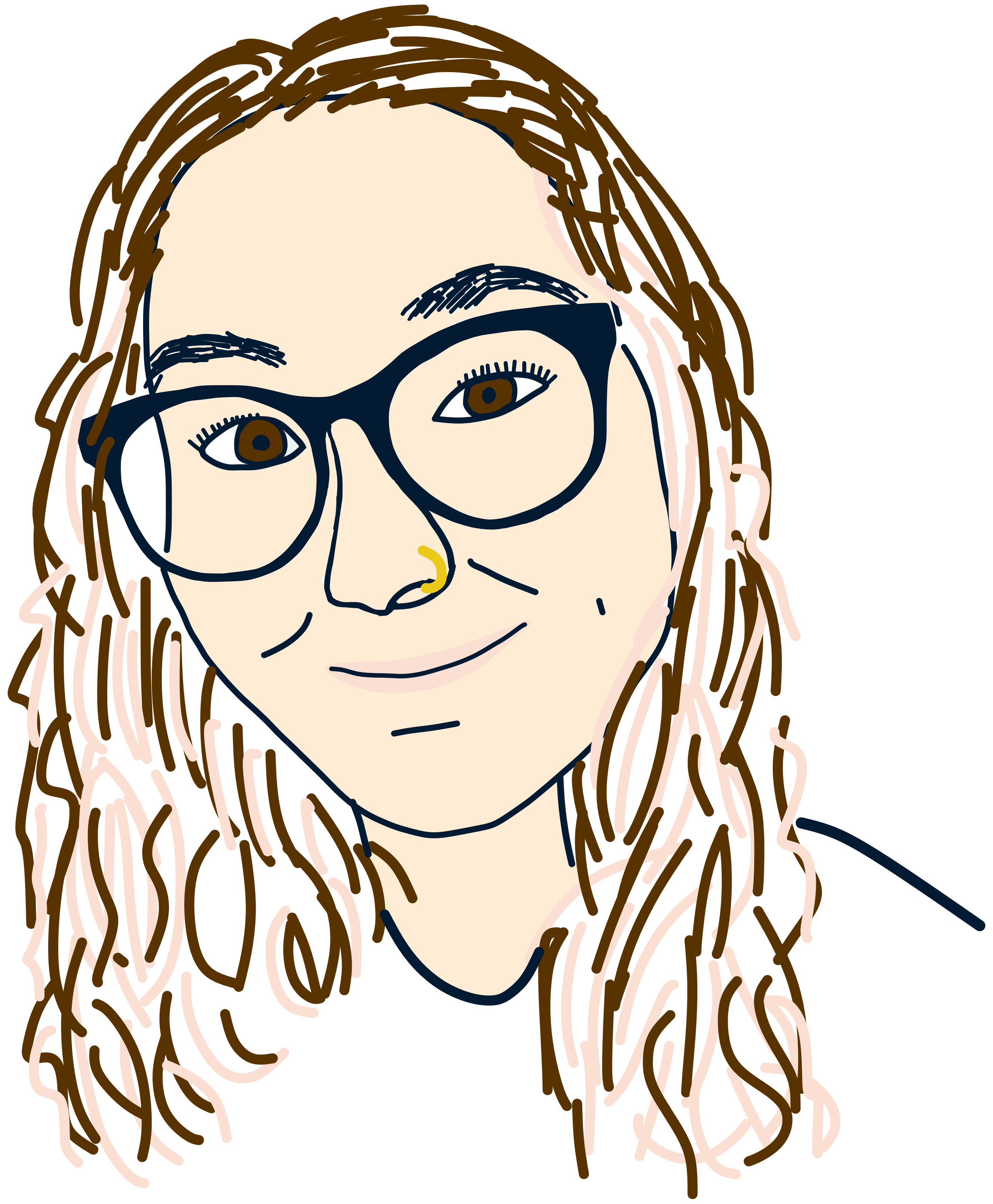
LAMP Stack:

- Low cost
- Allowed businesses to have an online presence
- Linux has historically been seen as “glue” for modern web development
- Different degrees of abstraction today

Focus

Glue work makes the dream work

1. Learn how to "outsource" well
2. Failure will happen, learn to handle it gracefully
3. Find ways to abstract the glue code



Thank You!

planetscale

@taylor_atx