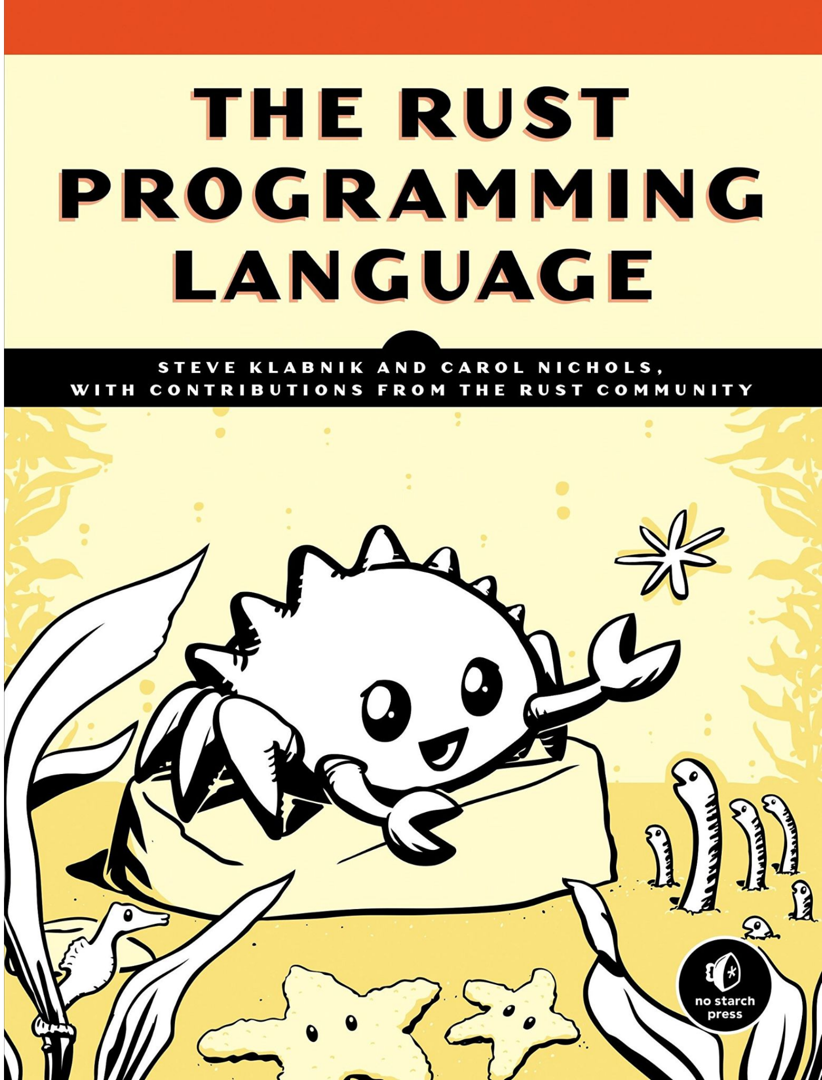
Rocket fueled by Diesel

How did I get here?

“Best Rust web frameworks”

AVIS

Google



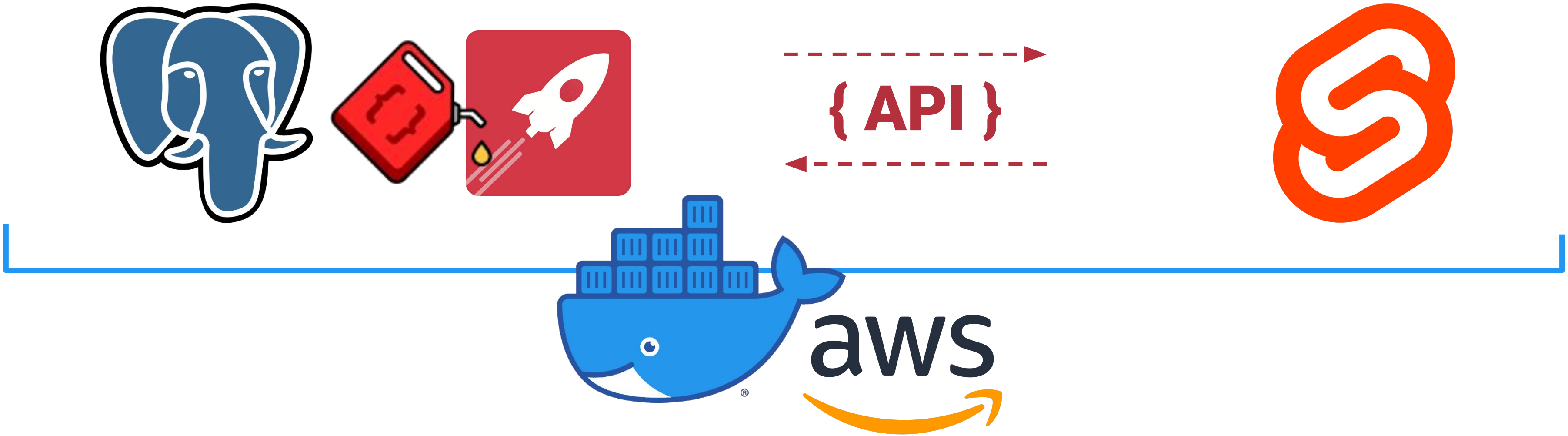
THE RUST PROGRAMMING LANGUAGE

STEVE KLABNIK AND CAROL NICHOLS,
WITH CONTRIBUTIONS FROM THE RUST COMMUNITY

no starch press

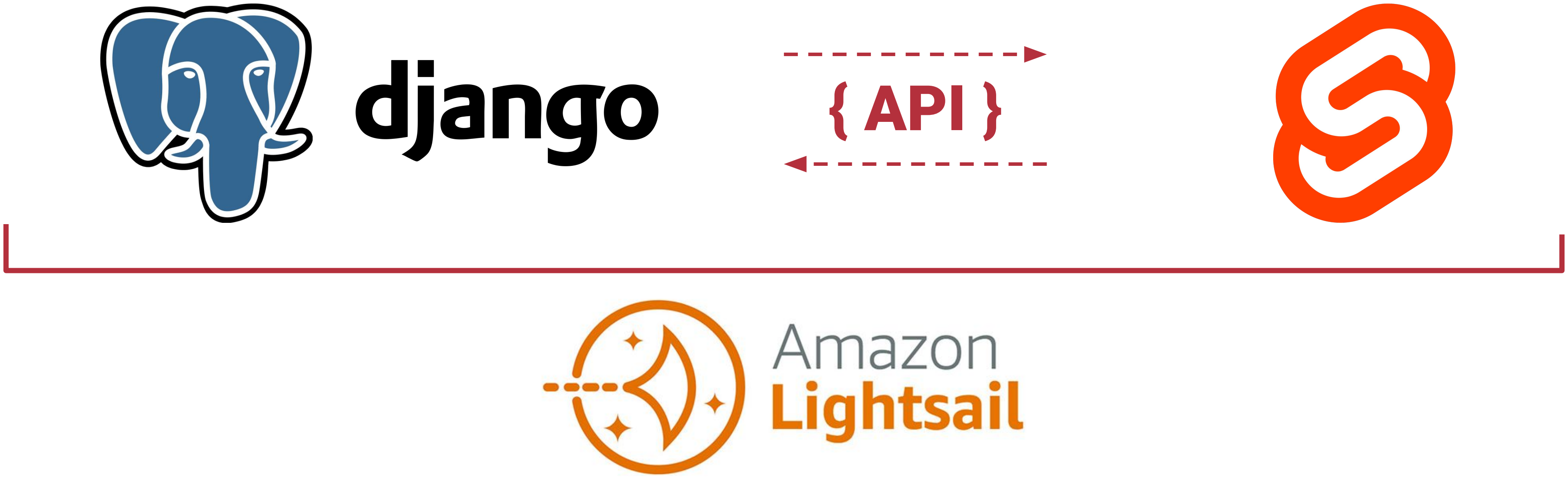
What I planned…

“Make it work, make it right, make it fast.” – Kent Beck (XP creator)



...what I ended up with

Use the right tool for the job!



{ API }

We Enjoy Typing!

WET isn’t** always **bad!

```
CREATE TABLE users (…);
```

```
table! {
    users (user_id) {
        …
    }
}
```

```
use schema::users::{
    self,
    dsl::*
};
```

```
pub struct User { … }

pub struct NewUser { … }

pub struct PutUser { … }
```

```
CREATE EXTENSION IF NOT EXISTS "uuid-ossp";

CREATE TABLE
    appointments (
        appointment_id UUID PRIMARY KEY DEFAULT uuid_generate_v4(),
        created_at TIMESTAMP DEFAULT NOW(),
        ...
    );
```

```
#[derive(Queryable, Identifiable,
Serialize)]
#[primary_key(appointment_id)]
pub struct Appointment {
    pub appointment_id: Uuid,
    pub at: NaiveDateTime,
    pub user_id: Option<Uuid>,

    pub created_at: Option<NaiveDateTime>,
    pub confirmed_at: Option<NaiveDateTime>,
    pub token: Option<Uuid>,
}
```

```rust
#[derive(Insertable, Deserialize)]
#[table_name = "appointments"]
pub struct NewAppointment {
    pub at: NaiveDateTime,
    pub user_id: Option<Uuid>,
}
```

```
let filters = …;

let mut query = appointments.into_boxed();

if filters.date.is_some() {
    query =
query.filter(date.eq(filters.date.unwrap()));
}

query.load::<Appointment>(db)
```

*“Remember, a **few hours** of **trial and error** can save you several **minutes** of looking at the **README**.”*

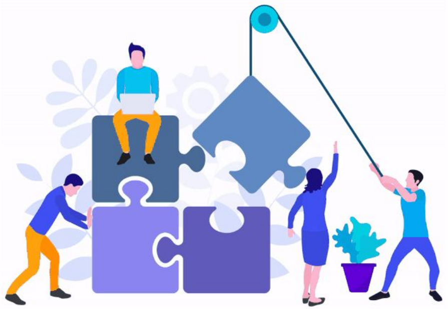
Advantages

Reusability

Control

Coherence

Integration



Disadvantages

Code duplication

Imports

Learning a new DSL

*"There are only two **hard** things in computer programming: cache invalidation and **naming things**"*

Diesel is a SYNC ORM

*And it **won't go ASYNC** in the near future!*

Diesel 2.0

Diesel 2.0

*More than **1700 commits** were submitted over a span of **3 years***

GROUP BY

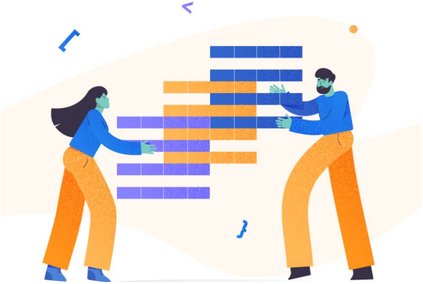
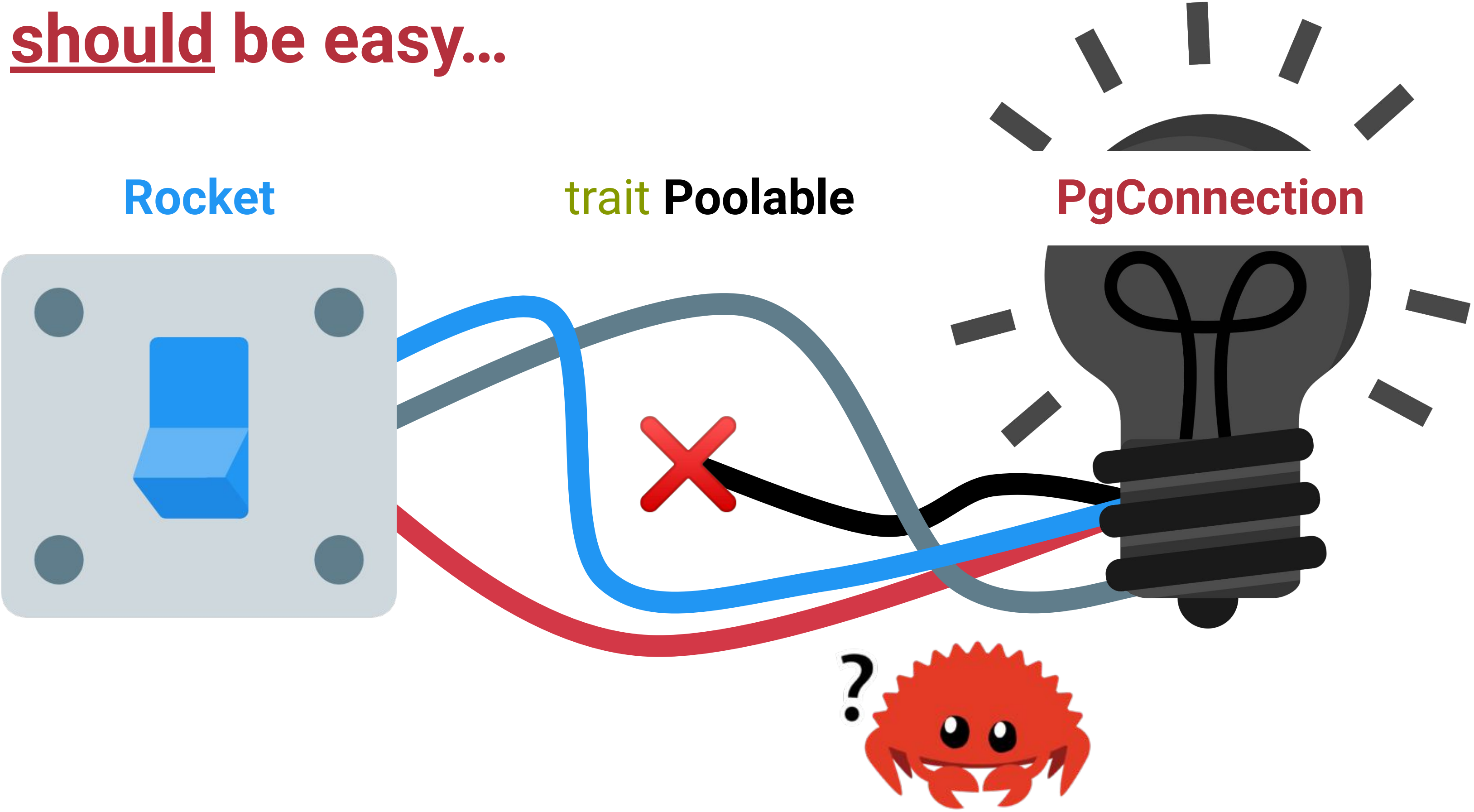


Table aliasing

No schema

LTS

It should be easy...



Rocket

trait Poolable

PgConnection

Coherence: orphan rule

“We can’t implement ***external traits*** *on* ***external types****”*

```
impl Poolable for PgConnection { … }
```

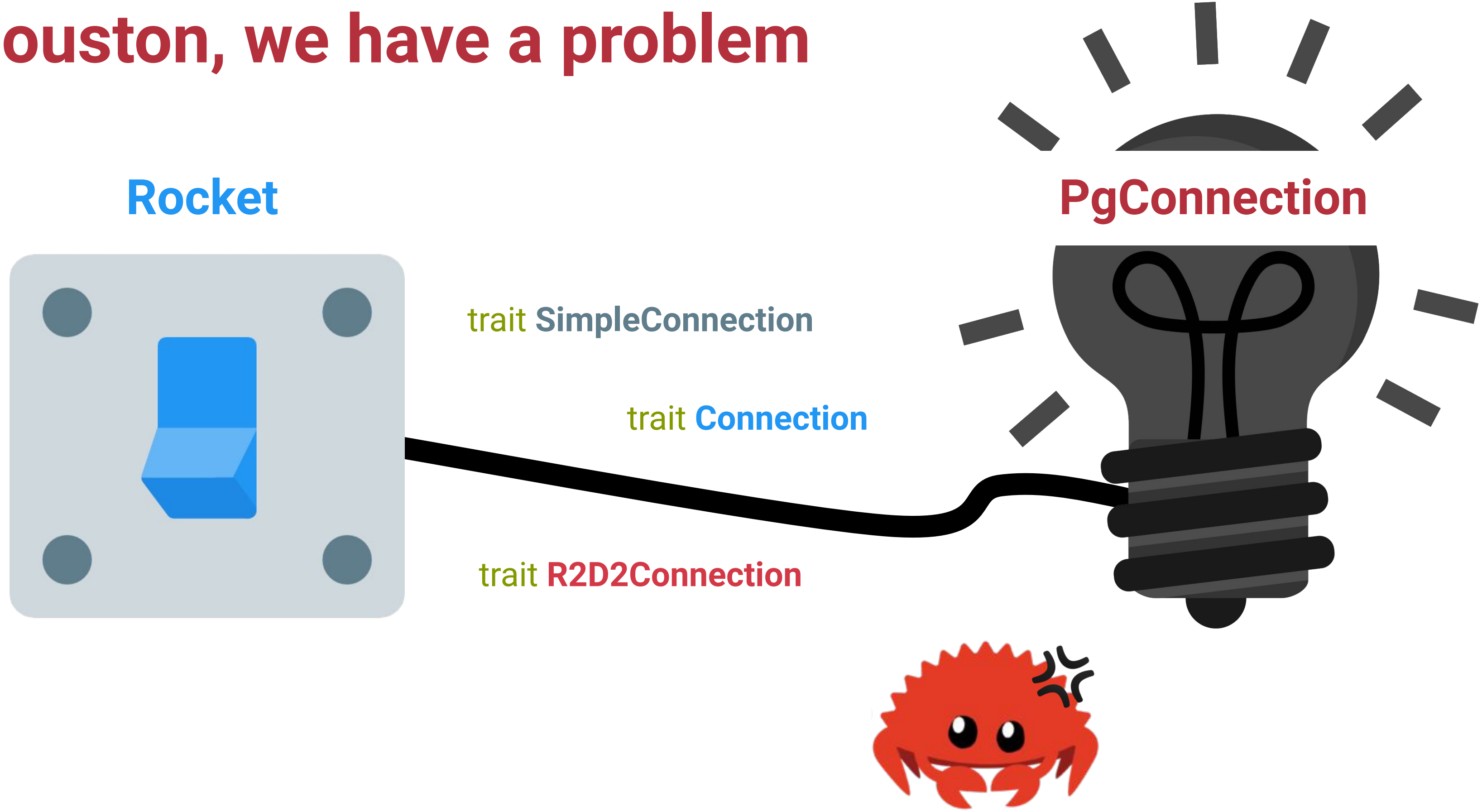
❌

```
pub struct MyConnection(PgConnection)

impl Poolable for MyConnection { … }
```

✔

Houston, we have a problem



Rocket

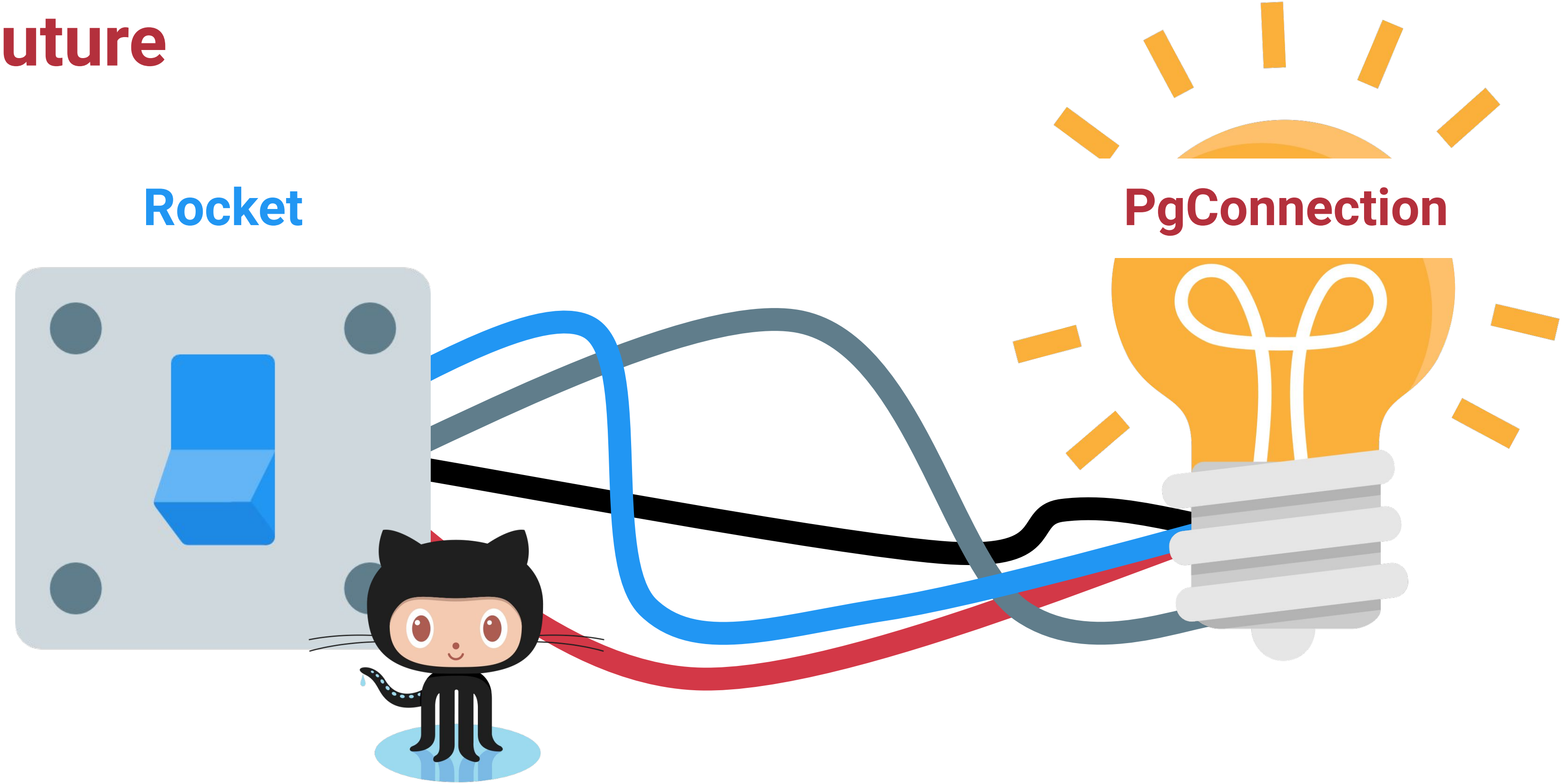
PgConnection

trait SimpleConnection

trait Connection

trait R2D2Connection

Future



Rocket

PgConnection

Migrating to 2.0

```
diesel = { version="1.4.8", features=[…, "uuidv07"]}

diesel = { version="2.0.0-rc.0", features=[…, "uuid"]}


uuid = ”1.0.0”
```

```
cargo install diesel_cli –version “2.0.0-rc.0” ...

rm diesel.toml

diesel print-schema
```

Development & Deployment

Development

“99 bugs in the code, 99 bugs in the code. Take one down and pass it around, 98 bugs in the code.”

Docker

Linux target

libpq

deploy

PowerShell

Amazon Lightsail

AWS sdk & ses

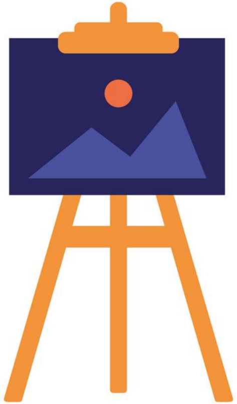
aws

Development pt. 2

“98 bugs in the code, 98 bugs in the code. Take one down and pass it around, 97 bugs in the code.”

Real time

Views?



Dynamic JSON

rust-analyzer

localhost

Rocket auth?

Is there an ***authentication framework*** *for Rust?*

Bibliography

- https://www.reddit.com/r/rust/comments/g07t7a/why_does_everyone_recommend_diesel_when_its_so/
- https://python-gino.org/docs/en/1.1b2/explanation/why.html
- https://thenewstack.io/code-n00b-naming-hardest-part/
- https://github.com/diesel-rs/diesel/issues/399
- https://stackoverflow.com/questions/58473606/cache-rust-dependencies-with-docker-build
- https://blog.mgattozzi.dev/caching-rust-docker-builds/