

SPA Server/Client

In a SPA:

- Client handles UI and UI State
- Server is "source of truth"

State can be duplicated between client and server!

- Server is "real"

Keeping State in sync

Client state can be "stale"

- out of date

Effort to keep in sync

- Depends on importance
- Can decide to always overwrite
- Can add and check an "as of" with data
- Remember: Don't trust client data!

SPA: Fetch sends commands

In SPA:

- Client sends commands to server
- Server
 - Decides if to do it
 - Decides response
 - Informs client
- Client updates state
 - Possibly from data in response
 - Possibly makes same change itself
 - Server response is usually better

Pagination

Our examples have all been small and fast

Real examples are often not

- "Give me all movies from 2000"

"Pagination" is where server sends partial response

- often includes a "page" or "cursor"
- use to request next "page" of data
- Does data in page change?

What kind of web dev?

Front-End

- HTML (Semantic?)
- CSS
 - SASS?
- JS in browser
 - babel? Webpack?
 - fetch (or other XHR)
- SPA?
- Progressive Enhancement?
- May need to run backend environment

Backend Web Dev

Backend

- May not be in JS
- May need to call other systems
 - Database
- Providing pages?
 - HTML in language/templates
- Providing services?
 - Have to consider clients that aren't you
 - CORS!
- Deploying changes

Full Stack Web Dev

- Often alone or small team
- Can cover all the bases
 - Connect the sides
- Often shallow or narrow in tech
- Language swapping impacts code style

Serverless

"Serverless" does not mean no server

Does mean limited server role

- cloud-based full virtual servers, or
- cloud-based static files, or
- cloud-based functions-as-a-service, or
- cloud-based database-as-a-service, or
- etc