



Signals vs React Hooks: The Battle for Clean and Performant Code

Henry LAGARDE



DATADOG





Henry Lagarde

Software Engineer @ Datadog

Agenda

01  How to handle states?

02  Signals VS Hooks

03  Let's play

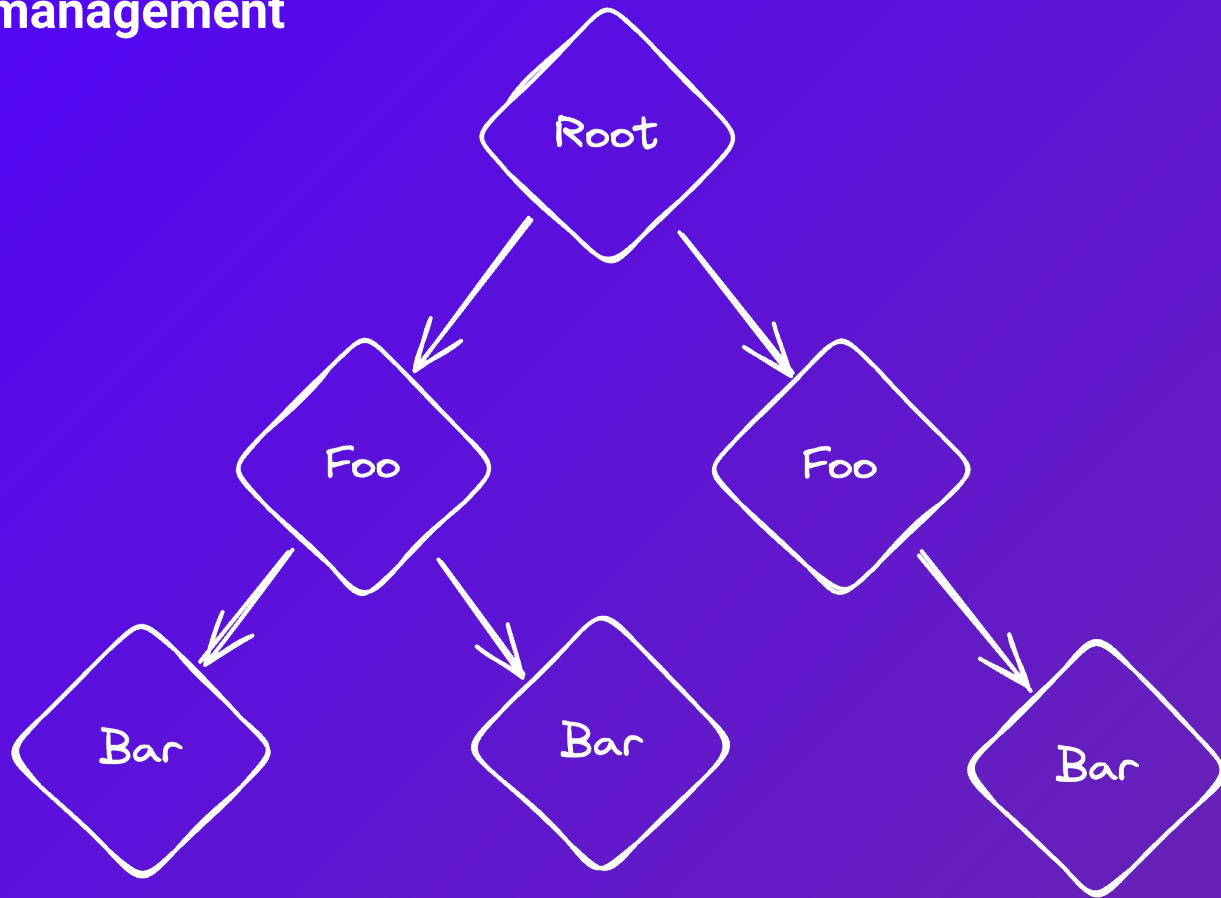


How to handle states ?



How to handle states ?

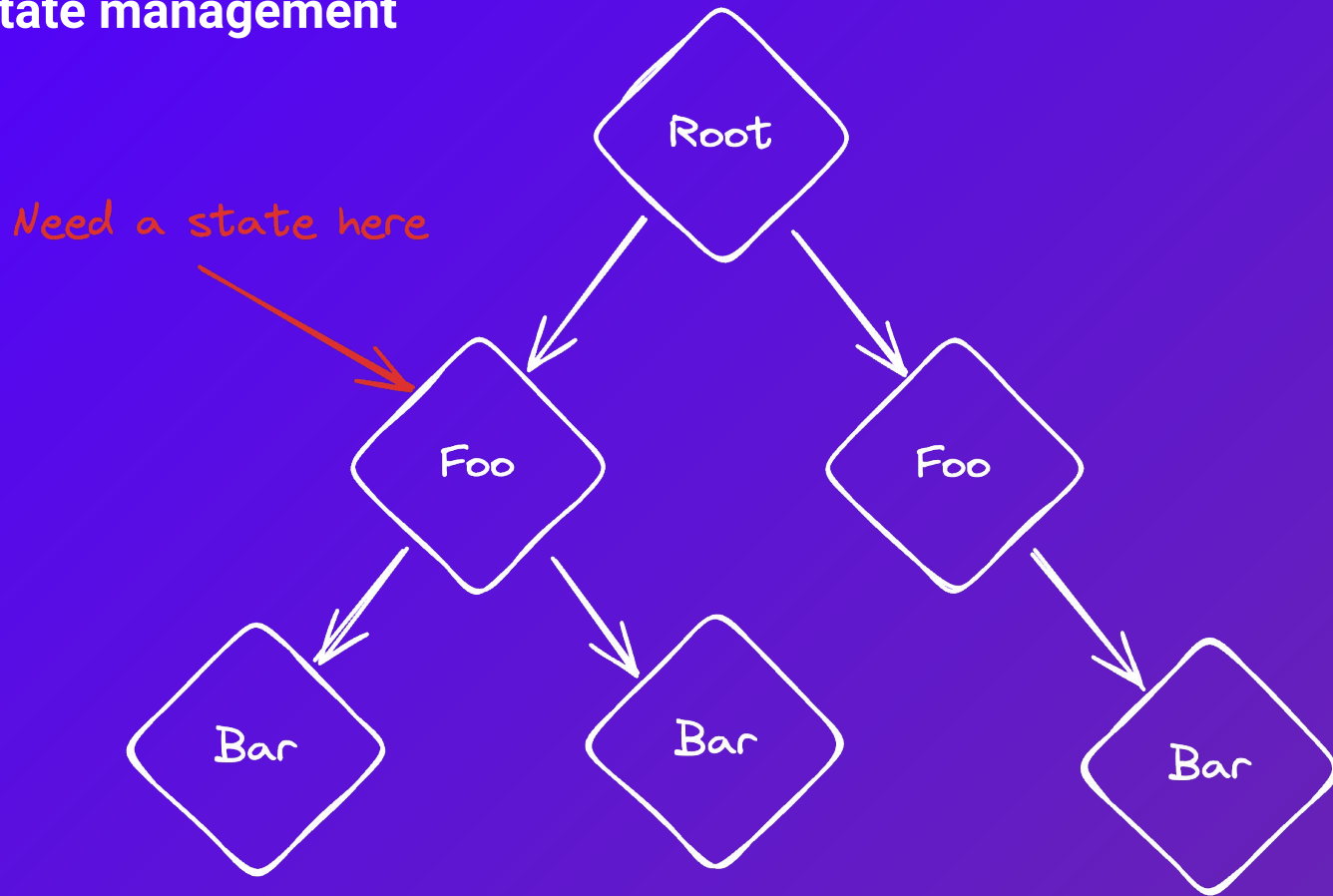
Basic state management





How to handle states ?

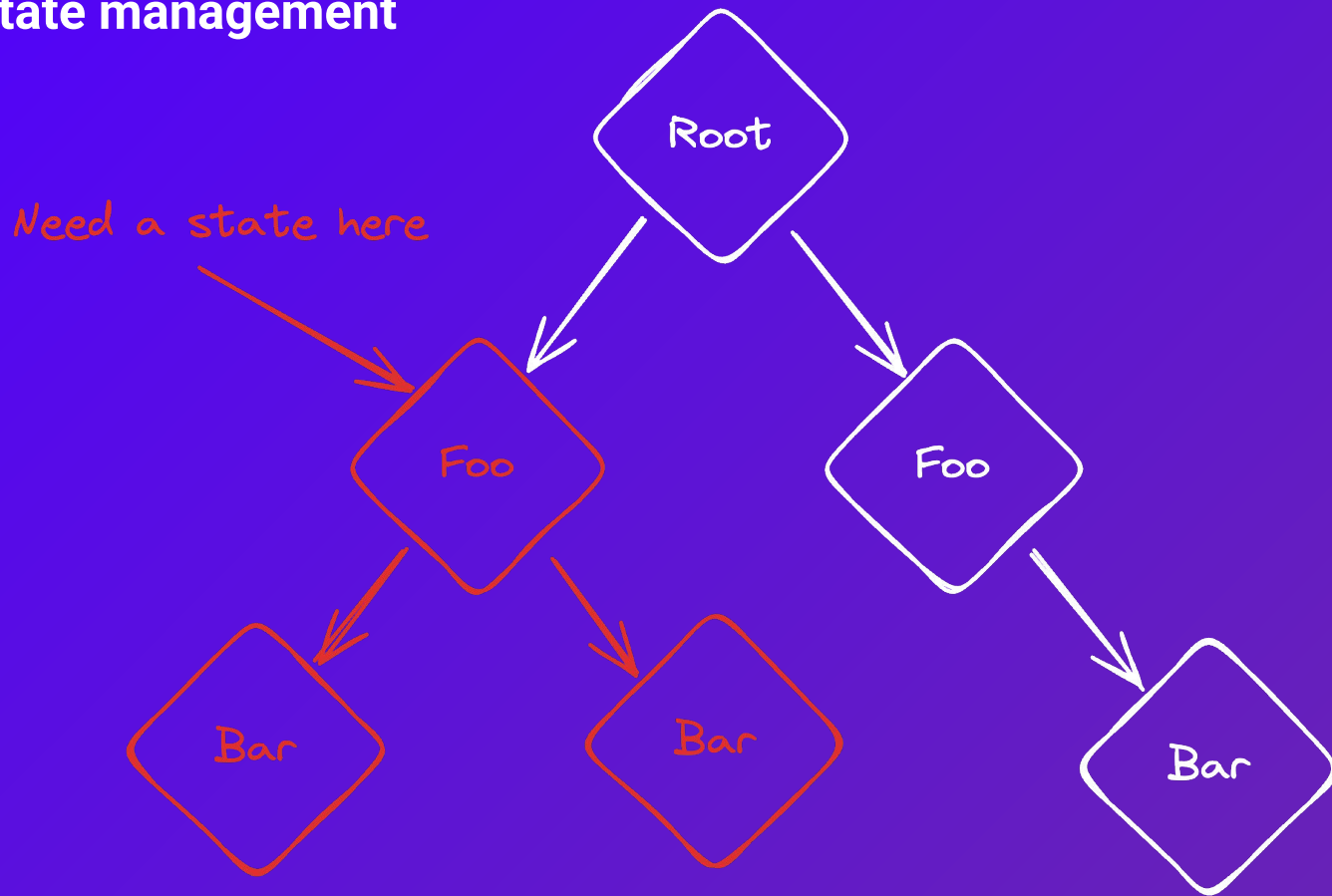
Basic state management





How to handle states ?

Basic state management

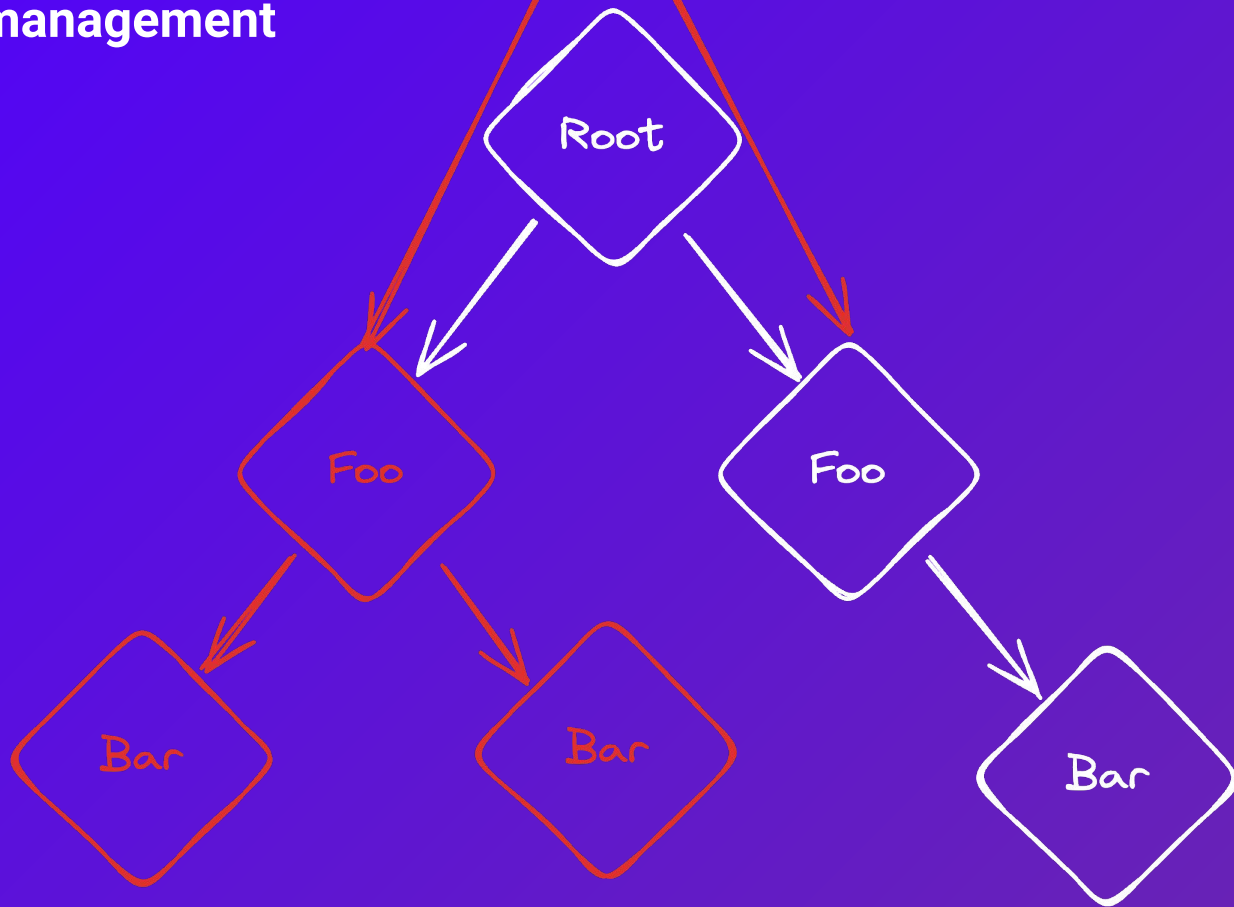




How to handle states ?

Basic state management

Need the same state here

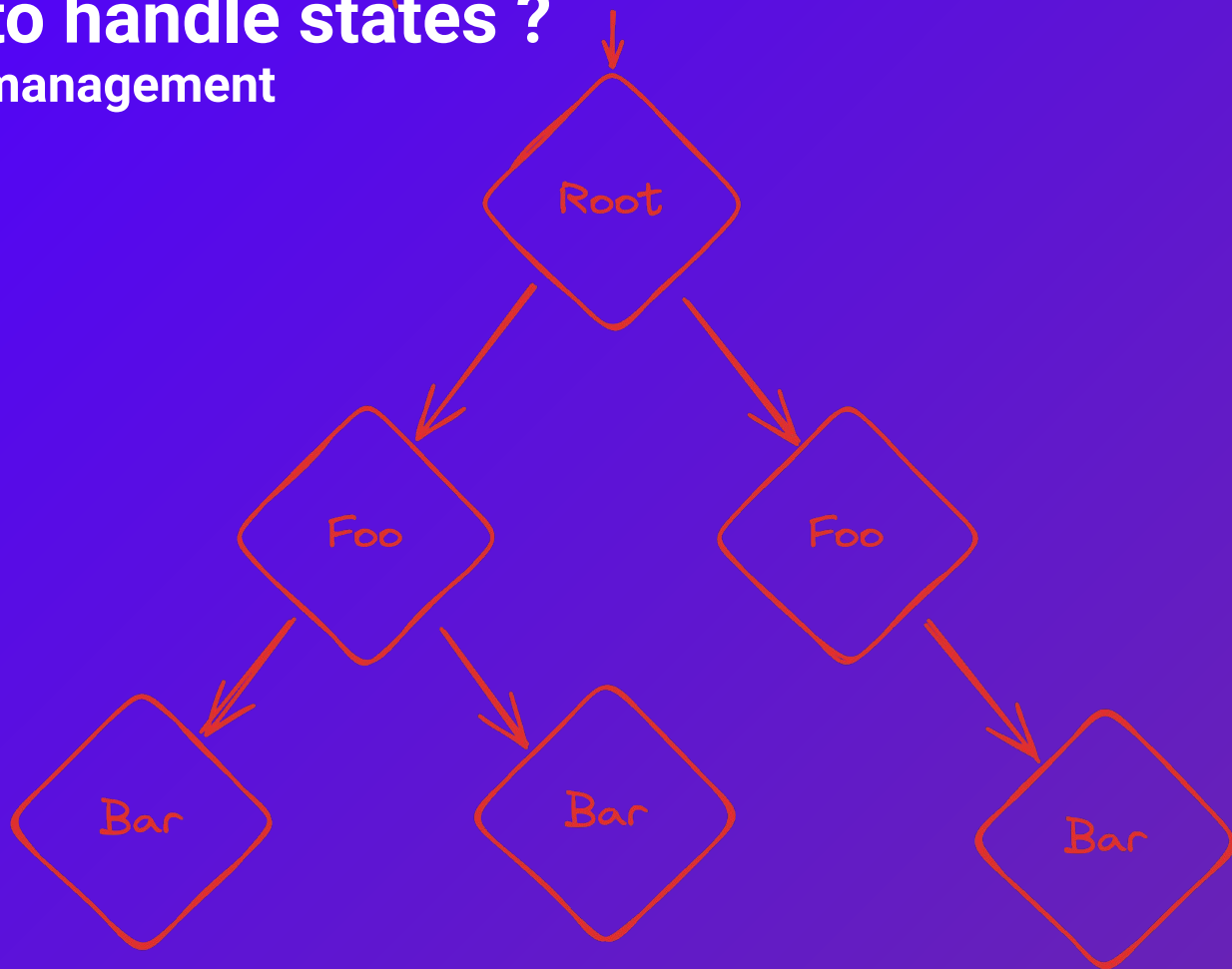




How to handle states ?

Basic state management

We put the state here 🤔

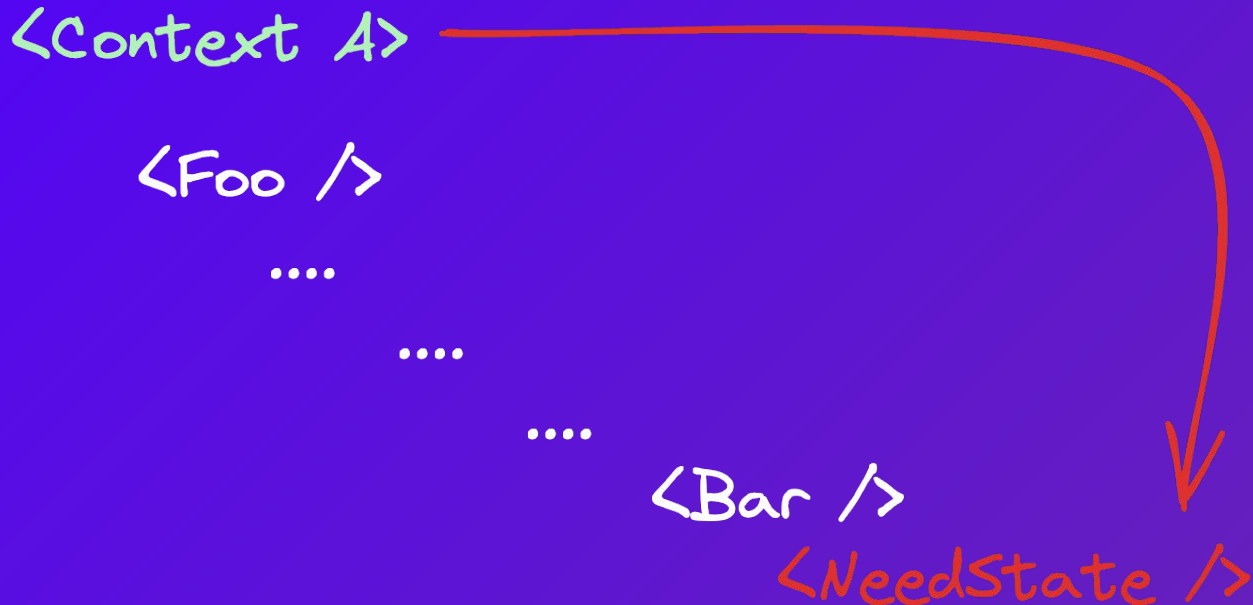




How to handle states ?

Here come the context

The context propagate
the state to the component





How to handle states ?

Here come the context

<Context A>

<ContextB>

<ContextC>

<Foo />

....

<Needs_AB />

....

....

<Bar />

<NeedState />

WELCOME TO

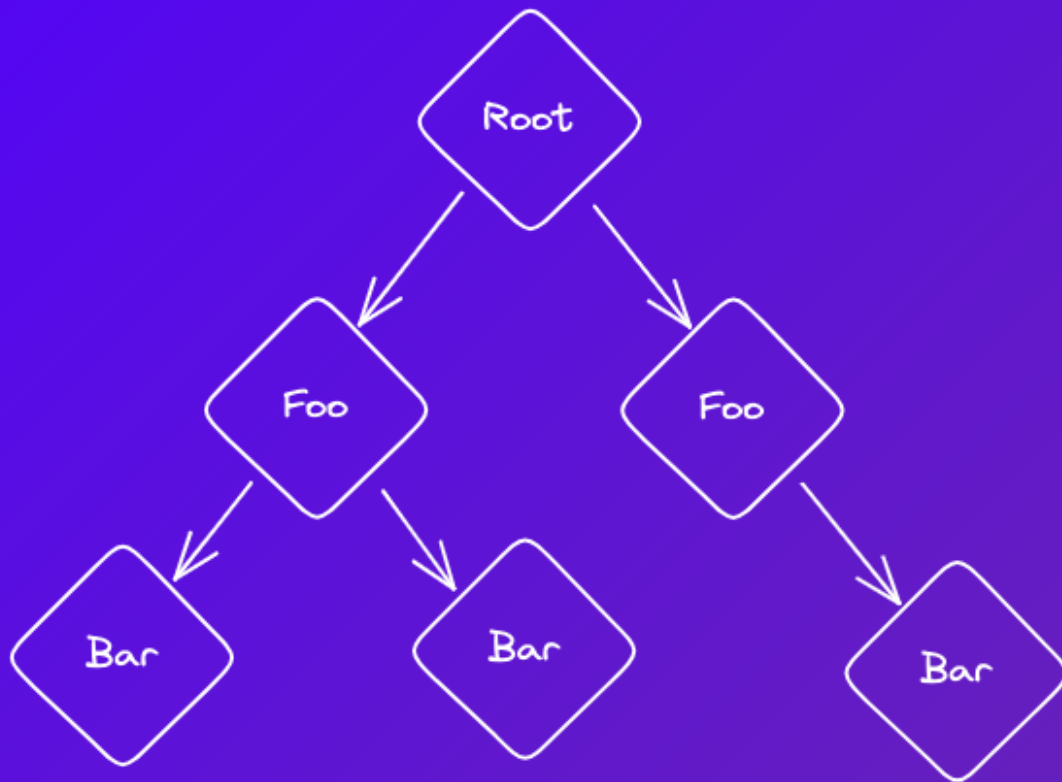
HELL

```
<FooContext.Provider value="foo">
  <BarContext.Provider value="bar">
    <BazContext.Provider value="baz">
      <AnotherProvider>
        <AgainAnotherProvider configHereAlso={someEnvironmentVar}>
          <AreYouKiddingMeProvider>
            <IHateReactContextHellProvider frustration={true}>
              <FinallyYourApp />
            </IHateReactContextHellProvider>
          </AreYouKiddingMeProvider>
        </AgainAnotherProvider>
      </AnotherProvider>
    </BazContext.Provider>
  </BarContext.Provider>
</FooContext.Provider>
```

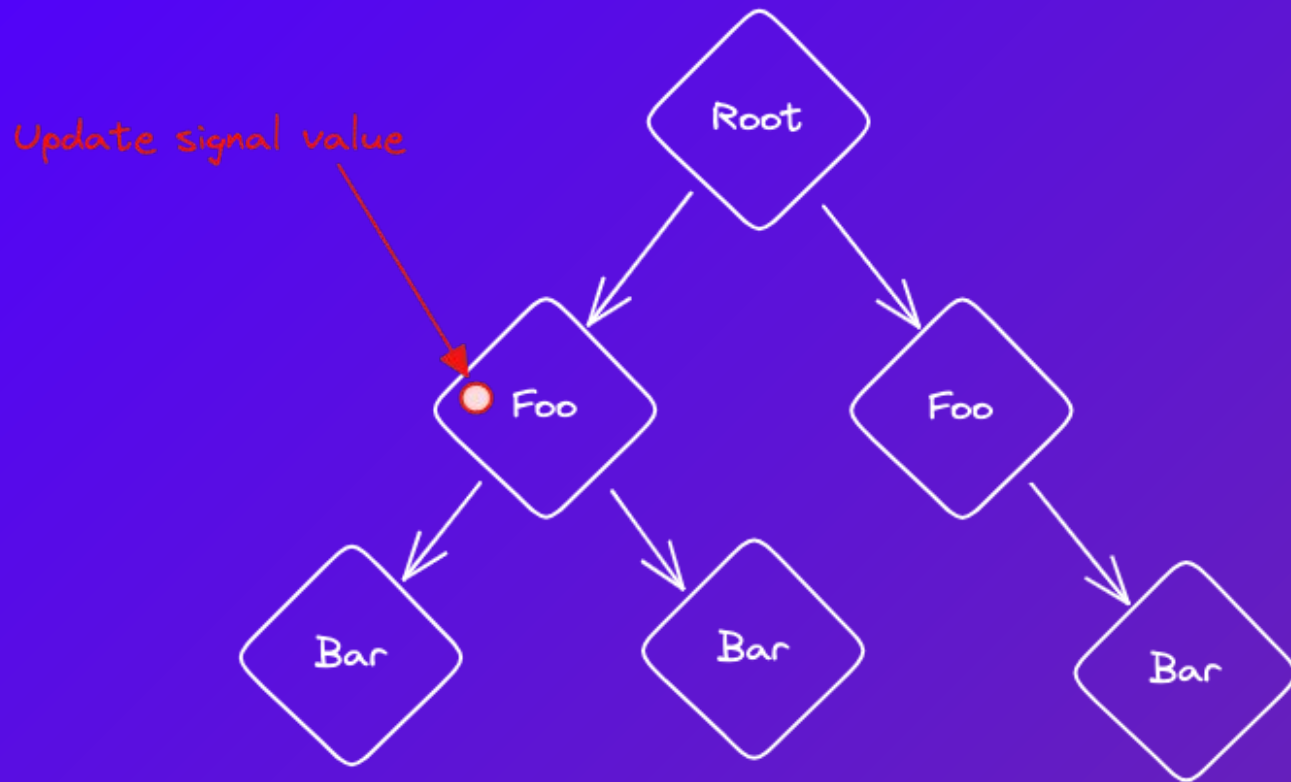



What are Signals?

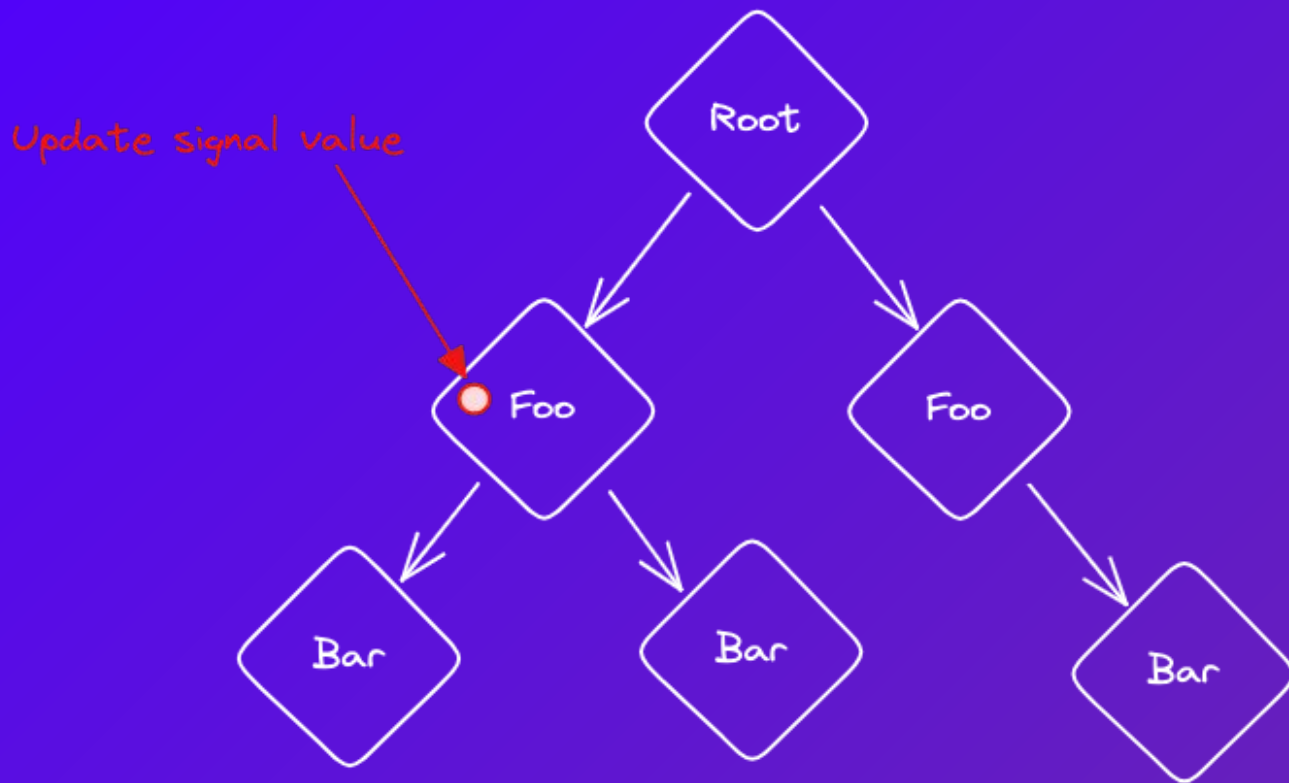
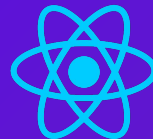
What are signals?



What are signals?



What are signals?





Signals vs Hooks

Hook

Reactively respond to changes in data or state

Component Rendering

React core feature since 2019

Available at components level

Need to declare dependencies

Signal

Data level Rendering

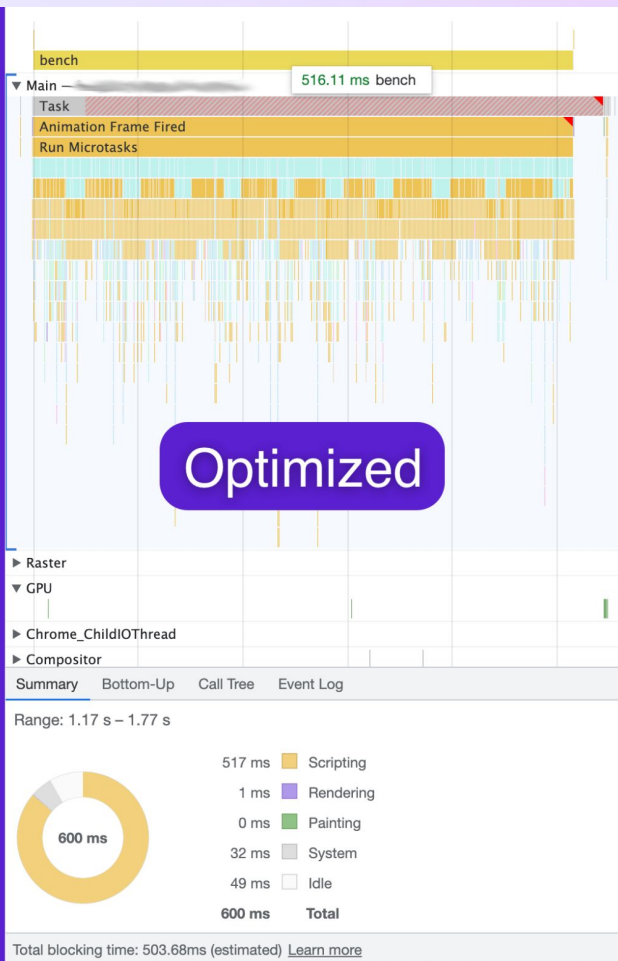
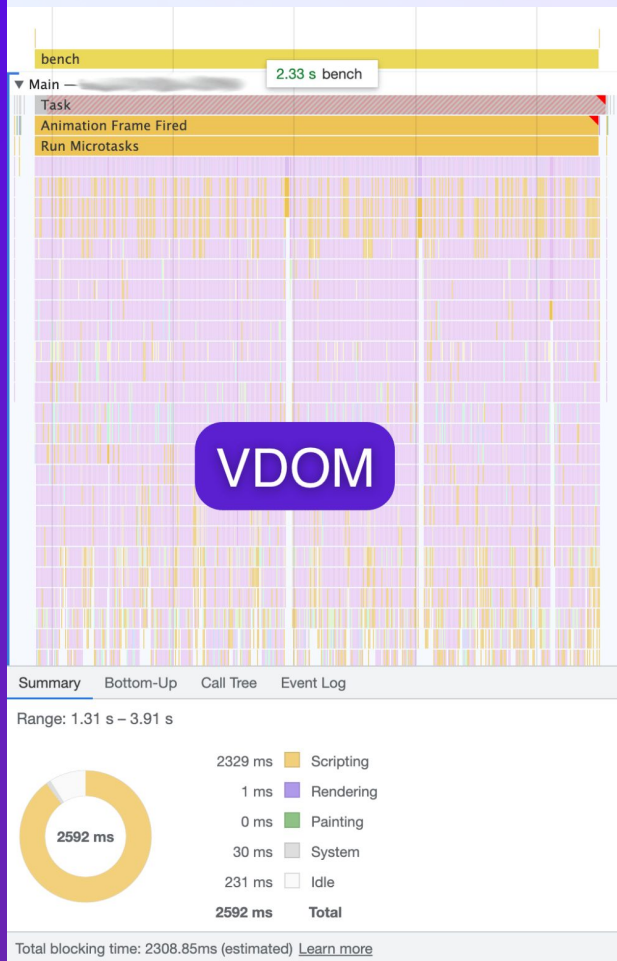
Recent concept from third-party library

Available everywhere

Automatic dependencies detection

Rendering improvements

- Lazy by default
- Optimal updates
- Optimal dependency tracking
- Direct access



Signals 101

```
import { signal } from "@preact/signals-react";

const count = signal(0);

const App = () => (
  <div>
    <button onClick={() => count.value++}>{count}</button>
  </div>
);
```

Signals 101

```
import { signal } from "@preact/signals-react";

const count = signal(0);

const App = () => (
  <div>
    <button onClick={() => count.value++}>{count}</button>
  </div>
);
```

```
import { count } from "./signals";

function AnotherComponent() {
  return (
    <div>
      <p>Current Count: {count}</p>
    </div>
  );
}
```

Signals 101

```
import { signal } from "@preact/signals-react";

const count = signal(0);

const App = () => (
  <div>
    <button onClick={() => count.value++}>{count}</button>
  </div>
);
```

```
import { count } from "./signals";

function AnotherComponent() {
  return (
    <div>
      <p>Current Count: {count}</p>
    </div>
  );
}
```

```
effect(() => {
  yourFunction(count.value);
});
```




Let's Play



Let's Play

- Create a simple todo App
- Migrate it with Signals instead of hooks



Conclusion



Thank you

Twitter: **@dev_lagarde**

LinkedIn : **Henry LAGARDE**



DATADOG