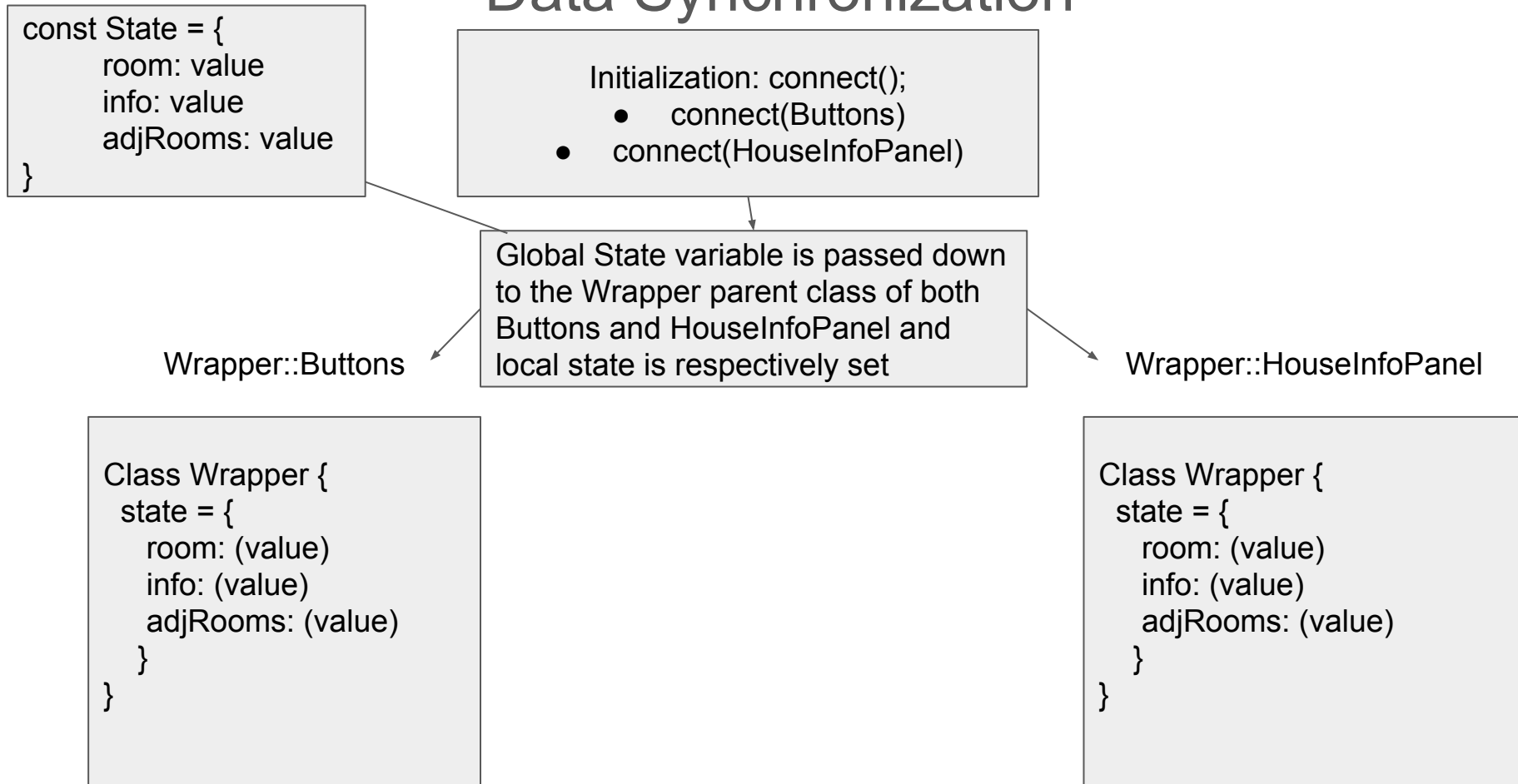


# Data Synchronization



\* When Buttons and HIP mount, the `componentDidMount()` call sends the value of `_listener` which is a call to `setState` into the `listeners` set, which is waiting on standby by for a button click a.k.a. `changeRoomButton()`

## Data Synchronization (cont)

```
const listeners = new Set()
```

```
componentDidMount() {  
  _listener = () => this.setState({etc...})  
}
```

Wrapper

Wrapper

```
class Buttons {  
  * Wrapper parent passes its state to  
  Buttons as props  
}
```

```
class HouseInfoPanel {  
  * Wrapper parent passes its state to HIP  
  as props  
}
```

# Data Synchronization (cont)

`changeRoom(roomSelection)`

```
graph TD; A[changeRoom(roomSelection)] --> B[1. Global State is changed to the selection<br/>2. Environment background is updated<br/>3. updateComponents is triggered]; B --> C[Wrapper of both Buttons and HIP gets its local state changed by the cb() function in updateComponents. Afterwards, the new local state is passed to the children from Wrapper as props and then the cycle waits until changeRoom is triggered again.];
```

1. Global State is changed to the selection
2. Environment background is updated
3. `updateComponents` is triggered

Wrapper of both Buttons and HIP gets its local state changed by the `cb()` function in `updateComponents`. Afterwards, the new local state is passed to the children from Wrapper as props and then the cycle waits until `changeRoom` is triggered again.