

Rx.js

rethink async with rx.js

Asynchronous code

Async code in javascript

- Some computation with delayed result.
(data fetch, animations, DOM operations)
- Reacting to events (event handlers)

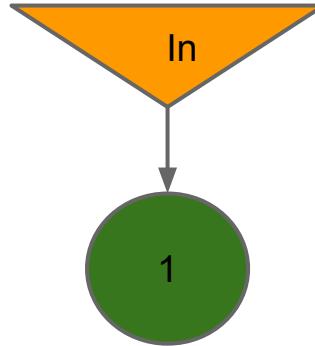
Sync vs Async dataflow

Synchronous dataflow

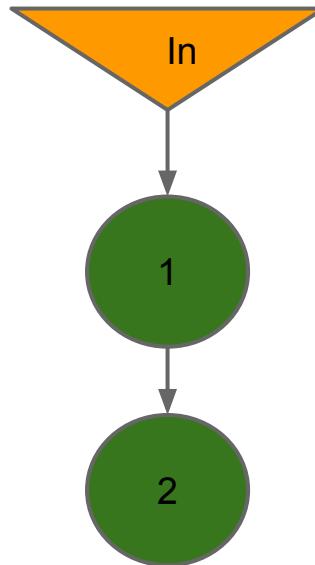
Synchronous dataflow



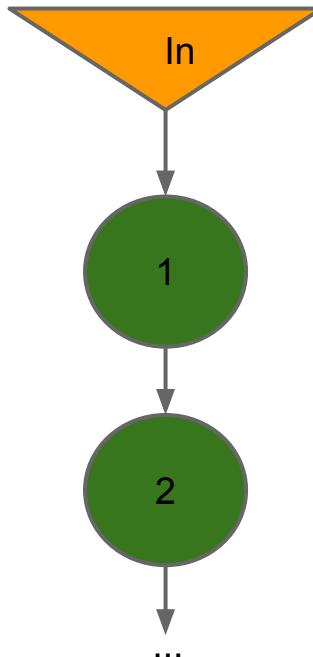
Synchronous dataflow



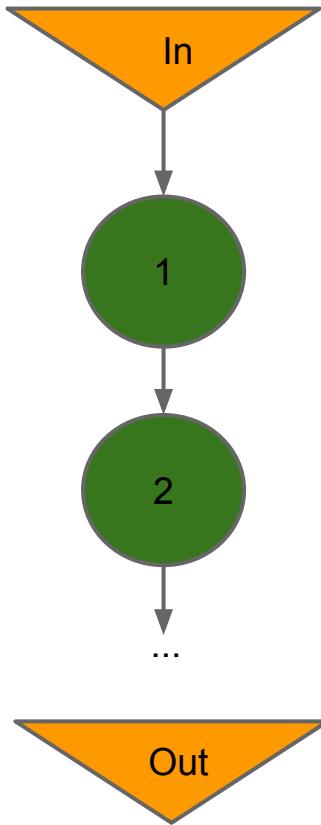
Synchronous dataflow



Synchronous dataflow



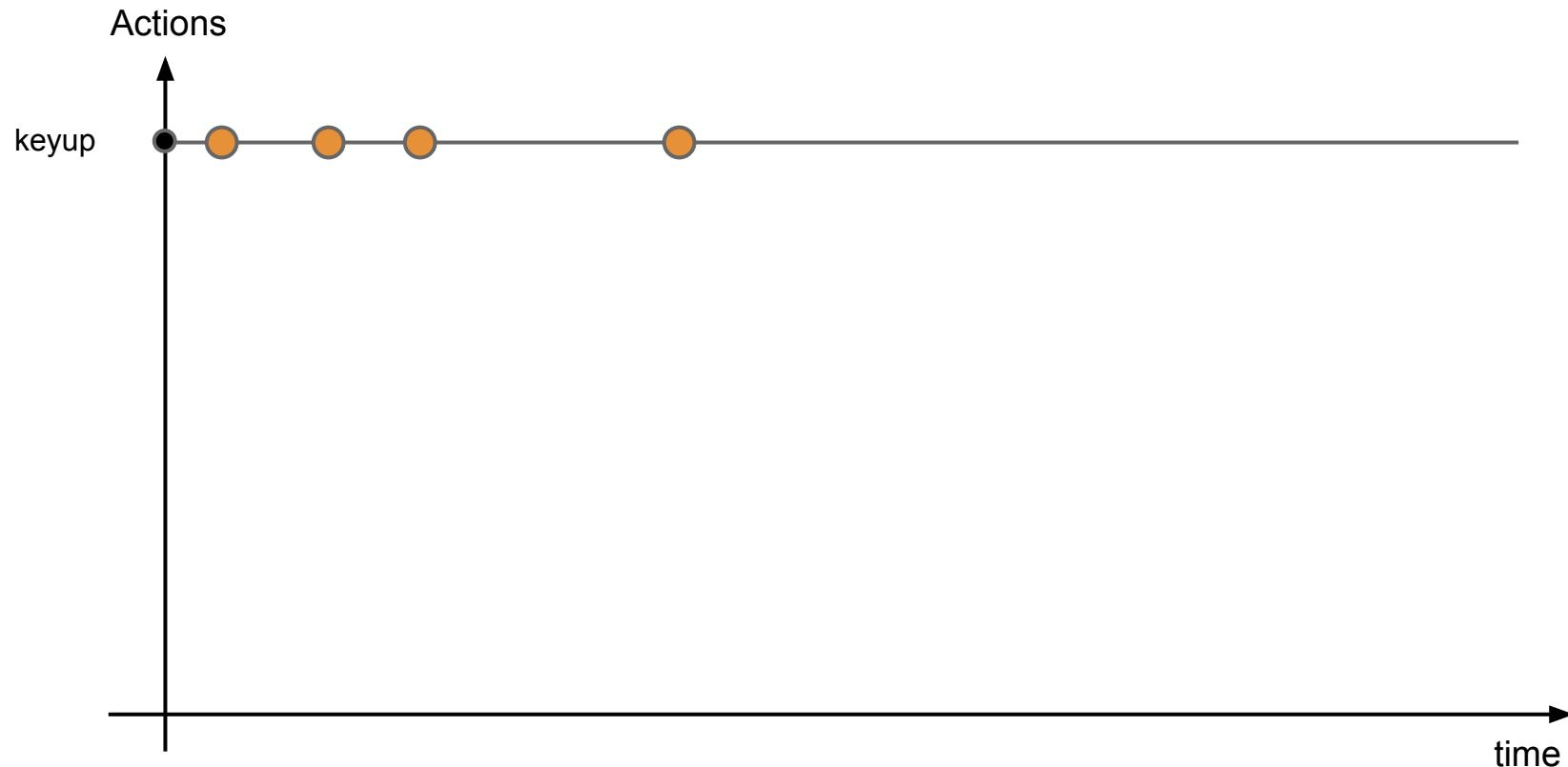
Synchronous dataflow



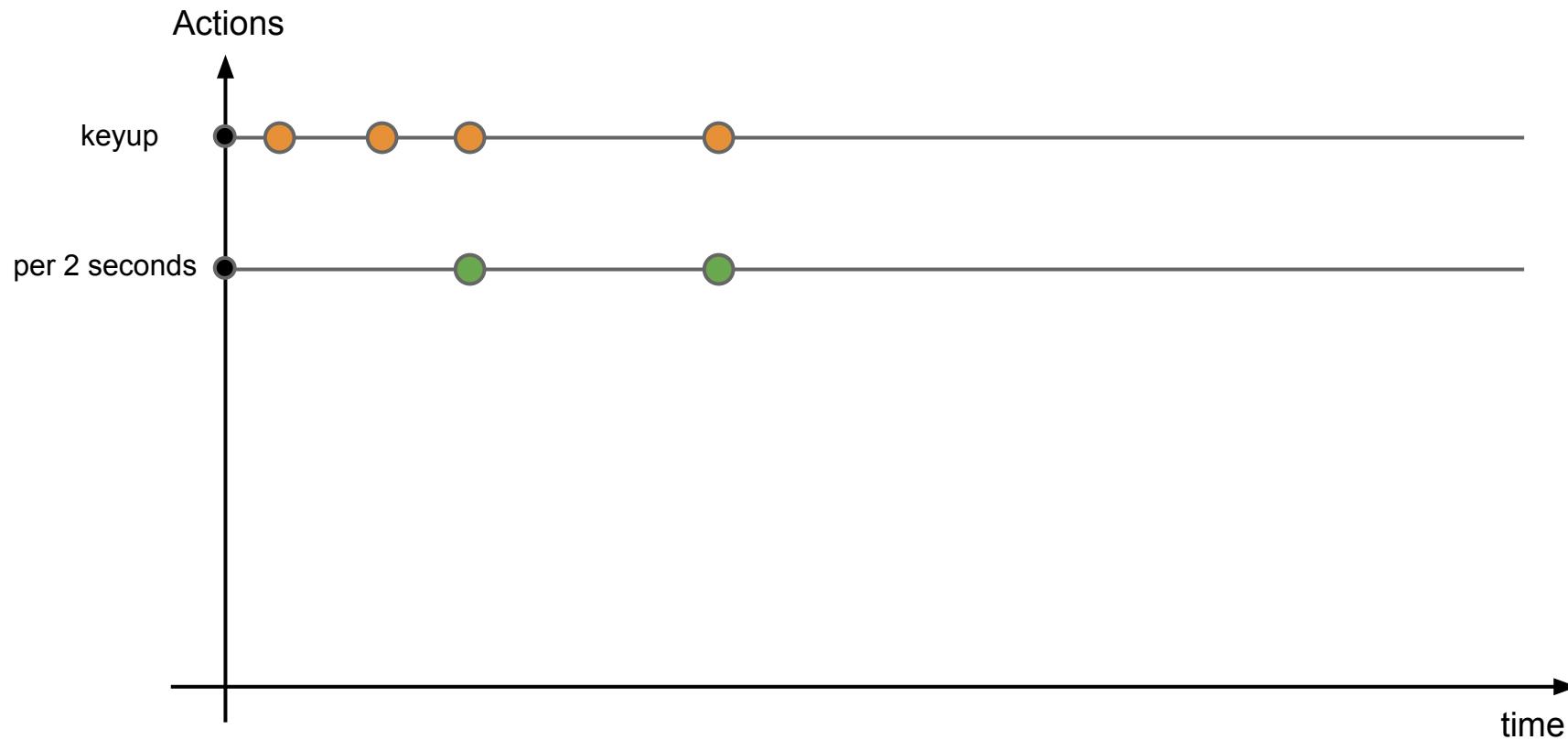
Asynchronous dataflow



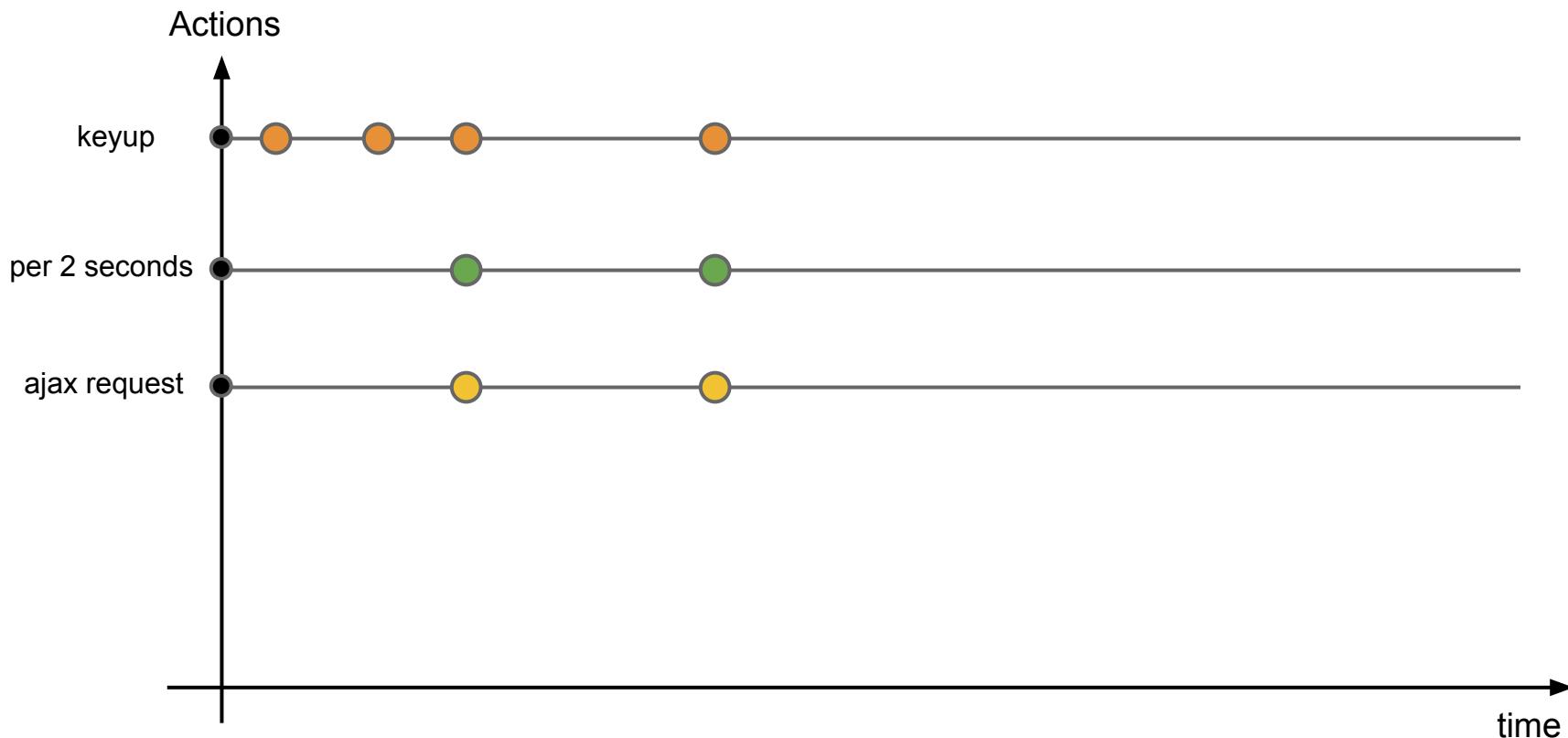
Asynchronous dataflow (autocomplete)



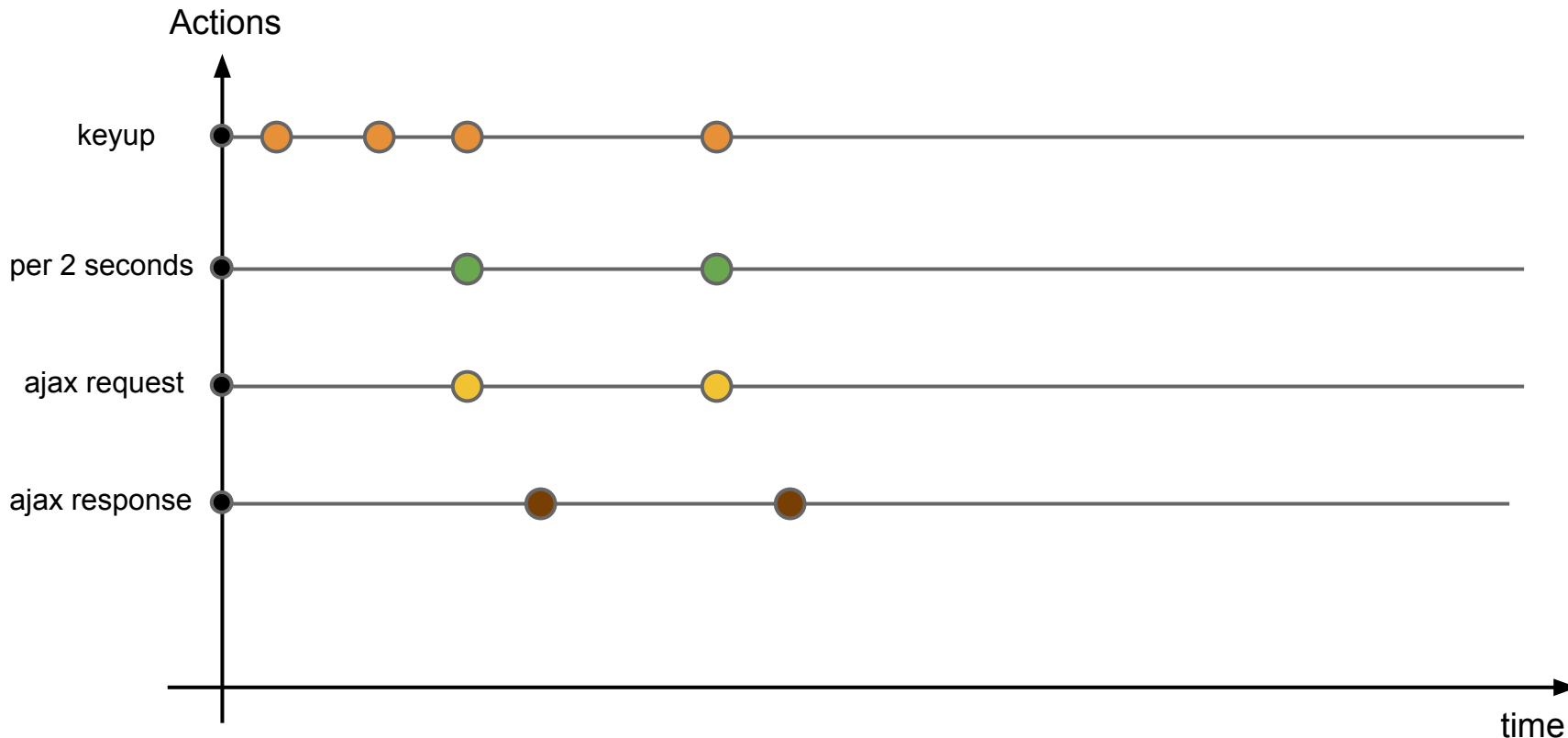
Asynchronous dataflow (autocomplete)



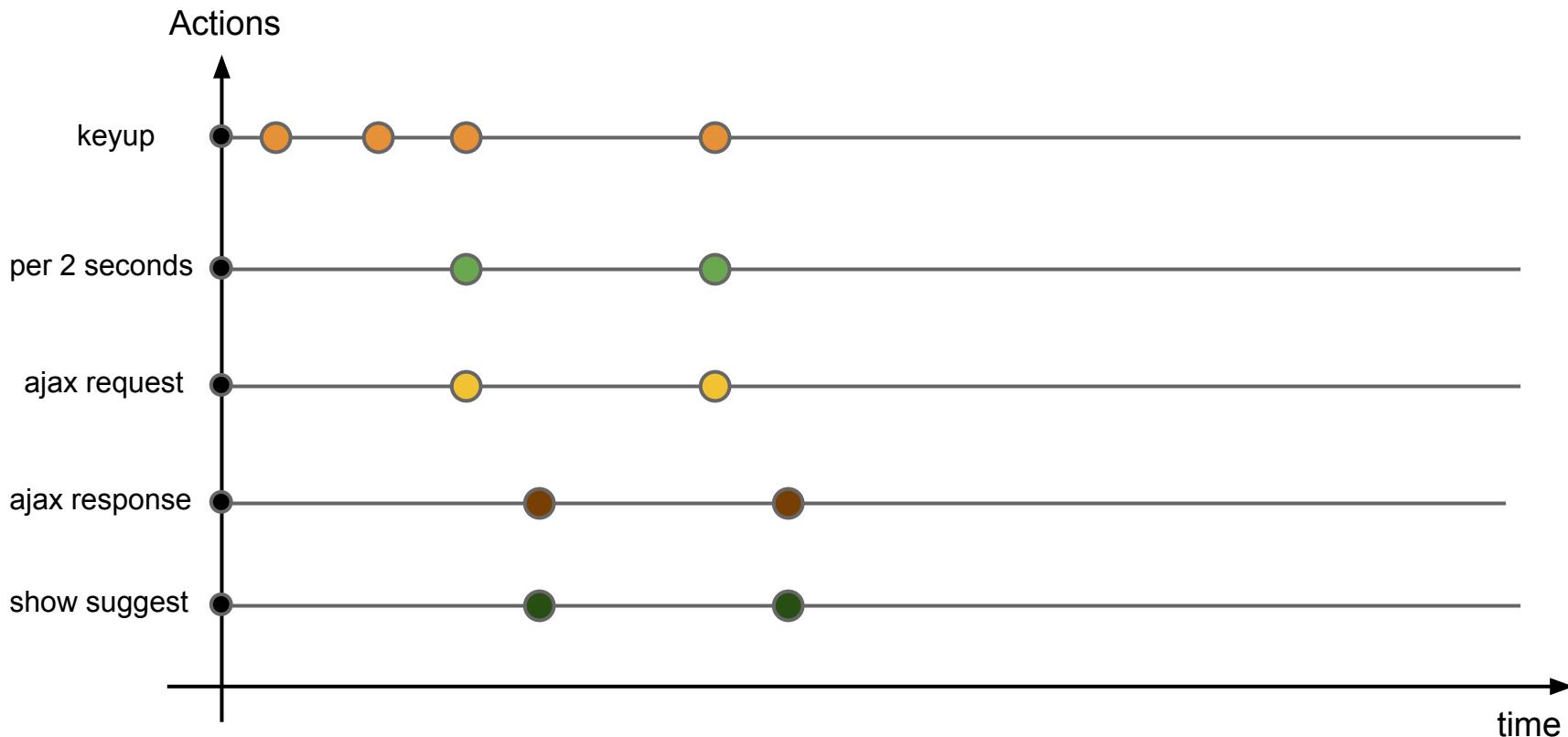
Asynchronous dataflow (autocomplete)



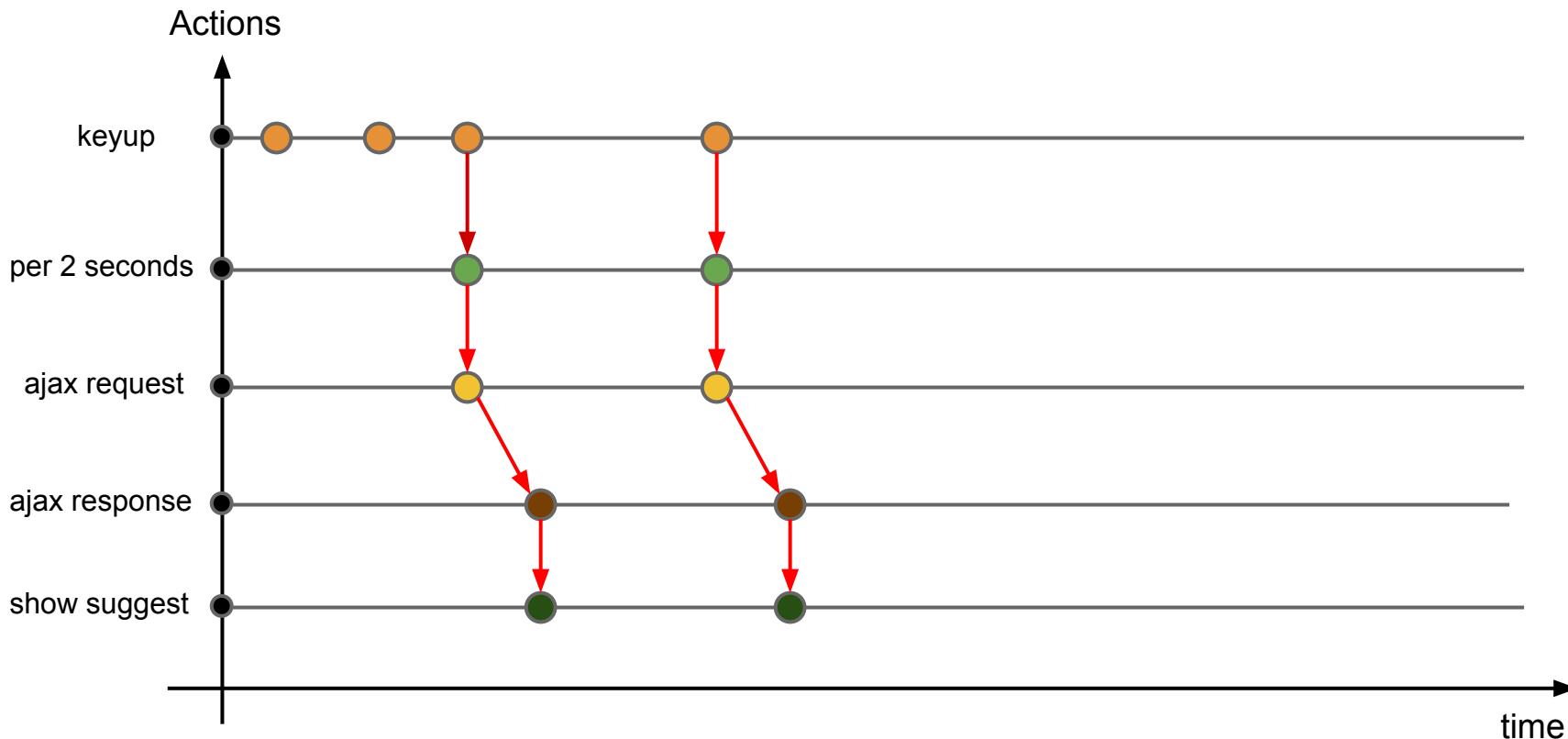
Asynchronous dataflow (autocomplete)



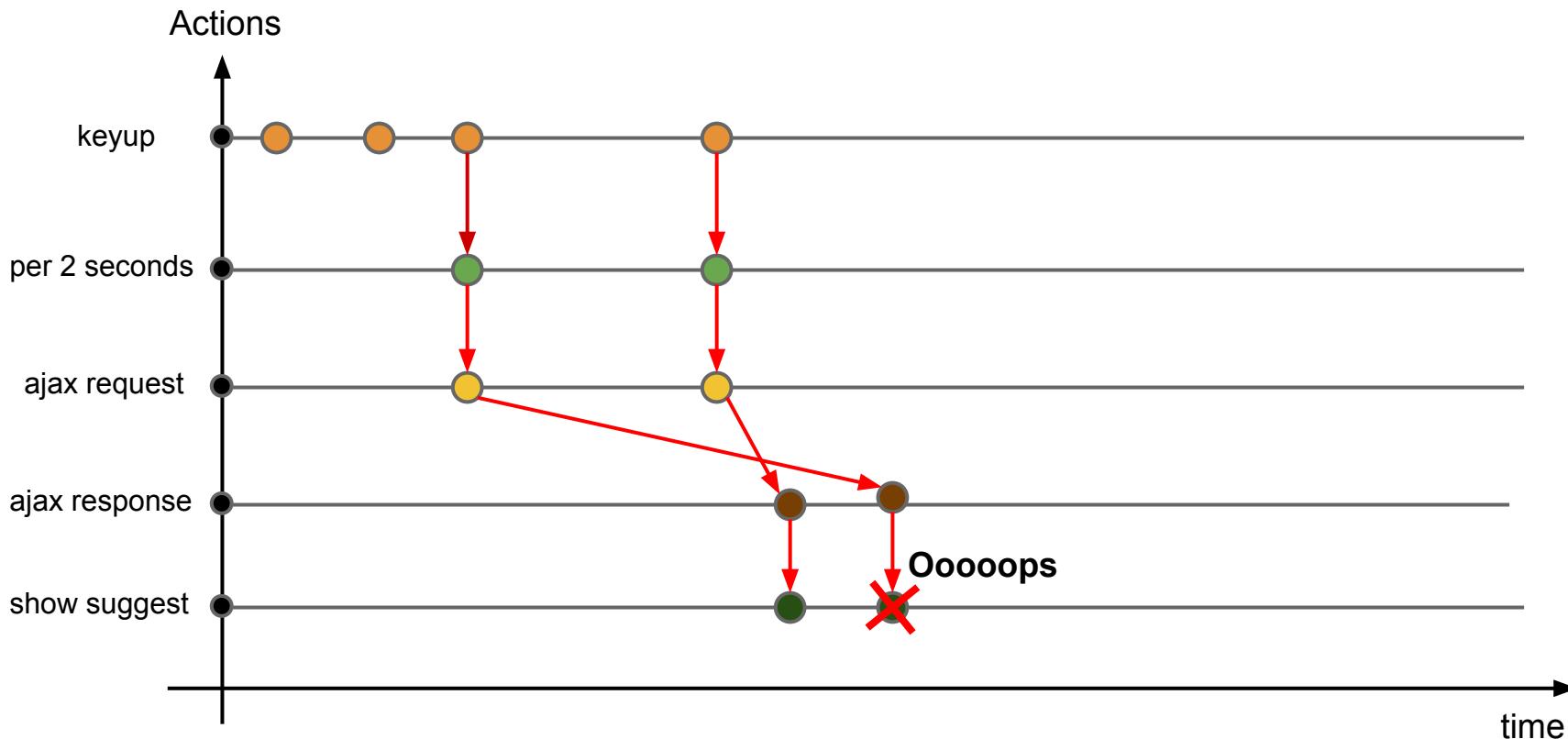
Asynchronous dataflow (autocomplete)



Asynchronous dataflow (autocomplete)



Asynchronous dataflow (autocomplete)



Problems in async code

1. How to describe async flow in code?
2. How to handle errors?
3. How to compose?

Callbacks

Event notification

Manage control flow

Callbacks

Event notification - Yes

Manage control flow - Nooooooooooooo

Callback hell

```
async_func1(function(err, data){  
    async_func2(function(err, data){  
        async_func3(function(err, data){  
            async_func4(function(err, data){  
                });  
            });  
        });  
    });  
});
```

Callback hell yeah !!!

```
async_func1(function(err, data){  
  if (!err) {  
    async_func2(function(err, data){  
      if (!err) {  
        async_func3(function(err, data){  
          if (!err) {  
            async_func4(function(err, data){  
              });  
            } else {  
              log(err);  
            }  
          });  
        } else {  
          log(err);  
        }  
      });  
    } else {  
      log(err);  
    }  
});
```

**MAN THAT REALLY
SUCKS**

BRO

Monad

A monad is just a monoid in the category of
endofunctors

Callback vs Black hole

```
some_async_func(function(err, result) {  
  // ... looks like we are in trap  
  another_wonderful_func(function(err, result){  
    // ... you are in trap  
  });  
});
```



Callback vs **Black Hole**

1. No return value (`undefined`)

Callback vs **Black Hole**

1. No return value (`undefined`)
2. One way ticket (`callback hell`)

How about Promise?

Promise

1. Step forward from callbacks for control flow managing

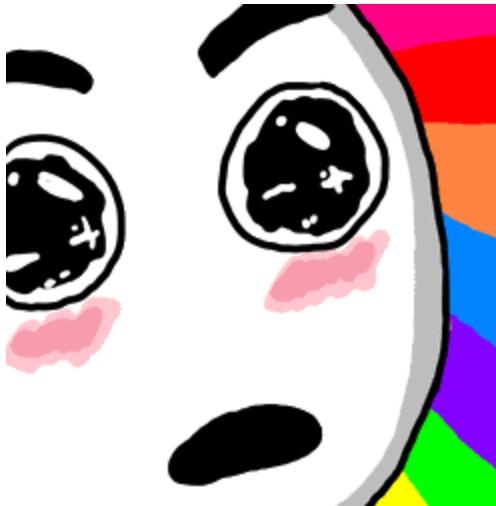
Promise

1. Step forward from callbacks for control flow managing
2. Error handling

Promise

1. Step forward from callbacks for control flow managing
2. Error handling
3. Works with single value :(
Observable is Promise++ :)

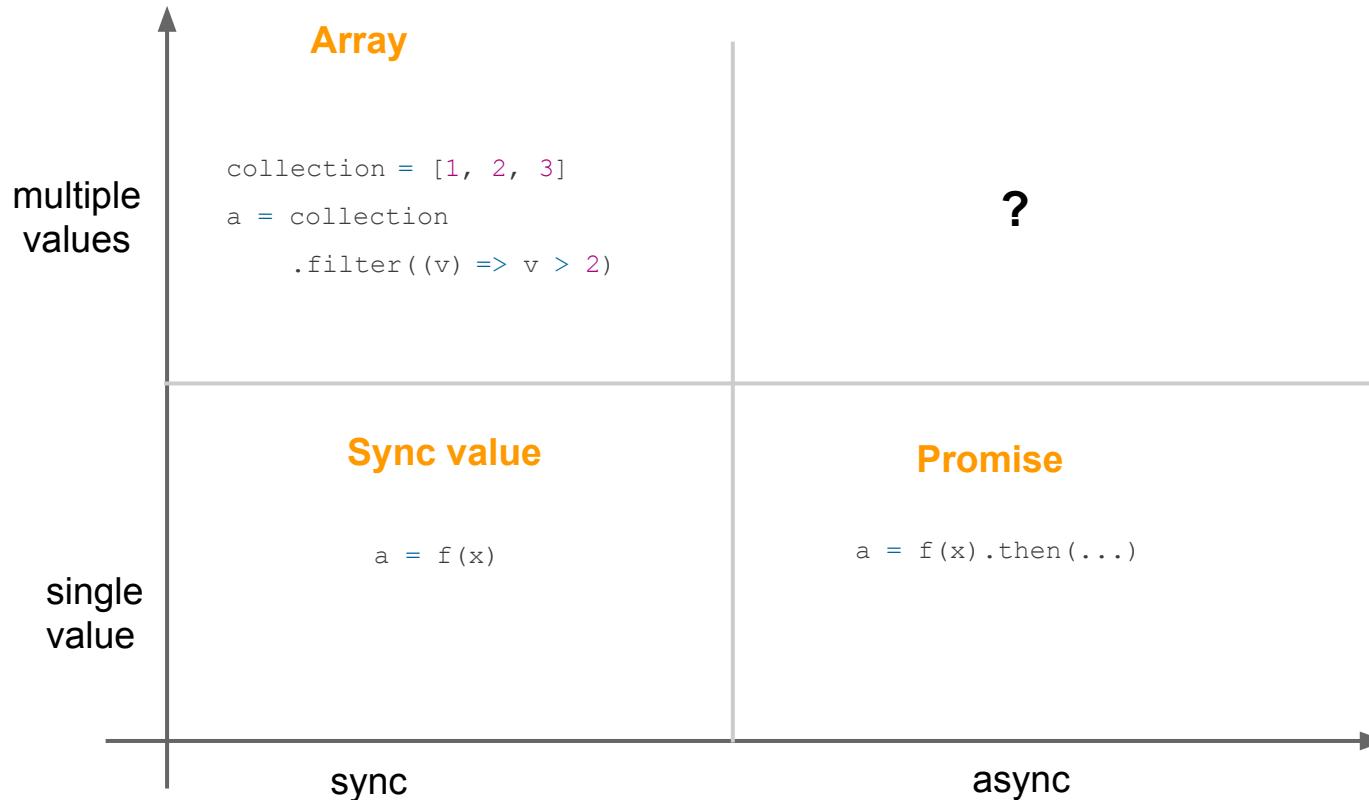
Promise is Observable only with a single value



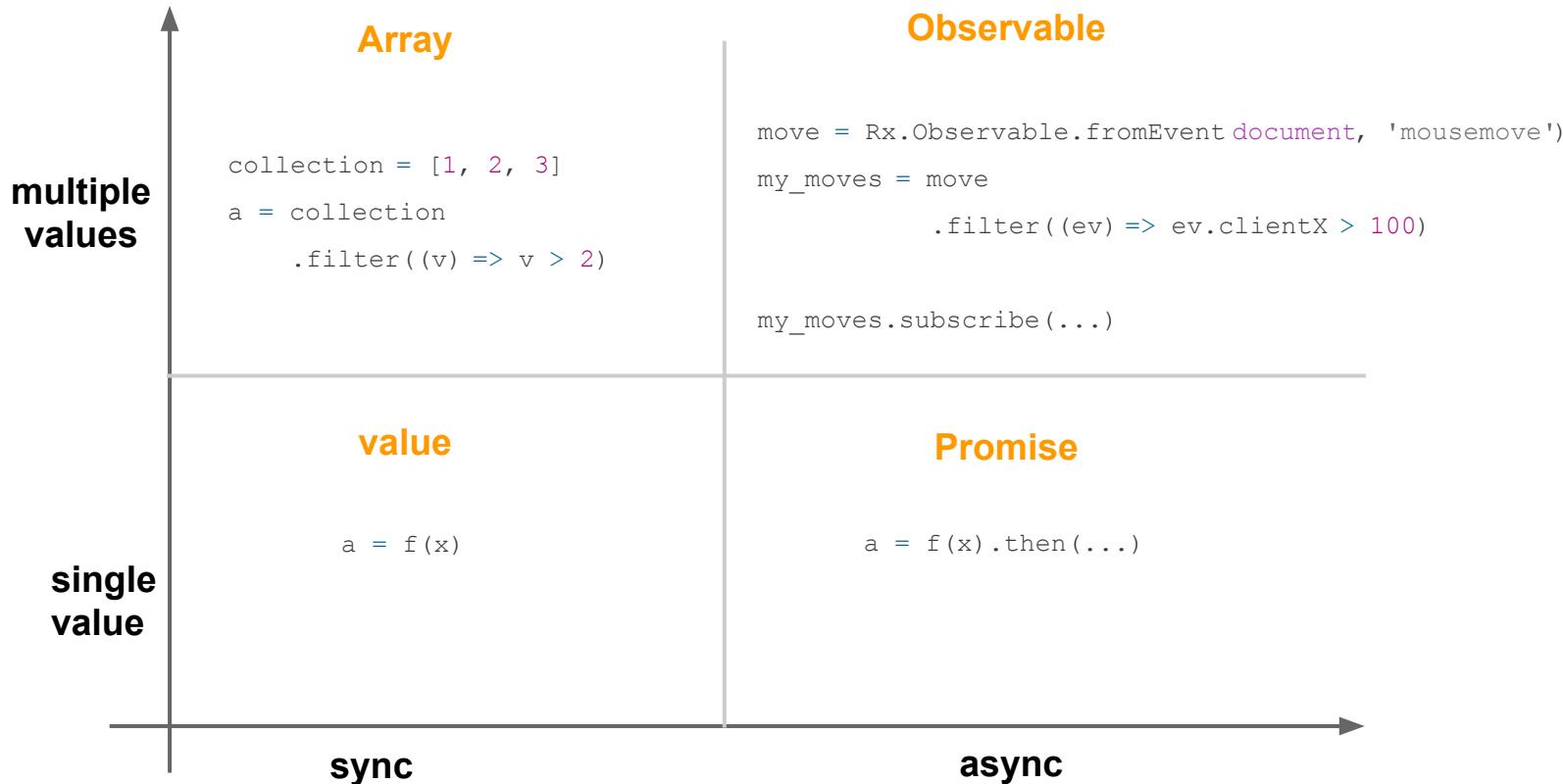
Observable

The **Observable** object represents a push based collection.

Asynchronous programming landscape



Asynchronous programming landscape



Drag & Drop example

What if i told you

That mouse move event is an array and you can map, filter over it?

<http://jsfiddle.net/AlexMost/2nhryfb1/>

Observable - first class events

Can be:

- passed as a parameter
- returned from a function
- assigned to a variable



Everything is a stream

How to create Observable

- Observable.create
- Observable.fromEvent
- Observable.fromNodeCallback
- Observable.fromArray
- Observable.fromPromise

Observable

```
observable.subscribe(  
    onNextCb,  
    onError,  
    onCompleted  
)
```

Observable is a collection

map

filter

reduce

zip

merge

flatMap

...

<https://github.com/Reactive-Extensions/RxJS/blob/master/doc/api/core/observable.md>

map



`map(x => 10 * x)`



<http://rxmarbles.com/#map>

filter



```
filter(x => x > 10)
```



<http://rxmarbles.com/#filter>

filter demo (drag and drop)

With filter we can narrow drag and drop area

<http://jsfiddle.net/AlexMost/2nhryfb1/3/>

delay



delay



<http://rxmarbles.com/#delay>

delay demo

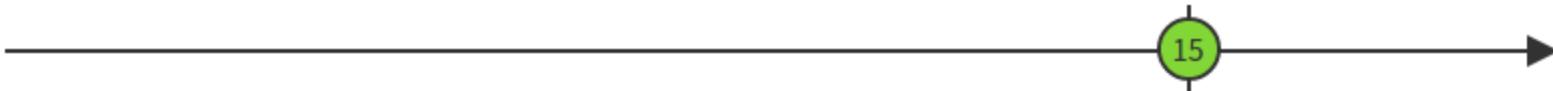
<http://jsfiddle.net/AlexMost/2nhryfb1/4/>

<http://jsfiddle.net/mattpodwysocki/9EjSQ/>

reduce



```
reduce((x, y) => x + y)
```



<http://rxmarbles.com/#reduce>

reduce demo

<http://jsfiddle.net/AlexMost/2nhryfb1/5/>

flatMap

Request stream



Response stream

merge



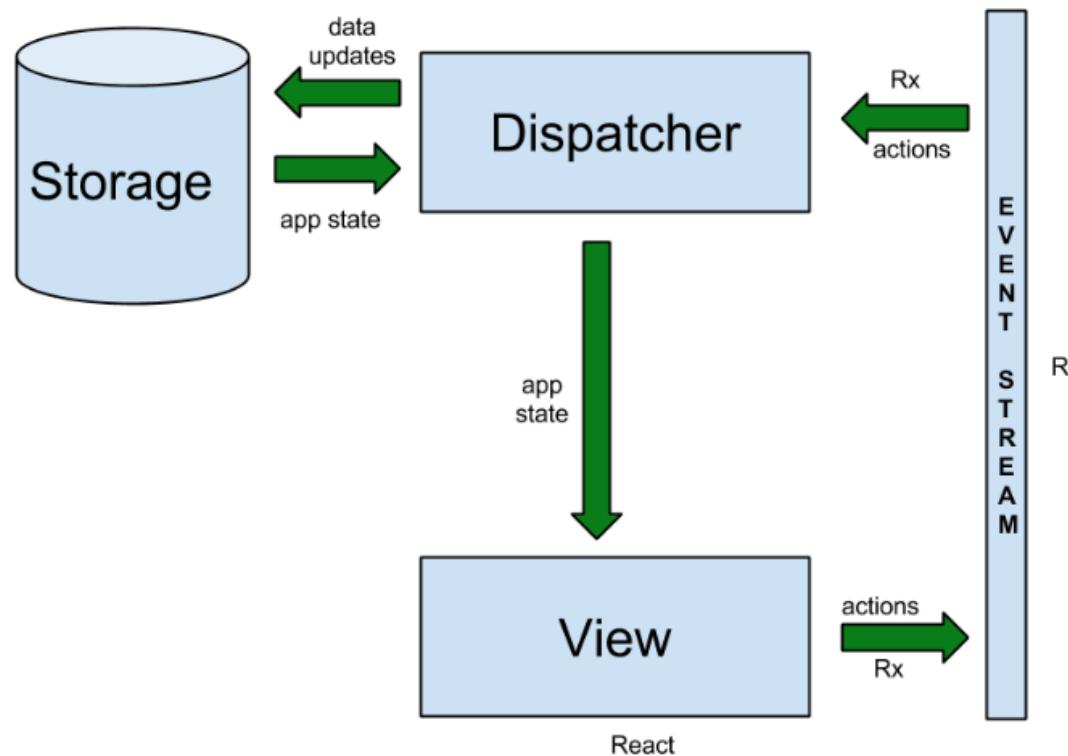
merge



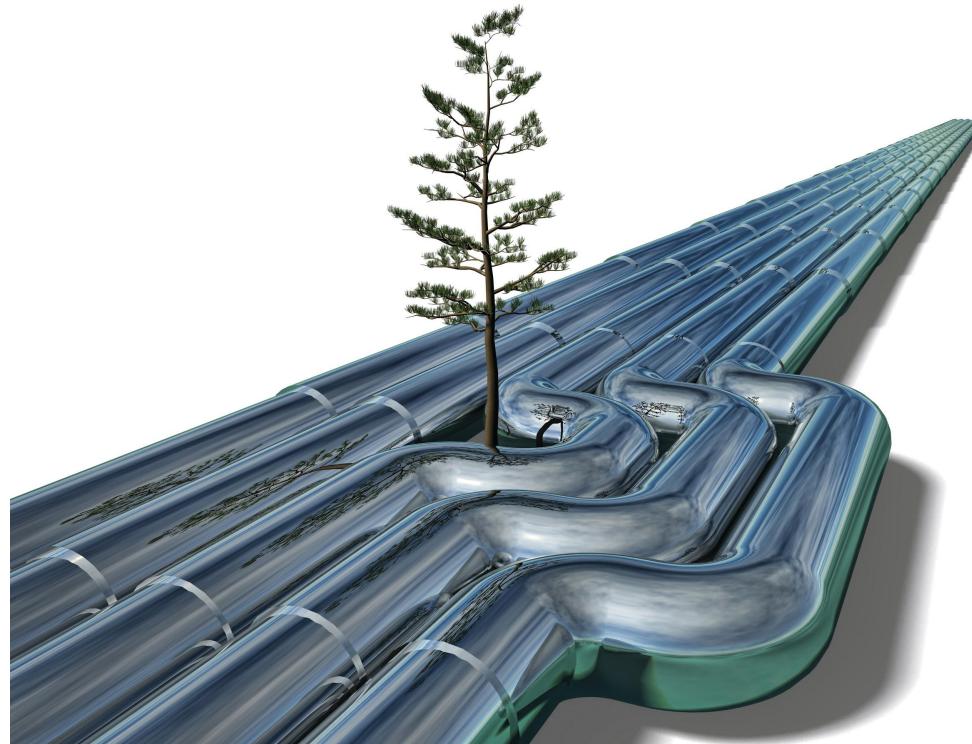
merge demo

<http://jsfiddle.net/AlexMost/2nhryfb1/6/>

Rx for javascript apps (RxReact)



dispatcher



Testing with time machine

Rx.TestScheduler



But it's another story ...

Resources

<http://rxmarbles.com/>

<http://jaredforsyth.com/rxvision/>

<https://github.com/Reactive-Extensions/RxJS/tree/master/examples>

Questions?