

# JavaScript on TV

Building full screen apps with Vue.js

**Welcome to multiplatform hell.**

# There's so many of them!

---

- LG Netcast
- LG WebOS
- Samsung Tizen
- Android TV
- Firefox OS (srsly.)
- Roku TV
- Fire TV
- tvOS
- ... and the list goes on.

# webapp!



It'll work everywhere.

Welcome to ~~multiplatform~~ cross-browser hell.

# There's still quite a lot of them!

---

- Mostly WebKit forks
- Mostly frozen in time
- Some APIs missing
- Some user agents are fake
- Video support is hilarious
- You look for matching desktop browsers

# How TV development looks

- Fixed wrapper
- Mashing on the keyboard
- Surrounded by TVs



# Off to a good start with vue-cli

---

- Webpack template
- Dev environment (linters, HMR, error overlay)
- Production builds
- Unit & E2E tests
- A router
- Vuex had to be added



**New problems**

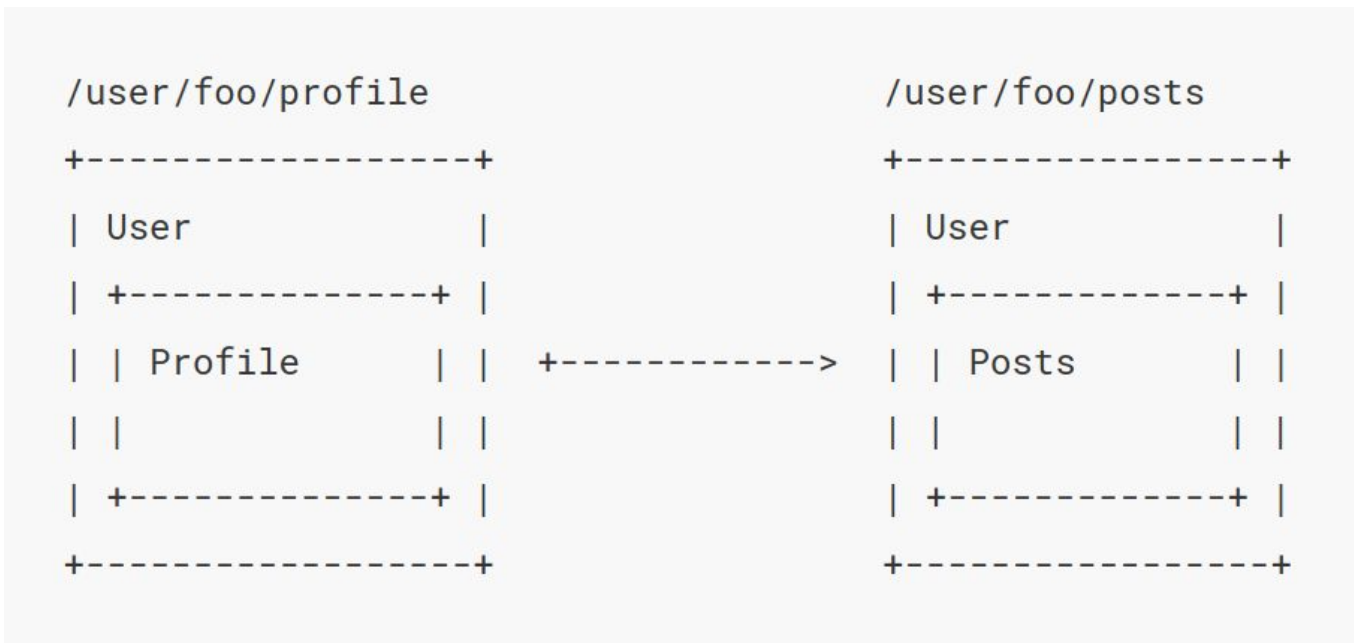
# Section switching

---

- Should I write a mini handler?
- Doh, router!
- Free transitions

# Navigation

- Router saves the day again
- Nested routes/partial templates





Home

Subscriptions

Uploads

Library

History

Watch later

Purchases

Settings

SUBSCRIPTIONS

# YouTube Red

## Recommended



What Are You Really Eating?  
Thoughty2 • 3 months ago

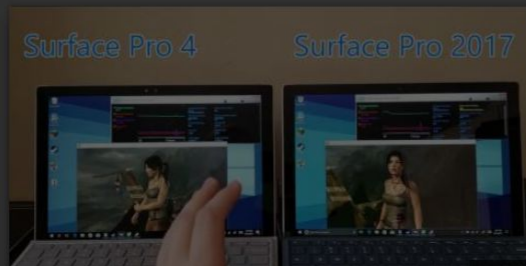


'Archer' Cast Makes Trading Cards For Th...  
WIRED • 2 months ago



Family Guy - Cons  
Jea Santi X Family C

## Technology



# Global state

---

- Vuex is really helpful
- App has a `_lot_` of global state
- Reactive programming greatly simplifies this

# Optimising for performance

---

- The TVs are slow
- The browsers are even slower
- We need to dig deep
- Followed a container/component structure from Redux
- Dumb components are easy to modify
- Architecture quality is not affected

# Remote control

---

- By far the most interesting problem
- Opposite flow of events affects architecture
- Constraints on what kind of states you can have
- Reactive programming, again, really helps

# The solution

---

Each page component:

```
export default {  
  ...  
  mixins: [registerKeyHandler],  
  methods: {  
    handleKey(key) { ... }  
  },  
  ...  
}
```

Main event handler:

```
if (specialCaseActive) {  
  specialCaseHandler();  
} else {  
  regularHandlers[currentRoute]()  
}
```

Mixin:

```
export default {  
  created() {  
    regularHandlers[currentRoute] =  
      this.handleKey;  
  }  
}
```



**Final verdict:  
surprisingly applicable**

**mato@zgajner.com, work done for united.cloud**