

stimulus reflex

by dbo

Build **reactive** apps with the Rails tooling you already know and love

- rails
- stimulus js
- action cable (web socket)

compatible with:

- turbolinks
- server rendered HTML
- russian doll caching
- ...

No need for a complex front-end framework

SPA x server rendered HTML

(one step further than turbolinks)

inspired by Phoenix Liveview

3 benefits over Rails HTTP request/response cycle:

- Communication happens on the ActionCable web socket
- The controller action is invoked directly
- DOM diffing is used to update the page (with `morphdom`)

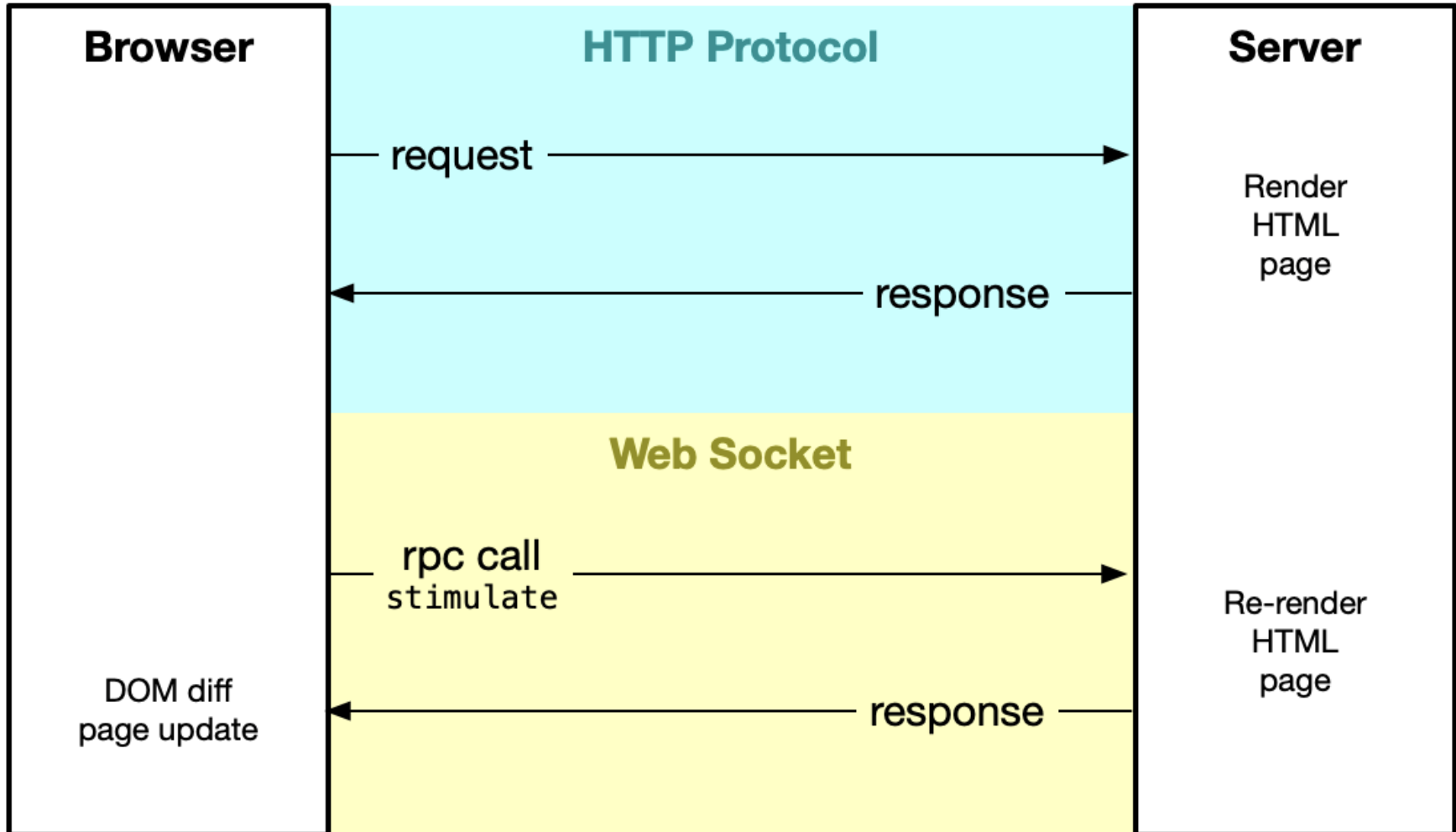
how does it work?

1. Render a standard Rails view
2. Use Stimulus and ActionCable to invoke a method on the server
3. Watch the page automatically render updates via fast DOM diffing
4. That's it...

yes, it really is that simple

Stimulus Reflex

A simple reactive model based on web standards



setup

```
yarn add stimulus_reflex
```

```
// app/javascript/packs/application.js  
// ...  
import StimulusReflex from 'stimulus_reflex';  
  
// ...  
// application -> stimulus app previously instantiated  
StimulusReflex.initialize(application);
```

```
# Gemfile  
gem "stimulus_reflex"
```

usage

demo 1

no JS

(also, the `reflex` `element` `property`)

demo 2

with ActiveRecord

demo 3

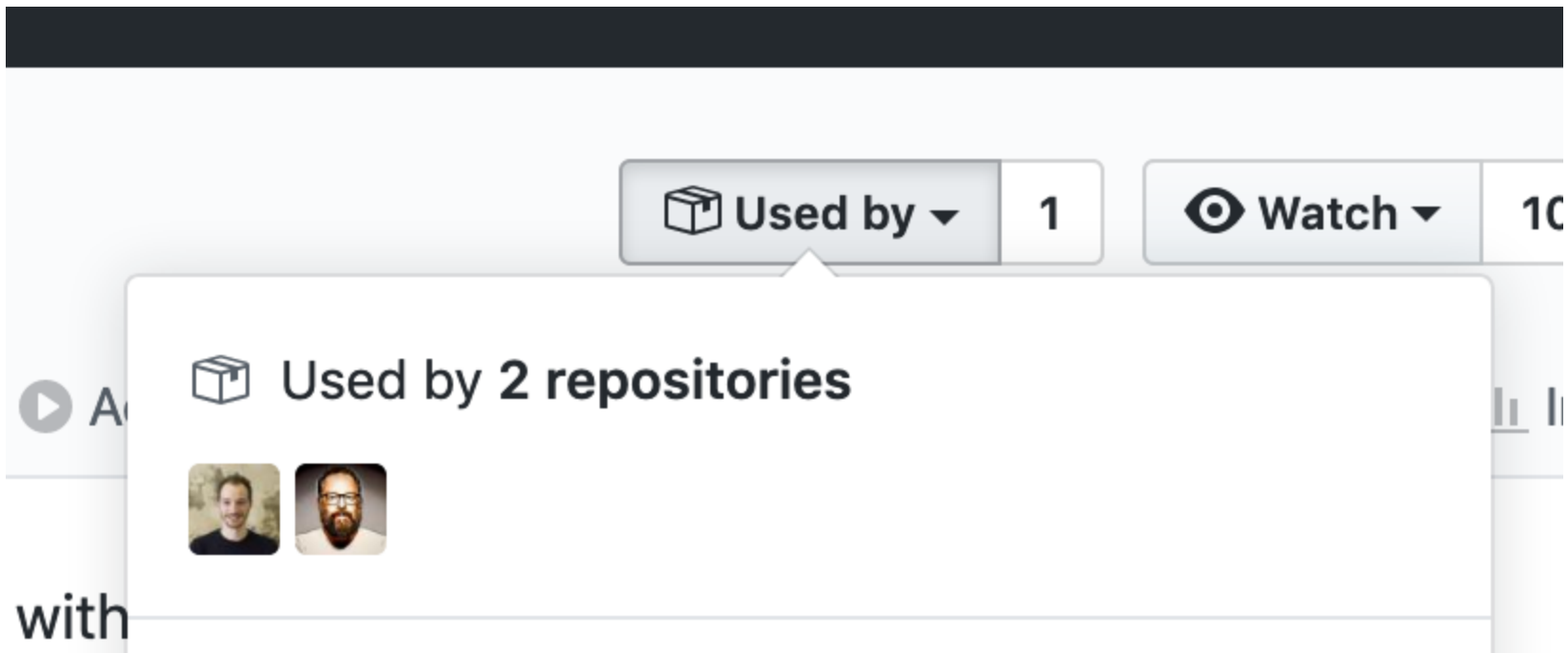
with Stimulus JS

all concerns (from managing state to rendering views) are handled
server side

how cool is that?

disclosure

this is still an early-stage project



important

- not a silver bullet (http request/response cycle is cool)
- harder to debug (WebSocket)
- triggered only from Client
- might be slow if page is heavy to render

but, devs, they want to have fun

- easy reactive app
- no action cable complexity
- no manual dom manipulation in JS
- keep the controller cycle
- no react / redux / state management

~_(ツ)_/~

IMHO

excellent new tool in addition to turbolinks + stimulus + rails

want to know more?

just try it

thank you