

НУЖЕН ЛИ ВАМ REACT SSR ?
Gentle introduction to react SSR

Dmitry Poddubniy a.k.a
[@mr47](#)

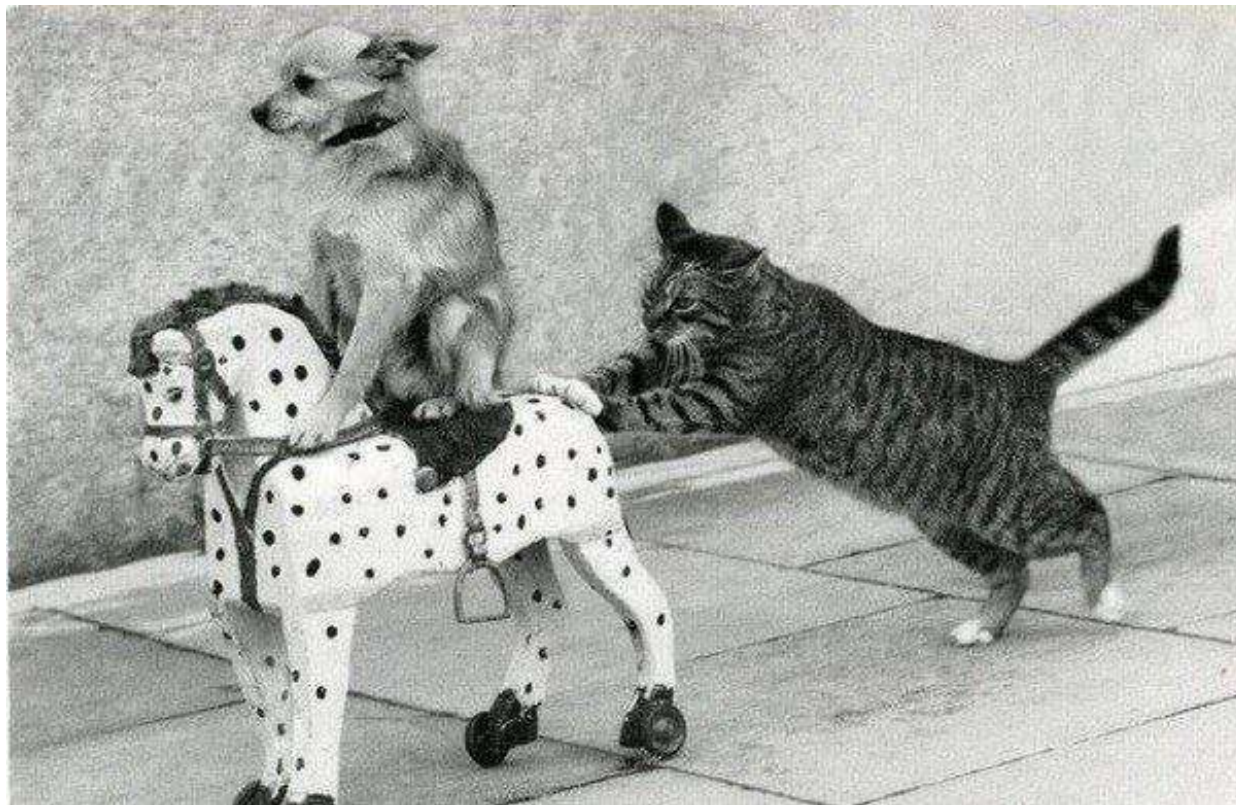


КОГДА НУЖНО ИСПОЛЬЗОВАТЬ SSR

- SEM, SEO - Baidu, Bing, Yahoo
- Хотим наилучшую производительность
- Убрать "лаг" на начальный рендеринг
- Письма / PDF
- Ваше приложение работает

Как же выглядит SSR на самом деле.

KAK TO BOT TAK -



REACTDOM.RENDERTOSTRING

```
const component = (props, context) => (  
  <div {...props}/>  
);
```

REACTDOM.RENDERTOSTRING

```
"use strict";
```

```
var component = function component(props, context) {  
  return React.createElement("div", props);  
};
```

REACTDOM.RENDERTOSTRING

```
<!-- code -->  
<div data-reactroot data-reactid="1" data-react-checksum="-111">  
  <div data-reactid="2"></div>  
</div>  
<!-- code -->
```


Что дает ReactDOM.renderToString:

- + Universal app
- Sync
- Rerender always
- CPU waste
- Memory waste

О ЧЕМ МАЛО КТО ГОВОРИТ:

- `ComponentDidMount`
- Компоненты не поддерживающие SSR
- Canvas / WebGL
- Двойной рендеринг
- Race conditions
- Передача начального состояния на клиент
- Позиционирование / масштабирование элементов
- Анимация
- `setState transaction`

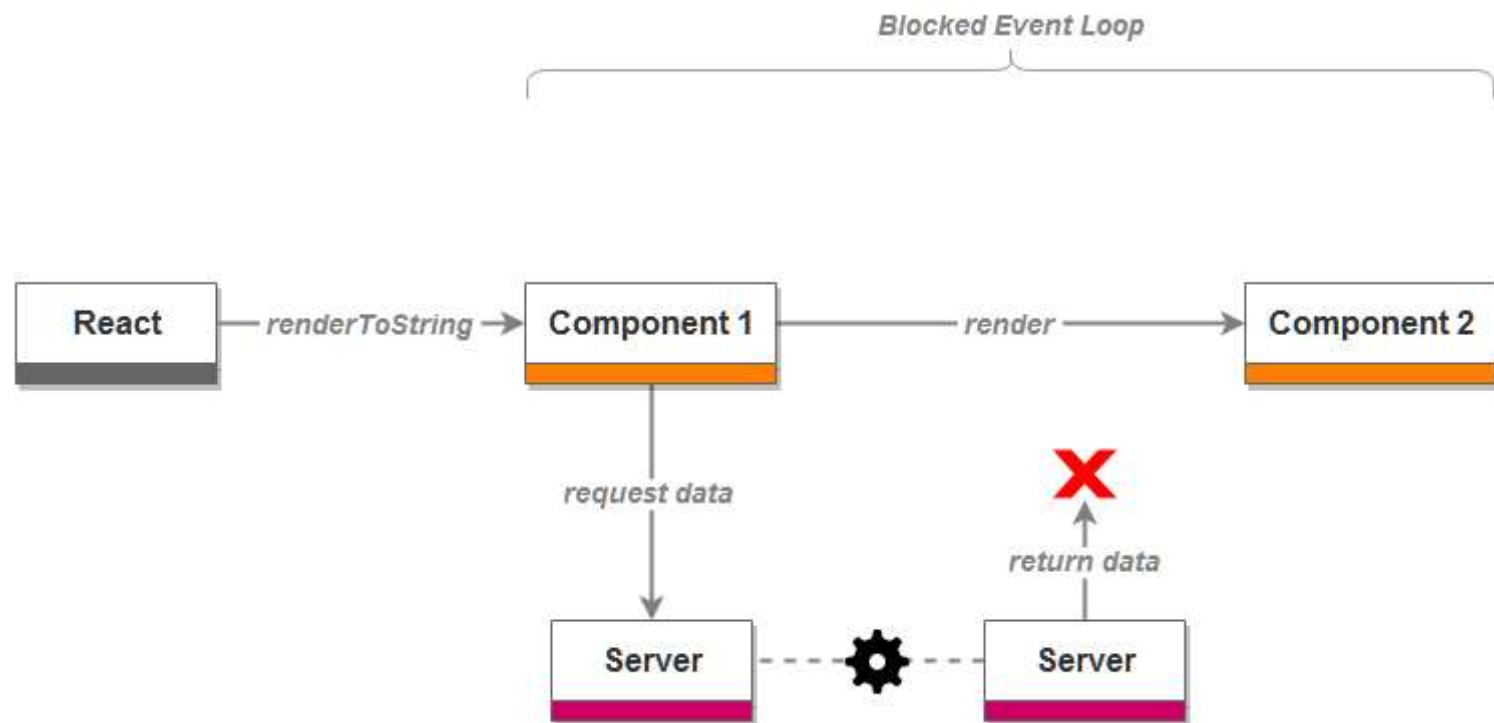
Warning: React attempted to reuse markup in a container but the checksum was invalid.

- «*console.log*»

ПРИМЕР ПРОБЛЕМЫ

```
class Slider extends React.PureComponent{
  constructor(props, context){
    super(props, context);
    this.state = { animated: false };
  }
  componentWillMount(){
    setTimeout(
      () => this.setState({ animated: true }),
      400)
  }
  render(){
    const { animated } = this.state;
    const c = animated ? 'my-div--animated' : '';
    return(
      <div className={`my-div ${c}`}>
        {/* ... Some code ... */}
      </div>
    );
  }
}
```

```
class Clock extends React.Component {
  constructor(props) {
    super(props);
    this.state = { date: new Date() };
    this.timers = {};
  }
  componentDidMount() {
    this.timers.timerID = setInterval(
      () => this.tick(),
      1000
    );
  }
  componentWillUnmount() {
    clearInterval(this.timers.timerID);
  }
  tick() {
    this.setState({ date: new Date() });
  }
  render() {
    return (
      <div>
        <h1>Hello, world!</h1>
        <h2>It is {this.state.date.toLocaleTimeString()}.</h2>
      </div>
    );
  }
}
```



В React 16.0.0-rc.2, проблема решена

componentDidCatch

ЕСЛИ Ы ВСЕ ЕЩЕ ХОТИТЕ SSR ТОГДА,



ЧТО ЖЕ НАМ ПОМОЖЕТ:

- `redux`
- `redux`
- `react-router`
- `no-ssr` component
- `ssr proxy` (Internal project)
- `react-jobs`
- `react-server`
- `react-ssr-optimization`
- less components do more

REACT-SERVER:

- Parallelize backend queries
- Bundle data for transfer to browser
- Stream HTML to the browser
- Wire up elements as they arrive
- Enable page-to-page transitions
- Render quickly, even when the backend is slow



| Все ради скорости.

- «*The Flash*»

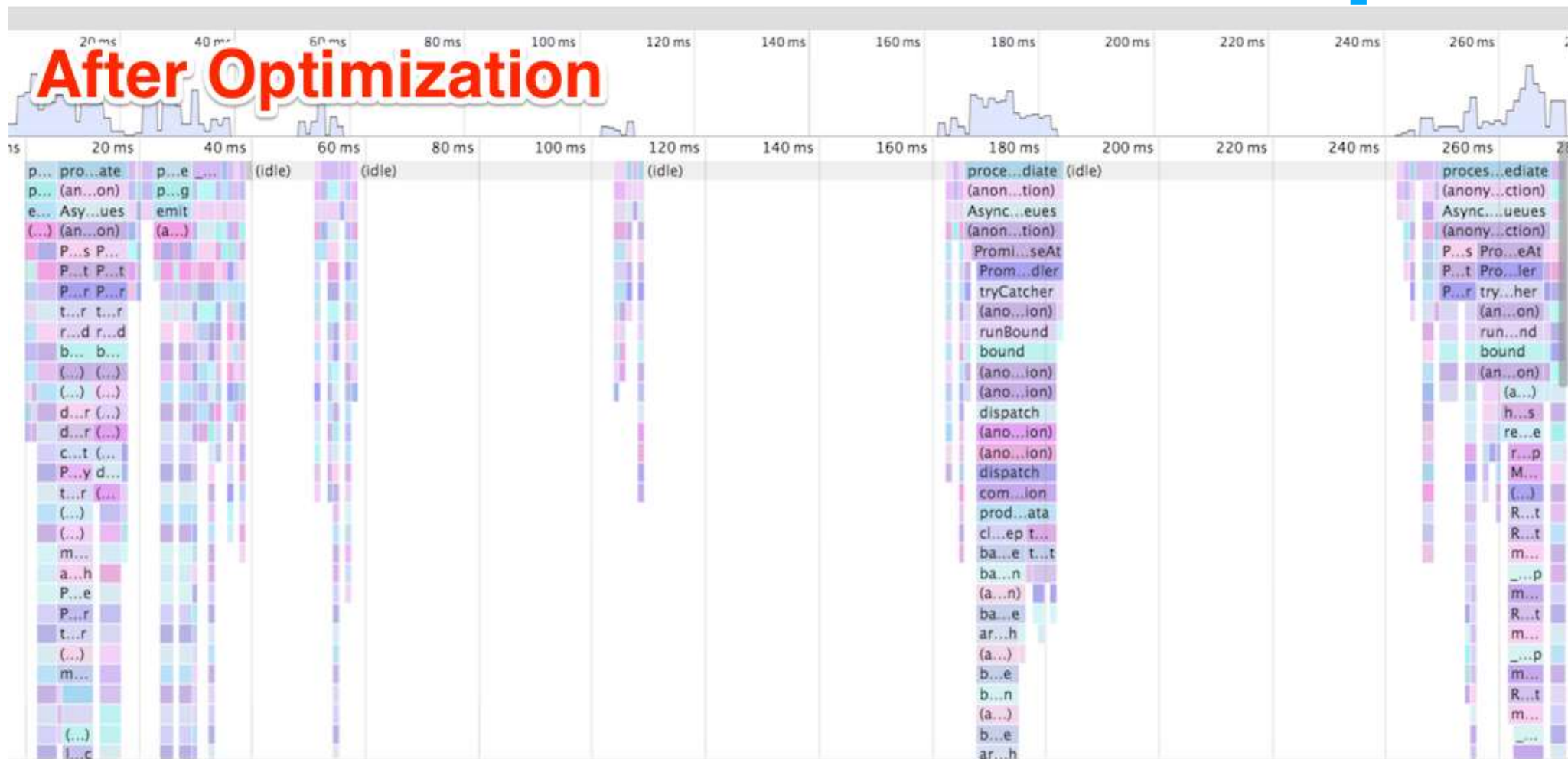
```
const component = (props) => (  
  <div className="product">  
    <img src={props.product.image}/>  
    <div className="product-detail">  
      <p className="name">{props.product.name}</p>  
      <p className="description">{props.product.description}</p>  
      <p className="price">Price: ${props.selected.price}</p>  
      <button type="button" onClick={props.addToCart} disabled={props.  
        {props.inventory ? 'Add To Cart' : 'Sold Out'}}  
      </button>  
    </div>  
  </div>  
>);
```

```
const component = () => (`
  <div className="product">
    <img src=${product_image}/>
    <div className="product-detail">
      <p className="name">${product_name}</p>
      <p className="description">${product_description}</p>
      <p className="price">Price: ${selected_price}</p>
      <button type="button" onClick={this.addToCart} disabled={this.props
        {this.props.inventory ? 'Add To Cart' : 'Sold Out'}}
      </button>
    </div>
  </div>
`);
```

Before Optimization

ReactDOMServer.renderToString(...)

ПОСЛЕ ОПТИМИЗАЦИИ





React's server-side rendering can become a performance bottleneck for pages requiring many virtual DOM nodes.

- *Walmart Labs*

The background is dark and textured, featuring a faint, glowing lightning bolt in the upper right quadrant and a subtle, ethereal profile of a person's face on the right side. The main text is centered and reads:

| У всего есть цена...

- «*The Flash*»

УВИДИМСЯ В БУДУЩЕМ!

