Up

REACT WITH ES6

BRIAN HOLT

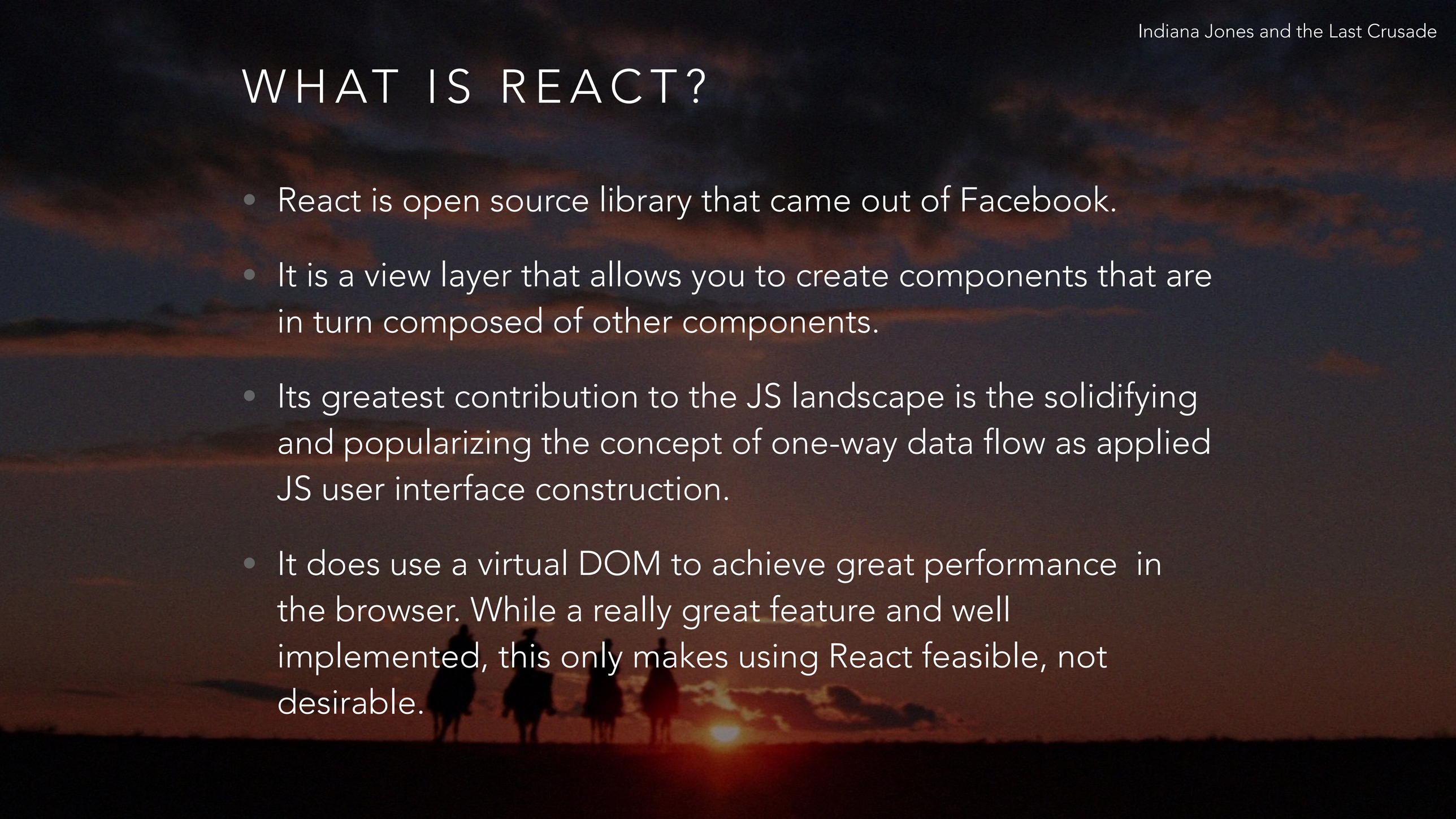
Frozen

@HOLTBT – BRI@NHO.LT

PREVIOUSLY

CURRENTLY

NETFLIX

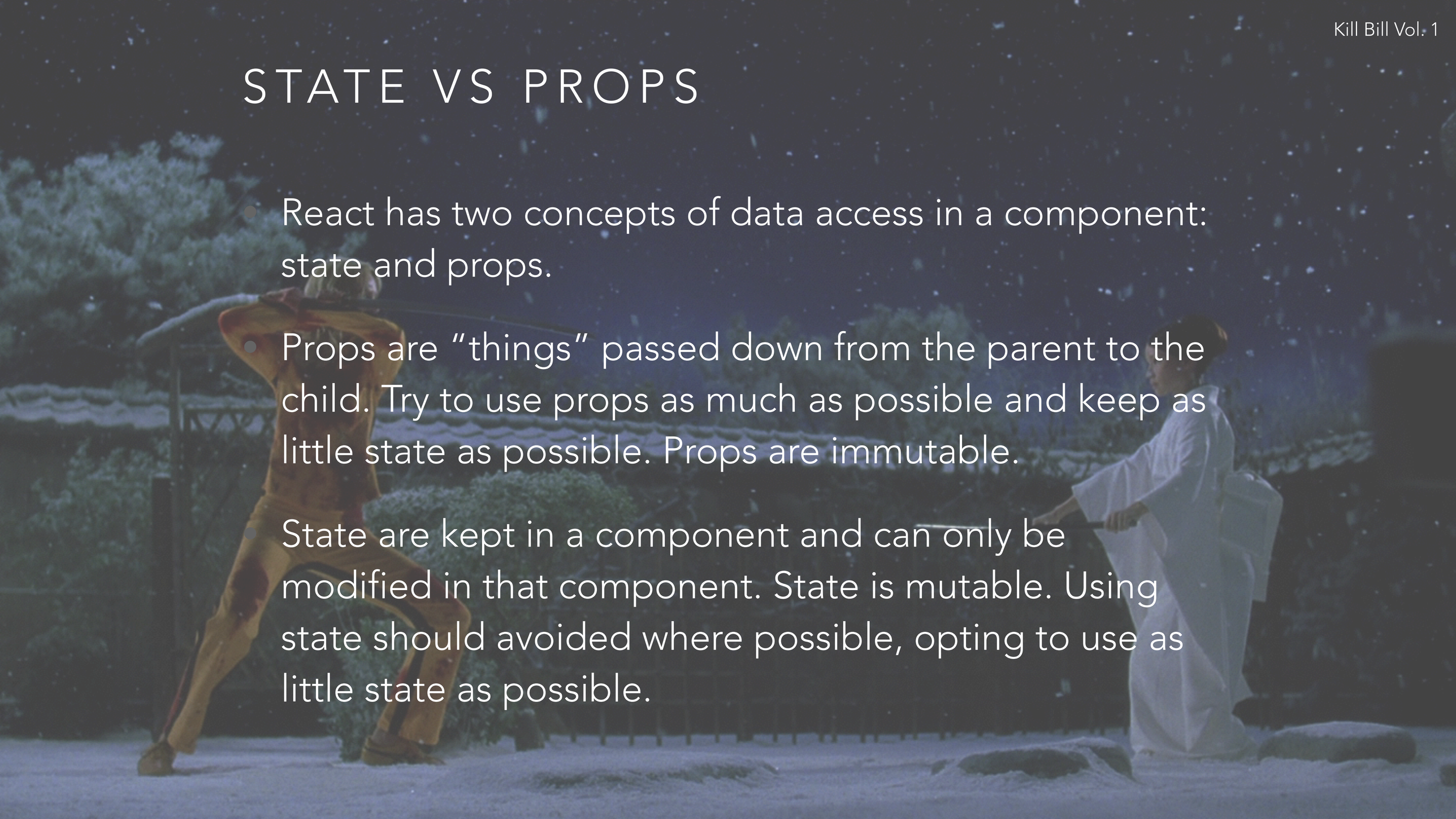


Indiana Jones and the Last Crusade

WHAT IS REACT?

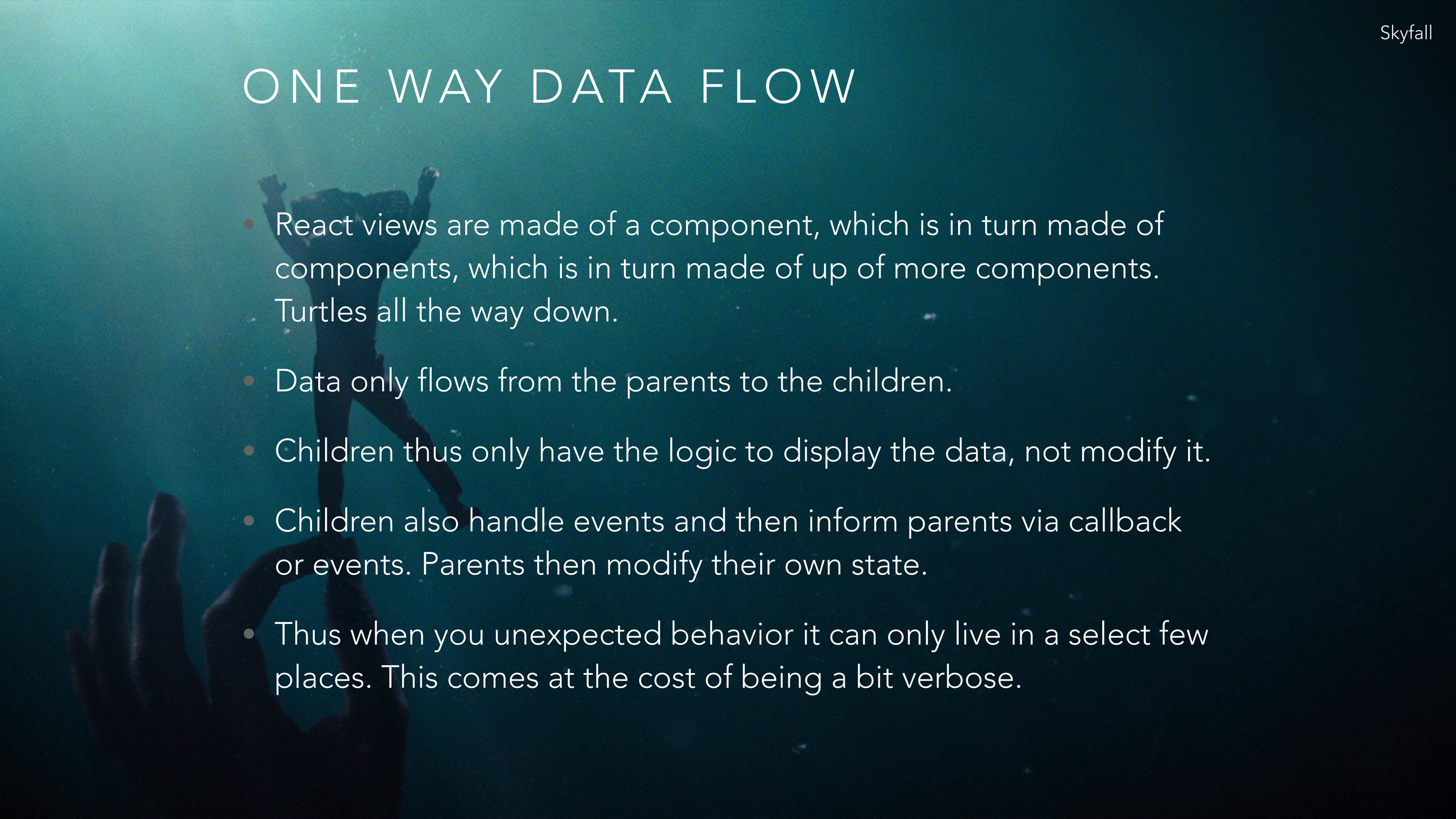
- React is open source library that came out of Facebook.
- It is a view layer that allows you to create components that are in turn composed of other components.
- Its greatest contribution to the JS landscape is the solidifying and popularizing the concept of one-way data flow as applied JS user interface construction.
- It does use a virtual DOM to achieve great performance in the browser. While a really great feature and well implemented, this only makes using React feasible, not desirable.


The Dark Knight
“STATE IS THE ENEMY OF A MAINTAINABLE APP.”
— BATMAN



STATE VS PROPS

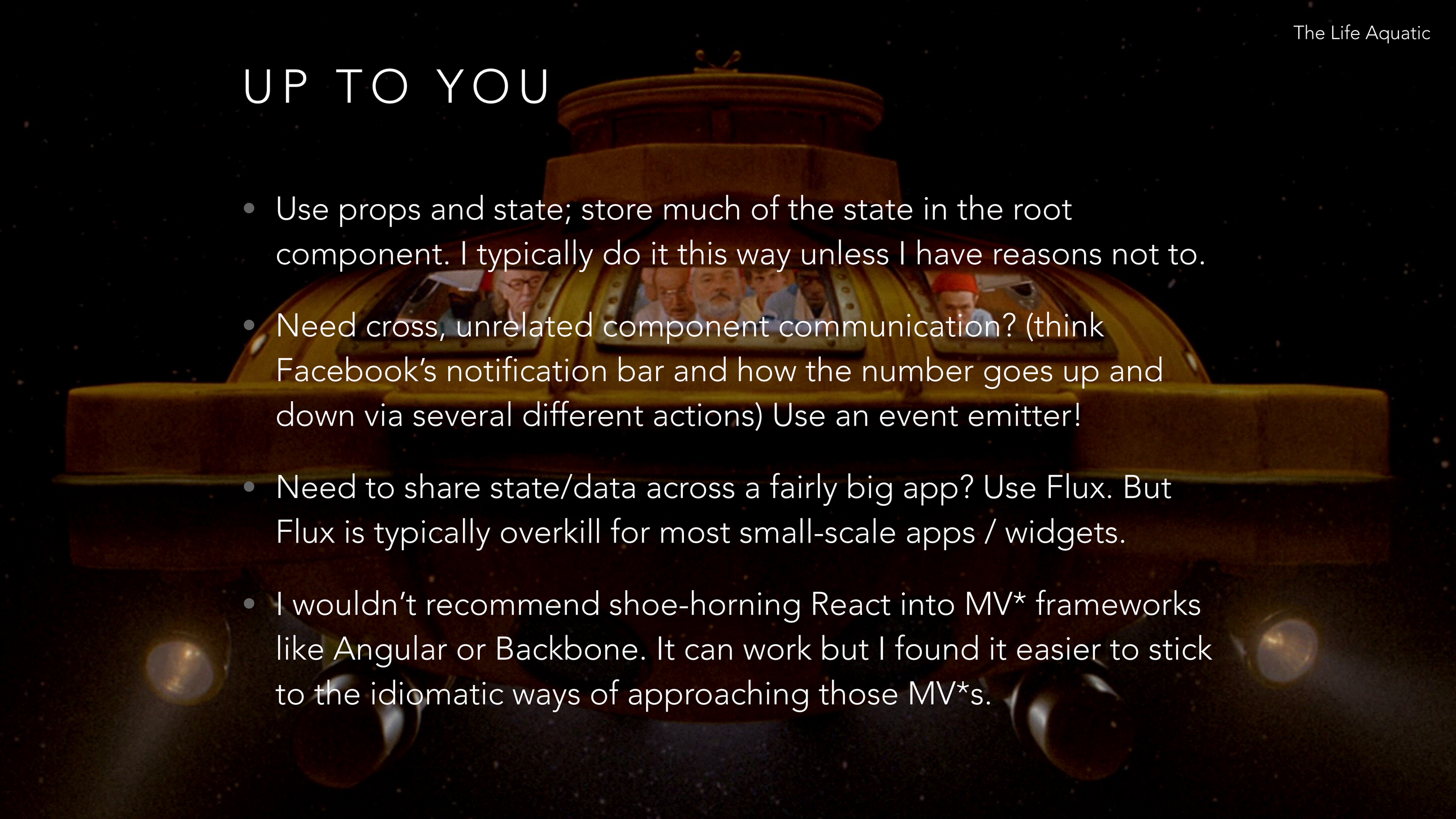
- React has two concepts of data access in a component: state and props.
- Props are “things” passed down from the parent to the child. Try to use props as much as possible and keep as little state as possible. Props are immutable.
- State are kept in a component and can only be modified in that component. State is mutable. Using state should avoided where possible, opting to use as little state as possible.



ONE WAY DATA FLOW

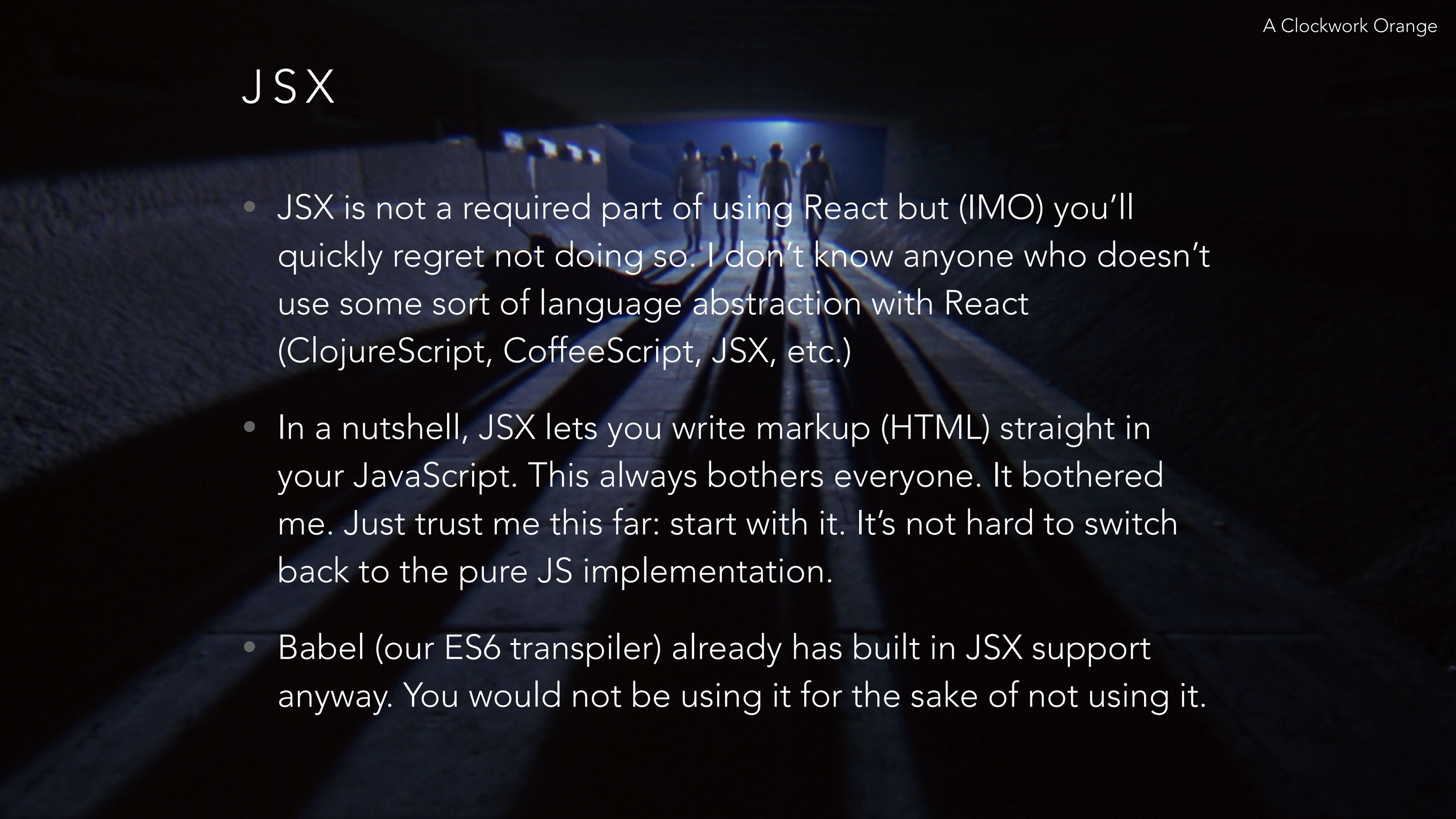
- React views are made of a component, which is in turn made of components, which is in turn made of up of more components. Turtles all the way down.
- Data only flows from the parents to the children.
- Children thus only have the logic to display the data, not modify it.
- Children also handle events and then inform parents via callback or events. Parents then modify their own state.
- Thus when you unexpected behavior it can only live in a select few places. This comes at the cost of being a bit verbose.


The Lego Movie
SO WHAT DO I DO WITH DATA IF REACT DOESN'T CARE ABOUT IT?
wtf state



UP TO YOU

- Use props and state; store much of the state in the root component. I typically do it this way unless I have reasons not to.
- Need cross, unrelated component communication? (think Facebook's notification bar and how the number goes up and down via several different actions) Use an event emitter!
- Need to share state/data across a fairly big app? Use Flux. But Flux is typically overkill for most small-scale apps / widgets.
- I wouldn't recommend shoe-horning React into MV* frameworks like Angular or Backbone. It can work but I found it easier to stick to the idiomatic ways of approaching those MV*s.



JSX

- JSX is not a required part of using React but (IMO) you'll quickly regret not doing so. I don't know anyone who doesn't use some sort of language abstraction with React (ClojureScript, CoffeeScript, JSX, etc.)
- In a nutshell, JSX lets you write markup (HTML) straight in your JavaScript. This always bothers everyone. It bothered me. Just trust me this far: start with it. It's not hard to switch back to the pure JS implementation.
- Babel (our ES6 transpiler) already has built in JSX support anyway. You would not be using it for the sake of not using it.


Indiana Jones and the Temple of Doom
lolwut
MASHING TOGETHER
MARKUP AND JAVASCRIPTS?
ISN'T THAT A BAD IDEA?

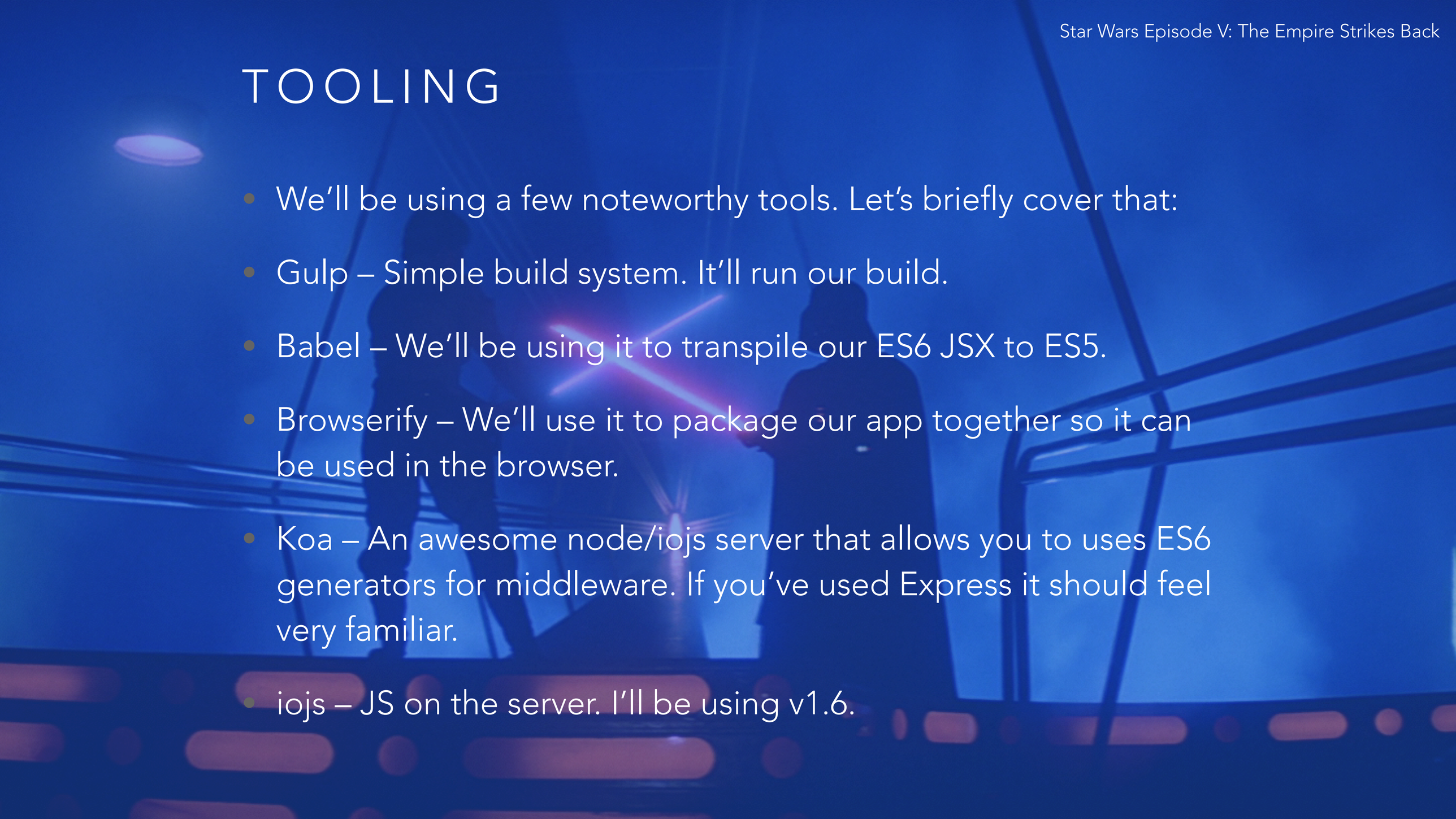
2001: A Space Odyssey

MUSHING TOGETHER CONCERNS

- Being used to MVC from the server, a lot of us get uncomfortable at the idea of mushing together concerns. I certainly was.
- What is good for the server is not good for the interface. Separating models out from controllers allows great code re-use for server. It's a mess on the client.
- Instead it makes sense to thinks of things in terms of components, which are made up of behaviors (or containers) and layouts. It often mimics your markup.

ECMASCRIPT 6

- Latest release of the JavaScript language. Now in final draft. Wide adoption is beginning but very incomplete.
- Babel is a transpiler that allows to write it now and have it translated to work on ES5 which is now widely adopted. You could have as easily used Tracer or another one. Babel just has JSX translation built in.
- We'll covering a small surface area of useful features in ES6 and specifically how it applies to React. React has new syntax that is specifically only for use with ES6.



TOOLING

- We'll be using a few noteworthy tools. Let's briefly cover that:
- Gulp – Simple build system. It'll run our build.
- Babel – We'll be using it to transpile our ES6 JSX to ES5.
- Browserify – We'll use it to package our app together so it can be used in the browser.
- Koa – An awesome node/iojs server that allows you to uses ES6 generators for middleware. If you've used Express it should feel very familiar.
- iojs – JS on the server. I'll be using v1.6.


Ghostbusters II
DEMO
github.com/btholt/es6-react-pres


Batman Begins
@HOLBT
BRI@NHO.LT
state is bad lol