

Mobile and The New Two Tiered Web Architecture

QCON SF
18 November 2011



Aditya Bansod



Sr. Director, Product Management



@adityabansod
aditya@senchacorp.com



Web Architecture is Changing

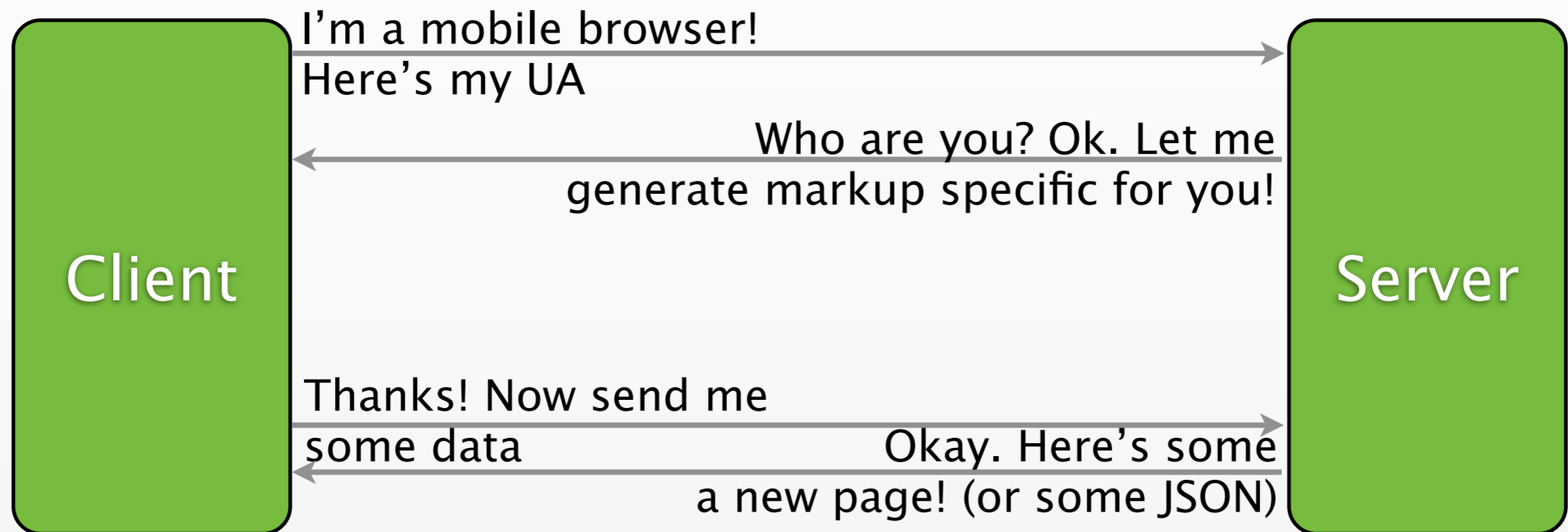


Web Architecture is
Changing

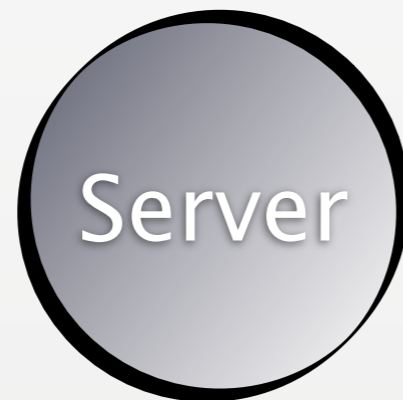
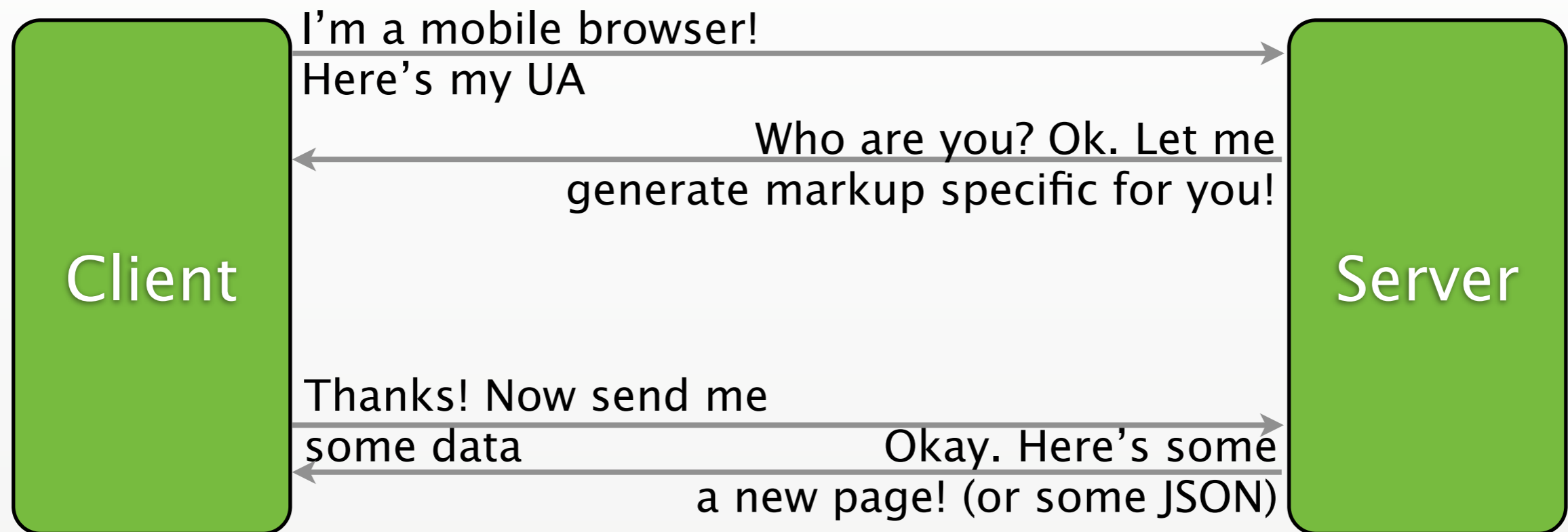
(changed?)



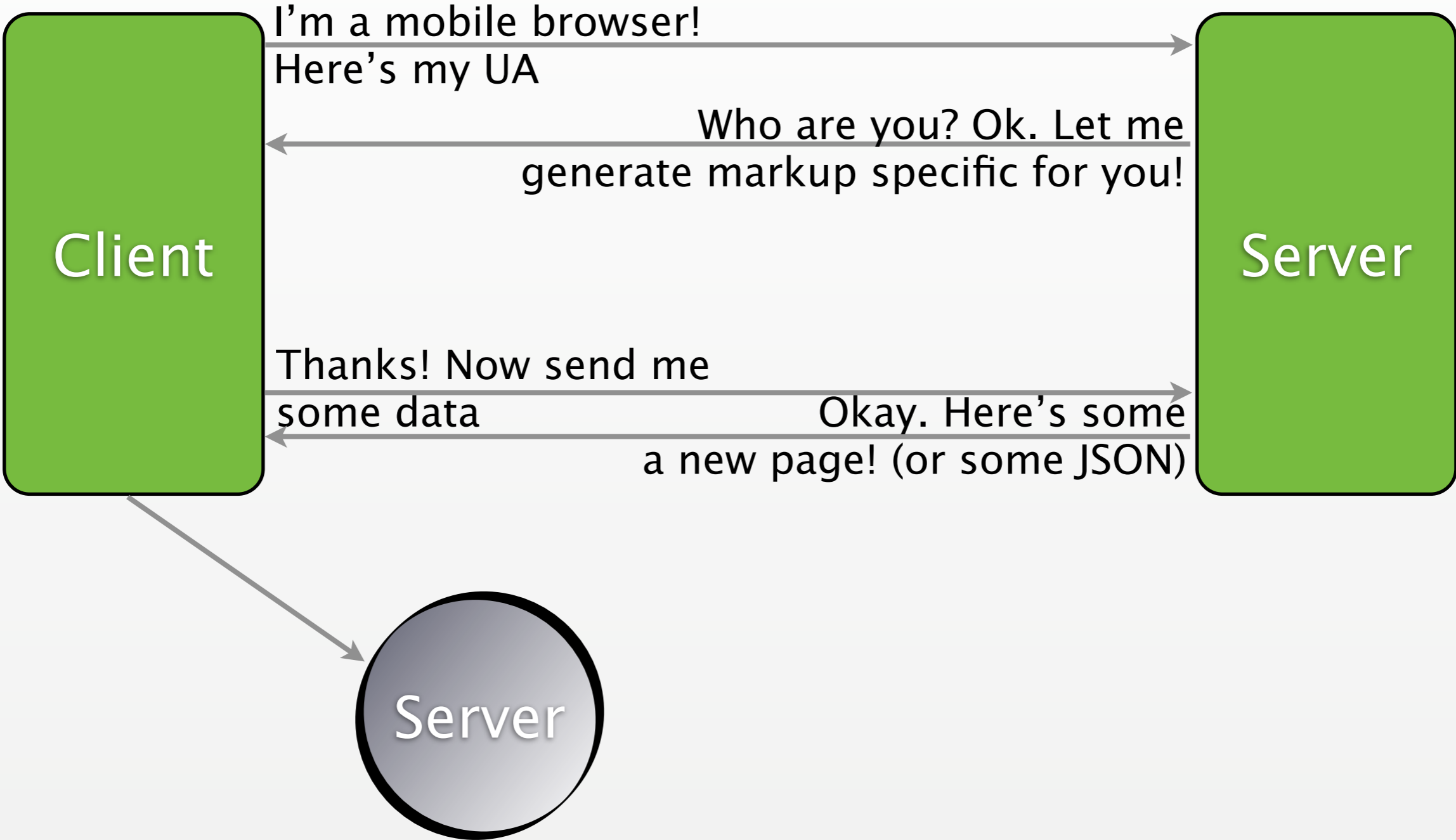
Mobile is Changing Web Architecture



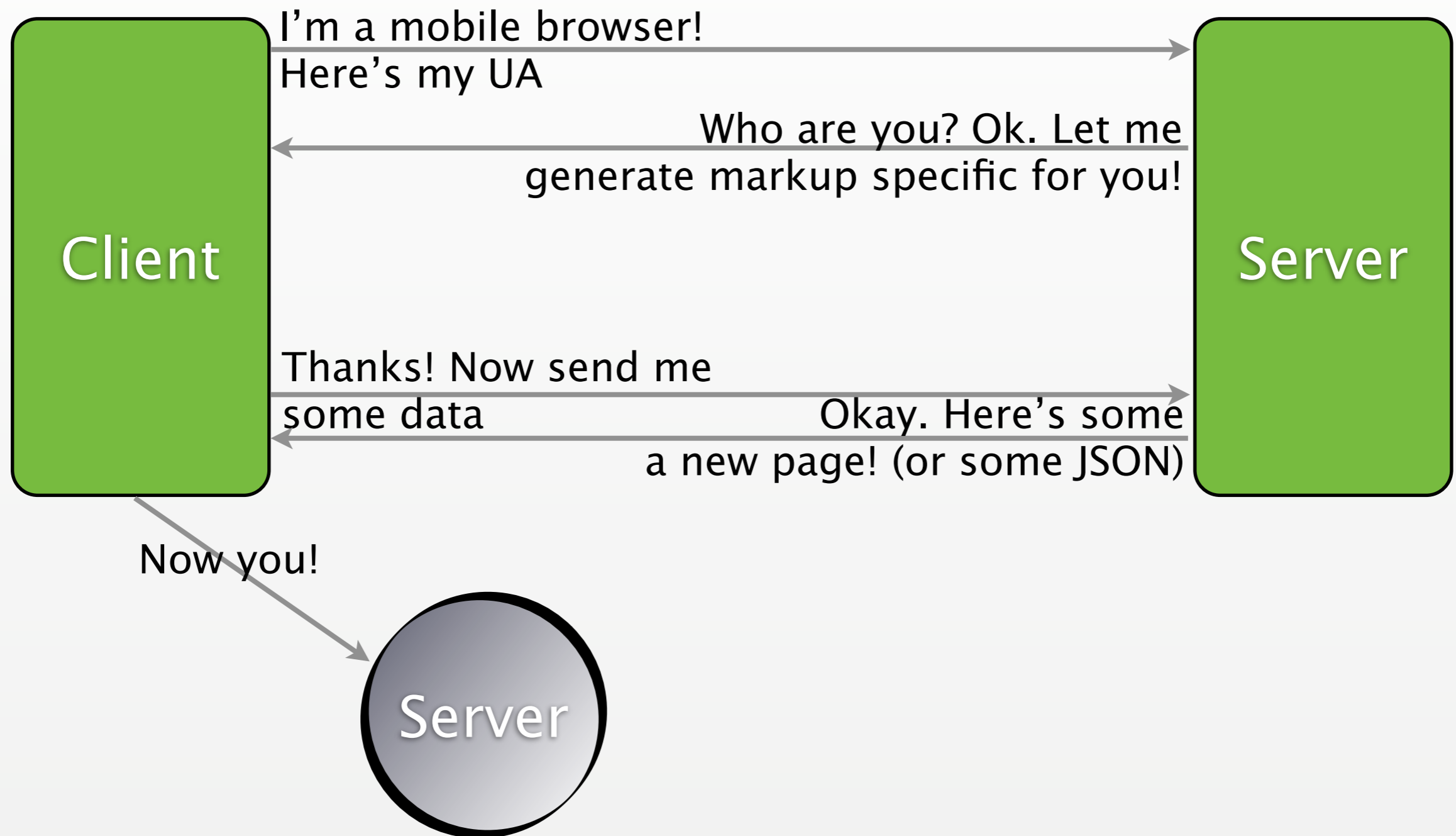
Mobile is Changing Web Architecture



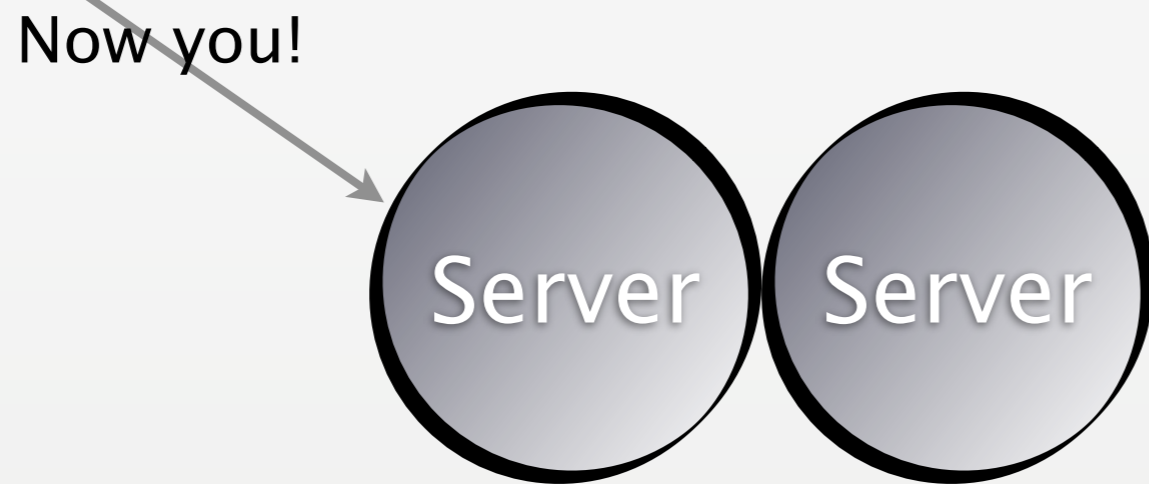
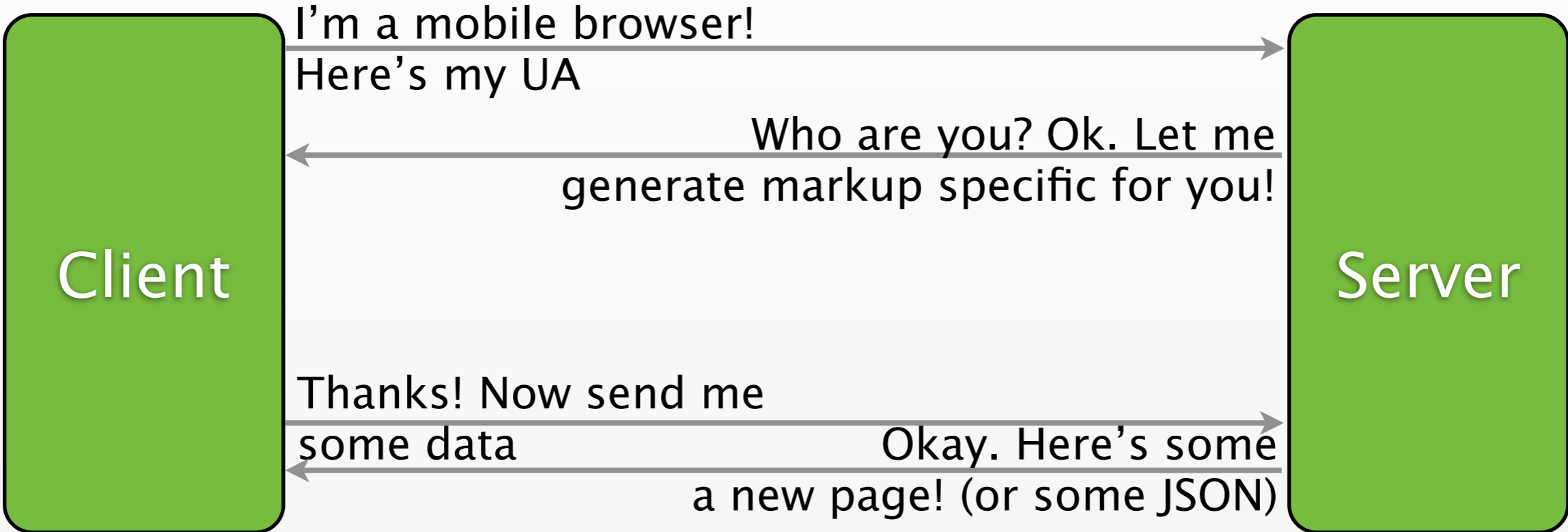
Mobile is Changing Web Architecture



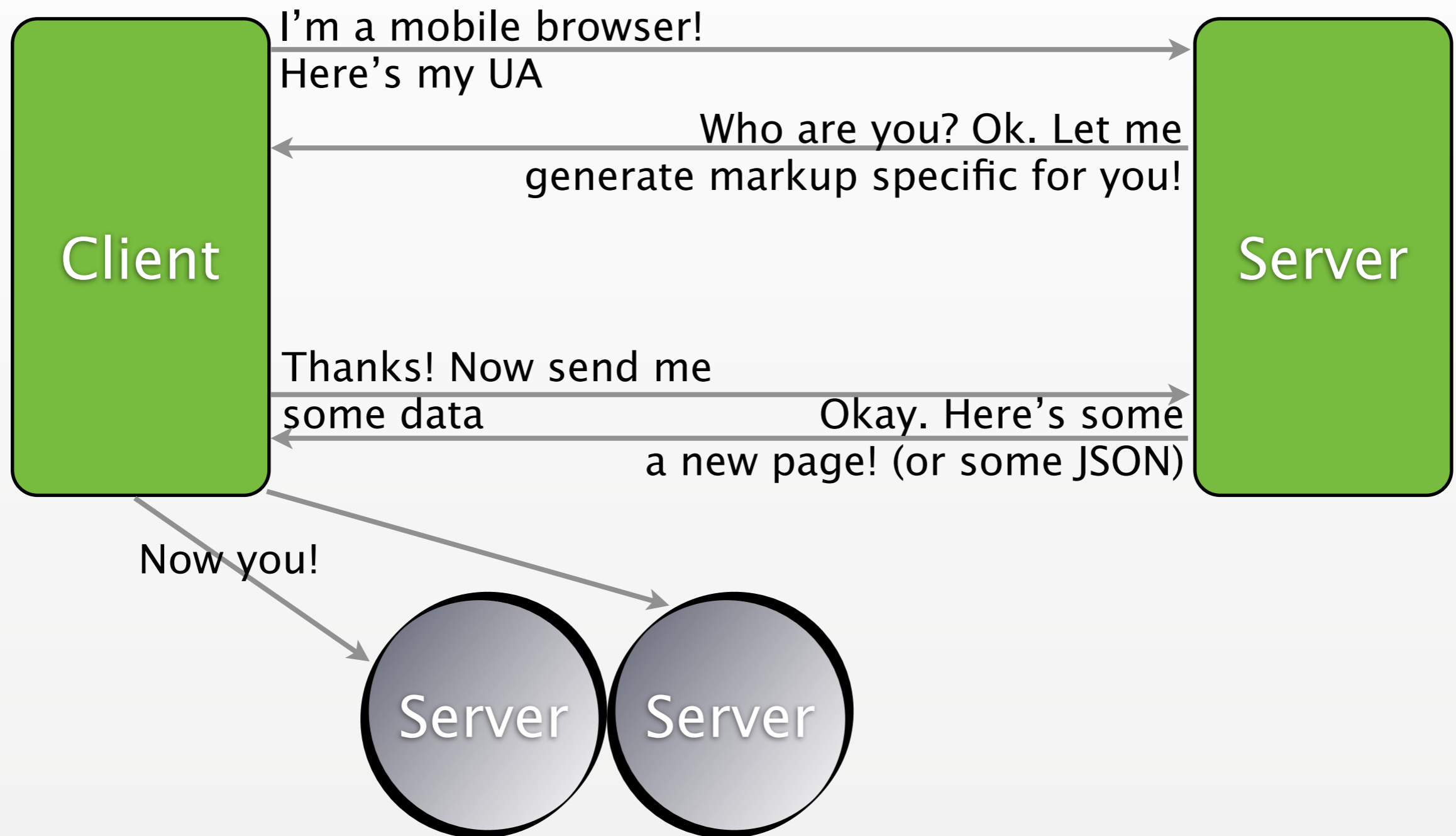
Mobile is Changing Web Architecture



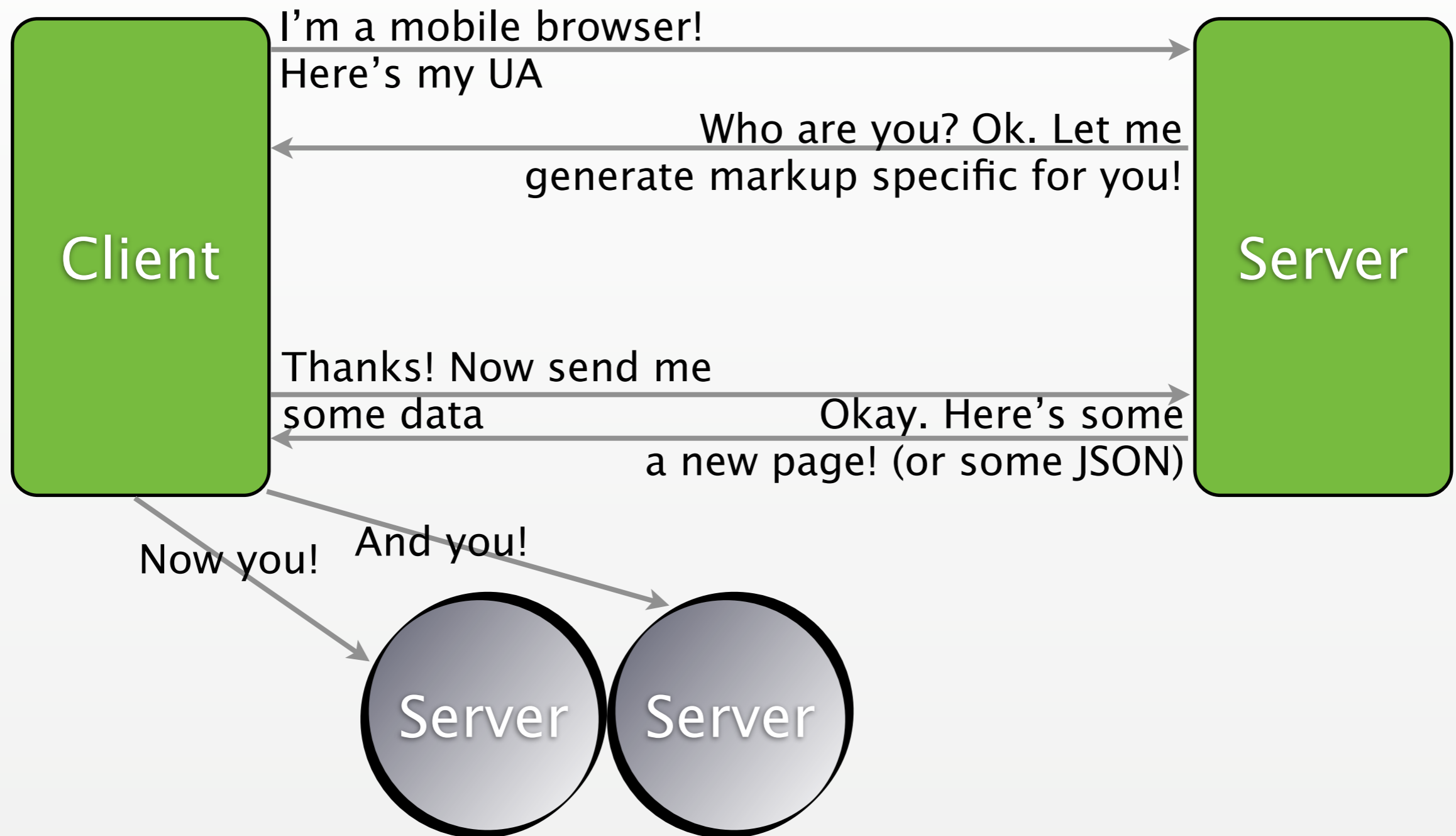
Mobile is Changing Web Architecture



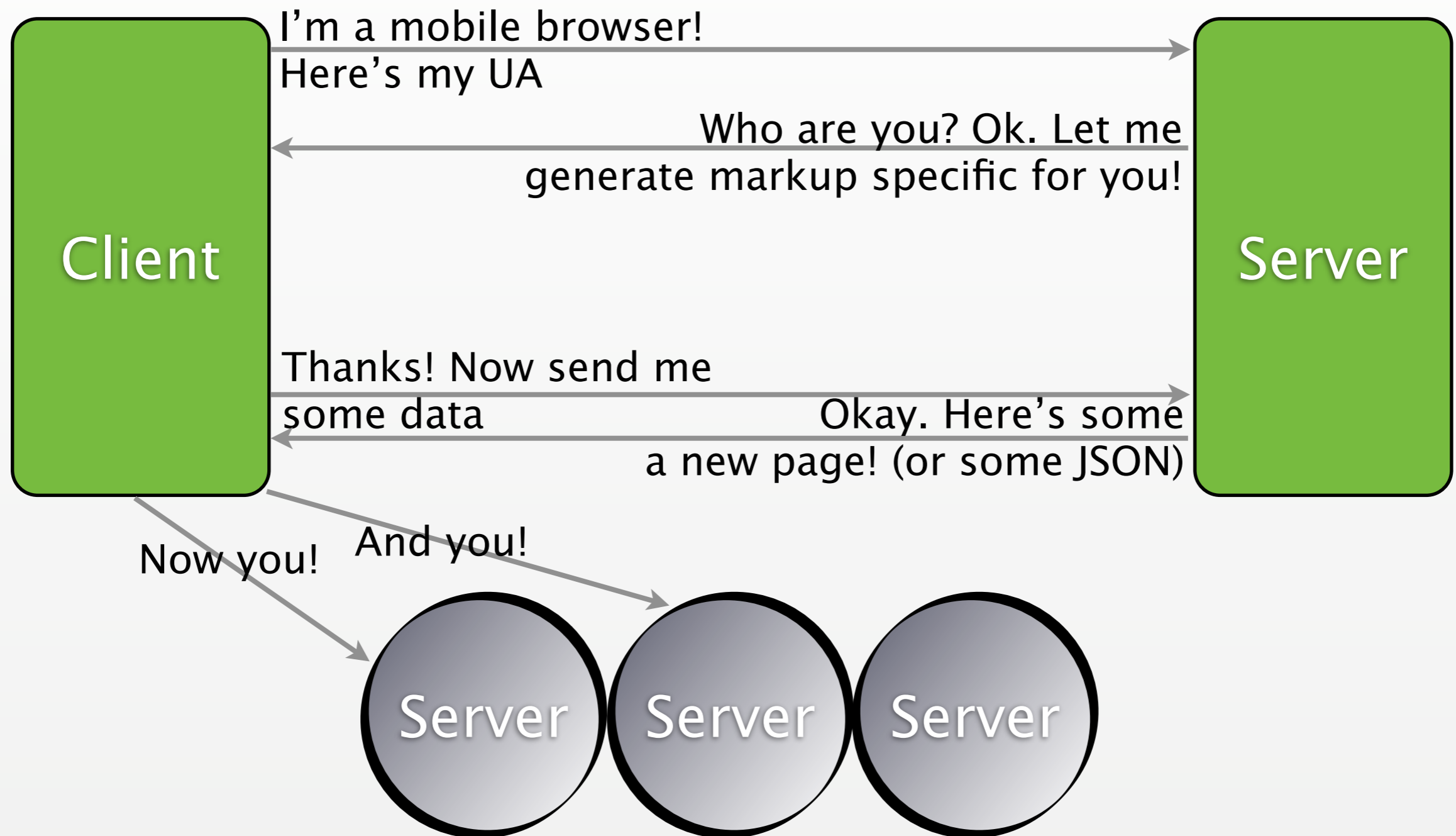
Mobile is Changing Web Architecture



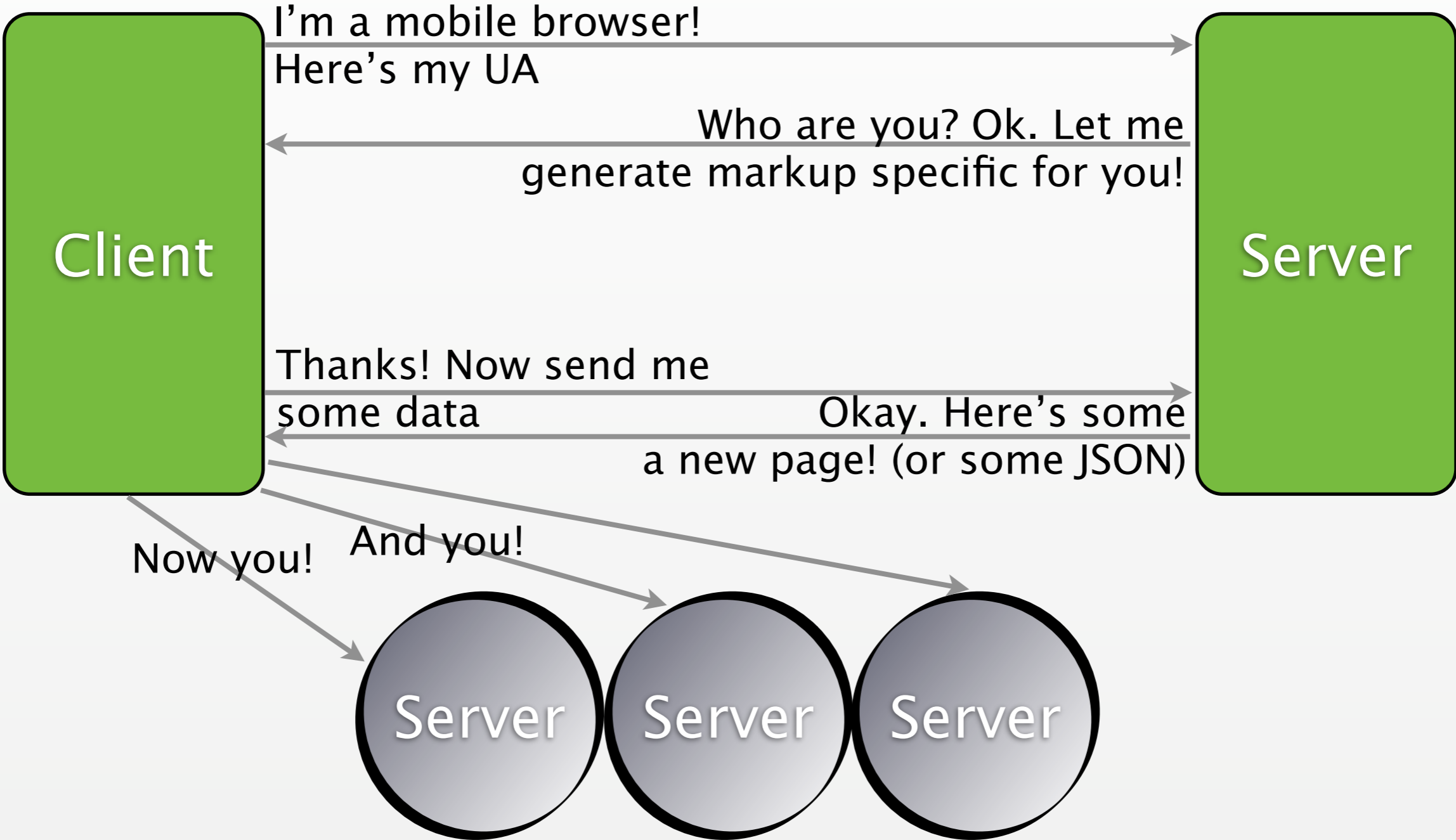
Mobile is Changing Web Architecture



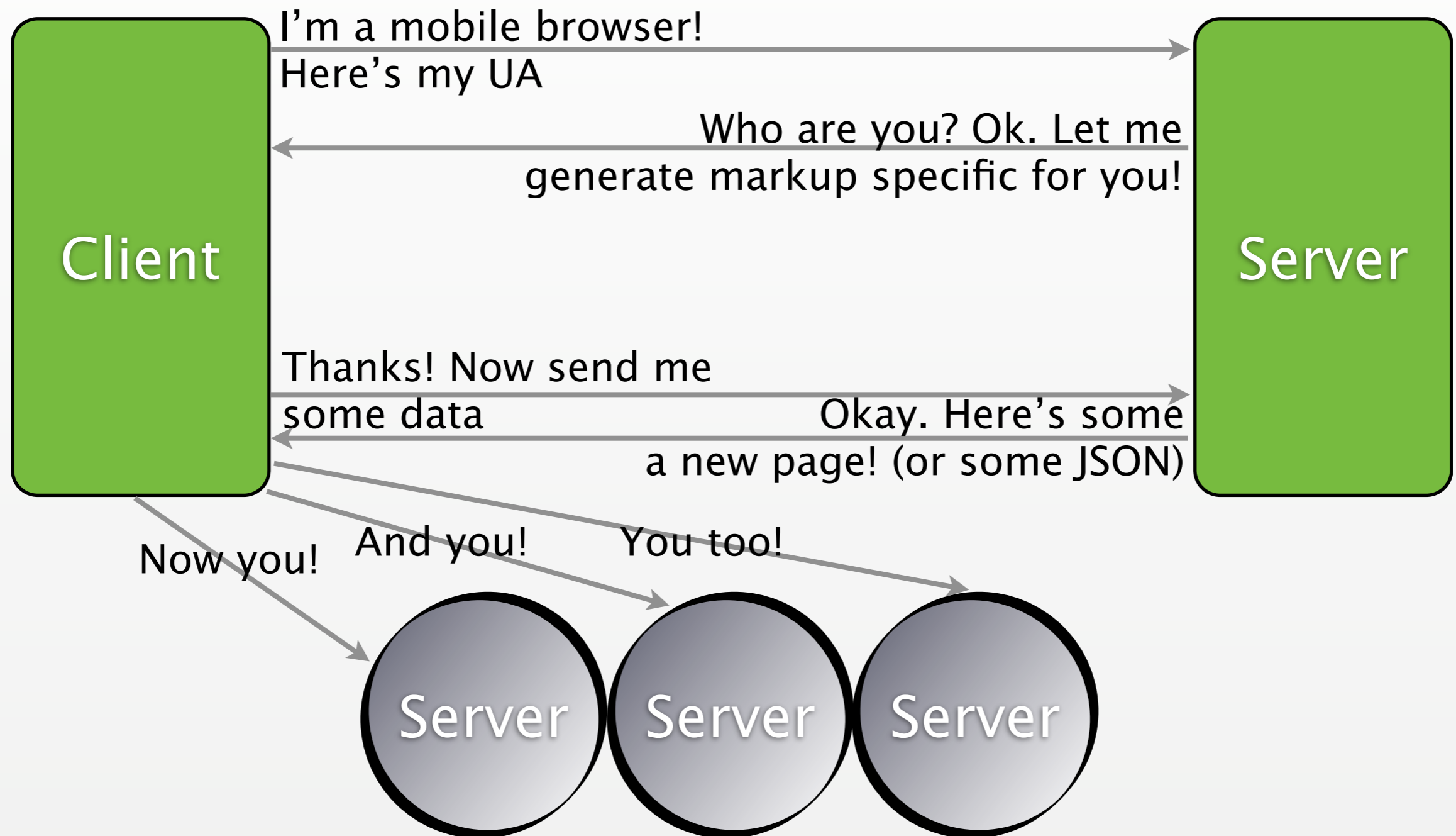
Mobile is Changing Web Architecture



Mobile is Changing Web Architecture



Mobile is Changing Web Architecture



The Problem

Your
Phon

Your
Serve



The Problem

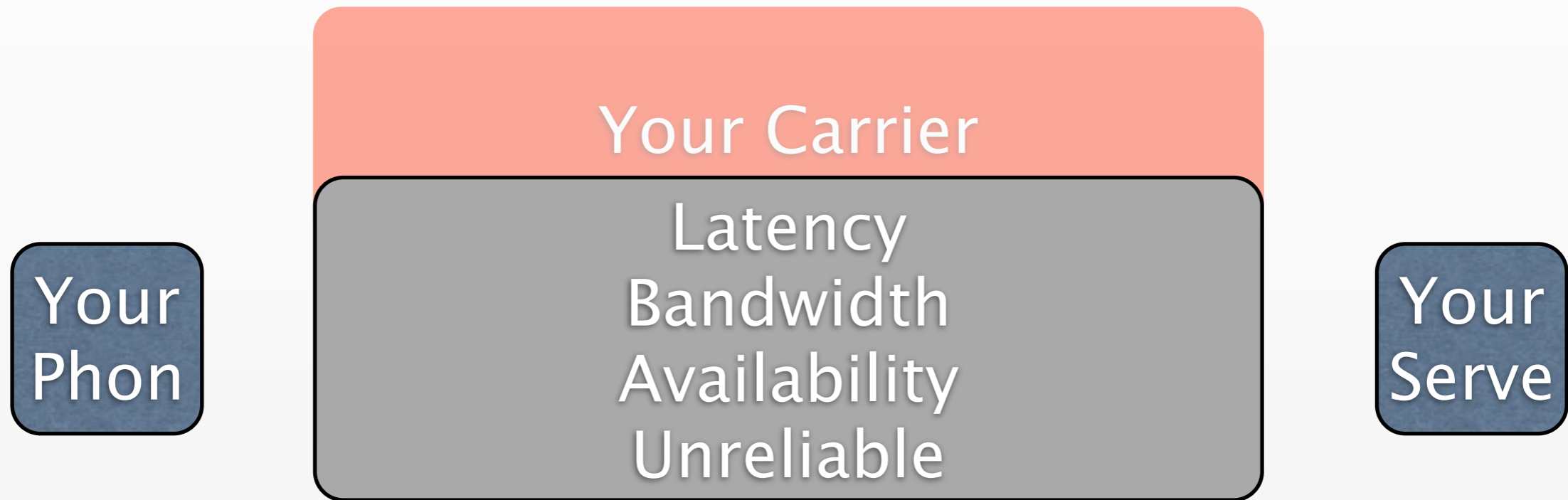
Your
Phon

Your
Carrier

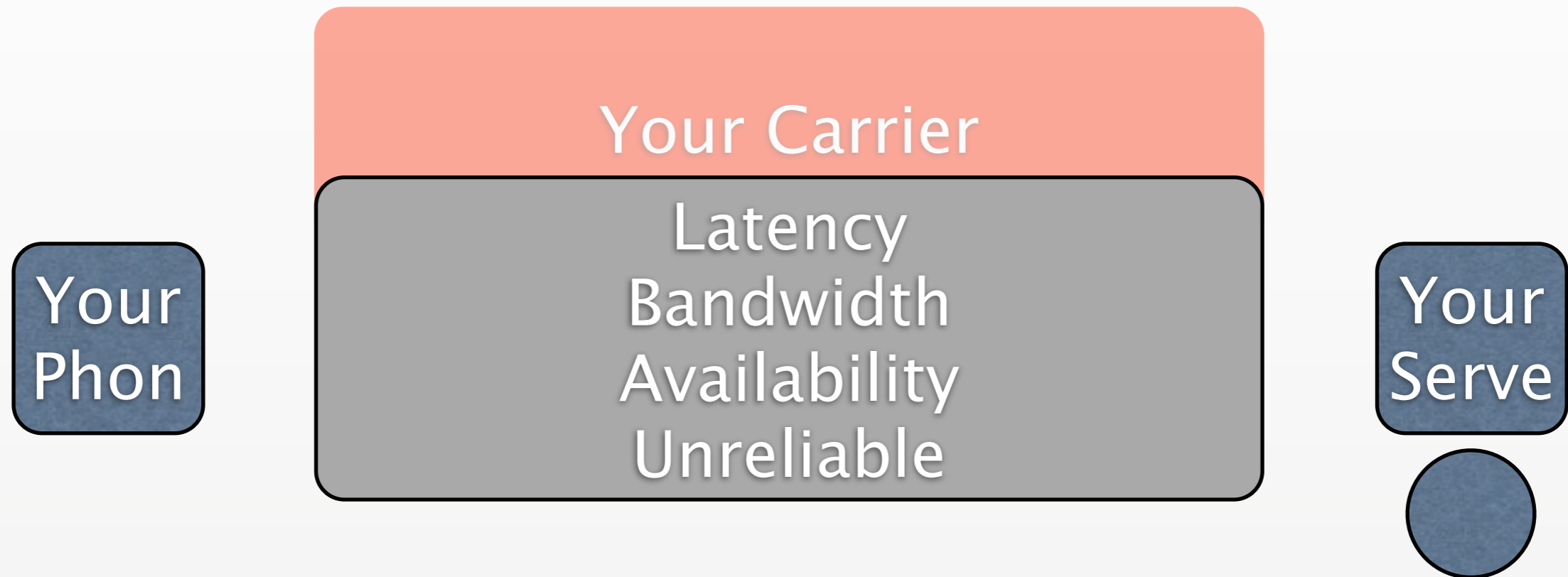
Your
Serve



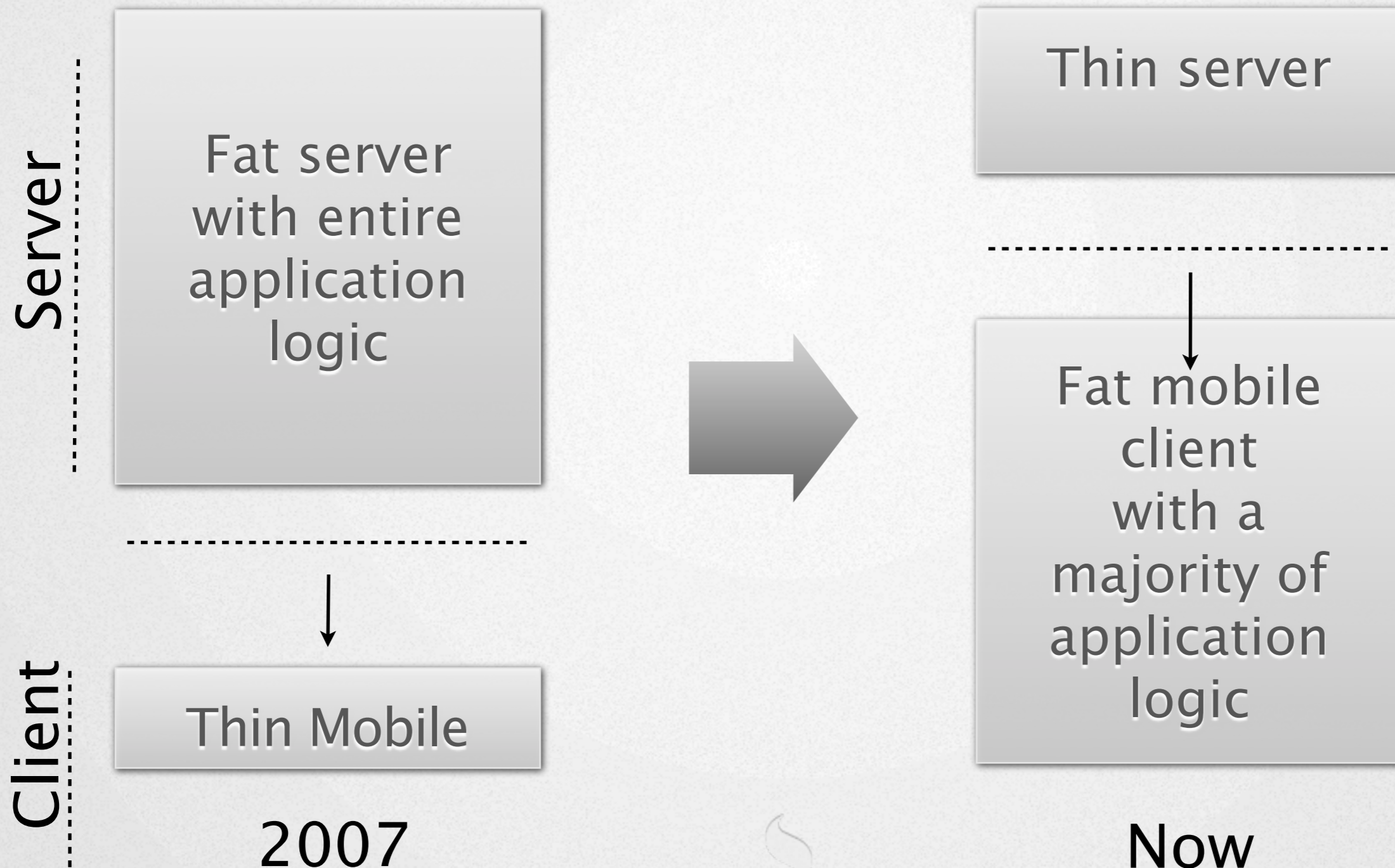
The Problem



The Problem



Mobile: New Architecture



Mobile is Changing Web Architecture

Then

- Servers generated markup specific for the browser
- Clients had JavaScript and CSS specific for the browser
- Clients pulled data from the server that generated the markup

Now

- Servers are thin, browsers are smart(er)
- Client frameworks deal with browser differences
- Clients pull data from many cloud services as well as “app server”



Camera
Location
Contacts
SMS
Orientation
Gyro

WebFont Video Audio Graphics

CSS Styling & Layout

JavaScript

Semantic HTML

File Systems
Databases
App Caches

Workers &
Parallel
Processing

Cross-App
Messaging

HTTP
AJAX
Events
Sockets
SSL

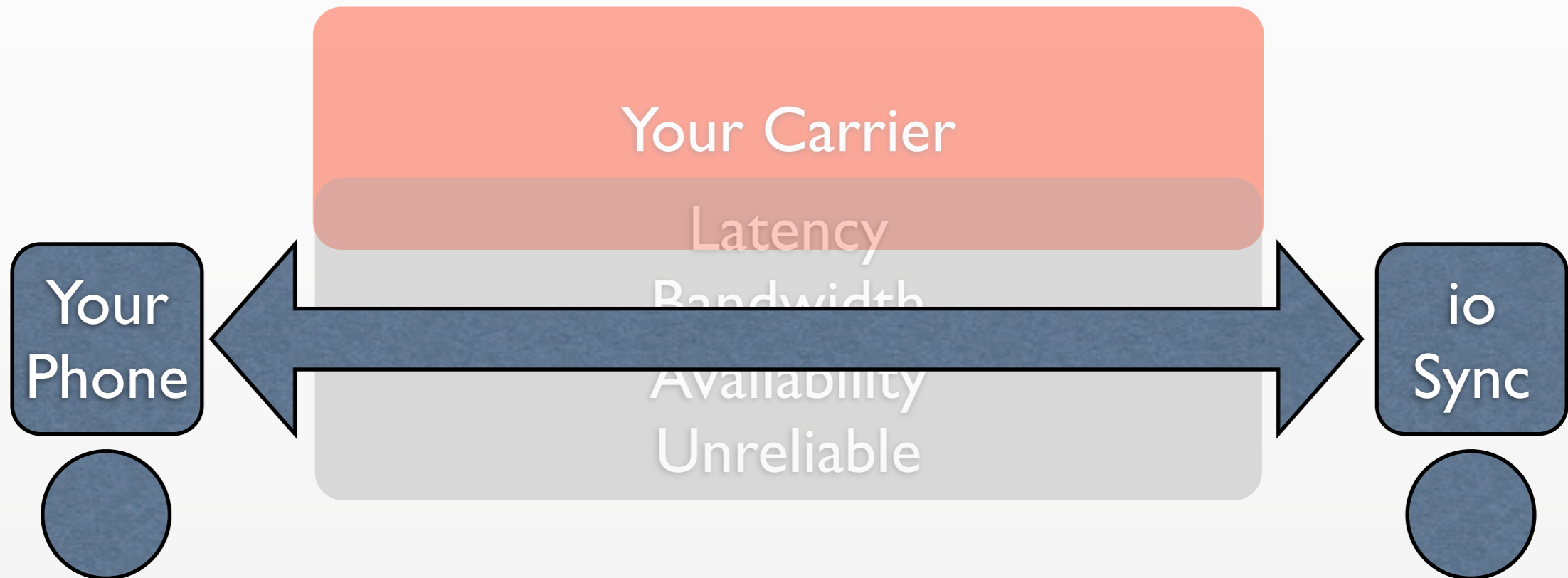


Cloud Solves The Network

- Be offline
- Data is backed up in the cloud
- Multi-master and shared



Cloud Solves The Network



The Mobile Cloud Imperative



Cloud Drivers

- Persistent and growing device diversity: display & storage
- Power optimization
- Bandwidth-aware behavior
- Complex geo-aware apps

Major Features

- Display-aware asset delivery
- Network-aware content delivery
- Peer-to-peer assist
- Geo-fencing, notifications



Sencha.io

Cloud for the
mobile web





Sencha.io



Sencha Designer

Project Ion



Ext JS/GWT

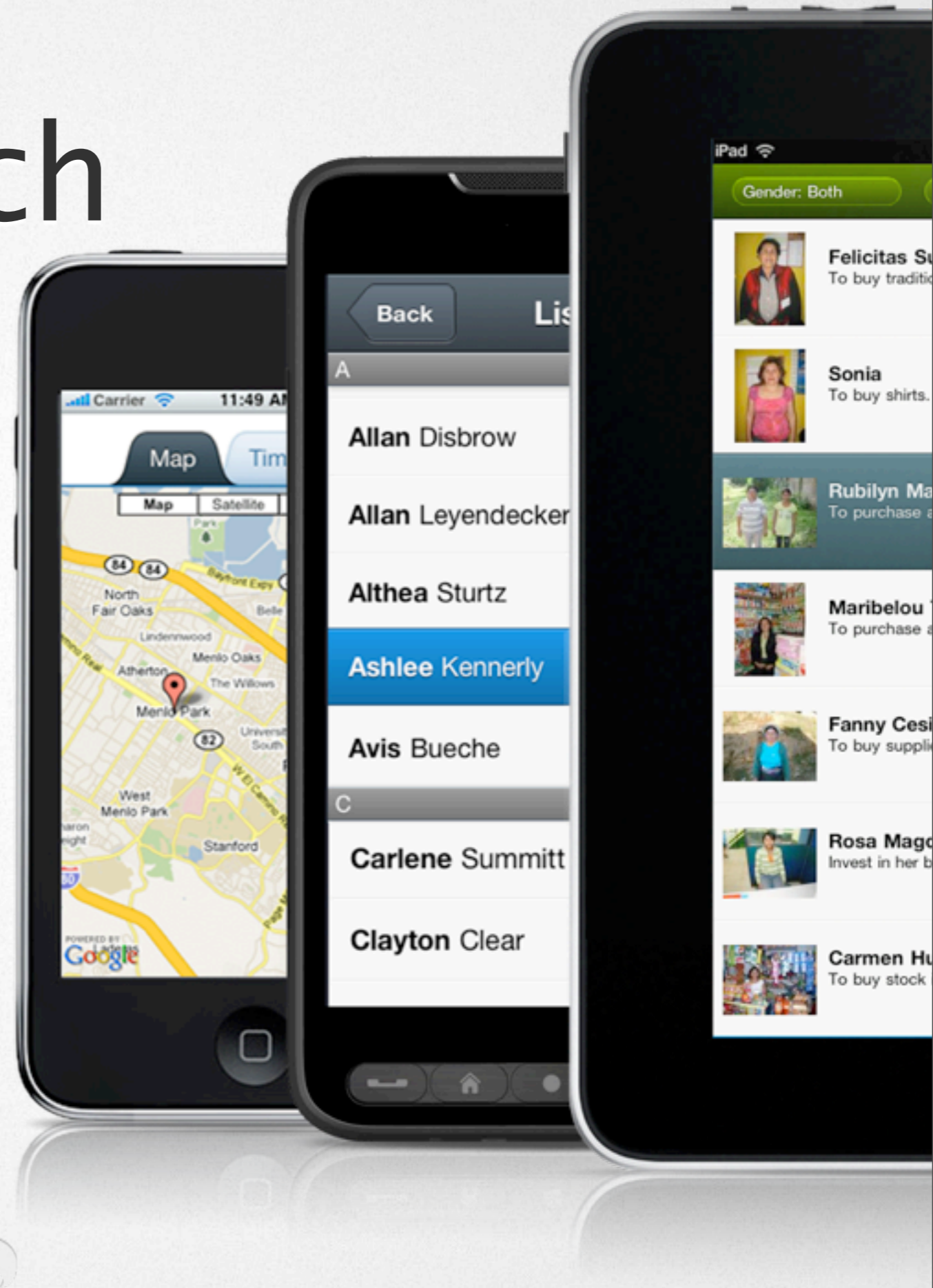
Sencha
Touch



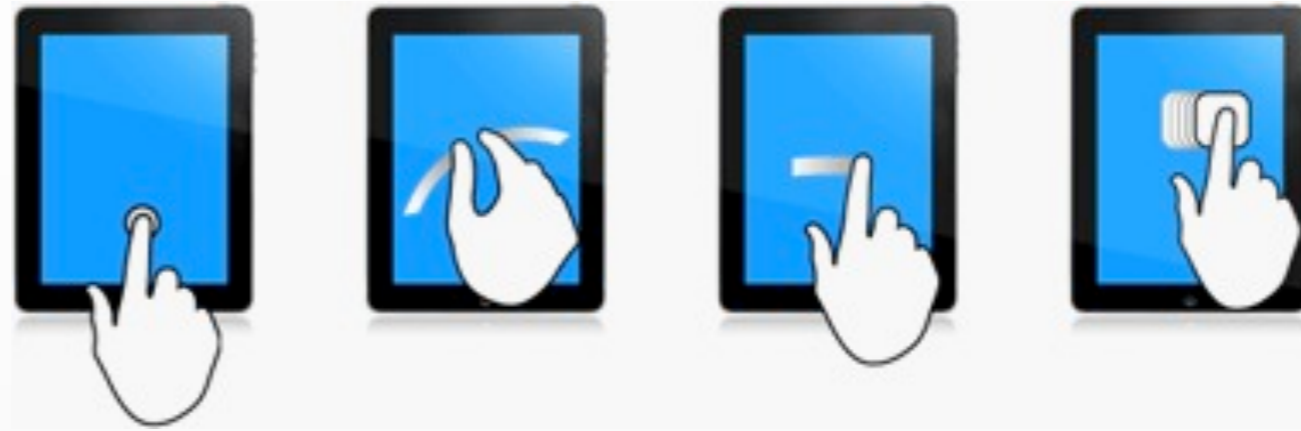
The Sencha Platform

Sencha Touch

- HTML5 mobile framework
- Cross-browser normalization
- Offline storage proxy
- Rich animations, styling, and themes
- Pre-built UI components
- Touch & Orientation







Enhanced Touch Events

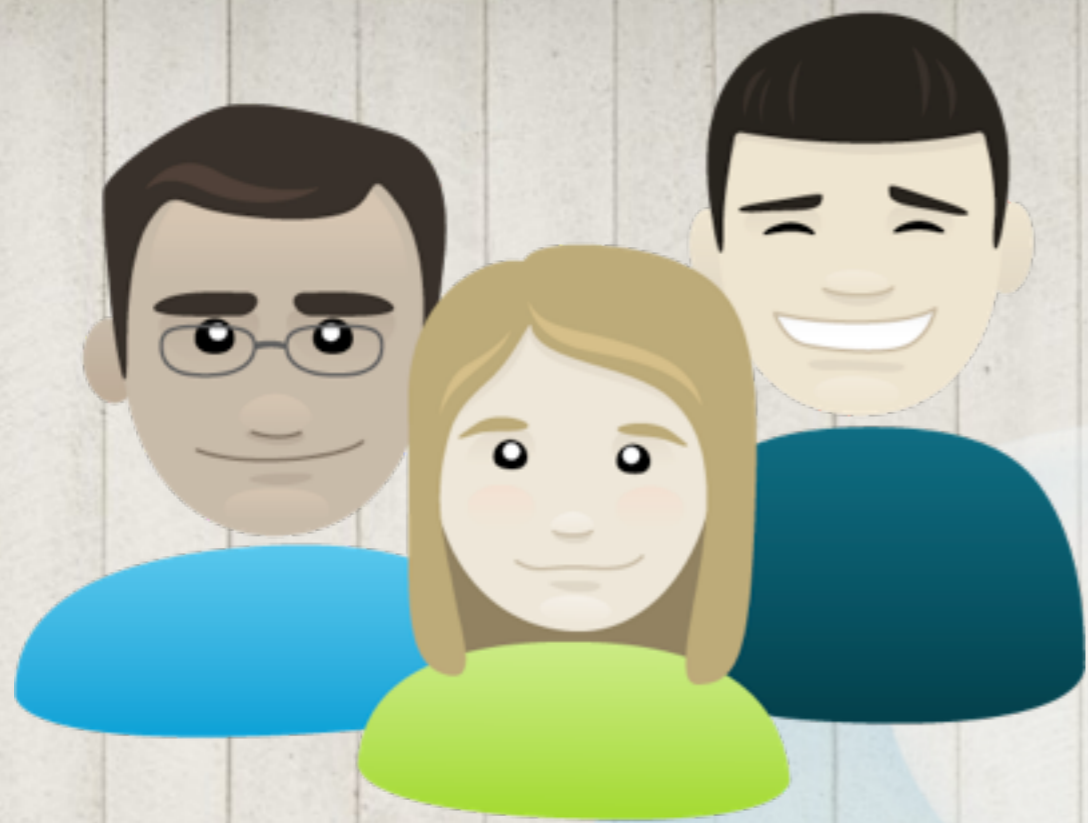
- As the core method of interaction in a touch-based app, robust touch events are vital to the framework's architecture.
- On top of the standard events supported by the browser like touchstart and touchend, we have added a long list of custom events that can be used like tap, double tap, swipe, tap and hold, pinch, and rotate.

Touch Charts

- Built with HTML5 & CSS3 for Sencha Touch
- Cross browser compatible
- Hardware accelerated for optimized performance
- Natural gestures







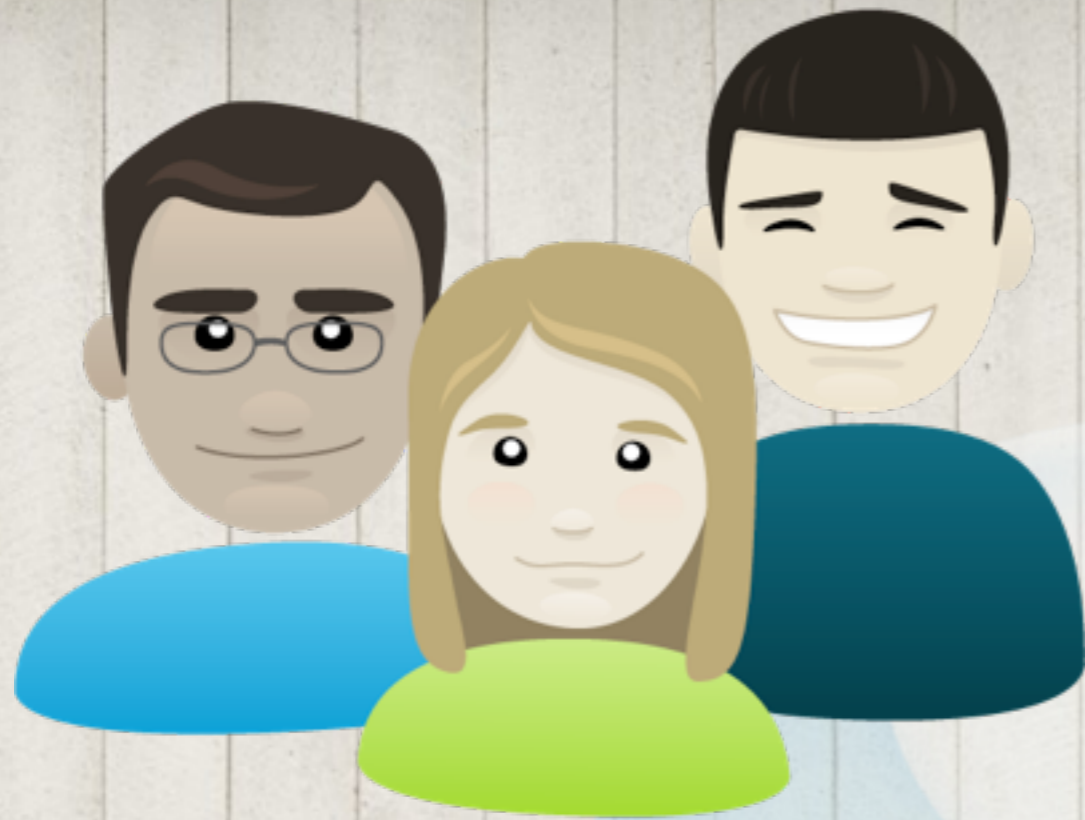
Login



Login



Data



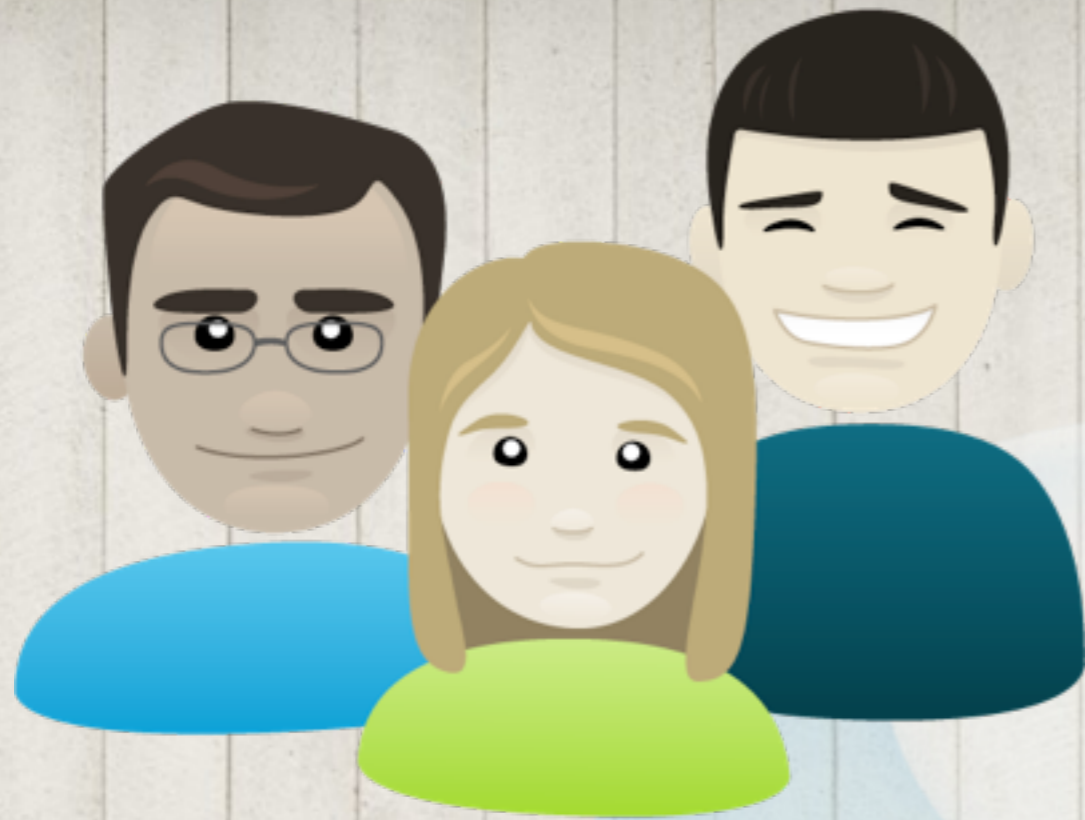
Login



Data



Messages



Login



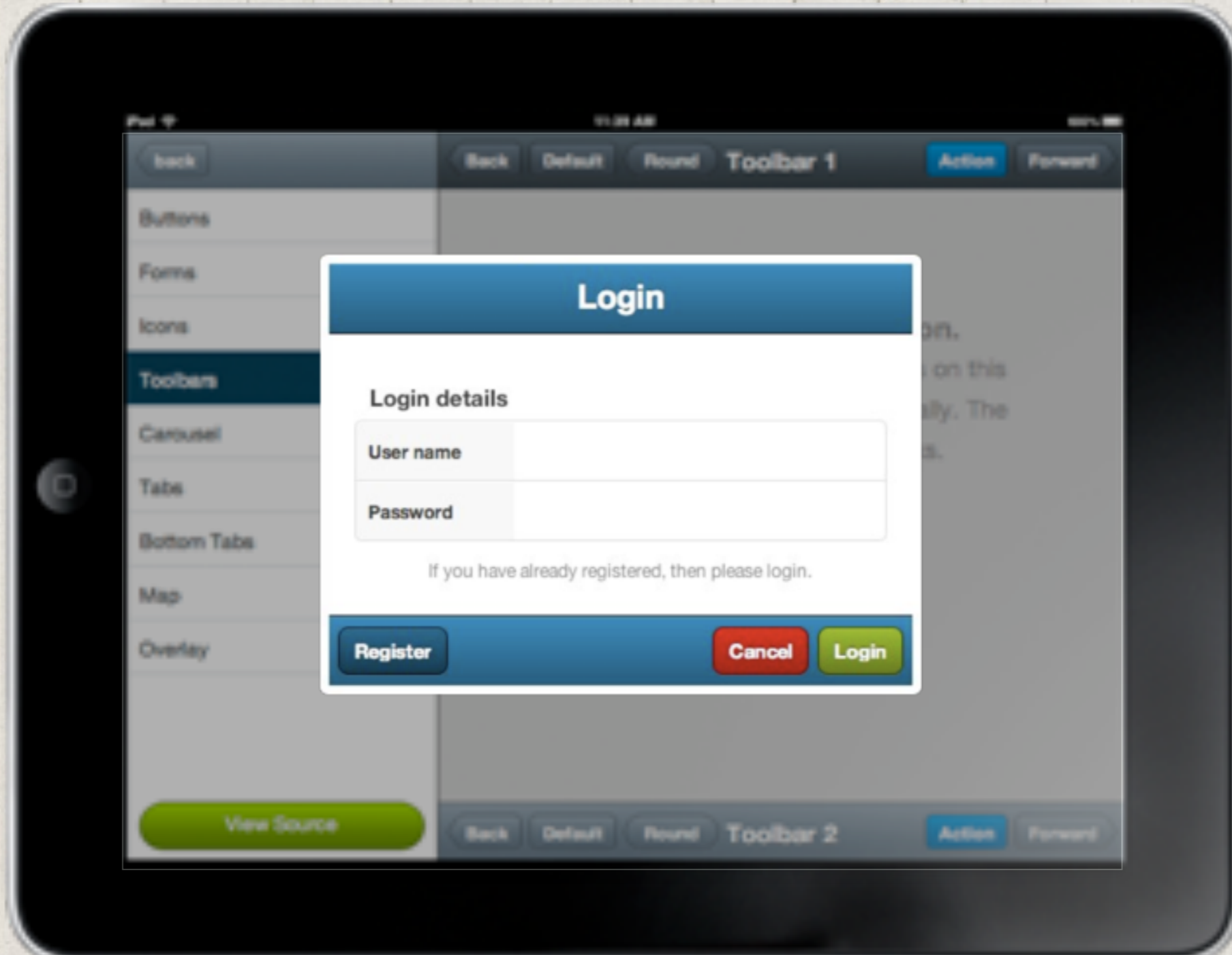
Data



Messages



Deployment



Sencha.io Data



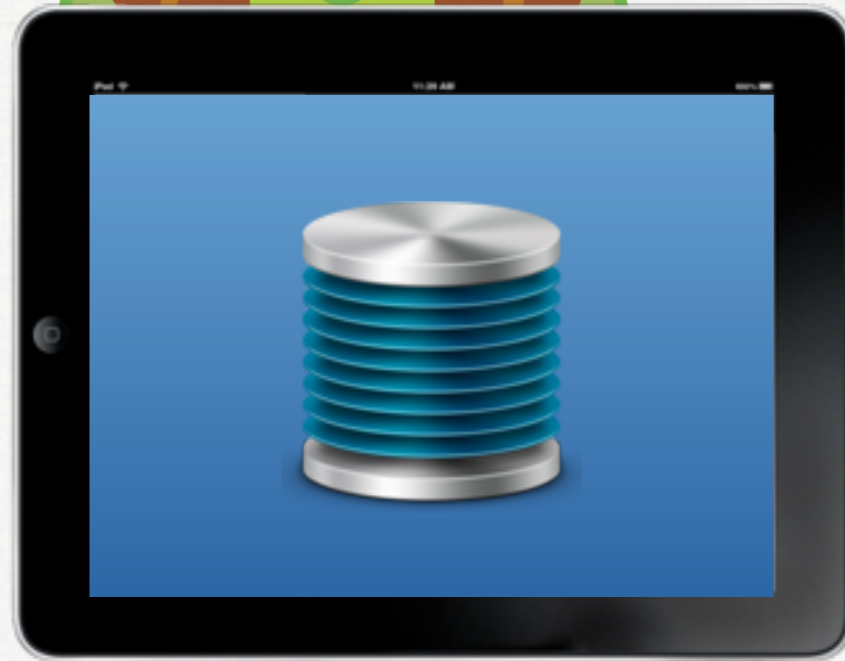
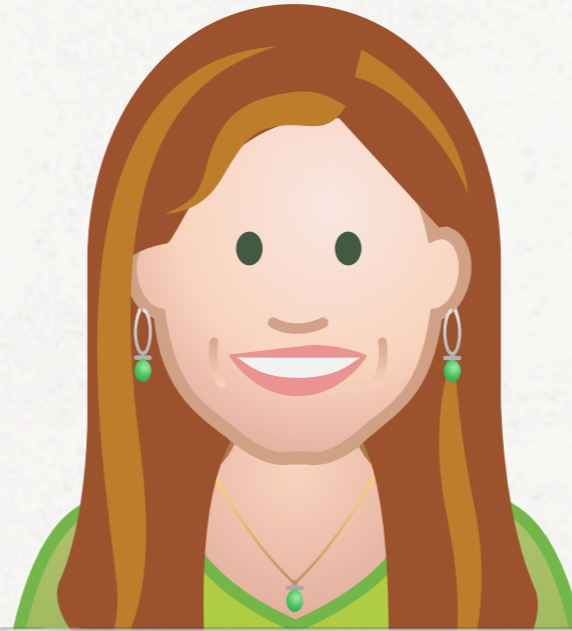


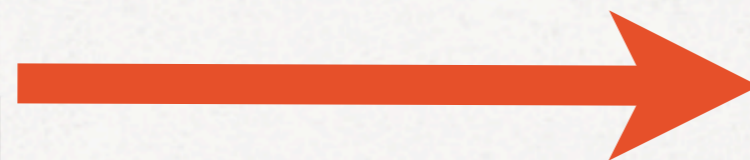
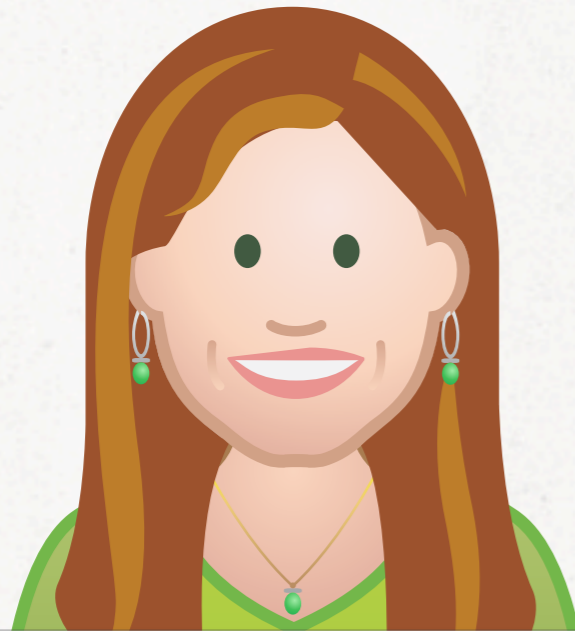


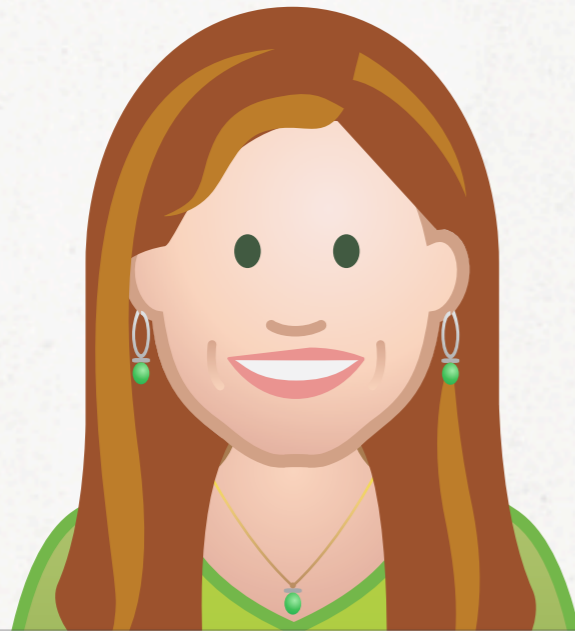


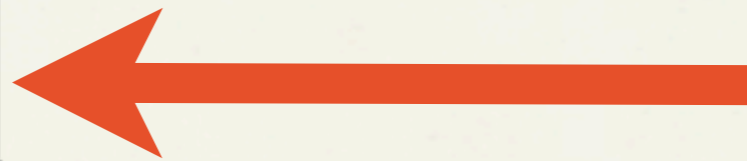










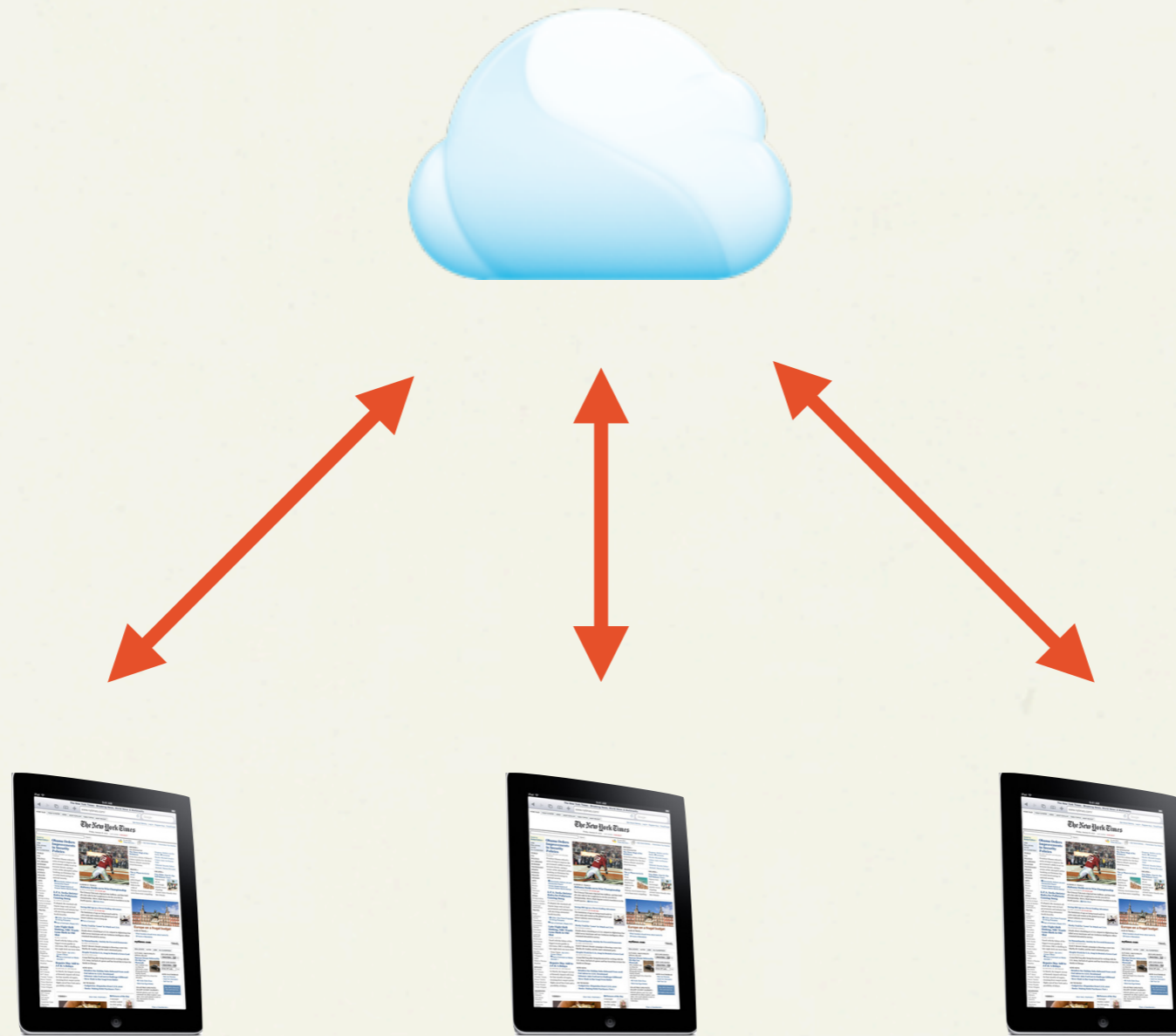




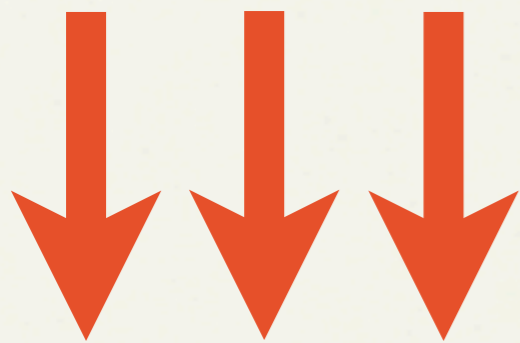
Store of Records

| user | name | shoe-size | ... |
|-------------|-------------|------------------|------------|
| johnm | John | 11 | ... |
| builder7 | Bob | 10,5 | ... |
| janesmith | Jane | 7 | ... |
| ... | ... | ... | ... |

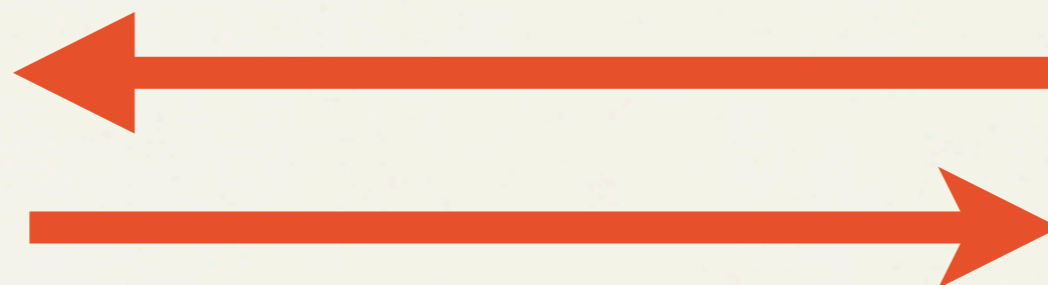
Many Replicas



Offline Operations

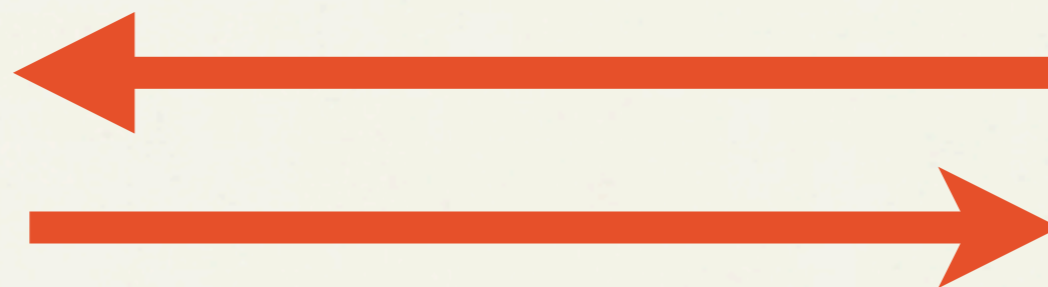


Async Sync



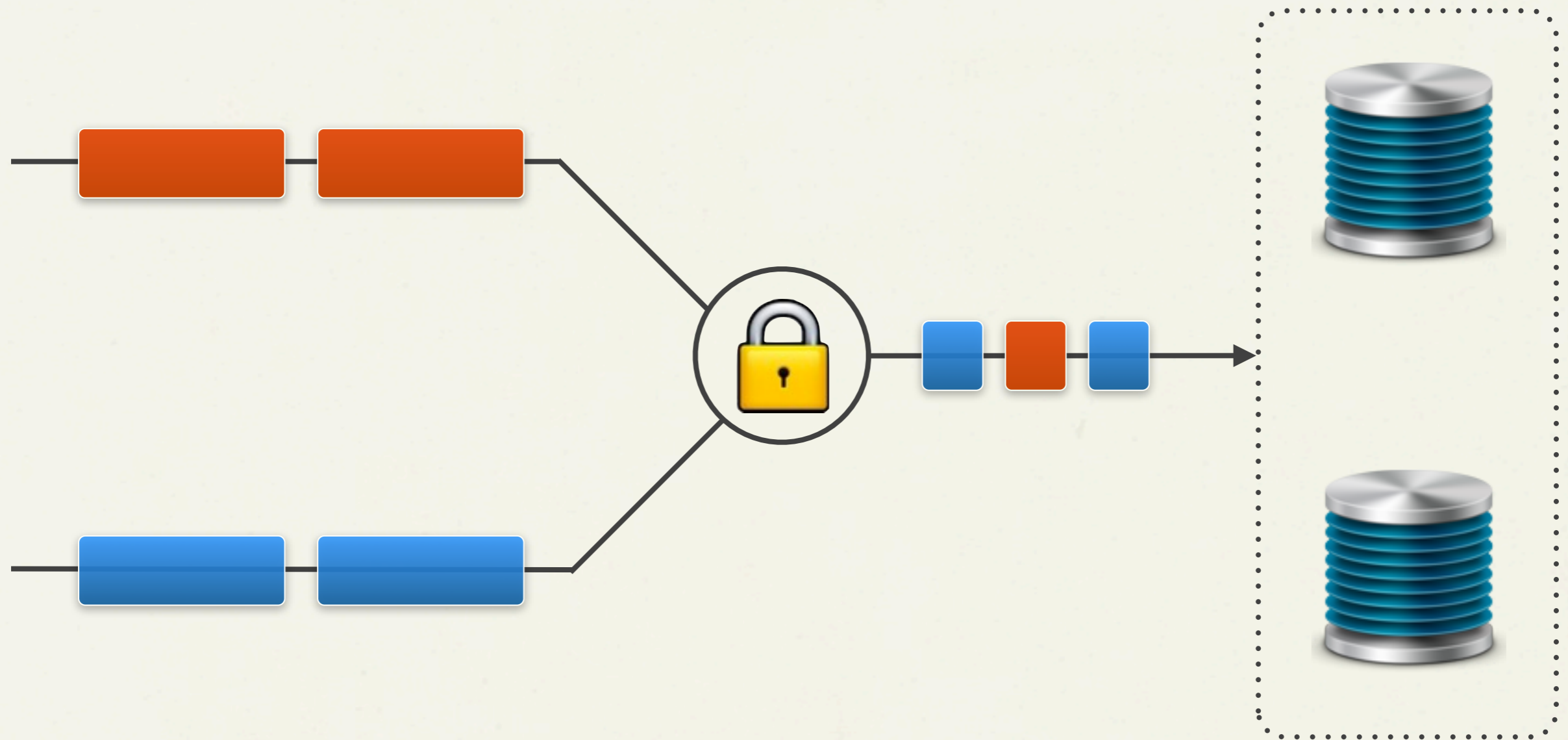
Efficient

$\delta \Delta$

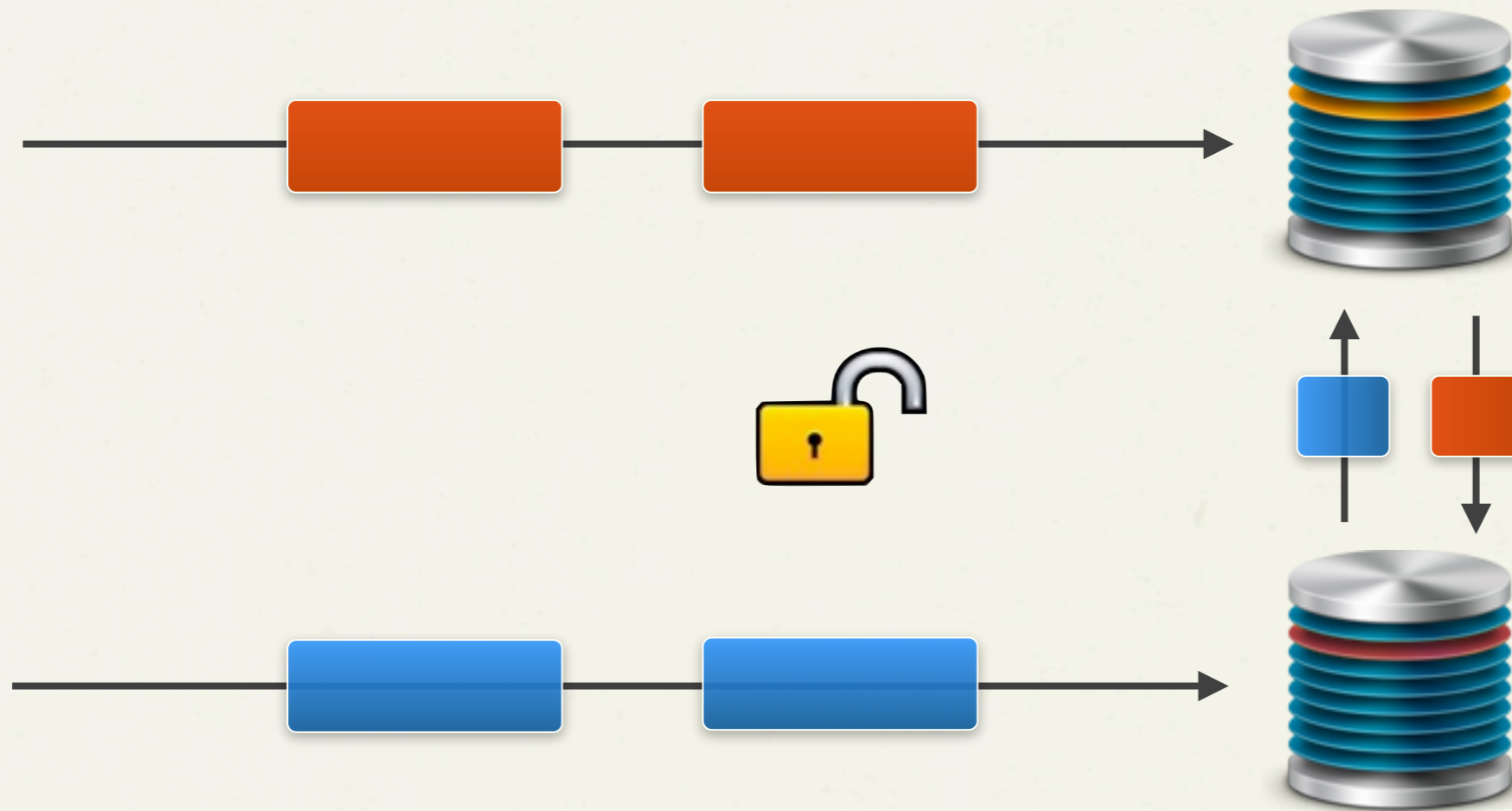


Consistency Models

Strongly Consistent



Eventually Consistent



Conflict Resolution

Cause least surprise for the user.

Cause least surprise for the user.

Preserve as much information as possible.

Cause least surprise for the user.

Preserve as much information as possible.

The policies are arbitrary, ...

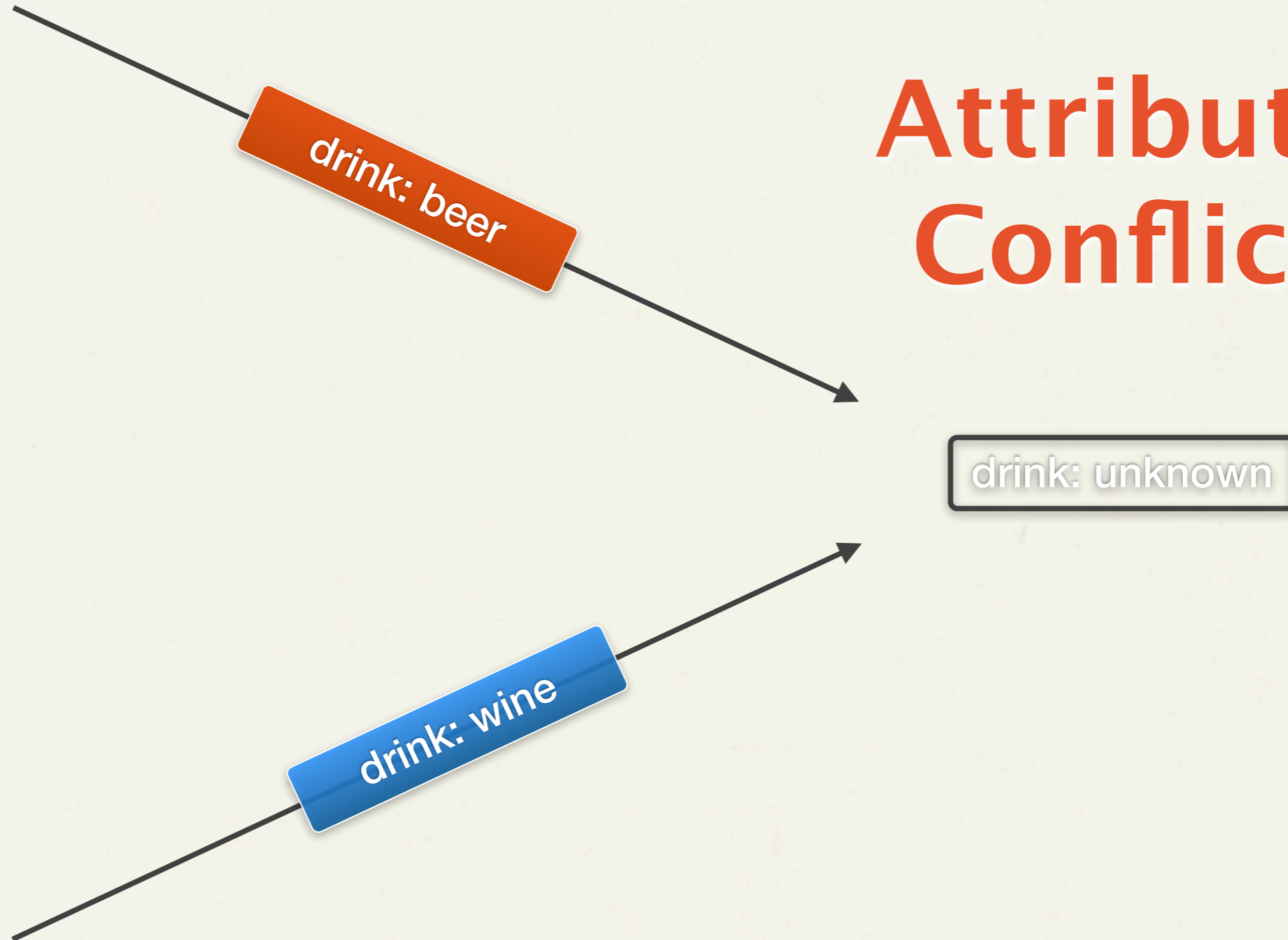
Cause least surprise for the user.

Preserve as much information as possible.

The policies are arbitrary, ...

... but consistently applied, everywhere.

Attribute Conflict





drink: beer



drink: wine

Resolved

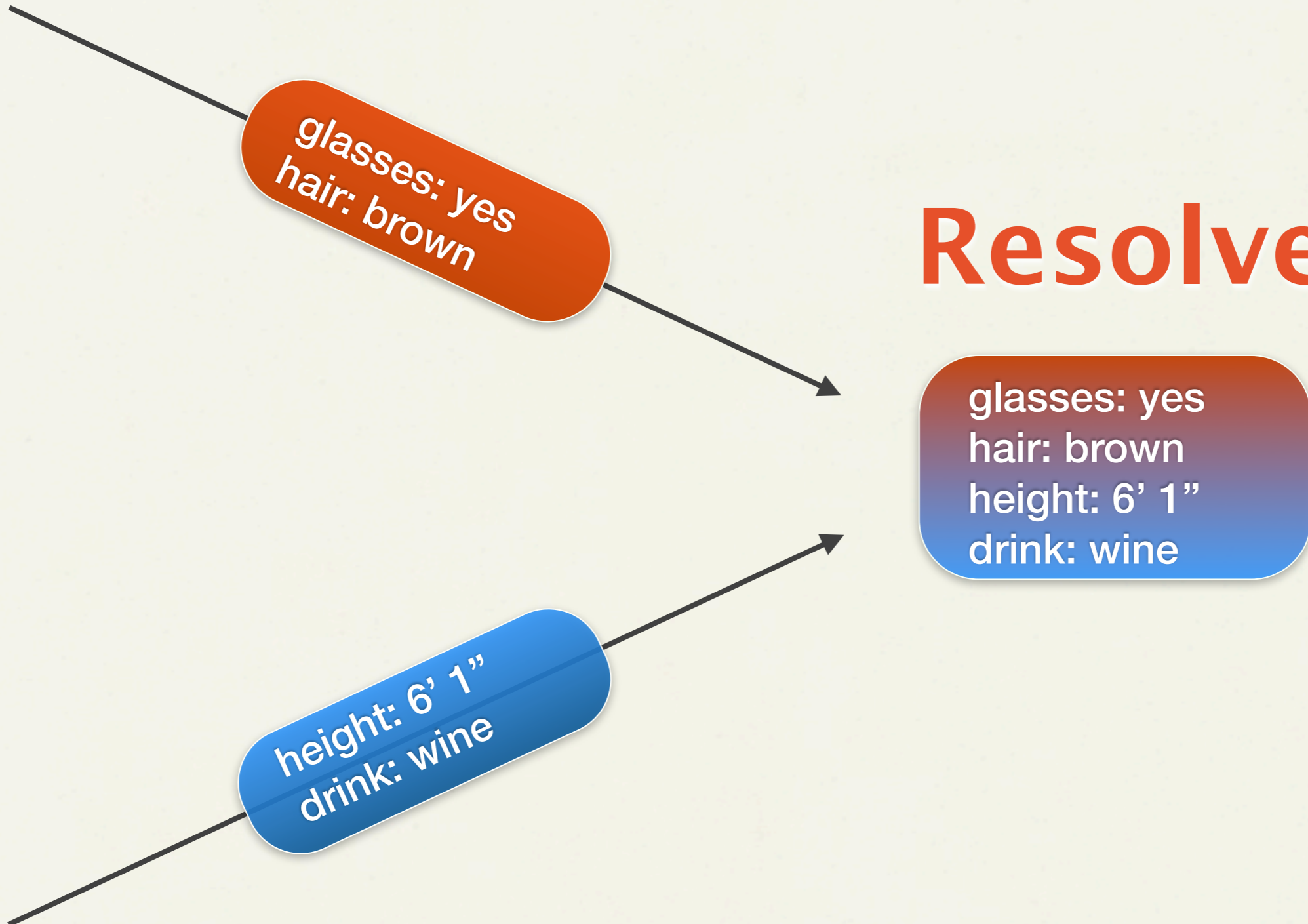
drink: beer
drink: wine

Object Conflict

glasses: yes
hair: brown

Unknown

height: 6' 1"
drink: wine



Resolved

Design Matters

sencha-touch.js

sencha-io.js

sencha-touch.js

Model Definition

```
Ext.define("Contact", {extend: "Ext.data.Model",
  fields: [
    {name: "firstName", type: "string"},
    {name: "lastName", type: "string"},
    {name: "email", type: "string"},
  ],
  proxy: {
    id: 'contacts',
    type: 'syncstorage'
  }
});
```

Store Creation

```
Ext.create('Ext.data.Store', {  
    model: 'Contact'  
});
```

Ext.Store

Ext.Store

Ext.io.SyncStorage

Ext.Store

Ext.io.SyncStorage

LocalStorage



Ext.Store

Ext.io.SyncStorage

LocalStorage

SyncProxy

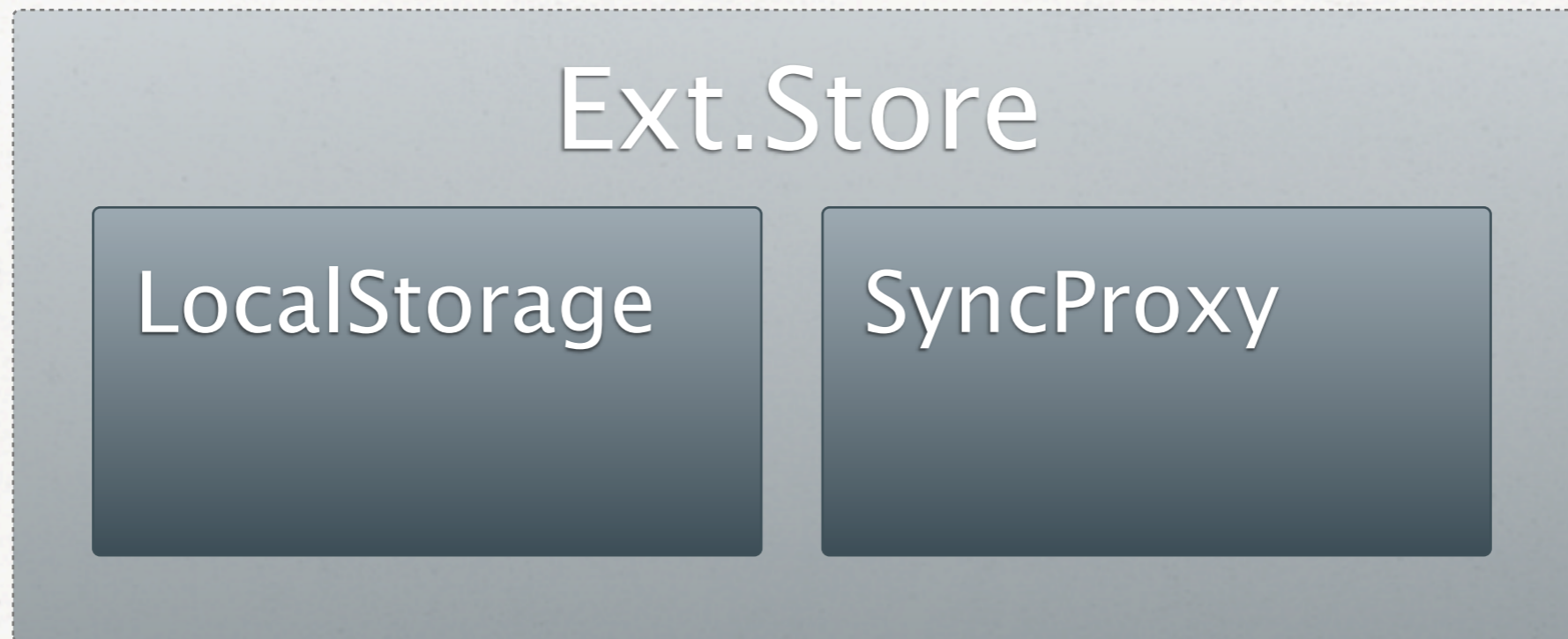


Ext.Store

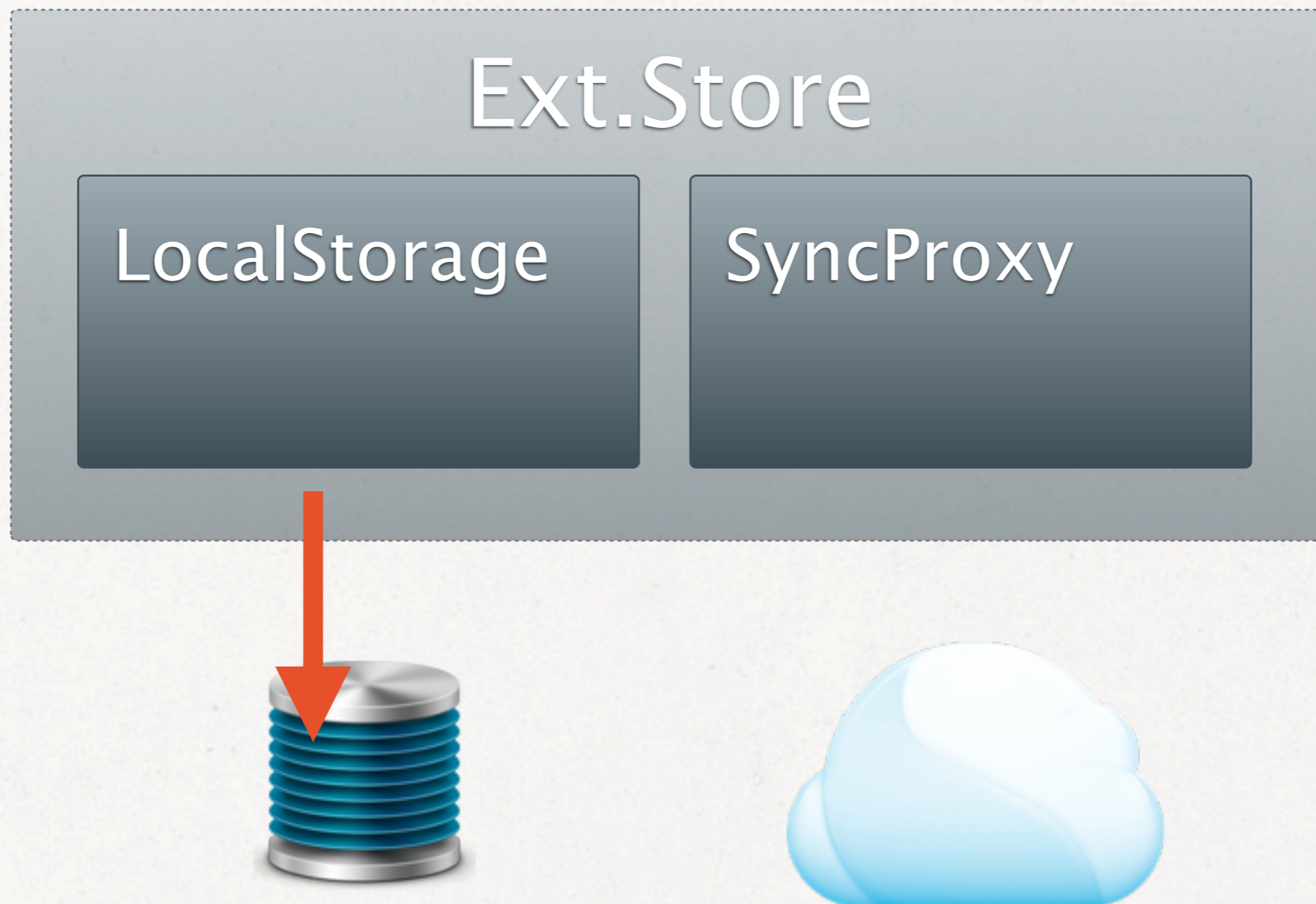
Ext.Store

```
create()  
read()  
update()  
destroy()  
sync()  
clear()
```

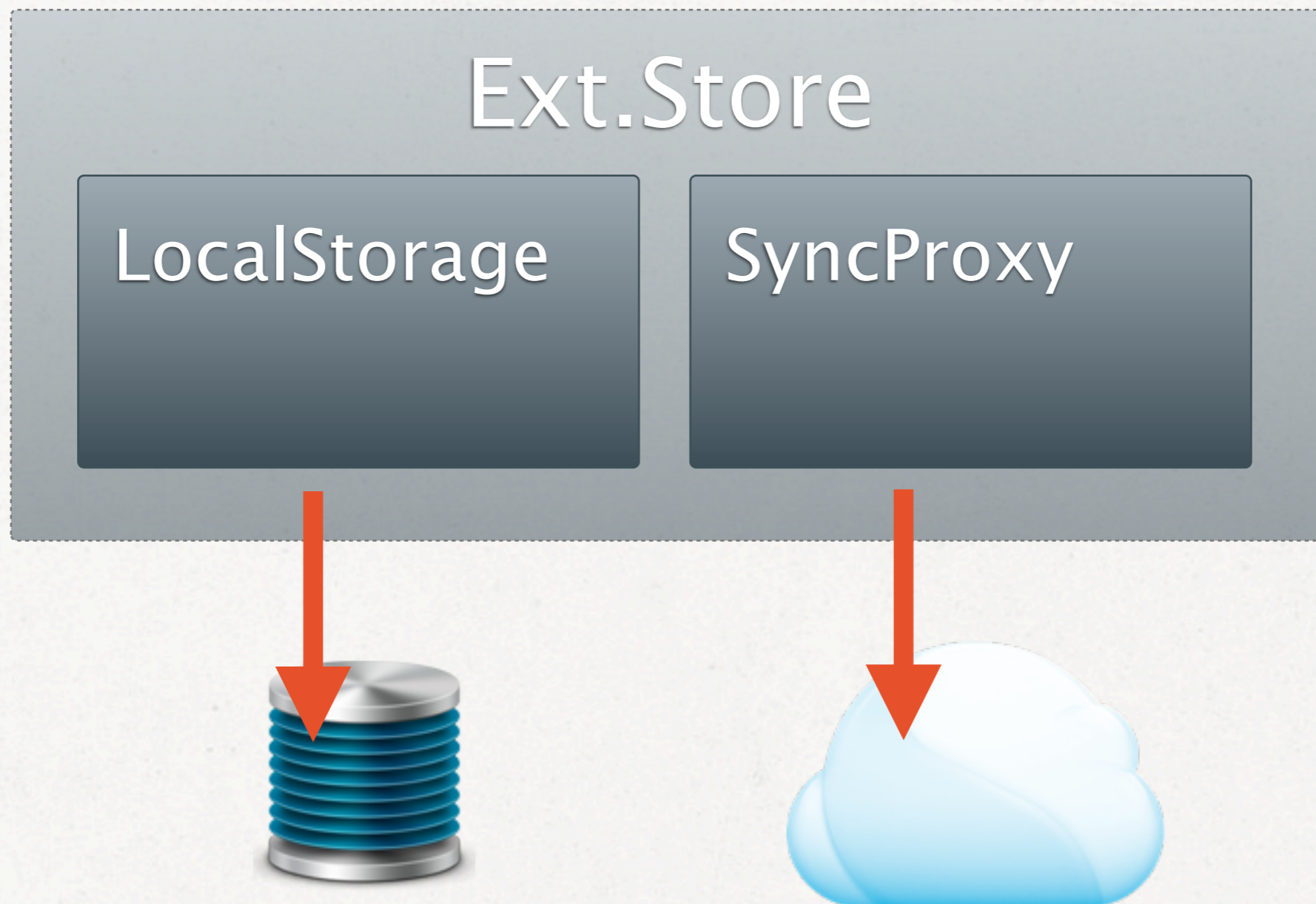
sync()



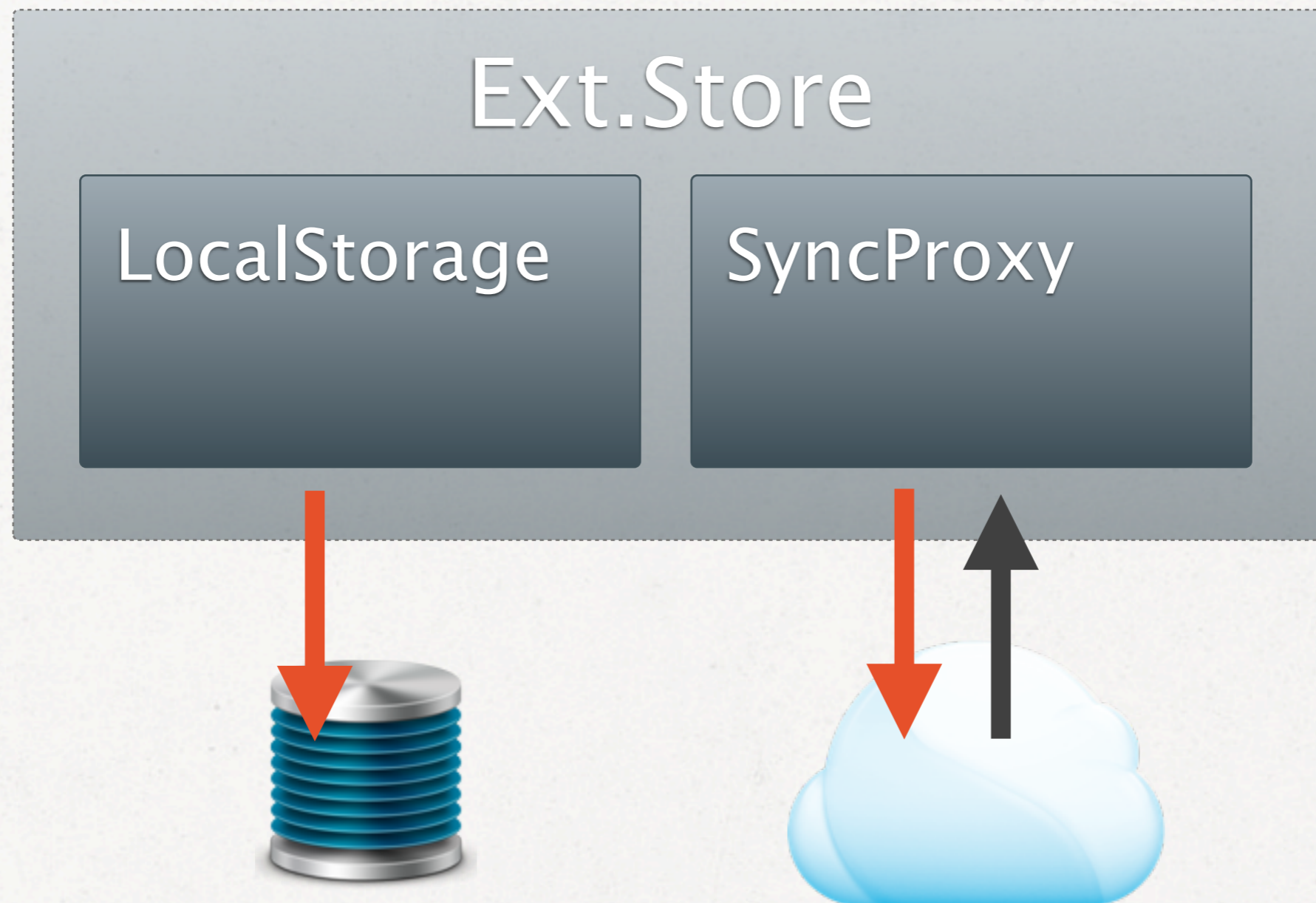
sync()



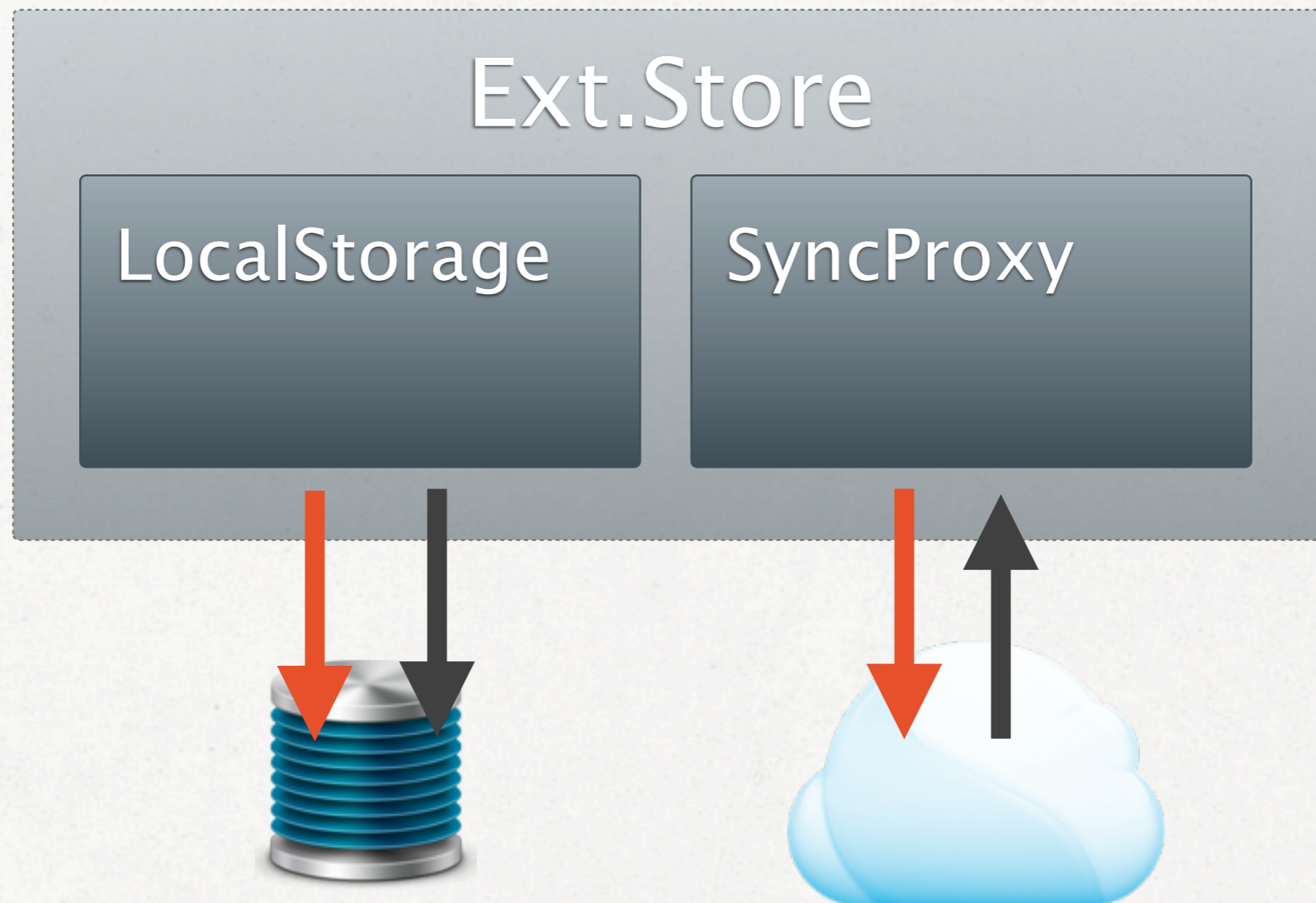
sync()



sync()

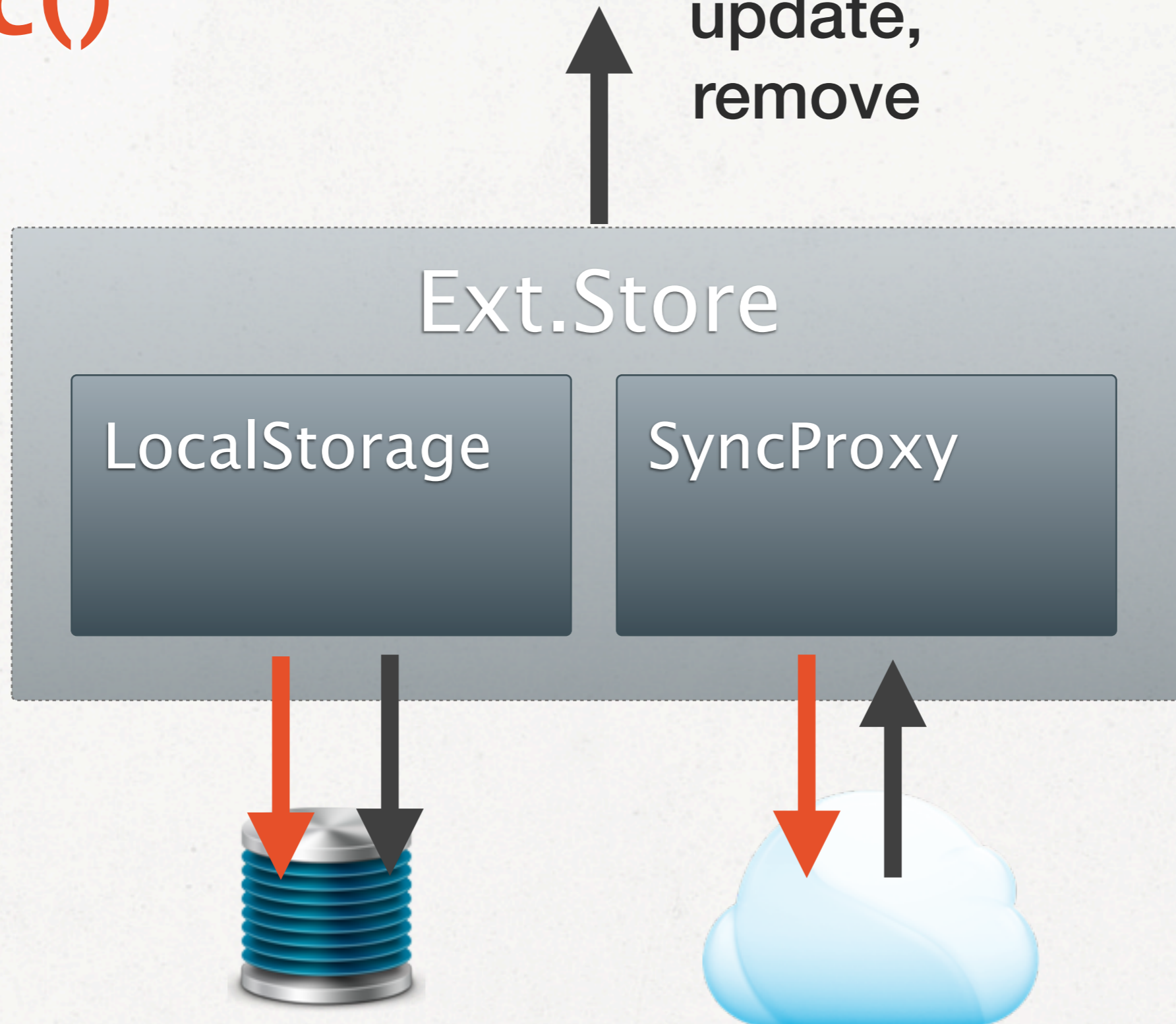




sync()



sync()

add,
update,
remove



Contacts  

B



Monroe Bode

C

Krystina Cartwright

H

Olen Hintz

Contacts  

B

Monroe Bode

C

Krystina Cartwright

H

Olen Hintz

W

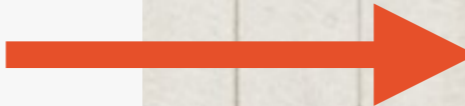
Nicolas Wolff

Contacts ↻ +

B
Monroe Bode

C
Krystina Cartwright

H
Olen Hintz



Contacts ↻ +

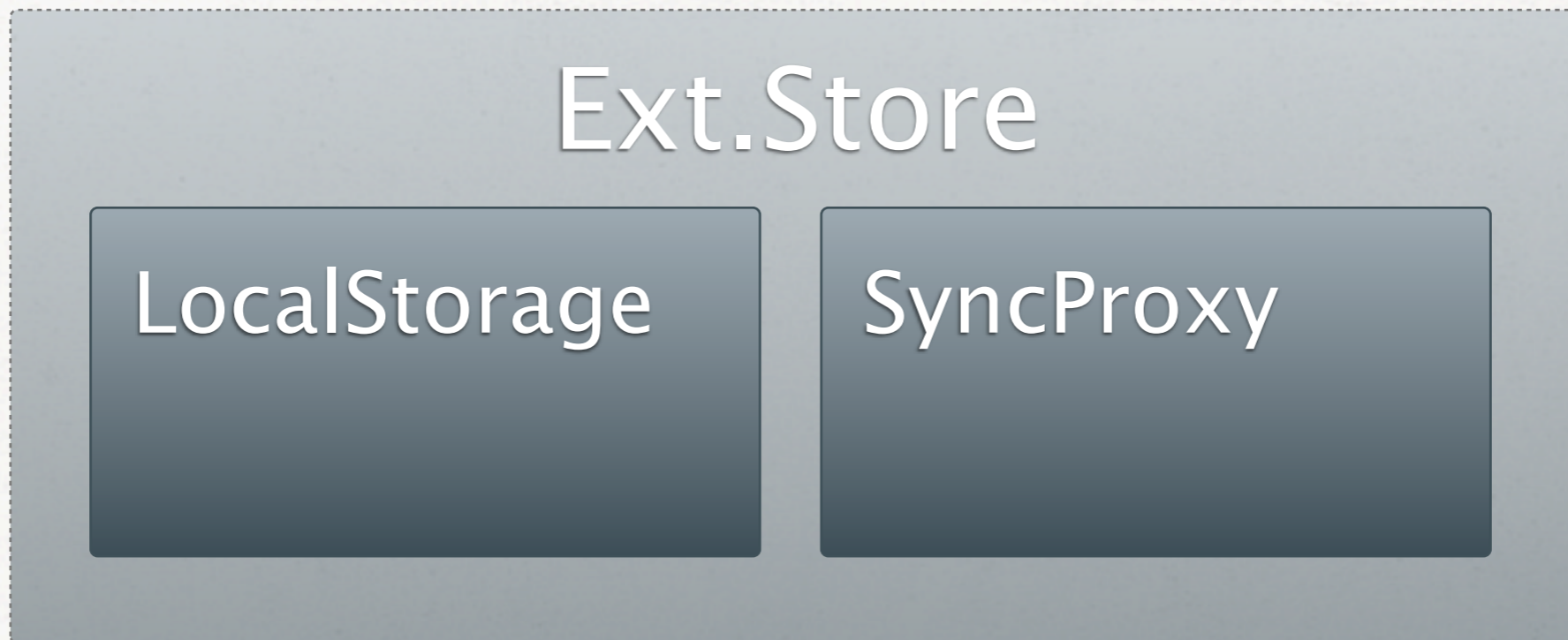
B
Monroe Bode

C
Krystina Cartwright

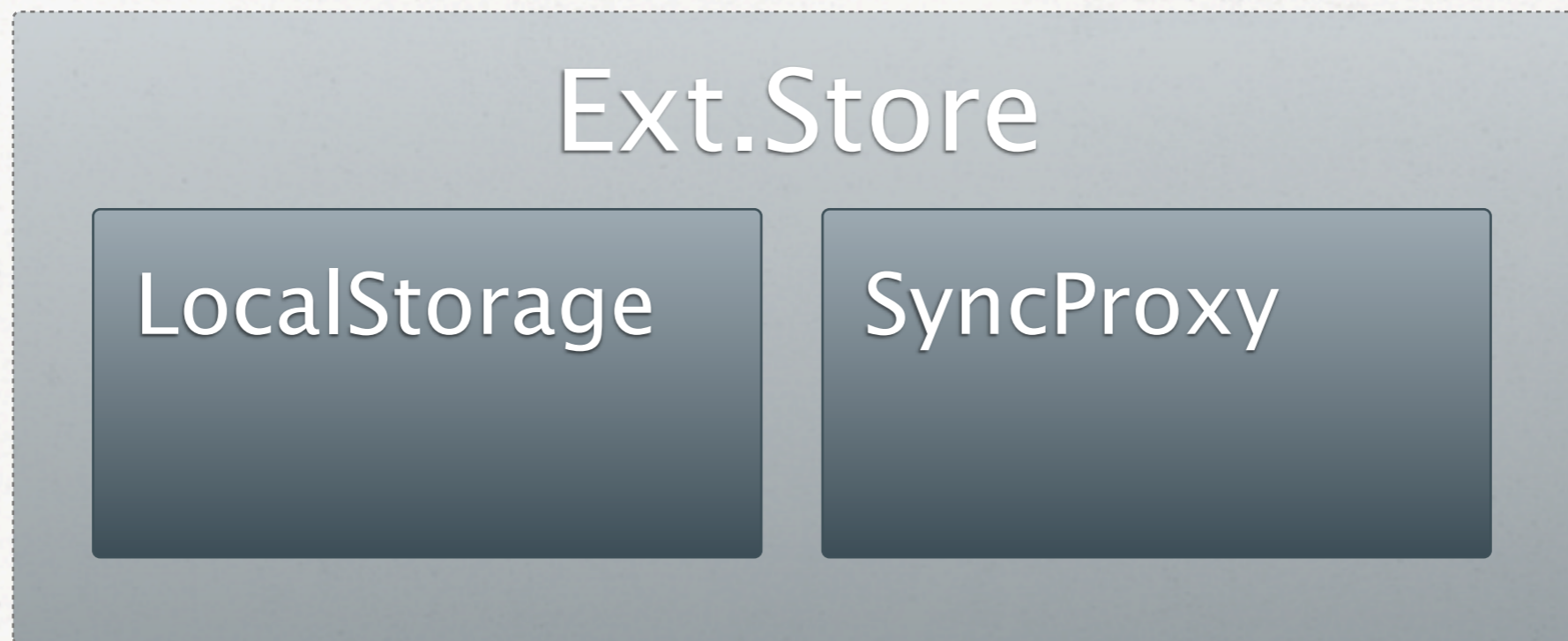
H
Olen Hintz

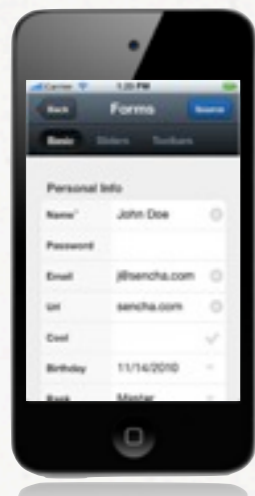
W
Nicolas Wolff

clear()



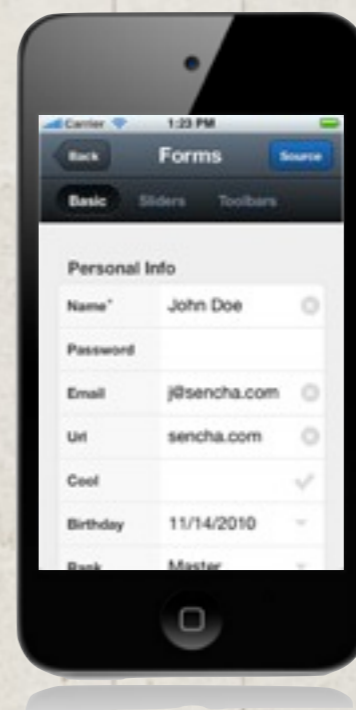
clear()





Sencha.io Messages









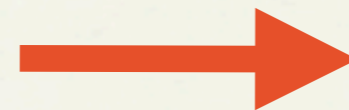
Sender could be offline...

Receiver could be offline...

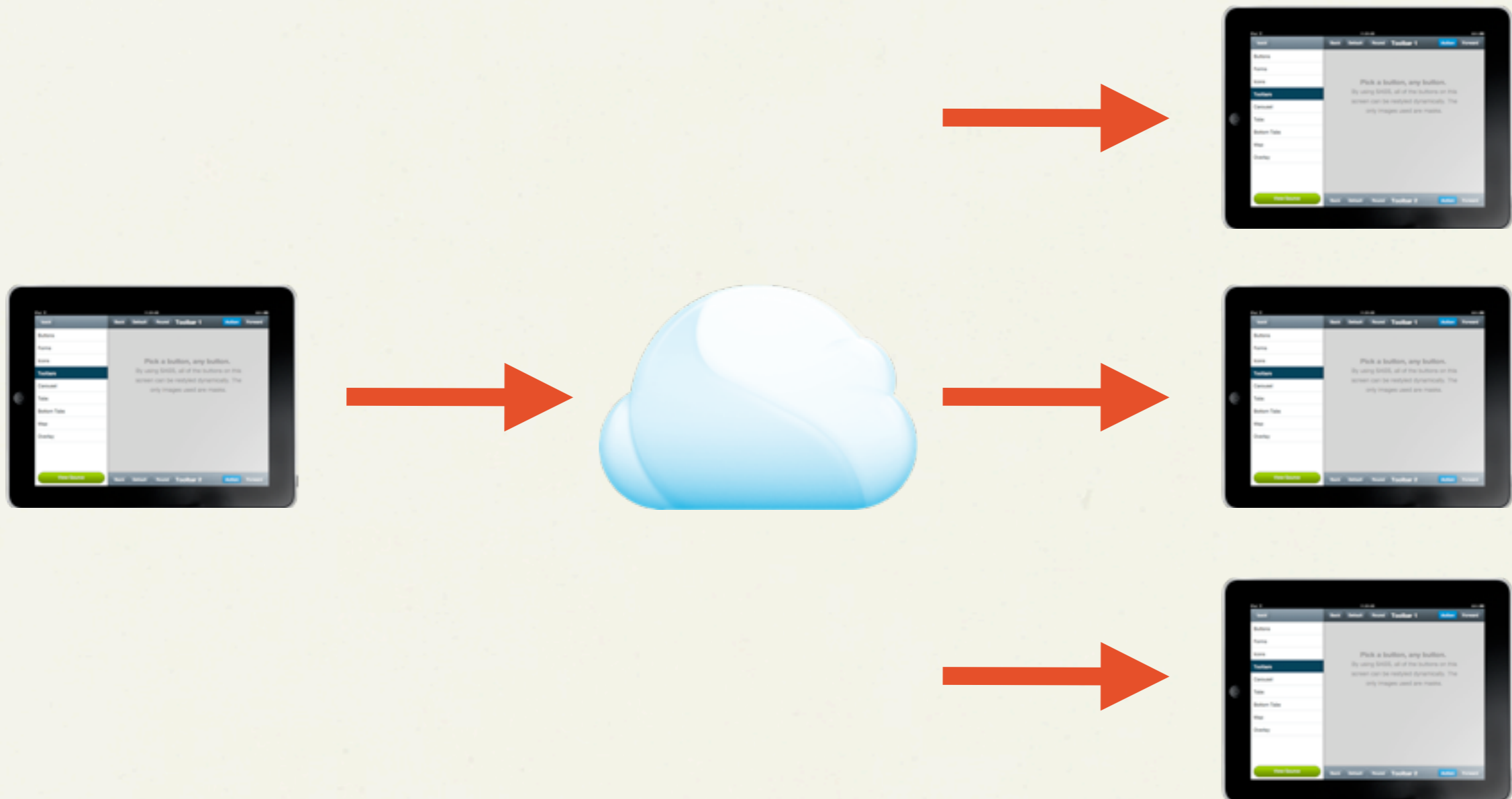




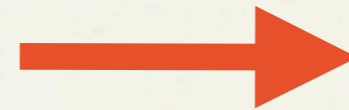
One to One



One to Many



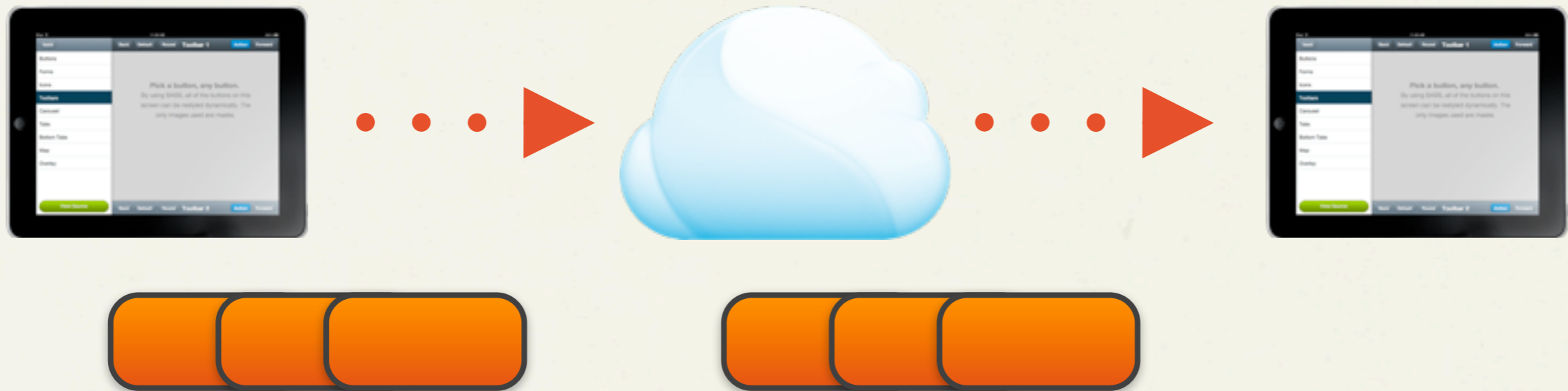
Reliable



Reliable



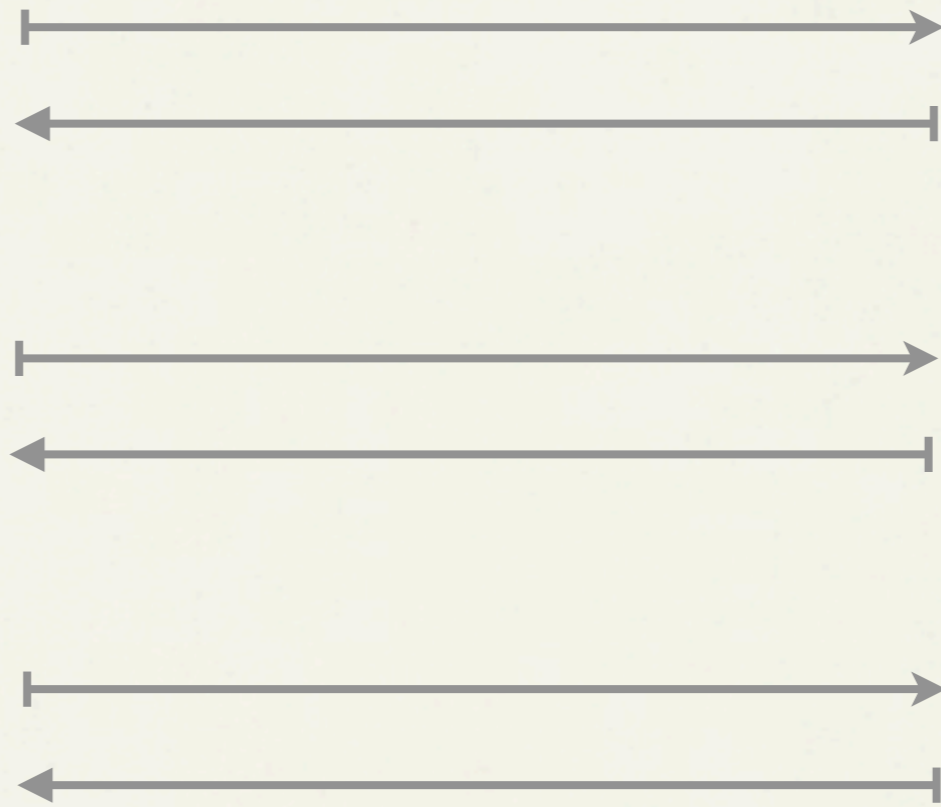
Reliable



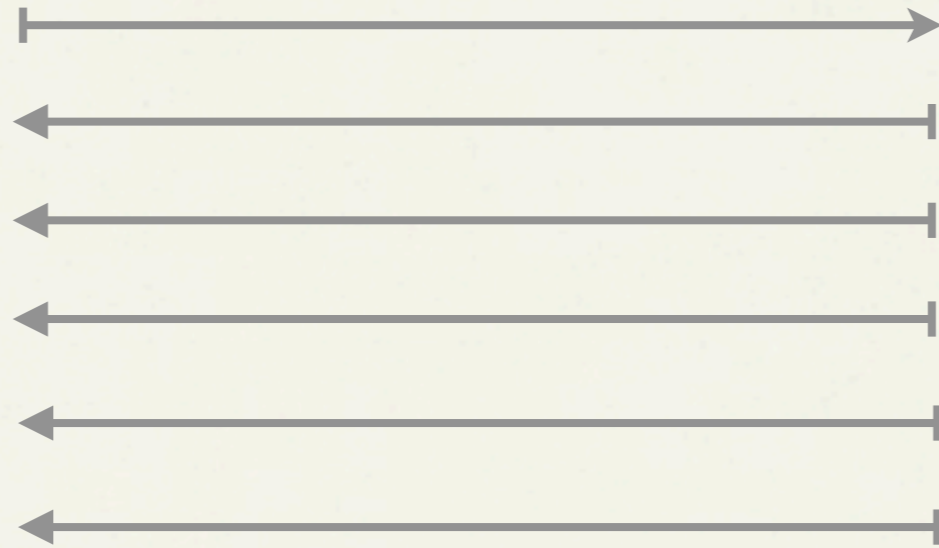
Protocol



Sync Poll



Async Push



How?

sencha-touch.js

sencha-io.js

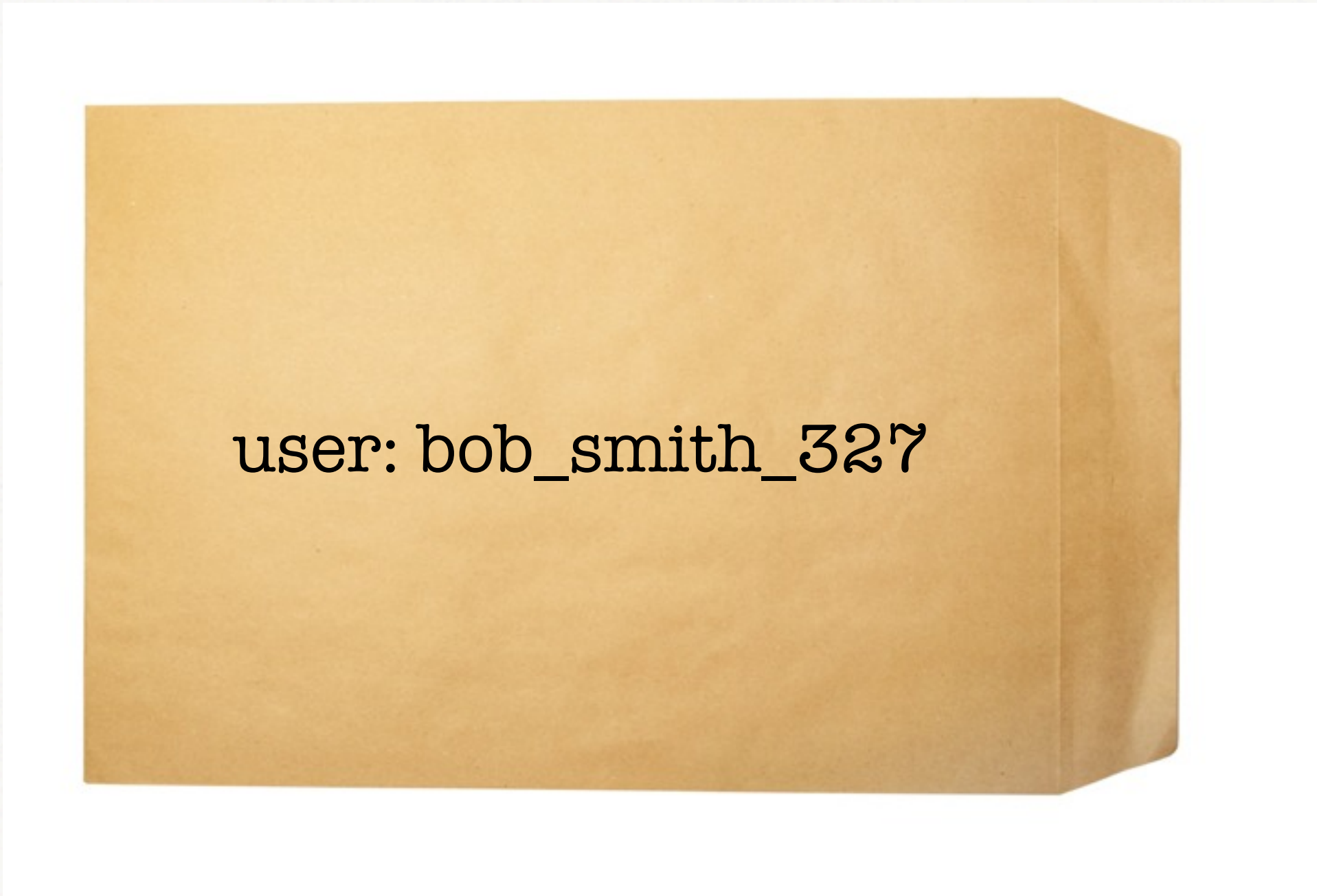
sencha-touch.js

```
{
```

```
  card: 'ace',
```

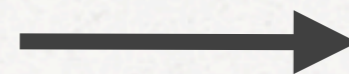
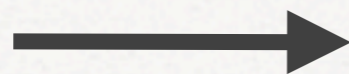
```
  suit: 'spades'
```

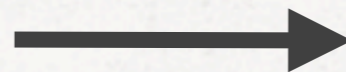
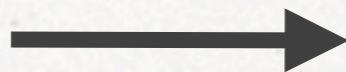
```
};
```



user: bob_smith_327

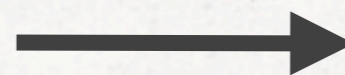
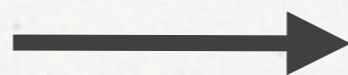
One to One





Ext.io.App

`findDevices(query)`

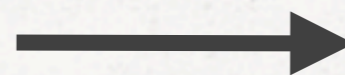
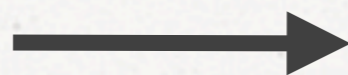


Ext.io.App

findDevices(query)

Ext.io.Device

send(message)



Ext.io.App

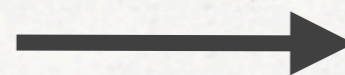
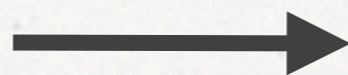
```
findDevices(query)
```

Ext.io.Device

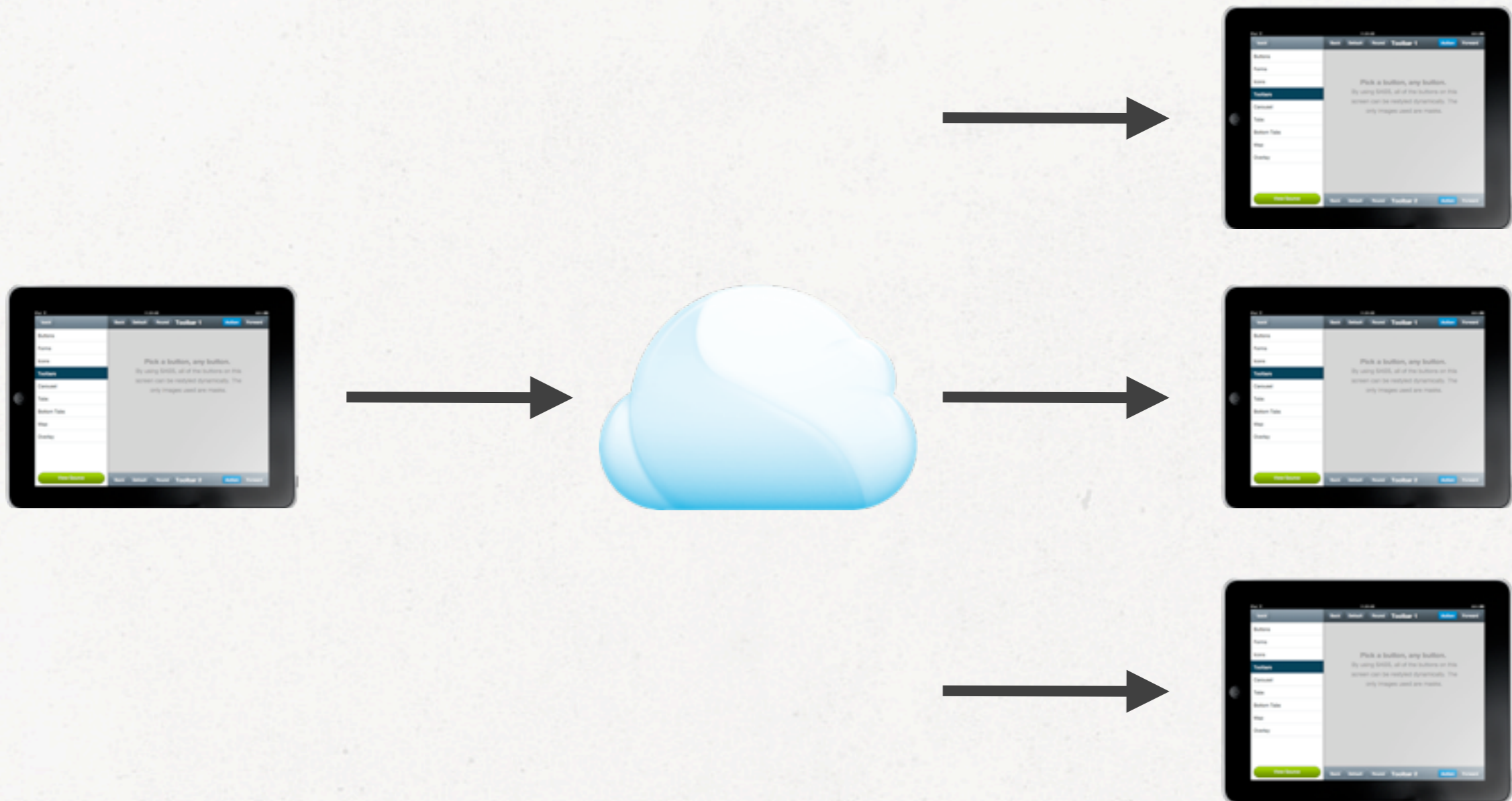
```
send(message)
```

Ext.io.Device

```
receive(  
  function(from,message) {  
  }  
);
```



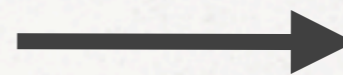
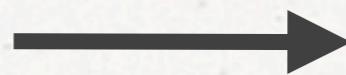
One to Many





Ext.io.App

getQueue()

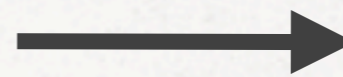
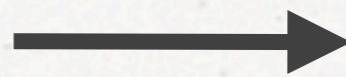


Ext.io.App

getQueue()

Ext.io.Queue

publish(message)



Ext.io.App

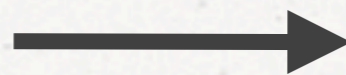
getQueue()

Ext.io.Queue

publish(message)

Ext.io.Queue

```
subscribe(  
  function(message) {  
  }  
);
```



sencha-touch.js

sencha-io.js

sencha-touch.js

Ext.io

Ext.io.App

Ext.io.Group

Ext.io.Device

Ext.io.User

Ext.io.Queue

Ext.io.SyncStorage

Ext.io

Ext.io

```
setup()  
getCurrentApp()  
getCurrentDevice()  
getCurrentGroup()  
getCurrentUser()  
getQueue()  
getService()
```

Ext.io.App

Ext.io.Group

Ext.io.Device

Ext.io.User

Ext.io.App

```
getGroup()  
findDevices()
```

Ext.io.Group

```
getApp()  
findUsers()  
register()  
authenticate()
```

Ext.io.Device

```
getApp()  
getUser()  
send()  
receive()  
update()
```

Ext.io.User

```
getDevices()  
getGroup()  
send()  
receive()  
update()
```

Ext.io.Queue

Ext.io.SyncStorage

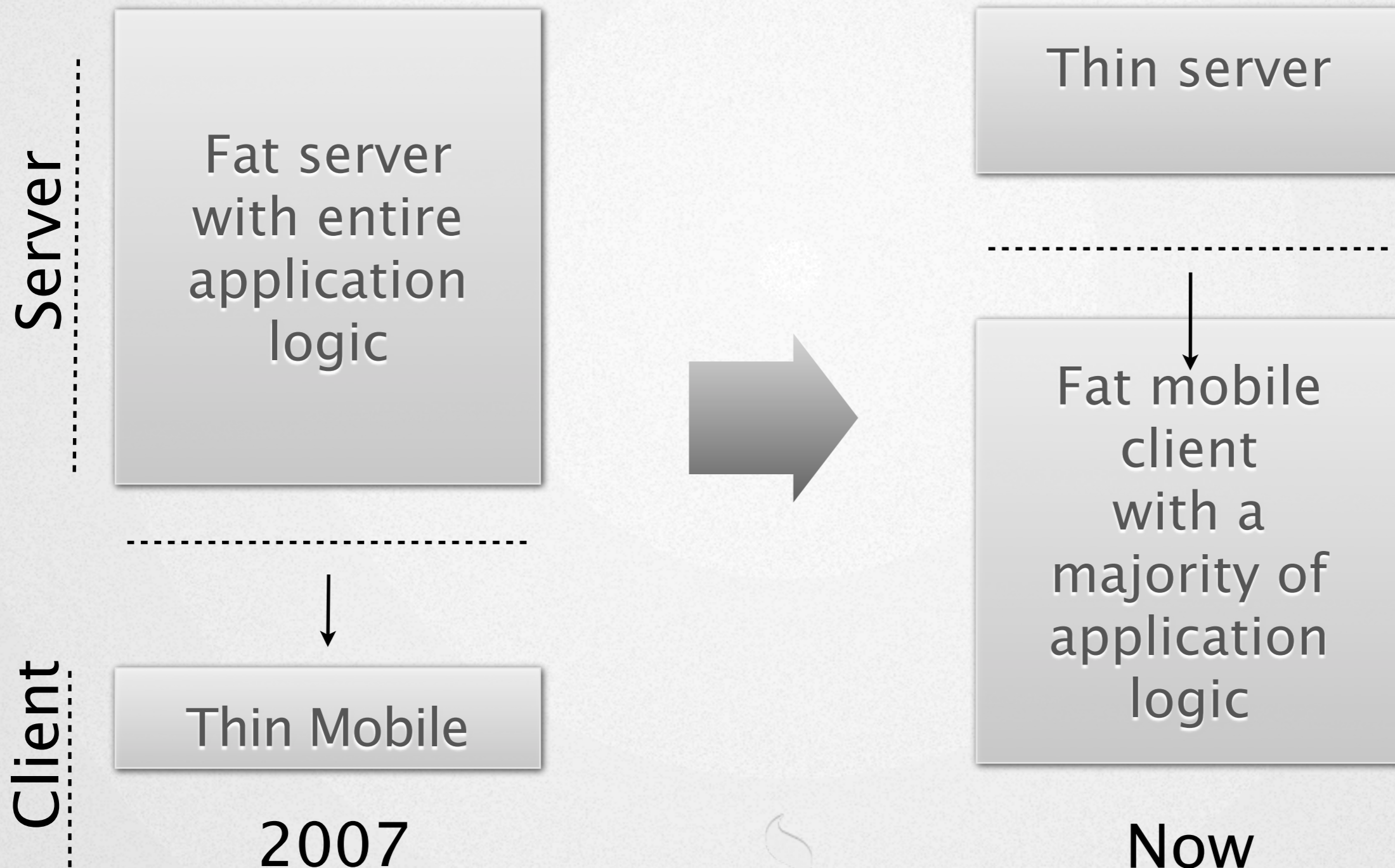
Ext.io.Queue

```
publish()  
subscribe()  
unsubscribe()
```

Ext.io.SyncStorage

```
create()  
read()  
update()  
destroy()  
sync()
```


Mobile: New Architecture





Sencha.io
Thank you!
@adityabansod

