

# HTMX

Make web development simple and fun again

My opinionated Odyssey

# History

1999: CGI Scripting with Shell and Perl

2001: Python

2008..today: Python and Django

# 2014: Angular

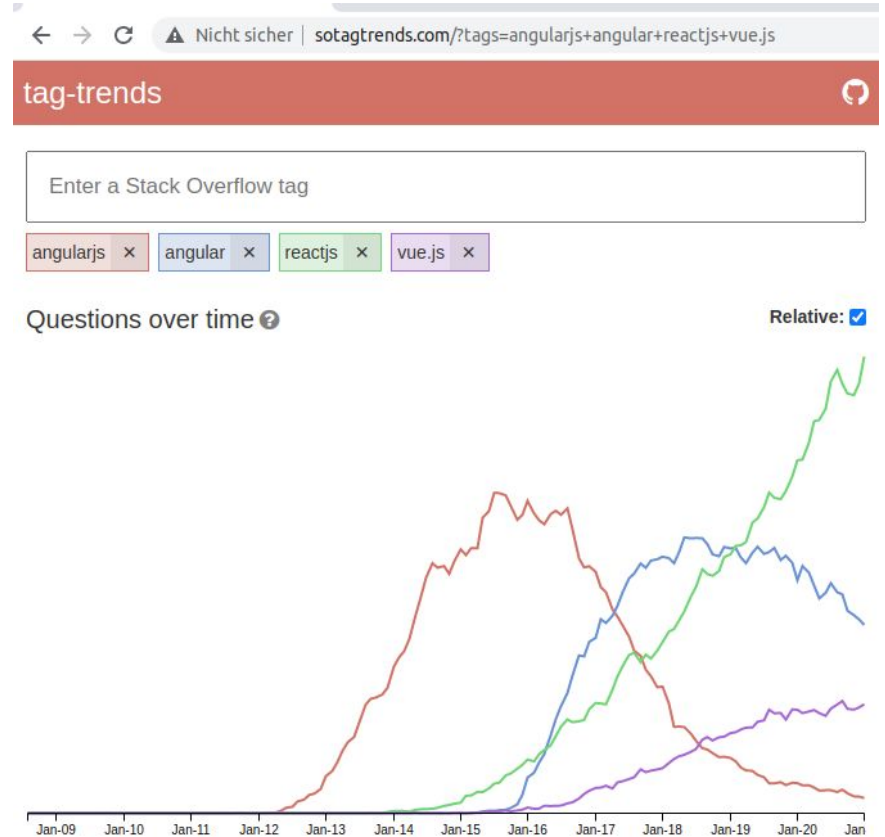
Wow, that's the future: The server sends JSON to the browser.

But customers kept us busy doing the things we did before (Django)

# 2020: React / Vue

I am glad, that I stucked to Django,  
and did not chose Angular.

I played with React/Vue ... but no “wow”



# Om - What do I like?

- I like the Django Forms library
  - Create HTML (on the server)
  - Validate user input (on the server)
- I like Django ORM
- HTTP & HTML

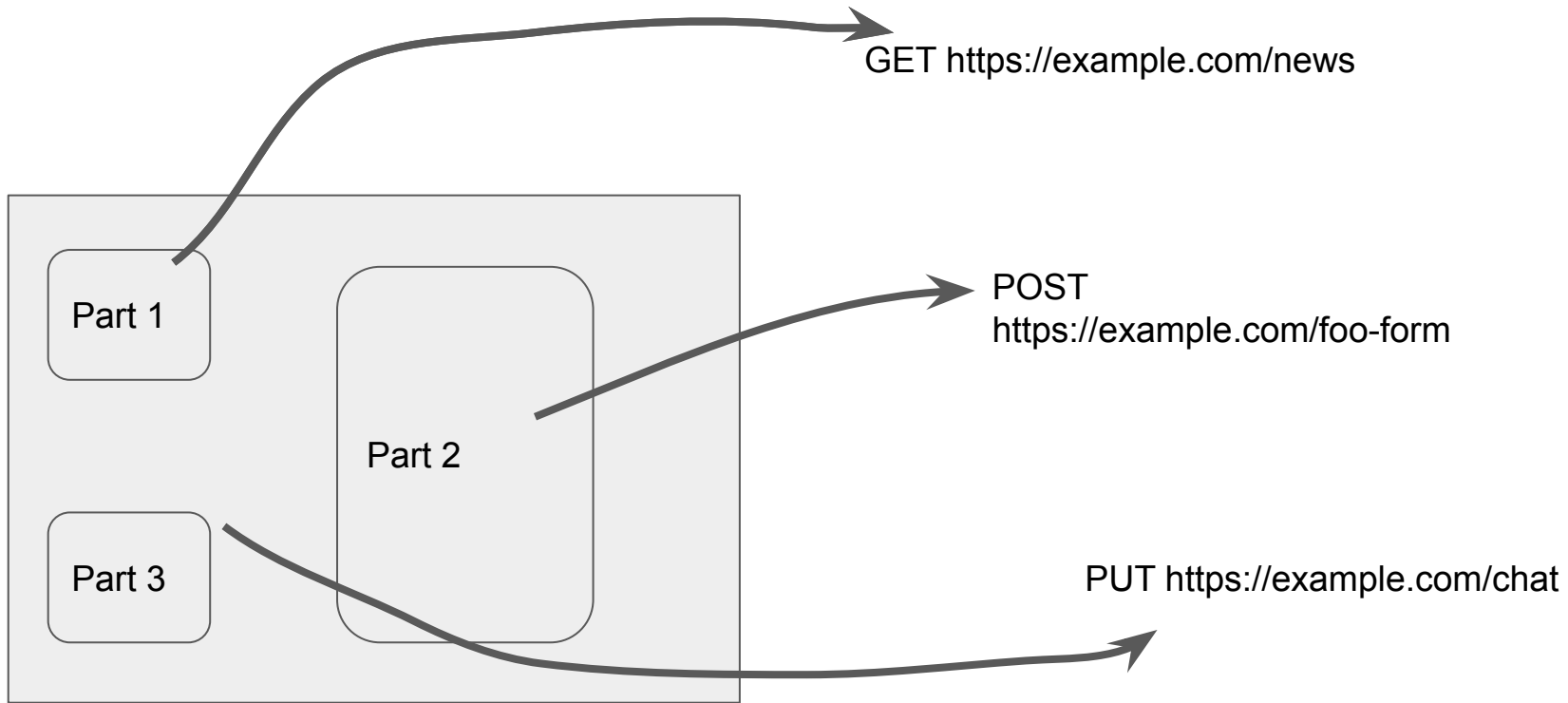
# What do I want to improve?

- Full Page Reload after sending a form. Grrrr
- Own ajax library for submitting/reloading snippets: No fun.

# One Page, several Parts







# SPA ?

Single Page Application?

..... no, not needed.

I just want to Submit/Reload parts of the page.

# State-Management

- State?
- Stateless has won:
  - CORBA → http
  - Server with long uptime → Container
- I don't want a valuable state on the client. Only GUI state (scrolling, folding)
- Every valuable state should be in the central database.

# Form Validation in the Browser?

..... ok, nice to have validation on the client side .... but  
.... but I need to validate on server, too.

Maybe I don't need validation on the client side?

# SSR Hype

- Looked at Server-Side-Rendering of React/Vue
- wait !!!
- This huge stack ....
- Do really want that?
- Are there alternatives?



If you prefer boring facts to emotions: [Rendering on the Web](#)

Declarative

vs

Imperative

HTML, CSS, SQL

vs

if/else, loops ...

Conditionless 



HTML 

Keep the Client ...

Simple & Stateless

# ~~JSON~~ HTML Fragments over the wire

- Unpoly: [unpoly.com](https://unpoly.com)
- Hotwire: [hotwire.dev](https://hotwire.dev)
- ...
- HTMX: [htmx.org](https://htmx.org)



# HTMX adds new Attributes

```
<button hx-post="/clicked" hx-swap="outerHTML">
```

Click Me

```
</button>
```

# HTML Fragments over the wire

URL /clicked

→ HTML Fragment/Snippet from server to client.

# HTMX

- Works with every language (server side)
- ~9k min.gz

# hx-trigger

```
<div hx-get="/clicked" hx-trigger="click">
```

Click Me

```
</div>
```

# [htmx.org/examples/](https://htmx.org/examples/)

[Click to Edit](#)

[Cascading Selects](#)

[Infinite Scroll](#)

[Tabs](#)





**HTMX**



---

**Hyperscript**

# You might not need JQuery?

## JQUERY

```
$.ajax({
  type: 'GET',
  url: '/my/url',
  success: function(resp) {

  },
  error: function() {

  }
});
```

## IE10+

```
var request = new XMLHttpRequest();
request.open('GET', '/my/url', true);

request.onload = function() {
  if (this.status >= 200 && this.status < 400) {
    // Success!
    var resp = this.response;
  } else {
    // We reached our target server, but it returned a
  }
};

request.onerror = function() {
  // There was a connection error of some sort
};

request.send();
```

# Hyperscript

Hyperscript is a fun little scripting language for doing front end web development.

[hyperscript.org](https://hyperscript.org)

# Hyperscript: Step by Step

No Async, No Await, No Promises

```
<button _="on click put 'I was clicked!' into my.innerHTML">
```

```
Click Me!
```

```
</button>
```

You might not need JS

End of Odyssey?



# HTMX + Hyperscript

Make web development simple and fun again

[Thomas Güttler](#), March 2021

