ASSIGNMENT REVIEW

OBSERVATIONS

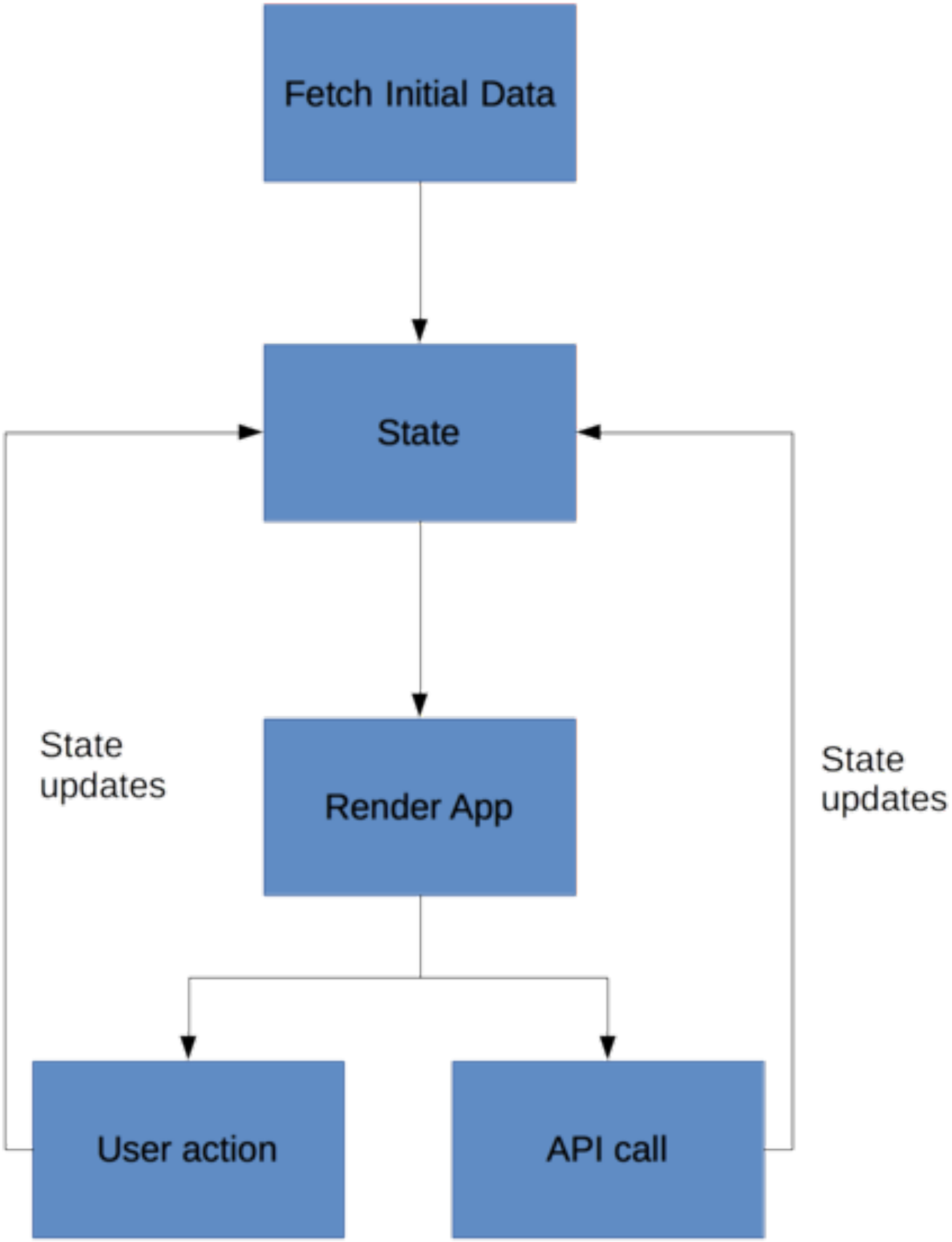
- Most people got ES6 and fetch, great job!
- Lots of people reached out for help
- We built an app!
- Codepen
- All JS in one script tag/html file
- Large functions

TIPS AND TRICKS

- Use an IDE!
- Use a linter
- Use many, smaller functions
- Separate async and sync code
- Use an architecture to structure your application

WEBAPP ARCHITECTURE

- There are many different architectures and design patterns and frameworks to help accomplish them
- I recommend the following:
  - Get initial data
  - State (global, all data in application we care about)
  - Render (Create html for user to interact with)
  - Data changes through user action or API call
  - Render again


Fetch Initial Data
State
State updates
Render App
State updates
User action
API call

BENEFITS

- Straightforward flow: data -> render -> something happens
- Easy to know where what code is, based on what it does
- Pushes async (generally more complicated) to the boundary of the app so we can handle it alone

CONCERNS

- Globals , isn’t that bad?
  - Yes, but can be okay if careful of concurrent updates
  - Technically should use framework like Redux which ensures single update
- Many functions to keep track of
  - Better than having large, complex functions
  - Can be alleviated with node and modules