COMP6080
Web Front-End Programming

Week 5.1
React Hooks - useContext

Managing State beyond a Component

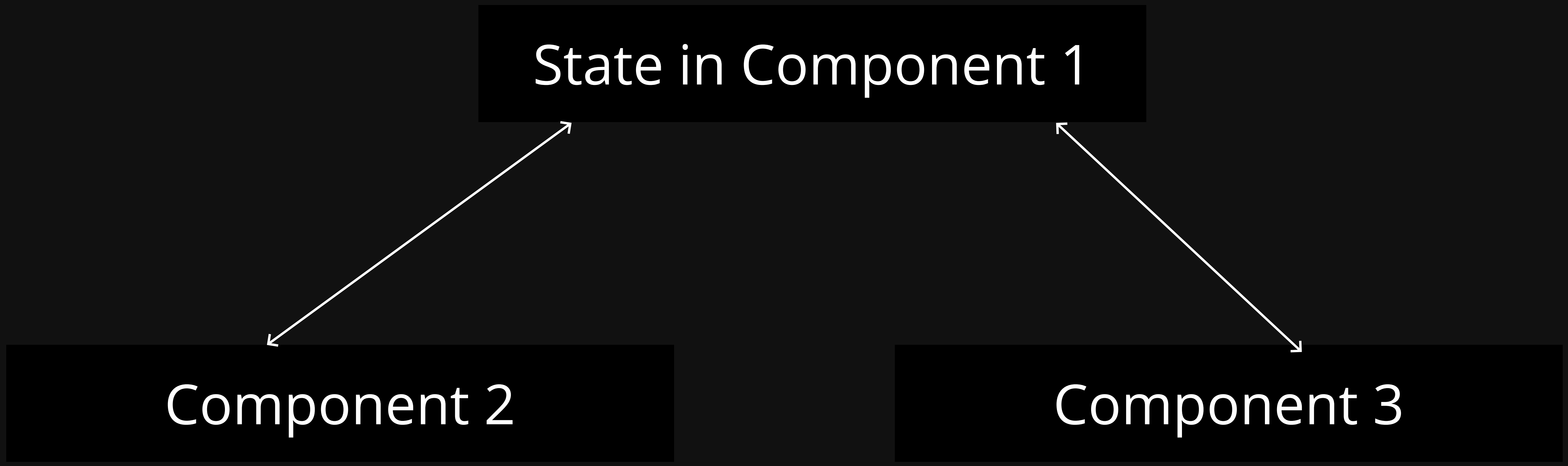
We've learned how to manage state within a component. However, what happens if we want to manage state between components? We have three key options:

1. Pass props down to subsequent components with "useState"
2. Use ReactJS "useContext" hook
3. Use a state management tool like **mobx** or **redux**

(3) is too complicated for the scope of this course. So we will discuss (1) and (2)

Passing props down

Let's do a demo.



State in Component 1

Component 2

Component 3

useContext hook

There are a large number of complex ways to use the useContext hook. We will demonstrate a very straightforward example through a demo that you can use in your own projects.

We will use this guide.

Course Updates: Notices

- Assignment 3 has been released, draft finalised tonight
- Assignment 2 marks intend to be released next Thursday
- Course content still not overdue, but also still delayed

Course Updates: Content for later weeks

The following are content taught in later weeks, so we won't touch today:

- Building your ReactJS app (for production)
- SASS
- Demonstrating people's ass2
- CSS for React, not in a single CSS file

Course Updates: Improvements

The following are areas we're looking to improve on or take note of:

- Asking tutors to setup lab marking more quickly
- Feedback passed on to tutors about how to run help sessions and labs better - so it's been heard
- More consistent recording quality (audio/video)
    - Student with long feedback, please email me!

Course Updates: Hard to fix

The following feedback items are hard to action

- Weighting assignment 3 more towards design/UX
- Using zoom instead of blackboard (zoom more stable)
- More consistent guest lecturer format
- More linear approach to teaching
- Teaching of JQuery in the course

Course Updates: 2021 Improvements

The following feedback items are being dealt with 2021

- Course will not have the same lab marking next year
- Less lengthy workload at the start of the course
- Covering CSS grids earlier on
- (Potentially) Longer but fewer help sessions (will poll)
- (Potentially) Reducing overall workload (will poll)
- Zain's lectures were some of the highest quality, but hardest to understand, will refine

Course Updates: Unsure what to do

The following feedback I (Hayden) am unsure what to do with:

- "Some lectures from external sources"
- "course coherence"
- "More relevant teaching for the completion of lab exercises"
- "Better use of livestreamed lectures, e.g. pre-planned tutorial-like exercises."
- "I think that the assignments/labs could be a bit more refined - they totally end up being refined, but this usually happens after the initial release"
- "The assignments feels like a massive step up from labs (especially ass2) so maybe a bit more helpful material or tutorial exercise will be appreciated"

Express Servers

The most common way to build an HTTP server with NodeJS (Javascript) is to build an express server. Let's do this with NPM.

Using a limited knowledge of HTTP, combined with your knowledge of Javascript so far, this can be a very simple library to make sense of.

Starting a ReactJS application

Now we're going to start a ReactJS application inside this repository. We'll use **create-react-app** for this, which will by default use **yarn** as the package manager.

Building Components, Talking to the API

A common design pattern for ReactJS apps is to use **react-router-dom** to manage your page routes, and then have two key folders in the `src` directory:

- "/components/": For all components that don't map directly to routes
- "/pages/": For all components that map directly to routes

Let's explore this together.

Using Libraries in ReactJS

One of the best aspects about working on these ReactJS projects is that you can easily leverage countless node packages. It is VERY rare that you will have to reinvest the wheel.

A simple example for assignment 3:

- Trying to setup a countdown timer

Assignment 3

Let's explore this together.