Introducing React Router

React Developer Series

Peter Munro

Introducing React Router

Introducing React Router

Why Routing?

We need routing in order to:

- maintain application state

- distinguish between logically separate areas of our app

- allow access to areas based on authorization rules

In addition, routing enables us to:

- bookmark a 'deep' location within an app and return to it later

- share such a URL with others

Client-side Routing Mechanisms

Two techniques are commonly used. With both:

- Page is not reloaded

- Forward and back buttons work

Hash-based Routing

- Anchor name to identify content: `<a name="about"><h1>About</h1></a>`

- Fragment identifier to go to that content: `http://example.com/#/about`

HTML5 History API

- Uses `history.pushState()` and friends to navigate

    - `history.pushState(stateObj, "page 2", "bar.html");`

React Router Concepts

- Dynamic Routing

- Routes as Nestable React Components

<Route>

`<Route>`:

- either renders its component or renders `null`

- only renders if `location` matches the `path` defined

```
<Route path="/profile" component={Profile} />
```