React Native: Under the hood

Alex Kotliarskyi
Facebook

Plan

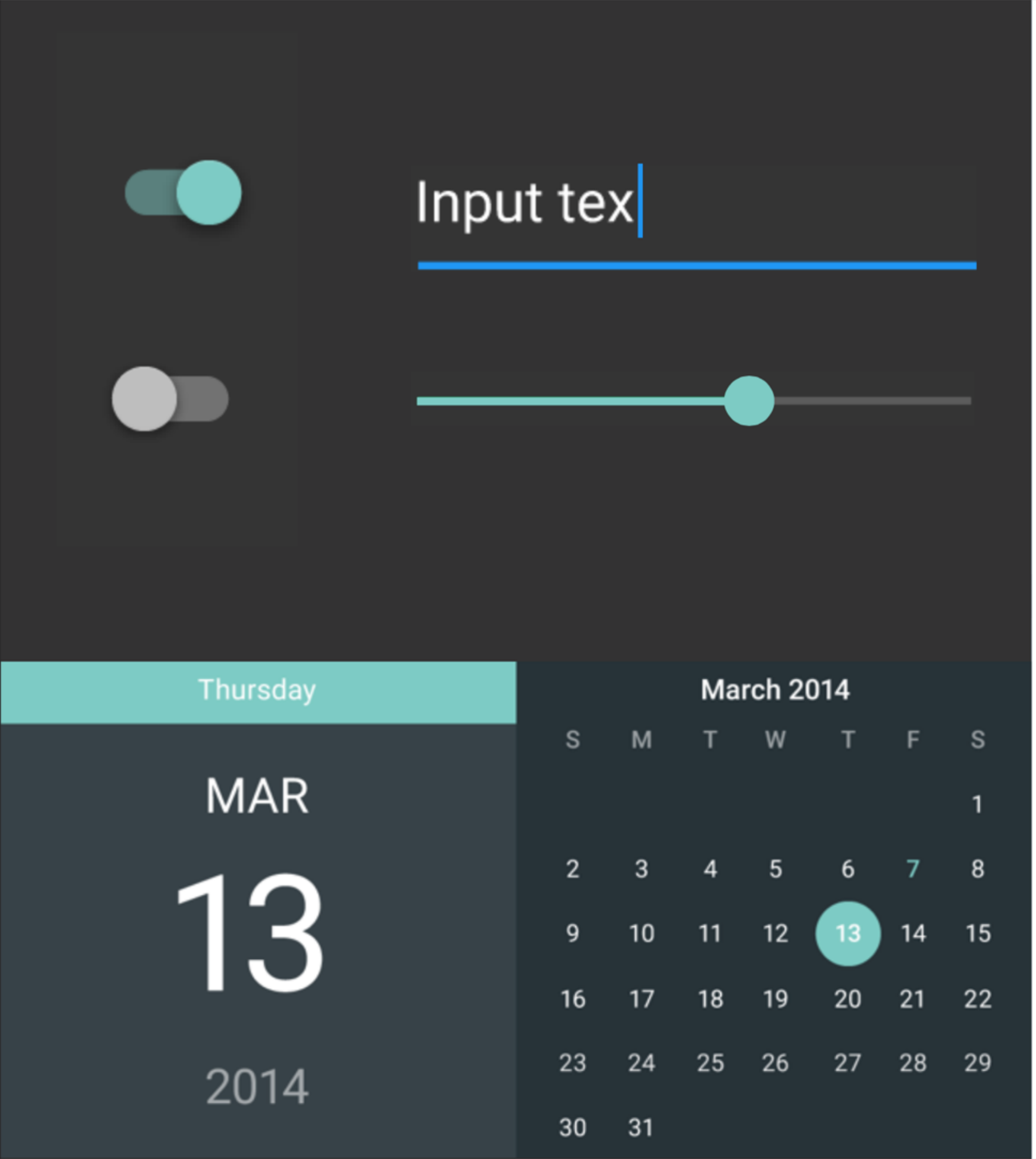
1. Why native apps matter?
2. How ReactJS works
3. Running ReactJS on native platforms

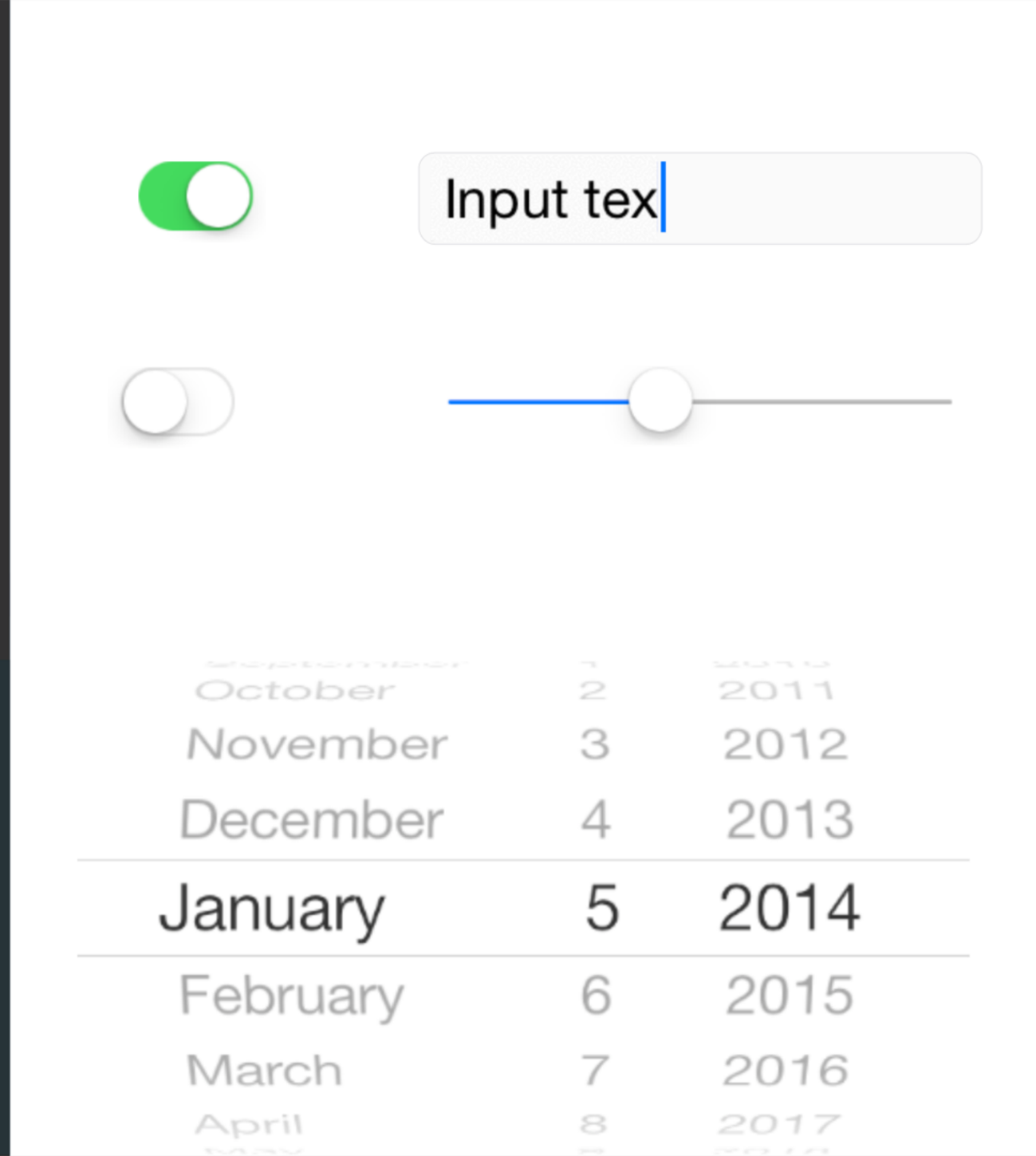
Why do we

native apps?

Native Apps

- Fast, responsive
- Complex gestures and smooth animations
- Consistent with platform


Input tex
Thursday
MAR
13
2014
March 2014
S M T W T F S
1
2 3 4 5 6 7 8
9 10 11 12 13 14 15
16 17 18 19 20 21 22
23 24 25 26 27 28 29
30 31


Input tex
October 2 2011
November 3 2012
December 4 2013
January 5 2014
February 6 2015
March 7 2016
April 8 2017

Building native apps is hard

- Different stacks of technologies
- No knowledge and code sharing
- Slow iteration speed
- Hard to scale

Web got this right

Web

- ~~Different stacks of technologies~~
- ~~No knowledge and code sharing~~
- ~~Slow iteration speed~~
- ~~Hard to scale~~

HTML / CSS / JS

Same code and tech

F5 / ⌘R

React!

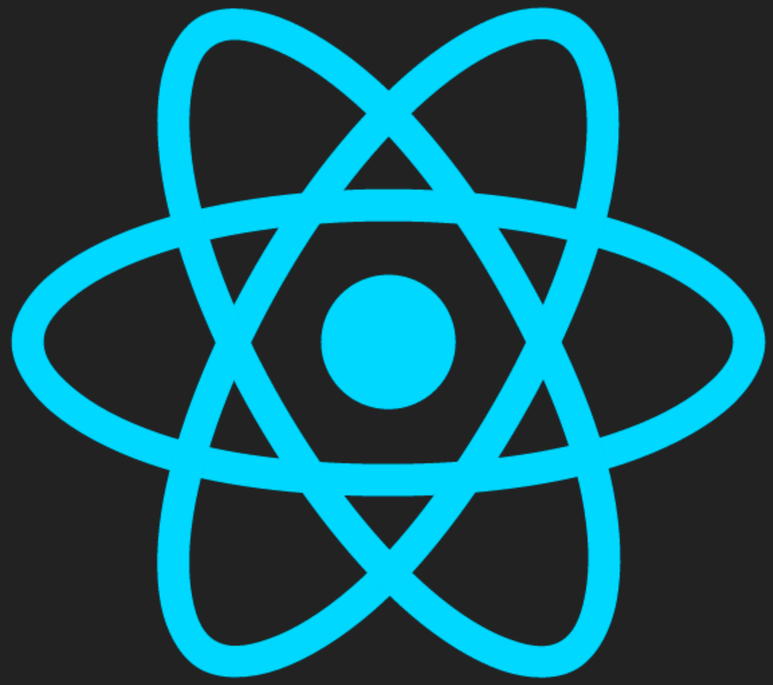
Web apps on the phone are not great

- Very hard to provide smooth experiences
- Not designed for complex interactions
- Impossible to embed native components

Development experience

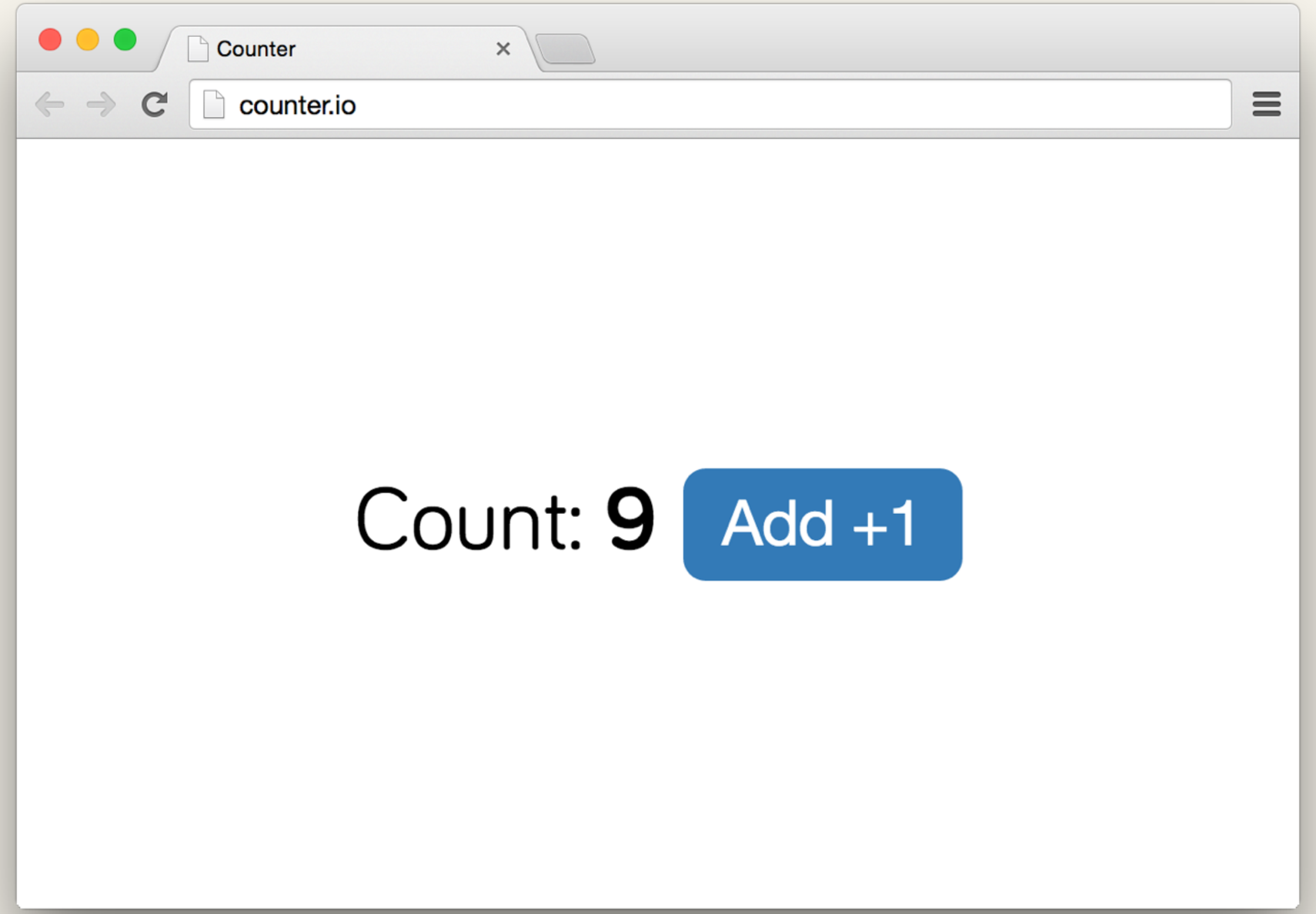
Awesome apps

React Native



$$UI = f^*(data)$$

* No side effects


Counter
counter.io
Count: 9
Add +1

$$UI = f(count)$$

```
UI = f(count) =
  div(
    span('Count ' + count),
    button('Add +1')
  )
```

```
render() {
  return (
    div(
      span(
        'Count: ' + b(this.state.count)
      ),
      button(
        'Add +1'
      )
    )
  )
}
```

```
render() {
  return (
    <div>
      <span>
        Count: <b>{this.state.count}</b>
      </span>
      <button>
        Add +1
      </button>
    </div>
  )
}
```

VirtualDOM

~~HTML~~

```
render() {
  return (
    <div>
      <span>
        Count: <b>{this.state.count}</b>
      </span>
      <button onClick={() => ??? }>
        Add +1
      </button>
    </div>
  )
}
```

Android

```
TextView text = (TextView)findViewByID(R.layout.label);
text.setText('10');
```

Objective-C

```
_label.text = @"10";
```

JavaScript

```
document.getElementByID('count').children[1].innerHTML = '10';
$('#counter b').html('10');
```

too complex

```
render() {
  var count = this.state.count;
  return (
    <div>
      <span>
        Count: <b>{count}</b>
      </span>
      <button onClick={() => ??? }>
        Add +1
      </button>
    </div>
  )
}
```

```
render() {
  var count = this.state.count;
  return (
    <div>
      <span>
        Count: <b>{count}</b>
      </span>
      <button onClick={() => this.setState({count: count + 1})}>
        Add +1
      </button>
    </div>
  )
}
```

setState

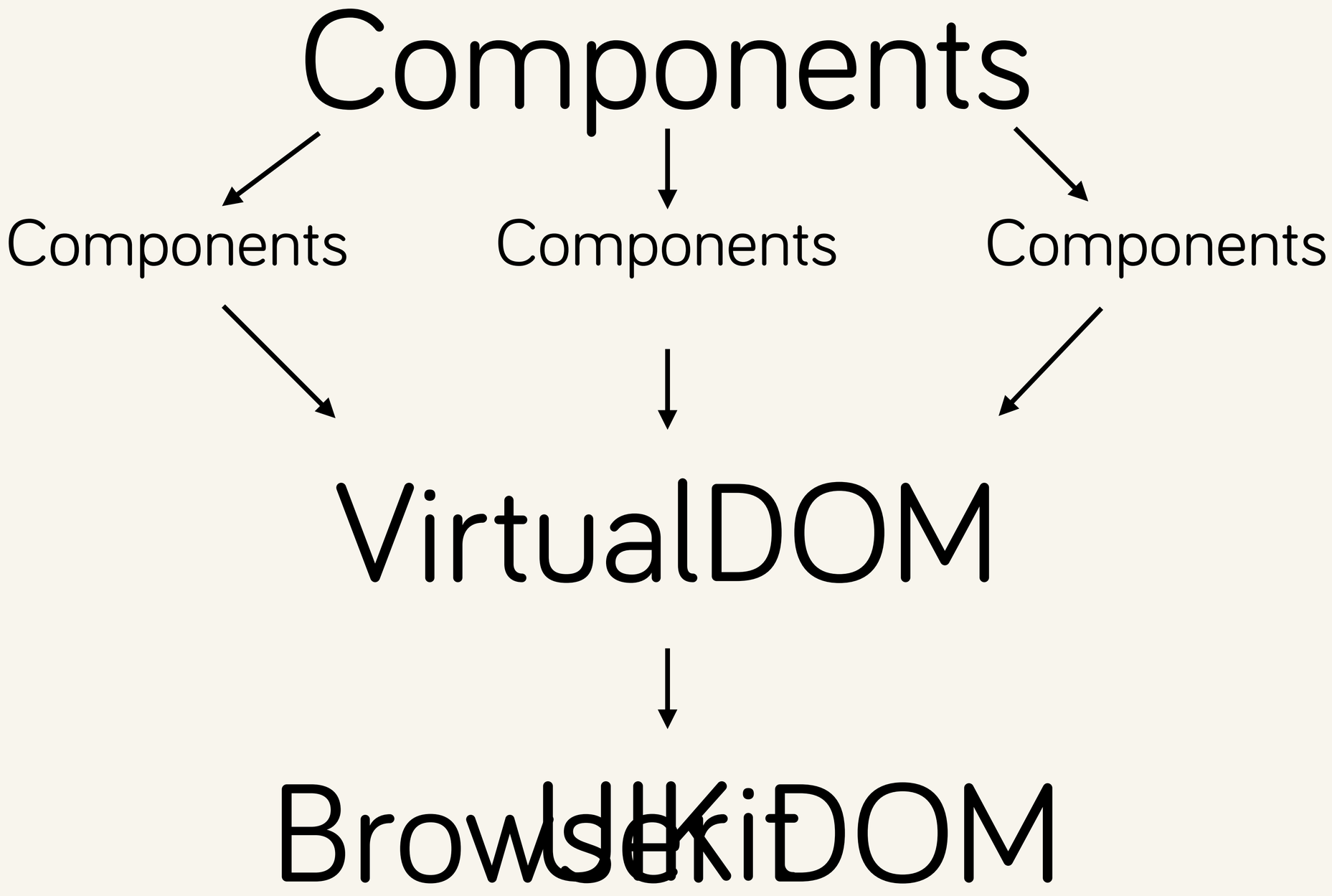
```
state = {count: 9}
```

```
<div>
   <span>
     Count: <b>9</b>
   </span>
   <button>
     Add +1
   </button>
 </div>
```

```
state = {count: 10}
```

```
<div>
   <span>
     Count: <b>10</b>
   </span>
   <button>
     Add +1
   </button>
 </div>
```

```
findDOMNode(b).innerHTML = '10';
```


Components
Components
Components
Components
VirtualDOM
BrowserKit
UIKit
DOM

1. Runtime
2. Base components
3. Calling native functions

ECMAScript 5

JavaScript Core

- Part of WebKit project
- Open Source
- Ships with iOS

Runtime

`<div>`	`<View>`
`<span>`	`<Text>`
`<img>`	`<Image>`
	`<ScrollView>`
	`<MapView>`
	`<TabBar>`
	`<DatePicker>`
	`...`

Base components

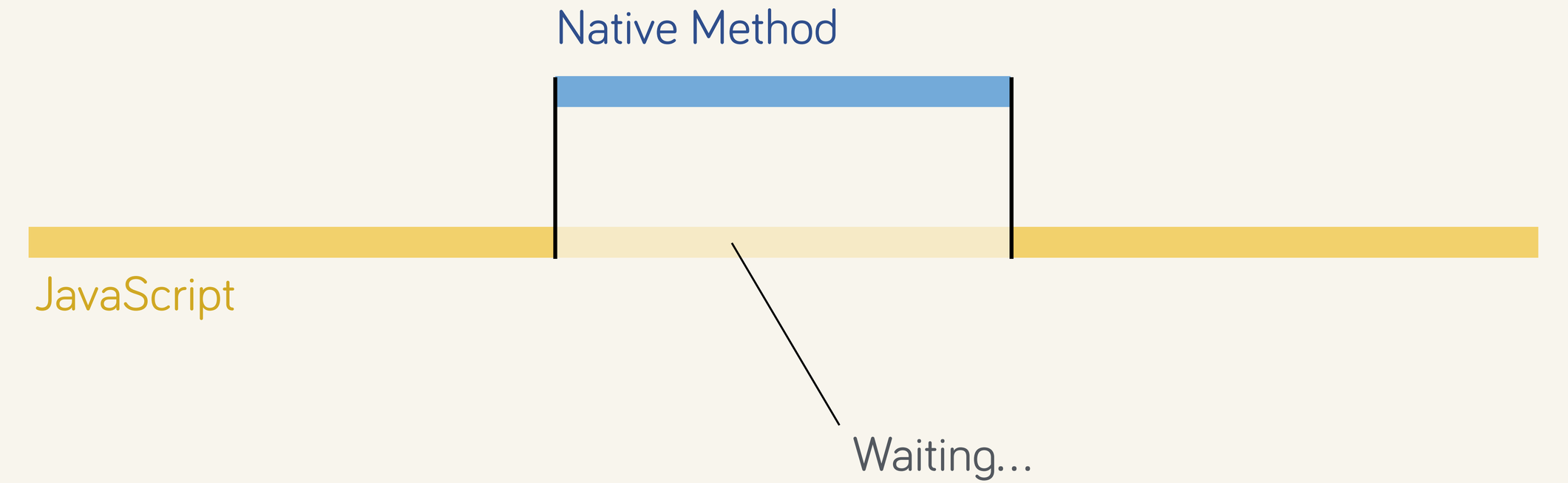
create(view, parent, attributes)*
update(view, attributes)
delete(view)

* actually React is more complex that that

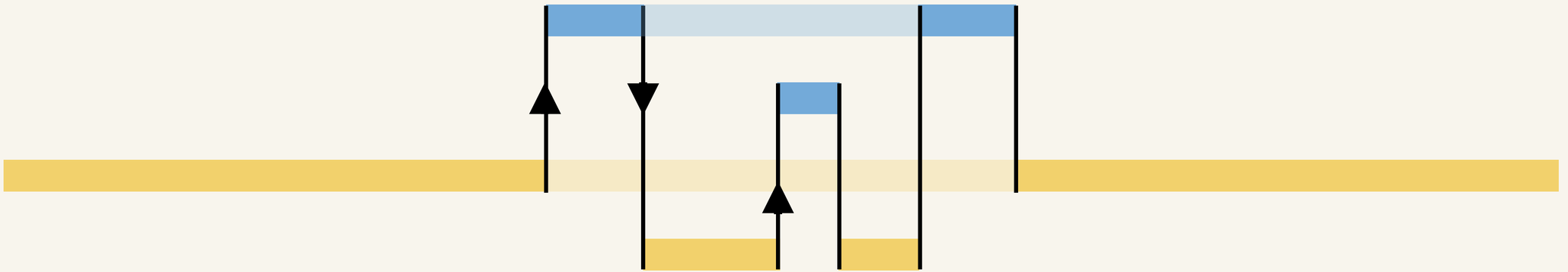
Just call native functions?

nope

Synchronous


Native Method
JavaScript
Waiting...
time

JavaScript → Native → JavaScript → Native



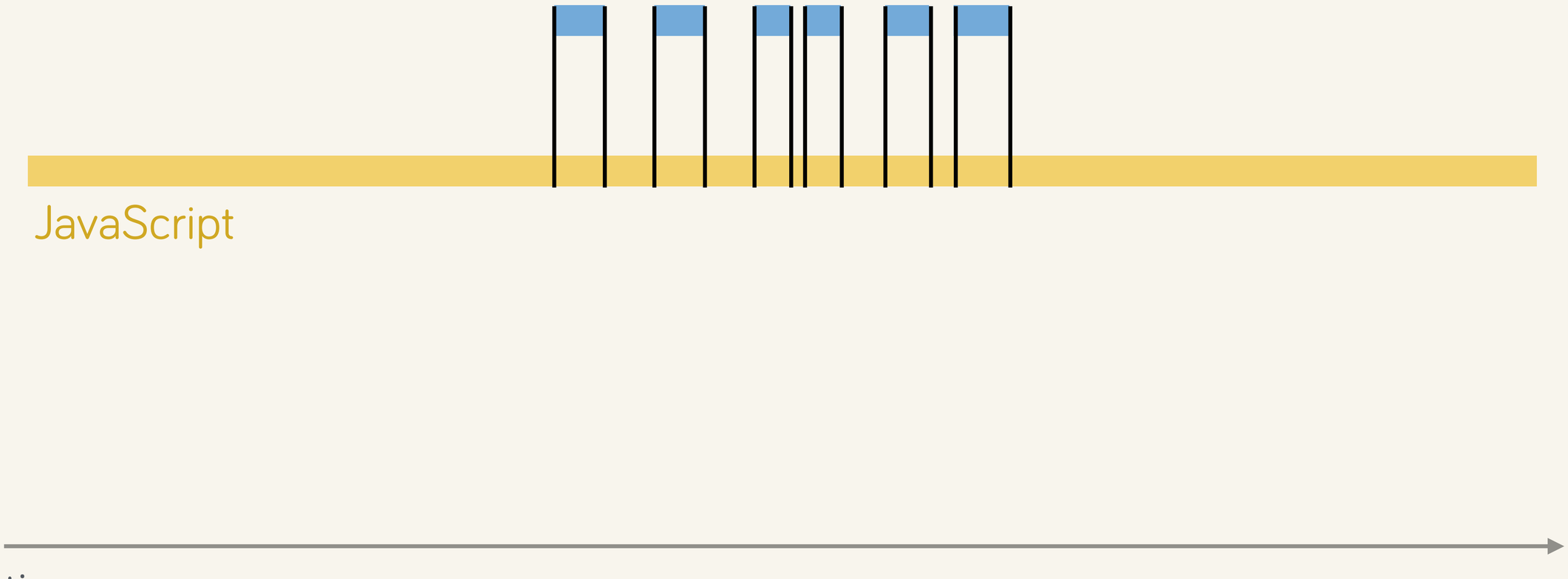
Asynchronous

~~Synchronous~~

Overhead of

every

native call


JavaScript
time

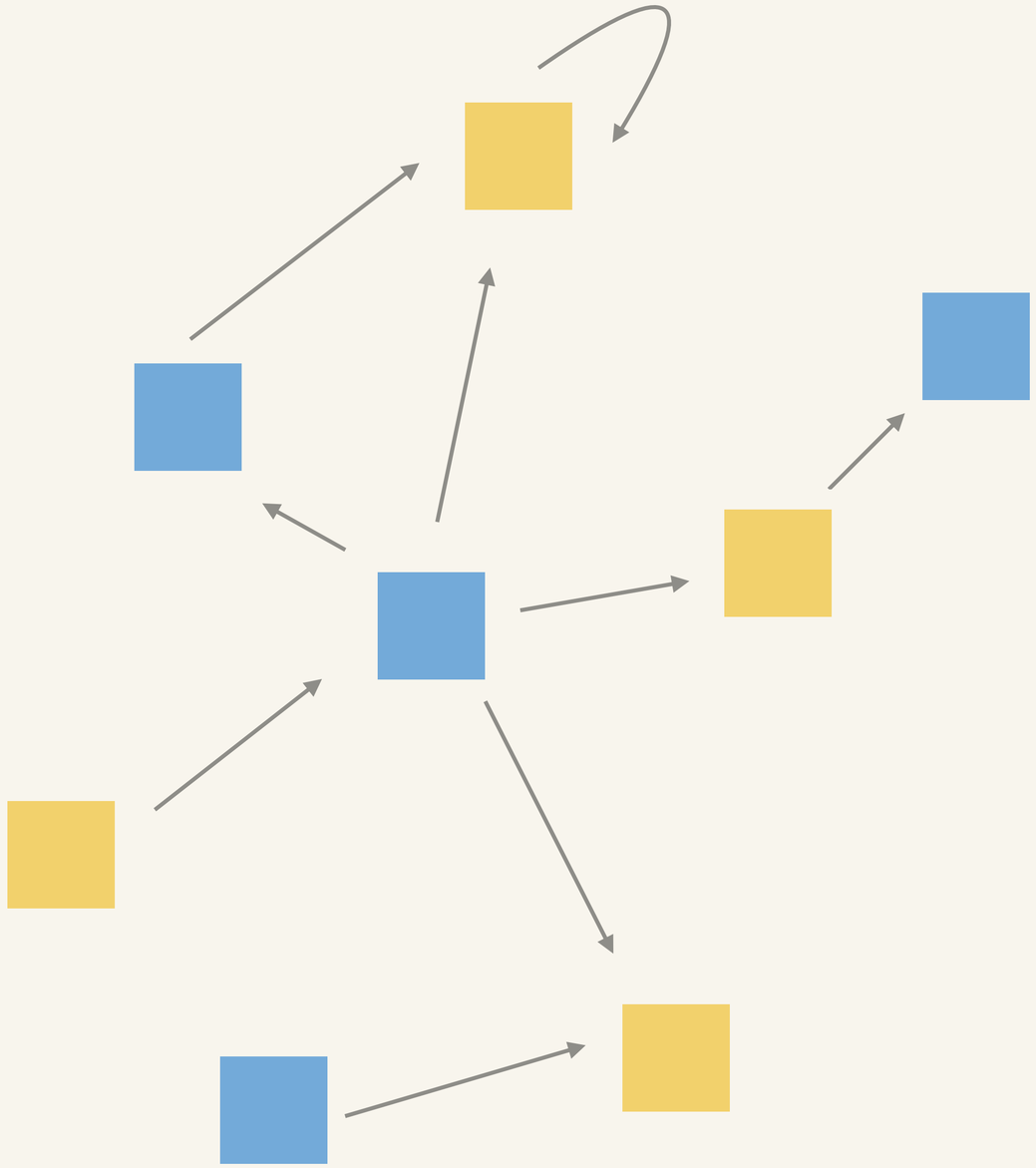

JavaScript


time

Batch
native calls

Shared
mutable
data

JavaScript Objects



Native Objects

Exchange
serializable
messages

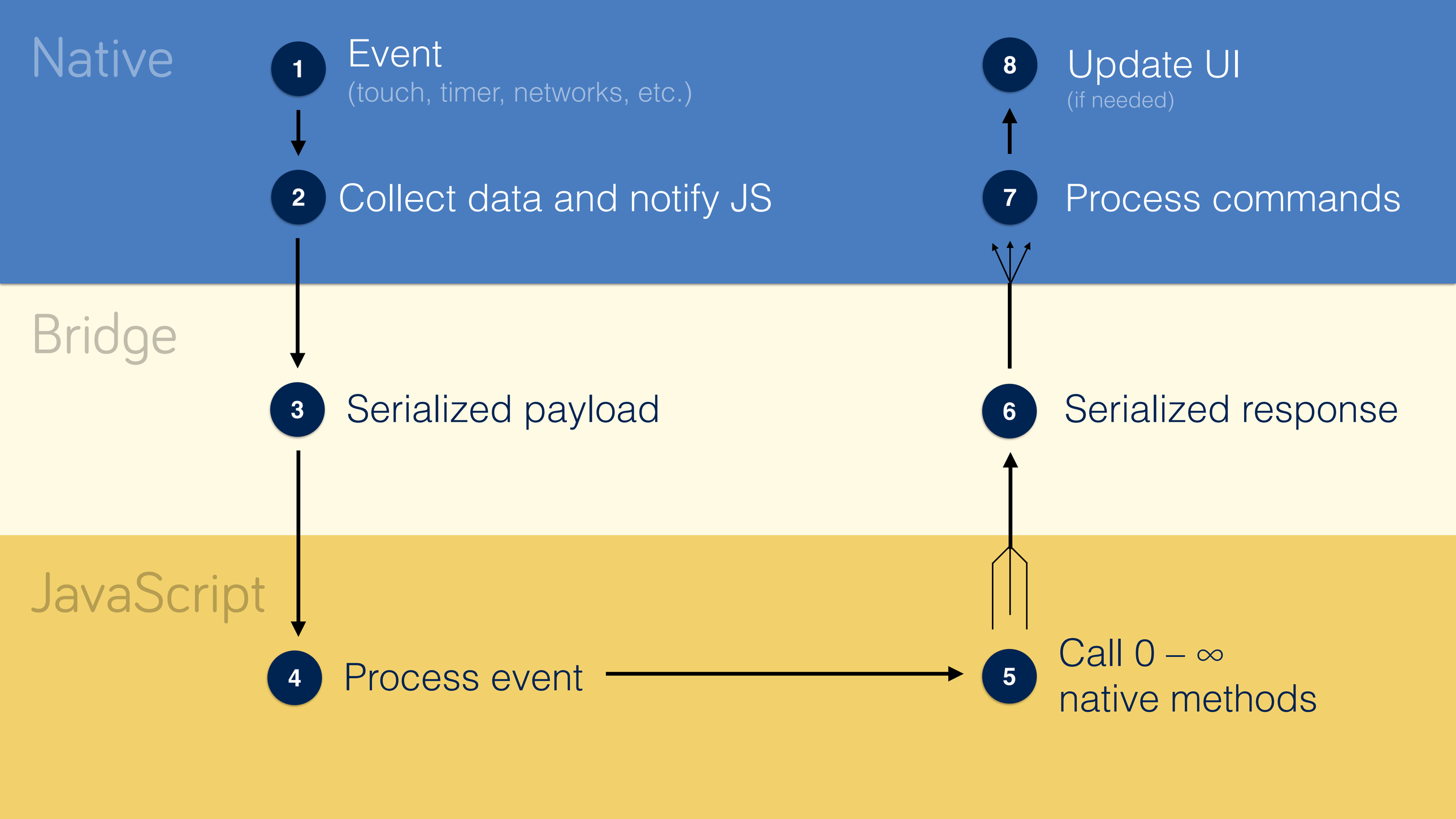
Asynchronous
Batched
Serializable

The Bridge

Native

Bridge

JavaScript

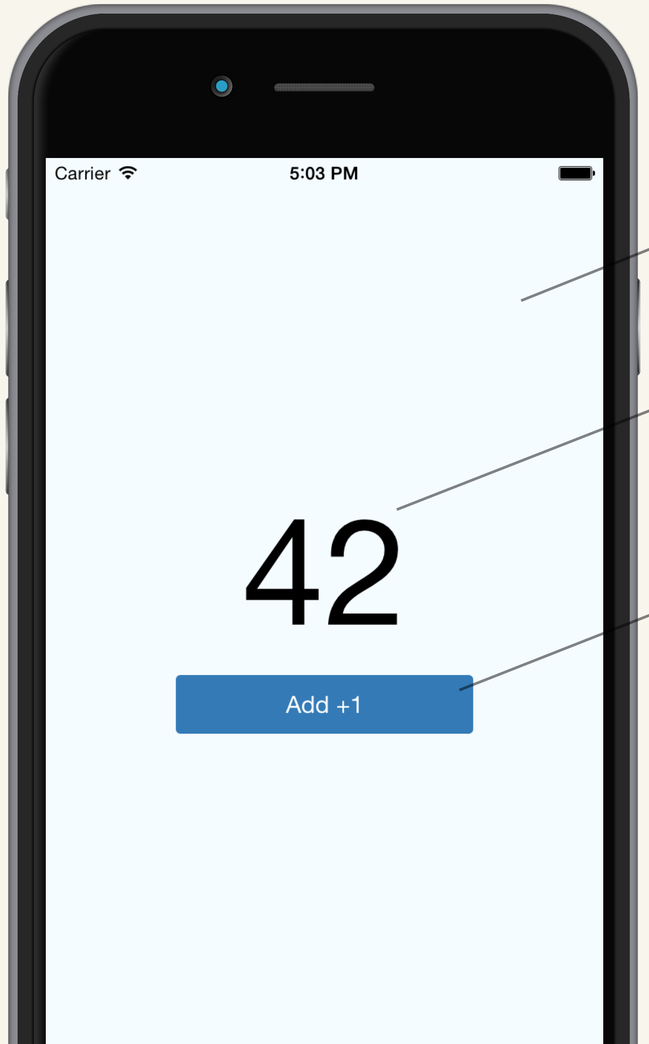

Native
1
Event
(touch, timer, networks, etc.)
2
Collect data and notify JS
8
Update UI
(if needed)
7
Process commands
Bridge
3
Serialized payload
6
Serialized response
JavaScript
4
Process event
5
Call 0 – ∞
native methods

JS is event-driven

Events →  → Commands

Example

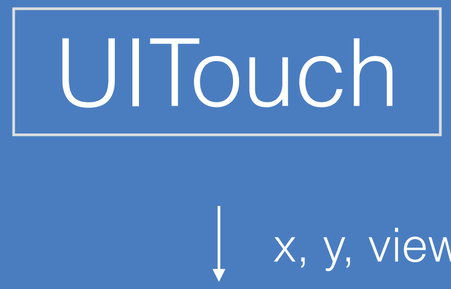
- Updates counter
- Sends data to web service




```
render() {
  return (
    <View style={styles.container}>
      <Text style={styles.value}>
        {this.state.count}
      </Text>
      <Button
        title="Add +1"
        onPress={() => this.inc()}
      />
    </View>
  );
}
```


```
inc() {
  var newCount = this.state.count + 1;
  this.setState({count: newCount});

  fetch(
    'https://api.conunter.io/',
    {
      method: 'post',
      body: 'value=' + newCount
    }
  );
}
```




```
[_bridge enqueueJSCall:@"EventEmitter.receiveTouches"
                  args:@[@"end",
                         @{@"x": @42, @"y": @106}]];
```

Native

Native

```
[_bridge enqueueJSCall:@"RCTEventEmitter.receiveTouches"
                  args:@[@"end",
                         @{@"x": @42, @"y": @106}]];
```

Bridge

```
[
  'EventEmitter', 'receiveTouches',
  ['end', {'x': 42, 'y': 106}]
]
```

JavaScript

```
call('EventEmitter', 'receiveTouches', [{x: 42, y: ...}])
```

```
function call(moduleName, methodName, args) {
  MessageQueue.init();

  var module = require(moduleName);          // EventEmitter
  module[methodName].apply(module, args);  // receiveTouches

  return MessageQueue.flush();
}
```

Message queue

Touch processing

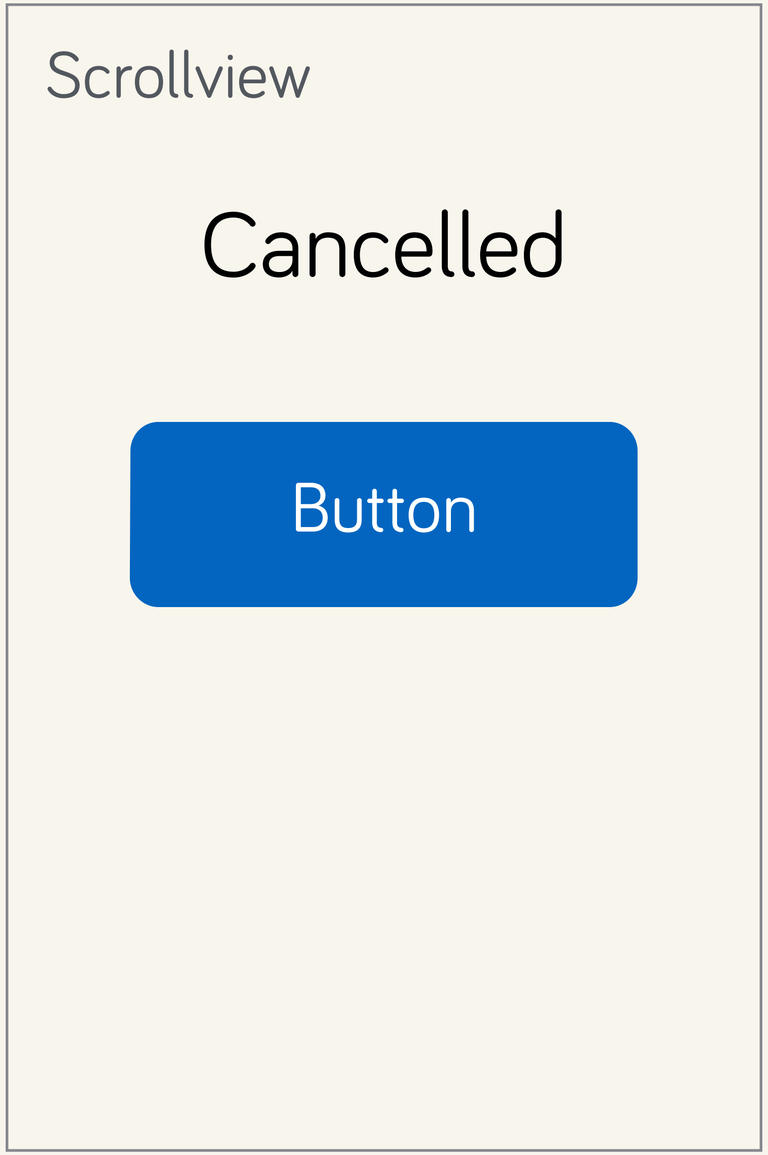
What **element**
should respond
to a given **event**?

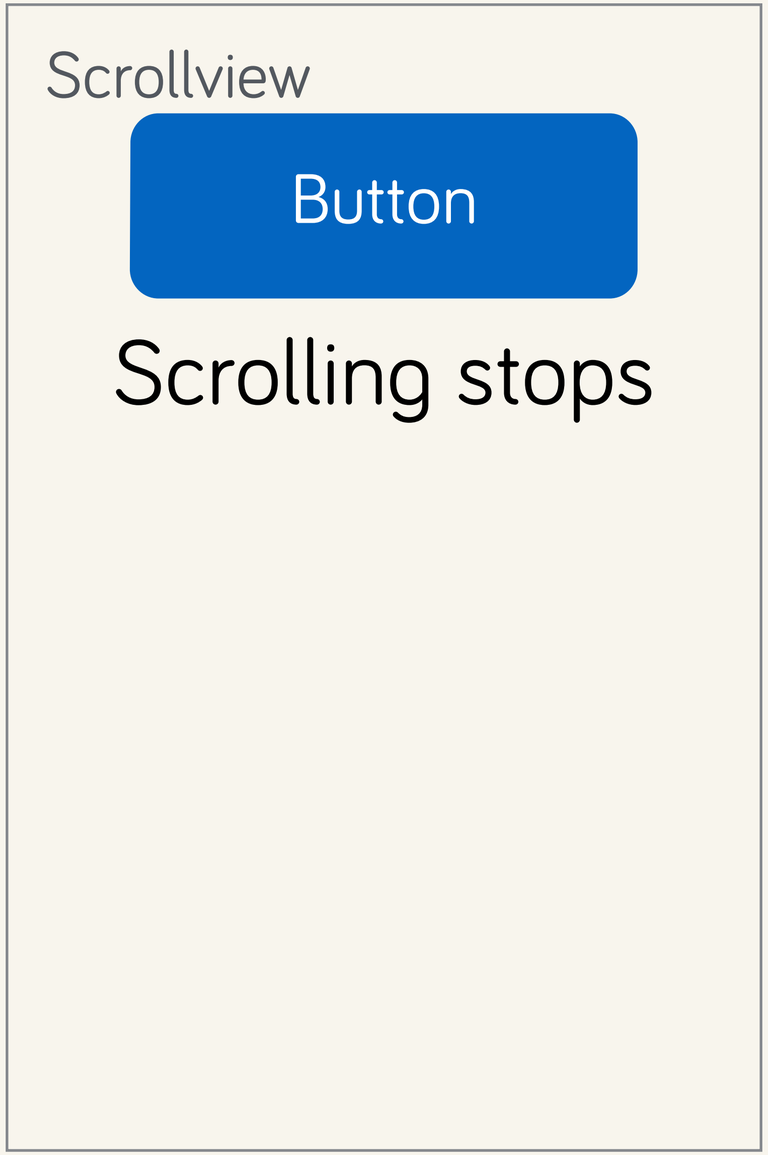
Button pressed!

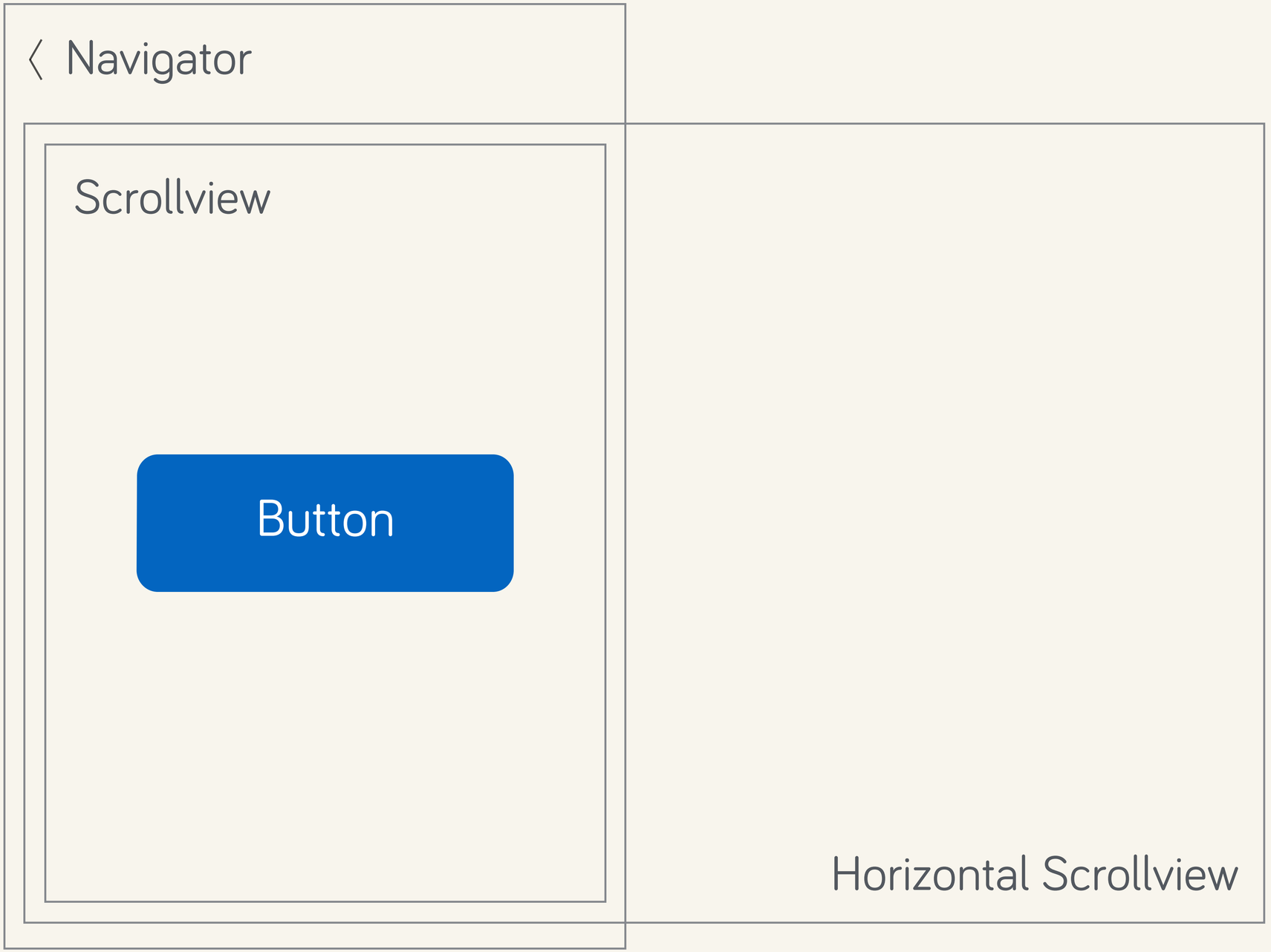
Button

Cancelled

Button


Scrollview
Cancelled
Button


Scrollview
Button
Scrolling stops


〈 Navigator
Scrollview
Button
Horizontal Scrollview

Responder System

onStartShouldSetResponder

onResponderTerminationRequest

onResponderGrant

onResponderMove

onResponderRelease

onResponderTerminate

...

<TouchableOpacity>

<TouchableHighlight>

<TouchableBounce>

<TouchableWithoutFeedback>

onPress

```
inc() {
  var newCount = this.state.count + 1;
  this.setState({count: newCount});

  fetch(
    'https://api.conunter.io/',
    {
      method: 'post',
      body: 'value=' + newCount
    }
  );
}
```

```
<View style={...}>
   <Text style={...}>
     42
   </Text>
   <Button
     title="Add +1"
     onPress={...}
   />
 </View>
```

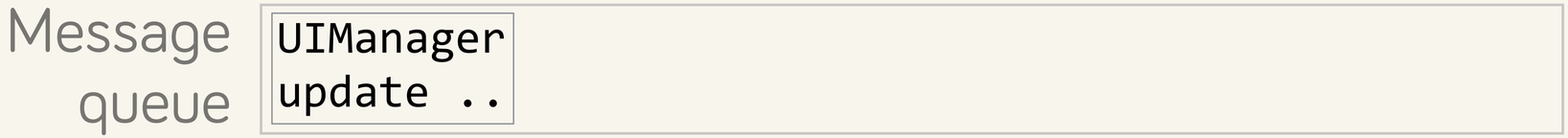
```
<View style={...}>
   <Text style={...}>
     43
   </Text>
   <Button
     title="Add +1"
     onPress={...}
   />
 </View>
```

Somewhere in React's internals:

```
var UIManager = require('NativeModules').UIManager;
UIManager.update(18, {text: '43'});
```

NativeModules

```
NativeModules.UIManager = {
  ...
  update: function(viewID, attributes) {
    MessageQueue.push(
      ['UIManager', 'update', [viewID, attributes]]
    );
  }
  ...
};
```



```
inc() {
  var newCount = this.state.count + 1;
  this.setState({count: newCount});

  fetch(
    'https://api.conunter.io/',
    {
      method: 'post',
      body: 'value=' + newCount
    }
  );
}
```

Message queue

UIManager	DataManager
update ..	query ....

```
function call(moduleName, methodName, args) {
  MessageQueue.init();

  var module = require(moduleName);
  module[methodName].apply(module, args);

  return MessageQueue.flush();
}
```

Message queue

UIManager	DataManager
update ..	query ....

Native

```
[UIManager updateView:18 props:@{@"text": @"43"}]
[DataManager query:@"post" url:@"http://..."]
```

Bridge

JavaScript

Message queue

UIManager	DataManager
update ..	query ....

```
[UIManager updateView:18 props:@{@"text": @"43"}]

addUIBlock:^() {
  UILabel *label = viewRegistry[18];
  label.text = @"43";
  [label markAsDirty];
}
```

Native

Layout

github.com/facebook/css-layout

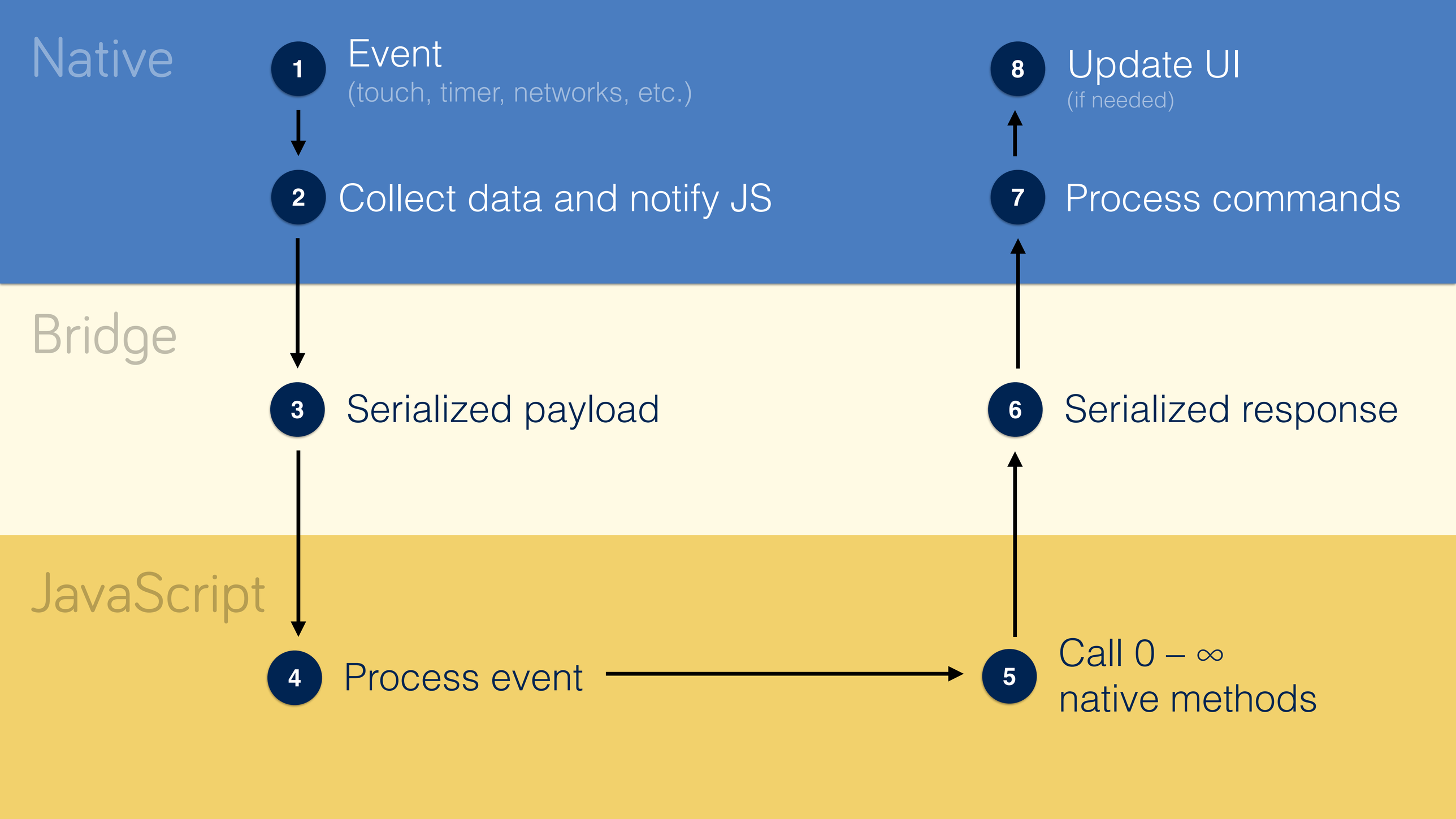
Flexbox

```
{
  margin: 20,
  borderBottomWidth: 2,
  flex: 1,
  alignContent: 'center',
}
```

Coordinates

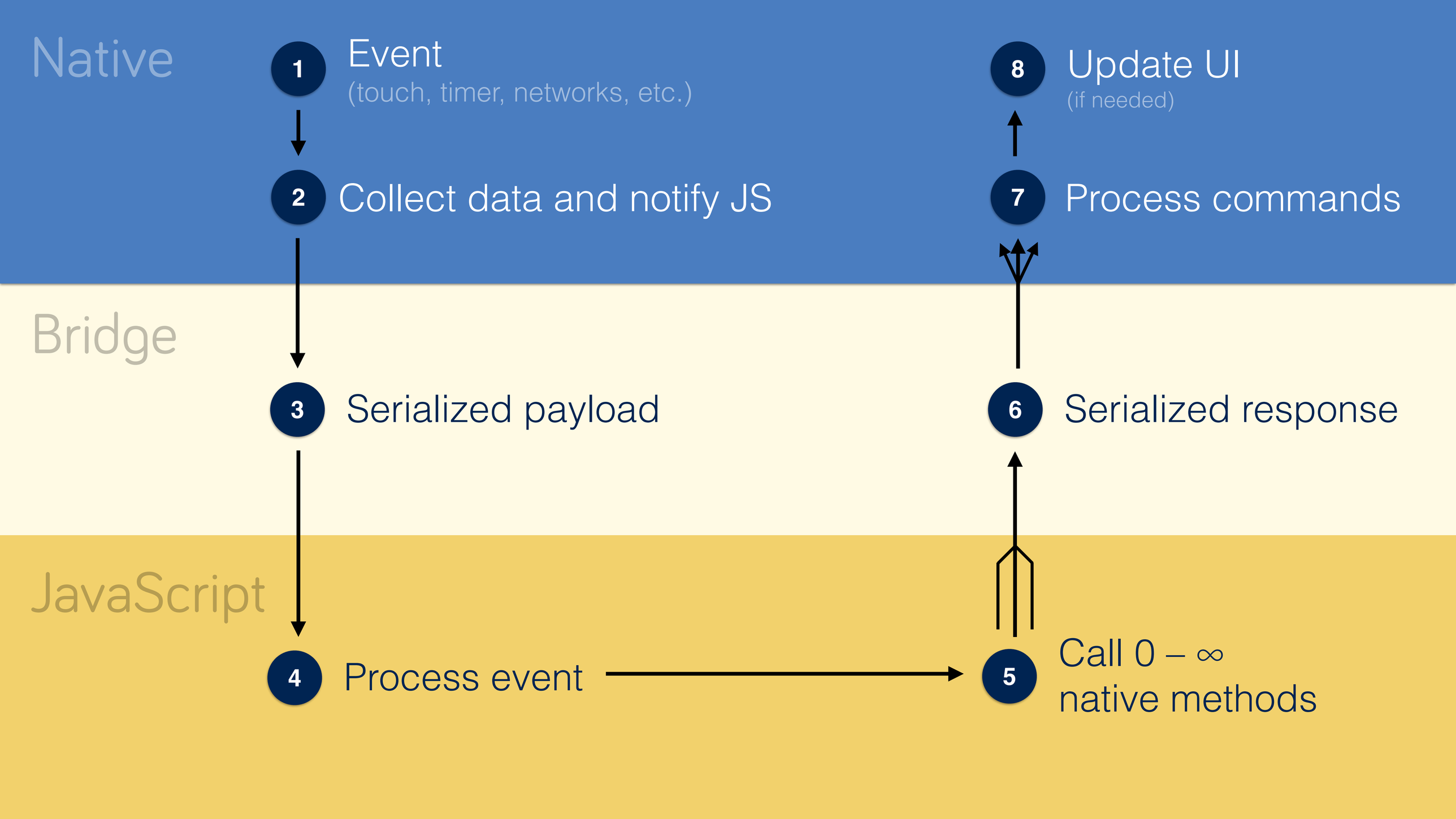
```
{
  left: 120,
  top: 220,
  width: 60,
  height: 60,
}
```

- User taps the button
- Counter is updated


Native
1
Event
(touch, timer, networks, etc.)
2
Collect data and notify JS
8
Update UI
(if needed)
7
Process commands
Bridge
3
Serialized payload
6
Serialized response
JavaScript
4
Process event
5
Call 0 – ∞
native methods

Benefits

Fast


Native
1
Event
(touch, timer, networks, etc.)
2
Collect data and notify JS
8
Update UI
(if needed)
7
Process commands
Bridge
3
Serialized payload
6
Serialized response
JavaScript
4
Process event
5
Call 0 – ∞
native methods

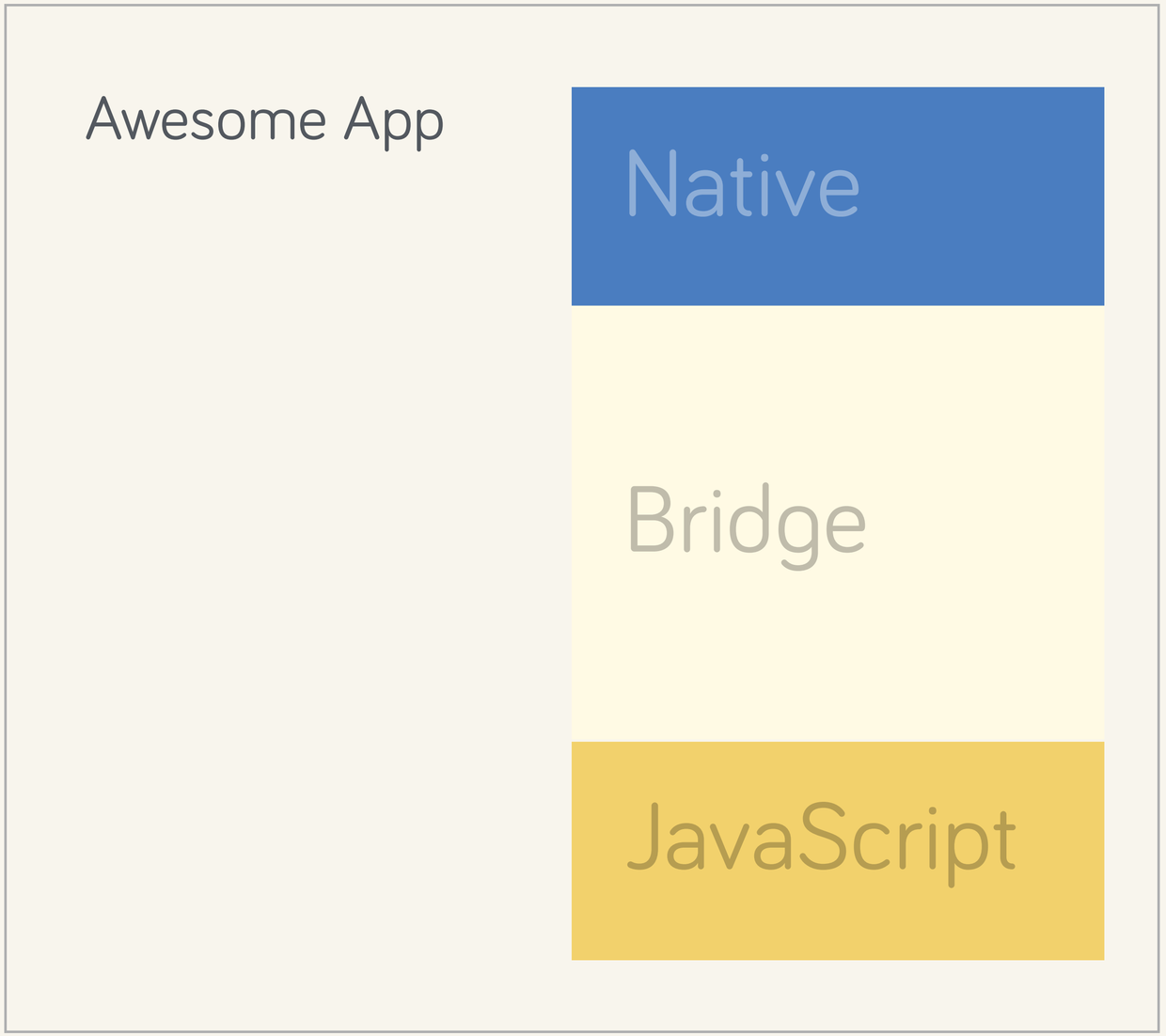
Record / Replay

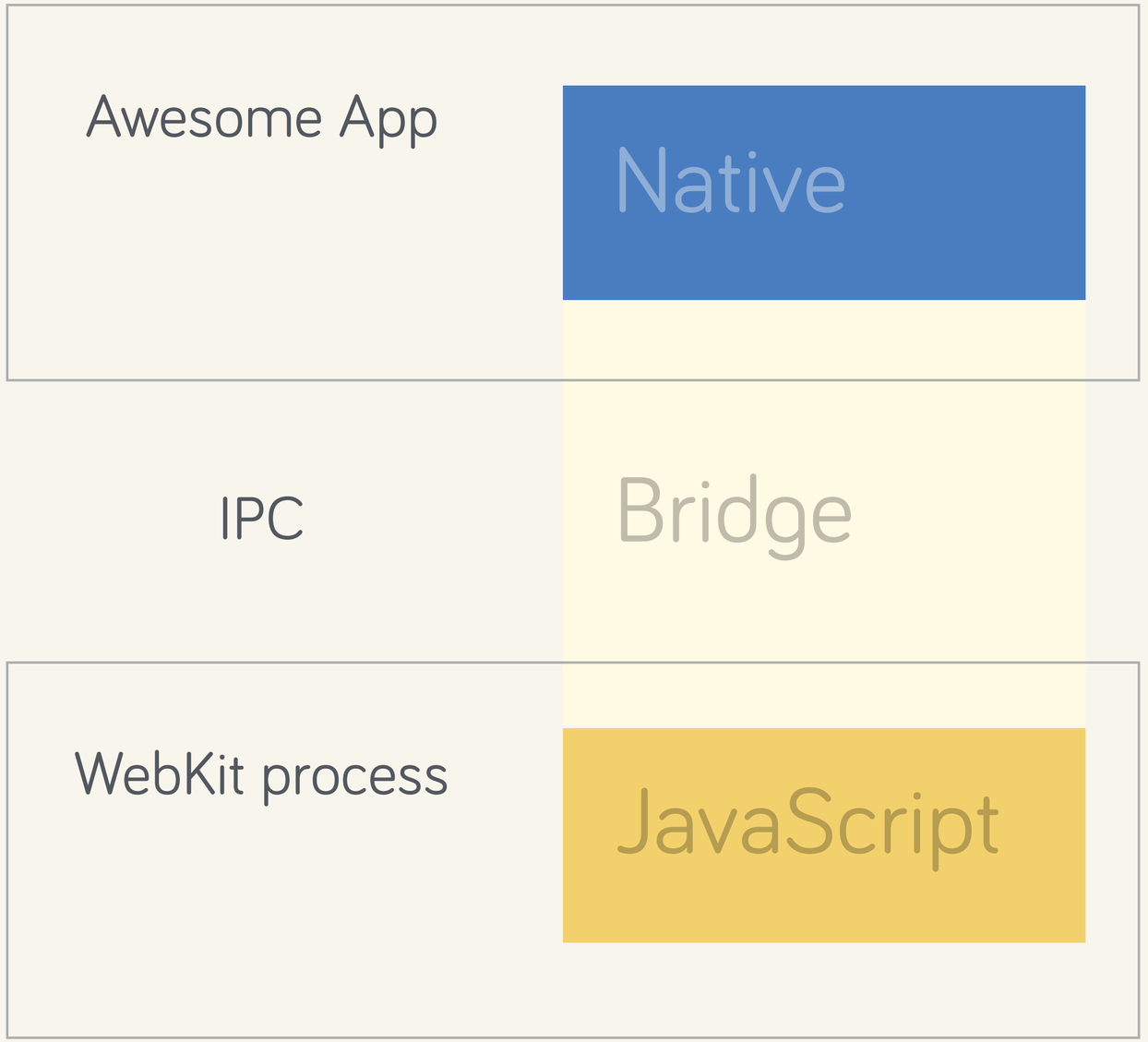
Flexible JS runtime

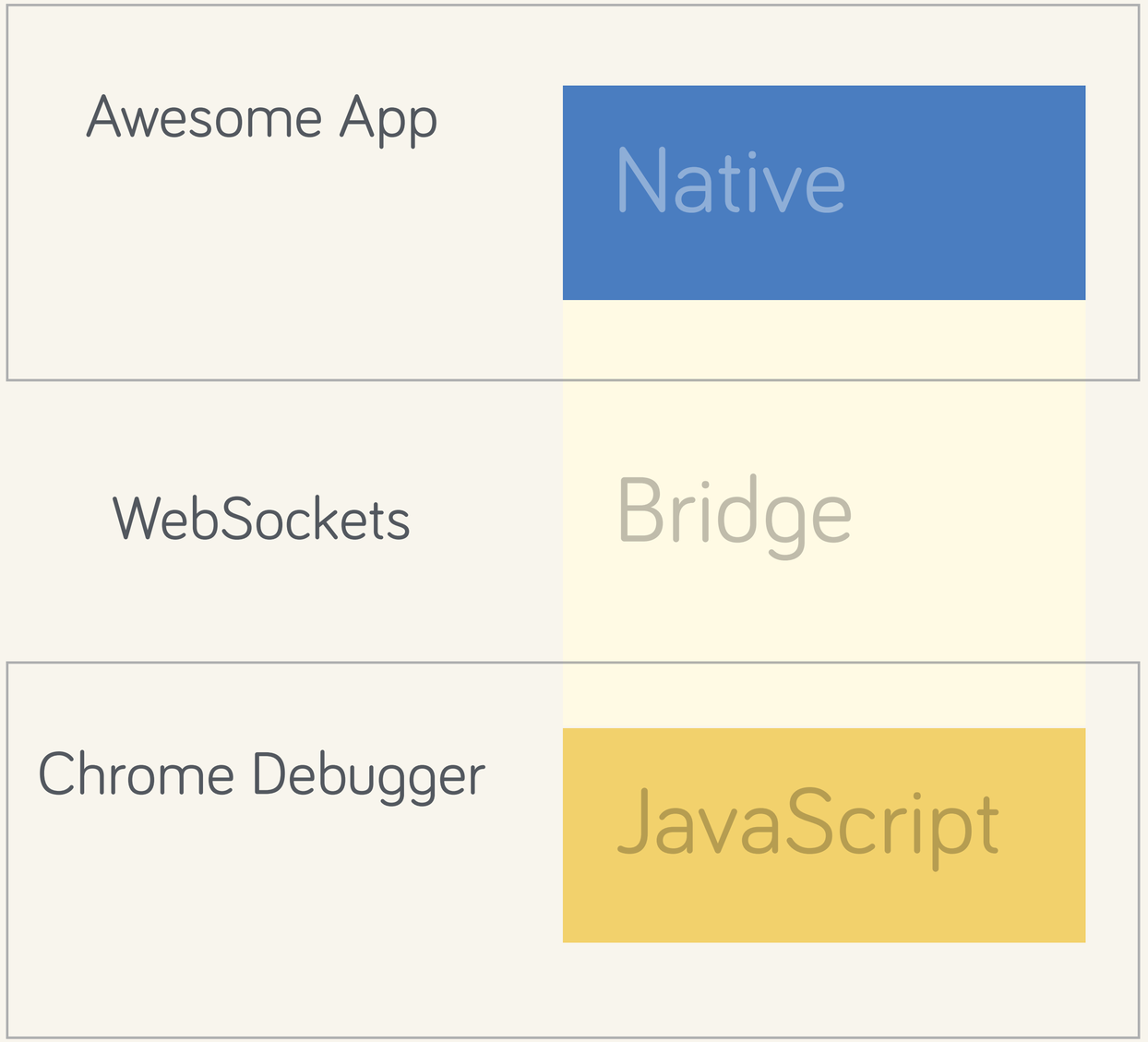
Native

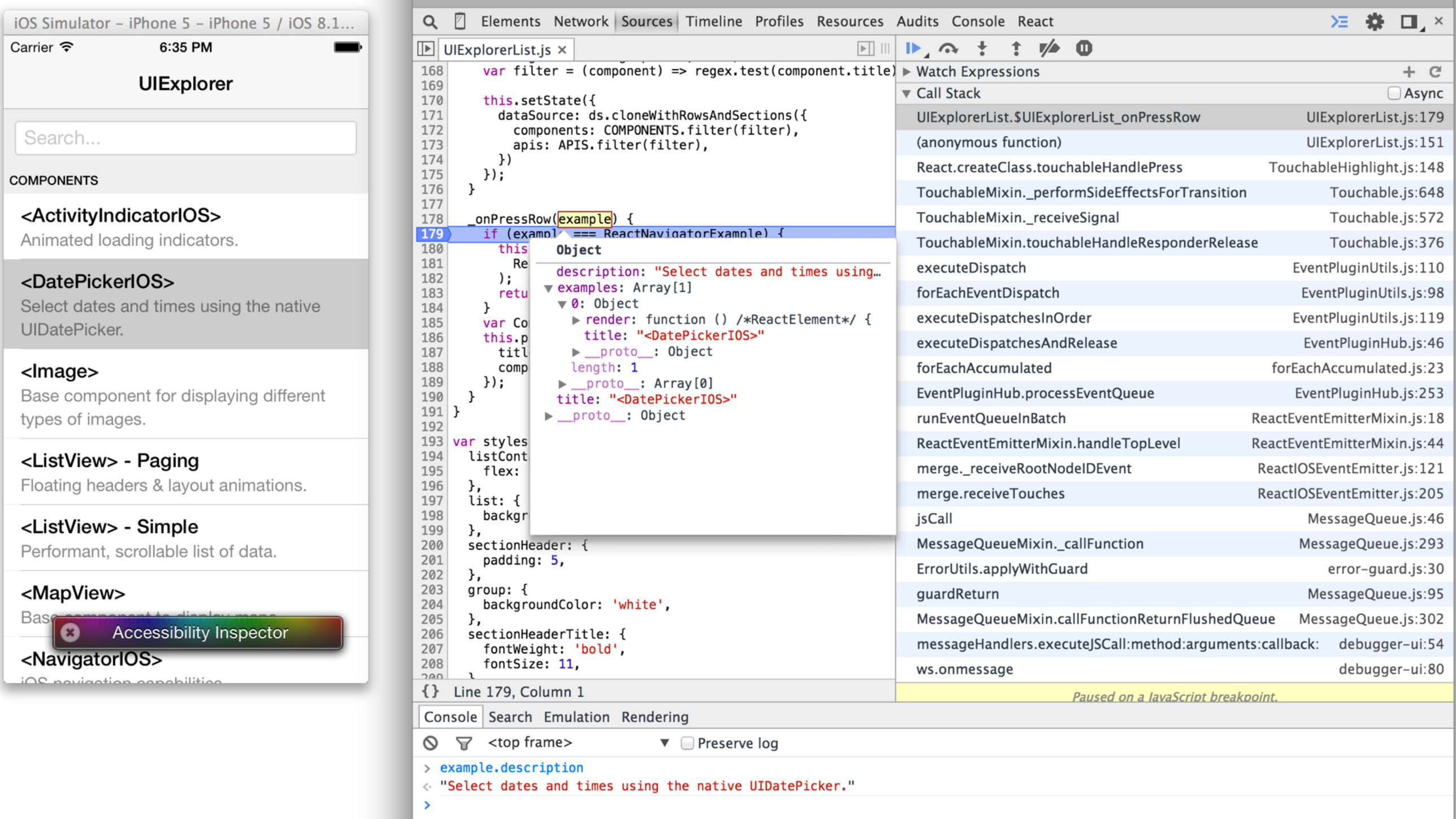
Bridge

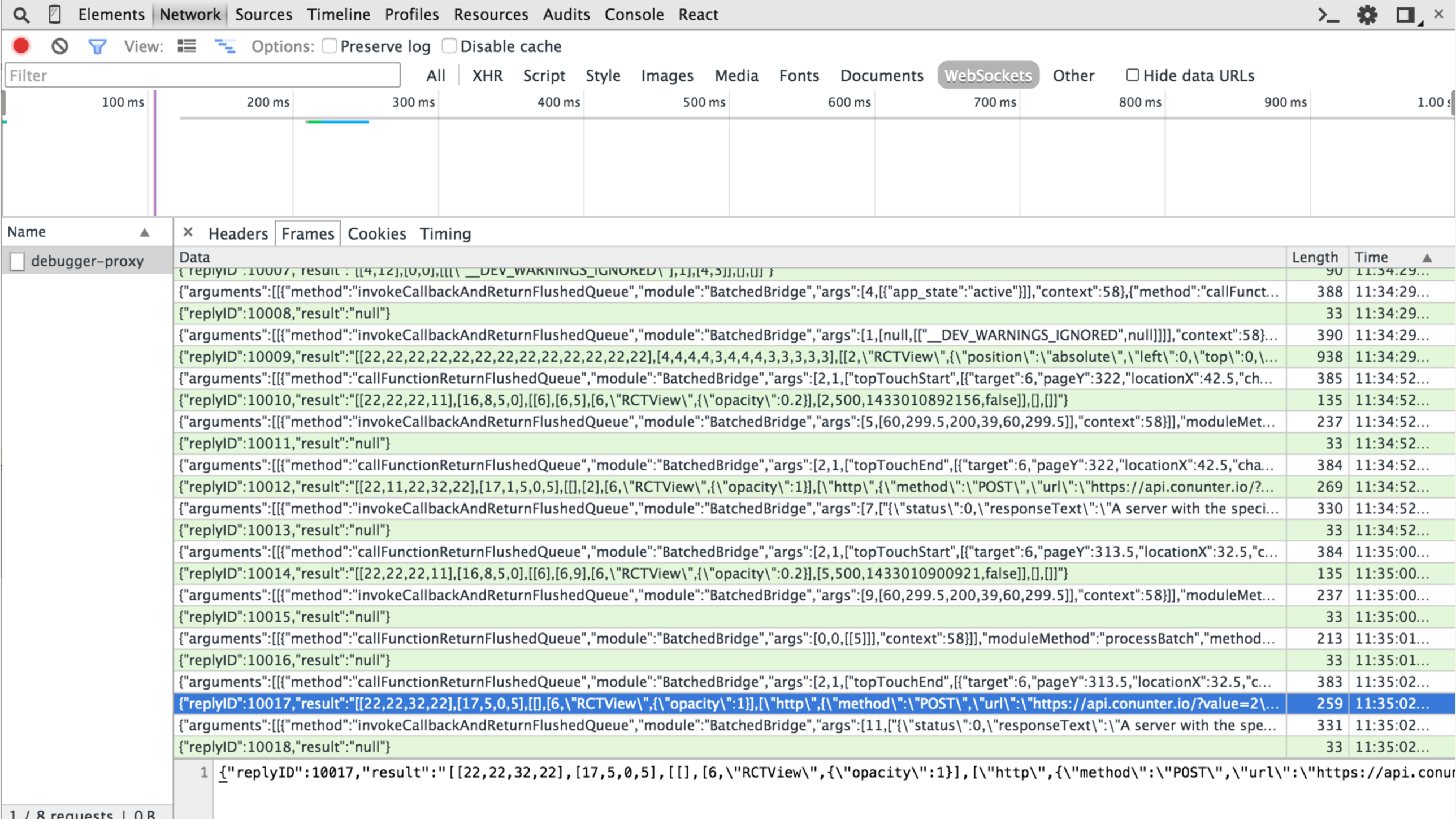
JavaScript

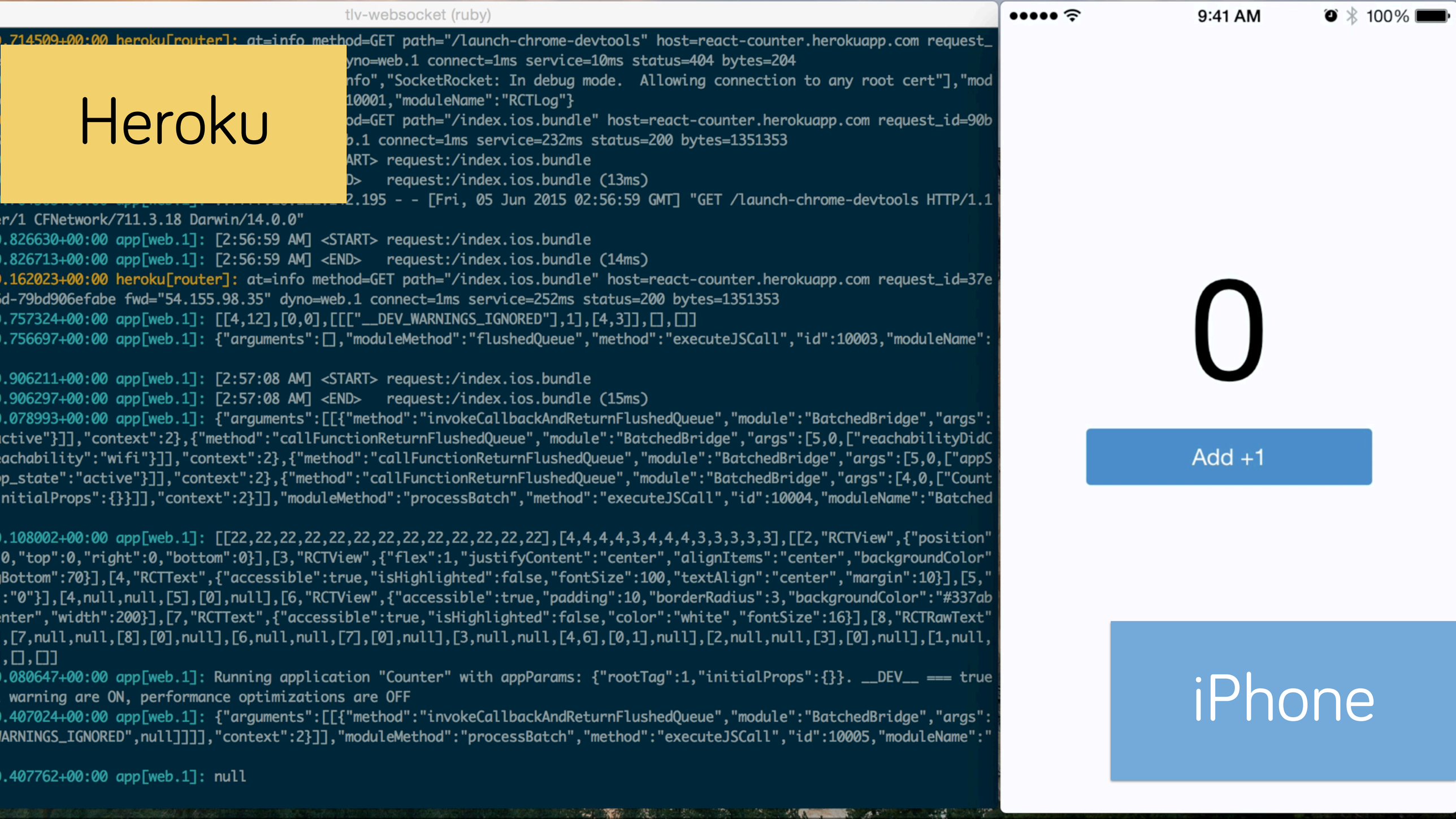

Awesome App
Native
Bridge
JavaScript


Awesome App
Native
IPC
Bridge
WebKit process
JavaScript

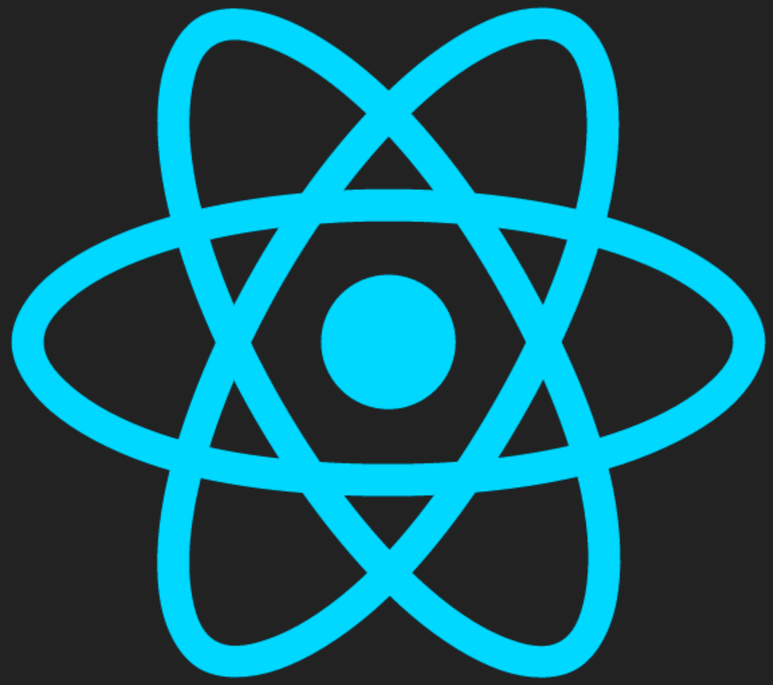

Awesome App
Native
WebSockets
Bridge
Chrome Debugger
JavaScript


iOS Simulator - iPhone 5 - iPhone 5 / iOS 8.1...
Carrier
6:35 PM
UIExplorer
Search...
COMPONENTS
<ActivityIndicatorIOS>
Animated loading indicators.
<DatePickerIOS>
Select dates and times using the native UIDatePicker.
<Image>
Base component for displaying different types of images.
<ListView> - Paging
Floating headers & layout animations.
<ListView> - Simple
Performant, scrollable list of data.
<MapView>
Accessibility Inspector
<NavigatorIOS>
Elements
Network
Sources
Timeline
Profiles
Resources
Audits
Console
React
UIExplorerList.js
var filter = (component) => regex.test(component.title)
this.setState({
dataSource: ds.cloneWithRowsAndSections({
components: COMPONENTS.filter(filter),
apis: APIS.filter(filter),
})
});
}
_onPressRow(example) {
if (examp === ReactNavigatorExample) {
Object
description: "Select dates and times using…
examples: Array[1]
0: Object
render: function () /*ReactElement*/ {
title: "<DatePickerIOS>"
__proto__: Object
length: 1
__proto__: Array[0]
title: "<DatePickerIOS>"
__proto__: Object
var styles
listCont
flex:
},
list: {
backgr
},
sectionHeader: {
padding: 5,
},
group: {
backgroundColor: 'white',
},
sectionHeaderTitle: {
fontWeight: 'bold',
fontSize: 11,
Line 179, Column 1
Watch Expressions
Call Stack
Async
UIExplorerList.$UIExplorerList_onPressRow UIExplorerList.js:179
(anonymous function) UIExplorerList.js:151
React.createClass.touchableHandlePress TouchableHighlight.js:148
TouchableMixin._performSideEffectsForTransition Touchable.js:648
TouchableMixin._receiveSignal Touchable.js:572
TouchableMixin.touchableHandleResponderRelease Touchable.js:376
executeDispatch EventPluginUtils.js:110
forEachEventDispatch EventPluginUtils.js:98
executeDispatchesInOrder EventPluginUtils.js:119
executeDispatchesAndRelease EventPluginHub.js:46
forEachAccumulated forEachAccumulated.js:23
EventPluginHub.processEventQueue EventPluginHub.js:253
runEventQueueInBatch ReactEventEmitterMixin.js:18
ReactEventEmitterMixin.handleTopLevel ReactEventEmitterMixin.js:44
merge._receiveRootNodeIDEvent ReactIOSEventEmitter.js:121
merge.receiveTouches ReactIOSEventEmitter.js:205
jsCall MessageQueue.js:46
MessageQueueMixin._callFunction MessageQueue.js:293
ErrorUtils.applyWithGuard error-guard.js:30
guardReturn MessageQueue.js:95
MessageQueueMixin.callFunctionReturnFlushedQueue MessageQueue.js:302
messageHandlers.executeJSCall:method:arguments:callback: debugger-ui:54
ws.onmessage debugger-ui:80
Paused on a JavaScript breakpoint.
Console
Search
Emulation
Rendering
<top frame>
Preserve log
example.description
"Select dates and times using the native UIDatePicker."


Elements Network Sources Timeline Profiles Resources Audits Console React
View: Options: Preserve log Disable cache
Filter All XHR Script Style Images Media Fonts Documents WebSockets Other Hide data URLs
100 ms 200 ms 300 ms 400 ms 500 ms 600 ms 700 ms 800 ms 900 ms 1.00 s
Name
debugger-proxy
Headers Frames Cookies Timing
Data Length Time
{"arguments":[[{"method":"invokeCallbackAndReturnFlushedQueue","module":"BatchedBridge","args":[4,[{"app_state":"active"}]],"context":58},{"method":"callFunct... 388 11:34:29...
{"replyID":10008,"result":"null"} 33 11:34:29...
{"arguments":[[{"method":"invokeCallbackAndReturnFlushedQueue","module":"BatchedBridge","args":[1,[null,[["__DEV_WARNINGS_IGNORED",null]]]],"context":58}... 390 11:34:29...
{"replyID":10009,"result":"[[22,22,22,22,22,22,22,22,22,22,22,22,22,22],[4,4,4,4,3,4,4,4,3,3,3,3,3],[[2,\"RCTView\",{\"position\":\"absolute\",\"left\":0,\"top\":0,\... 938 11:34:29...
{"arguments":[[{"method":"callFunctionReturnFlushedQueue","module":"BatchedBridge","args":[2,1,["topTouchStart",[{"target":6,"pageY":322,"locationX":42.5,"ch... 385 11:34:52...
{"replyID":10010,"result":"[[22,22,22,11],[16,8,5,0],[[6],[6,5],[6,\"RCTView\",{\"opacity\":0.2}],[2,500,1433010892156,false]],[],[]]"} 135 11:34:52...
{"arguments":[[{"method":"invokeCallbackAndReturnFlushedQueue","module":"BatchedBridge","args":[5,[60,299.5,200,39,60,299.5]],"context":58}]],"moduleMet... 237 11:34:52...
{"replyID":10011,"result":"null"} 33 11:34:52...
{"arguments":[[{"method":"callFunctionReturnFlushedQueue","module":"BatchedBridge","args":[2,1,["topTouchEnd",[{"target":6,"pageY":322,"locationX":42.5,"cha... 384 11:34:52...
{"replyID":10012,"result":"[[22,11,22,32,22],[17,1,5,0,5],[[],[2],[6,\"RCTView\",{\"opacity\":1}],[\"http\",{\"method\":\"POST\",\"url\":\"https://api.conunter.io/?... 269 11:34:52...
{"arguments":[[{"method":"invokeCallbackAndReturnFlushedQueue","module":"BatchedBridge","args":[7,["{\"status\":0,\"responseText\":\"A server with the speci... 330 11:34:52...
{"replyID":10013,"result":"null"} 33 11:34:52...
{"arguments":[[{"method":"callFunctionReturnFlushedQueue","module":"BatchedBridge","args":[2,1,["topTouchStart",[{"target":6,"pageY":313.5,"locationX":32.5,"c... 384 11:35:00...
{"replyID":10014,"result":"[[22,22,22,11],[16,8,5,0],[[6],[6,9],[6,\"RCTView\",{\"opacity\":0.2}],[5,500,1433010900921,false]],[],[]]"} 135 11:35:00...
{"arguments":[[{"method":"invokeCallbackAndReturnFlushedQueue","module":"BatchedBridge","args":[9,[60,299.5,200,39,60,299.5]],"context":58}]],"moduleMet... 237 11:35:00...
{"replyID":10015,"result":"null"} 33 11:35:00...
{"arguments":[[{"method":"callFunctionReturnFlushedQueue","module":"BatchedBridge","args":[0,0,[[5]]],"context":58}]],"moduleMethod":"processBatch","method... 213 11:35:01...
{"replyID":10016,"result":"null"} 33 11:35:01...
{"arguments":[[{"method":"callFunctionReturnFlushedQueue","module":"BatchedBridge","args":[2,1,["topTouchEnd",[{"target":6,"pageY":313.5,"locationX":32.5,"c... 383 11:35:02...
{"replyID":10017,"result":"[[22,22,32,22],[17,5,0,5],[[],[6,\"RCTView\",{\"opacity\":1}],[\"http\",{\"method\":\"POST\",\"url\":\"https://api.conunter.io/?value=2\... 259 11:35:02...
{"arguments":[[{"method":"invokeCallbackAndReturnFlushedQueue","module":"BatchedBridge","args":[11,["{\"status\":0,\"responseText\":\"A server with the spe... 331 11:35:02...
{"replyID":10018,"result":"null"} 33 11:35:02...
1 {"replyID":10017,"result":"[[22,22,32,22],[17,5,0,5],[[],[6,\"RCTView\",{\"opacity\":1}],[\"http\",{\"method\":\"POST\",\"url\":\"https://api.conur
1 / 8 requests | 0 B


tlv-websocket (ruby)
Heroku
nfo","SocketRocket: In debug mode. Allowing connection to any root cert"],"mod
10001,"moduleName":"RCTLog"}
od=GET path="/index.ios.bundle" host=react-counter.herokuapp.com request_id=90b
b.1 connect=1ms service=232ms status=200 bytes=1351353
ART> request:/index.ios.bundle
D>   request:/index.ios.bundle (13ms)
2.195 - - [Fri, 05 Jun 2015 02:56:59 GMT] "GET /launch-chrome-devtools HTTP/1.1
er/1 CFNetwork/711.3.18 Darwin/14.0.0"
.826630+00:00 app[web.1]: [2:56:59 AM] <START> request:/index.ios.bundle
.826713+00:00 app[web.1]: [2:56:59 AM] <END>   request:/index.ios.bundle (14ms)
.162023+00:00 heroku[router]: at=info method=GET path="/index.ios.bundle" host=react-counter.herokuapp.com request_id=37e
6d-79bd906efabe fwd="54.155.98.35" dyno=web.1 connect=1ms service=252ms status=200 bytes=1351353
.757324+00:00 app[web.1]: [[4,12],[0,0],[["__DEV_WARNINGS_IGNORED"],1],[4,3]],□,□]
.756697+00:00 app[web.1]: {"arguments":□,"moduleMethod":"flushedQueue","method":"executeJSCall","id":10003,"moduleName":
.906211+00:00 app[web.1]: [2:57:08 AM] <START> request:/index.ios.bundle
.906297+00:00 app[web.1]: [2:57:08 AM] <END>   request:/index.ios.bundle (15ms)
.078993+00:00 app[web.1]: {"arguments":[[{"method":"invokeCallbackAndReturnFlushedQueue","module":"BatchedBridge","args":
ctive"}]],"context":2},{"method":"callFunctionReturnFlushedQueue","module":"BatchedBridge","args":[5,0,["reachabilityDidC
eachability":"wifi"}]],"context":2},{"method":"callFunctionReturnFlushedQueue","module":"BatchedBridge","args":[5,0,["appS
p_state":"active"}]],"context":2},{"method":"callFunctionReturnFlushedQueue","module":"BatchedBridge","args":[4,0,["Count
nitialProps":{}}]],"context":2}]],"moduleMethod":"processBatch","method":"executeJSCall","id":10004,"moduleName":"Batched
.108002+00:00 app[web.1]: [[22,22,22,22,22,22,22,22,22,22,22,22,22,22],[4,4,4,4,3,4,4,4,3,3,3,3,3],[[2,"RCTView",{"position"
0,"top":0,"right":0,"bottom":0}],[3,"RCTView",{"flex":1,"justifyContent":"center","alignItems":"center","backgroundColor"
Bottom":70}],[4,"RCTText",{"accessible":true,"isHighlighted":false,"fontSize":100,"textAlign":"center","margin":10}],[5,"
:"0"}],[4,null,null,[5],[0],null],[6,"RCTView",{"accessible":true,"padding":10,"borderRadius":3,"backgroundColor":"#337ab
nter","width":200}],[7,"RCTText",{"accessible":true,"isHighlighted":false,"color":"white","fontSize":16}],[8,"RCTRawText"
,[7,null,null,[8],[0],null],[6,null,null,[7],[0],null],[3,null,null,[4,6],[0,1],null],[2,null,null,[3],[0],null],[1,null,
,□,□]
.080647+00:00 app[web.1]: Running application "Counter" with appParams: {"rootTag":1,"initialProps":{}}. __DEV__ === true
 warning are ON, performance optimizations are OFF
.407024+00:00 app[web.1]: {"arguments":[[{"method":"invokeCallbackAndReturnFlushedQueue","module":"BatchedBridge","args":
ARNINGS_IGNORED",null]]]],"context":2}]],"moduleMethod":"processBatch","method":"executeJSCall","id":10005,"moduleName":"
.407762+00:00 app[web.1]: null
9:41 AM
100%
0
Add +1
iPhone

Streaming



$$UI = f(data)$$

Thank you!

github.com/frantic/tlv-2015

@alex_frantic