

Sharing Components Across Projects

By Anselm Christophersen
[@anselmdk](https://twitter.com/anselmdk) / www.anselm.dk

Danske Bank

Copenhagen React Meetup, 5th December 2018

Agenda

- About me/my Employer
- Motivation
- The Journey & Solution

Anselm Christophersen

FRONTEND LEAD IN FUTURE FINANCING

Moonshot in Danske Bank



Danske Bank in the House

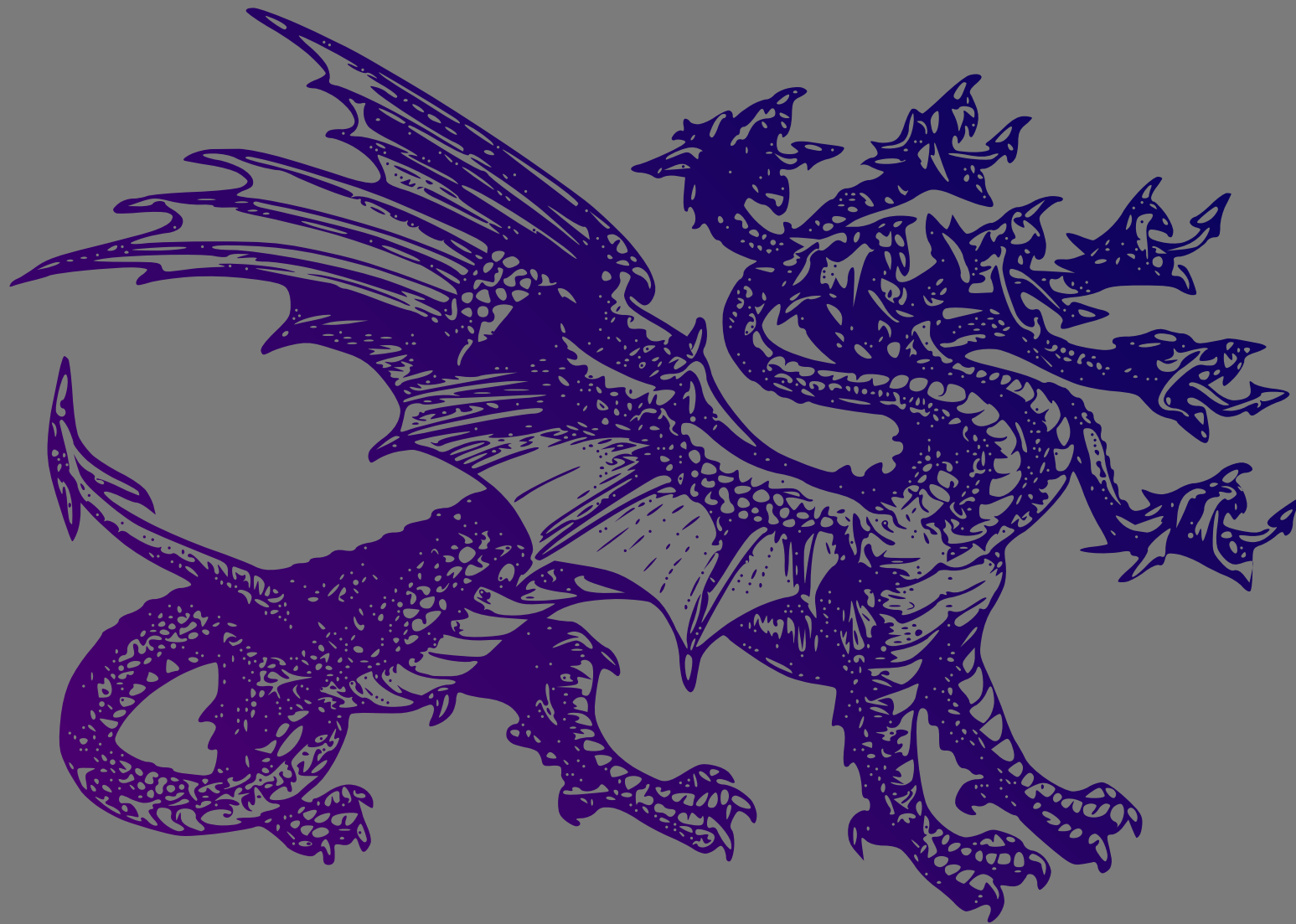
Motivation

- We have 2 clients
- Want to share code
 - Design system
 - Common libraries

How ?

Google is your friend

"JAVASCRIPT SHARED CODE BASE"



Lerna

Say no more



But...

I JUST WANT TO SHARE SOME
COMPONENTS

Stating the obvious
USE NPM!

Checklist for shared libraries

- Be accessible as a normal npm package
- Follow Semver
- Be pre-built

Approach

- Utilize the `prepare` section in the `scripts` section
 - `prepare` is run on the package after it's installed via `npm install`
- Babel for compilation
- `Docz` for preview and development
 - Could also be Storybook, Styleguidist or [any other solution](#)

Let's get started

```
mkdir shared-component  
cd shared-component  
npm init  
npm install --save react
```

A Simple Component

src/HelloWorld.js

```
import React from 'react'  
const HelloWorld = () => (  
  <h1>Hello world from React Component!</h1>  
)  
export default HelloWorld
```


Testing/developing the Component

1. INSTALL DOCZ

```
npm install docz --save-dev
```

2. ADD TO `scripts` SECTION IN `package.json`

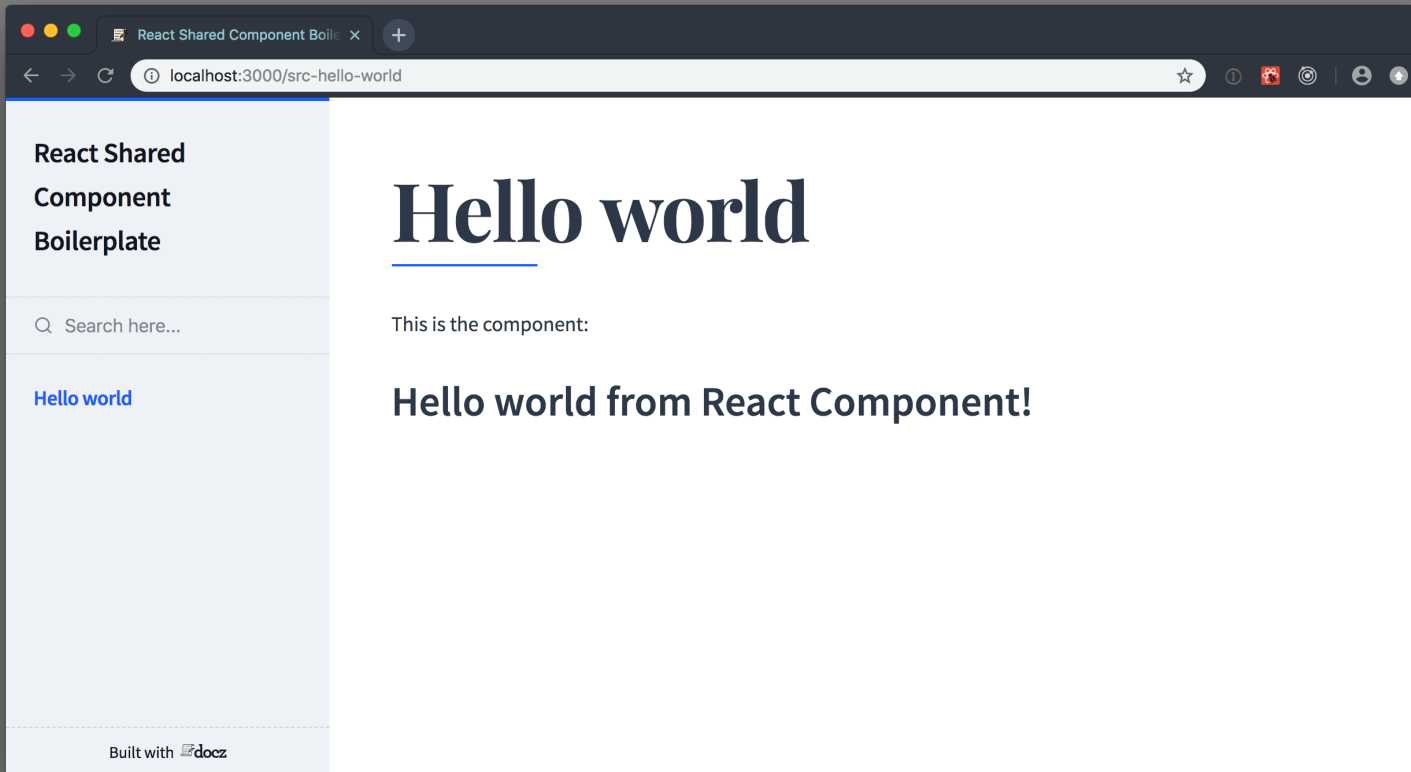
```
"start": "docz dev"
```

3. CREATE src/HelloWorld.mdx

```
---  
name: Hello world  
---  
  
import HelloWorld from './HelloWorld'  
  
# Hello world  
  
This is the component:  
  
<HelloWorld />
```

4. RUN DOCS

```
npm start
```



React Shared Component Boilerplate


🔍 Search here...

[Hello world](#)

Hello world

This is the component:

Hello world from React Component!

Built with  docz

Building for Release

1. INSTALL BABEL AND PRESETS

```
npm install --save-dev @babel/core @babel/cli @babel/preset-env  
@babel/preset-react
```

2. CREATE .babelrc

```
{  
  "presets": [  
    "@babel/env",  
    "@babel/preset-react"  
  ]  
}
```

3. ADD TO `scripts` SECTION IN `package.json`

```
"build": "babel src -d build"
```

4. RUN THE BUILD

```
npm run build
```

Publishing

1. CREATE `.gitignore`

```
/build  
/node_modules  
/.docz
```

2. CREATE `.npmignore`

```
/src  
.babelrc
```

3. ADD TO `scripts` SECTION IN `package.json`

```
"prepare": "npm run build"
```

4. COMMIT AND PUSH

```
git init
git add *
git commit -m "Initial Commit"
git remote add origin git@github.com:anse1mdk/react-shared-
component-boilerplate.git
git push -u origin master
```

Utilizing the Component

Component has now been published to

github.com/anselmdk/react-shared-component-boilerplate

HOW TO UTILIZE IT?

- **Option 1: Point directly to the git repository**
- **Option 2: Set up an npm registry**

Creating a Simple React Project

```
npx create-react-app example-project  
cd example-project  
npm start
```

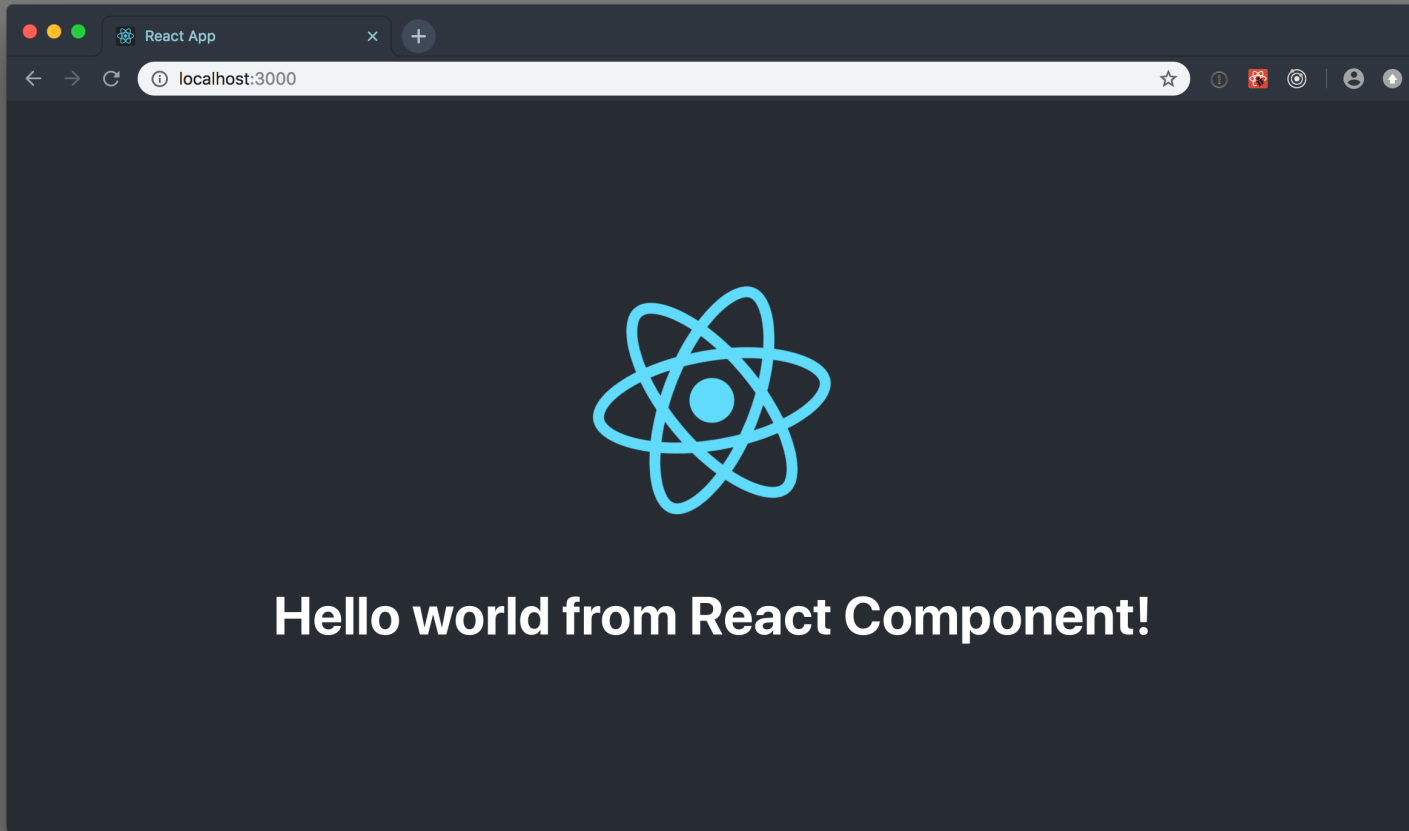
Adding the Shared Component

1. INSTALL THE SHARED COMPONENT

```
npm install --save anseimdk/react-shared-component-boilerplate
```

2. ADD THE FOLLOWING BITS TO TO THE App.js FILE

```
import { HelloWorld } from 'react-shared-component-boilerplate'  
<HelloWorld />
```



Hello world from React Component!

Thank you

SLIDES

anselm.dk/talks



SOURCE CODE

<https://github.com/anselmdk/react-shared-component-boilerplate>
<https://github.com/anselmdk/react-shared-component-example>



Next Steps

SET UP AN NPM REGISTRY

- npmjs.com - USD 7/month
- Enterprise solutions, e.g. [Artifactory](#)
- Set up your own:
 - [verdaccio](#)
 - [cnpm](#)

MONOREPO & LERNA