

MOBILE-FIRST ARCHITECTURES

Alexander Stigsen
Realm

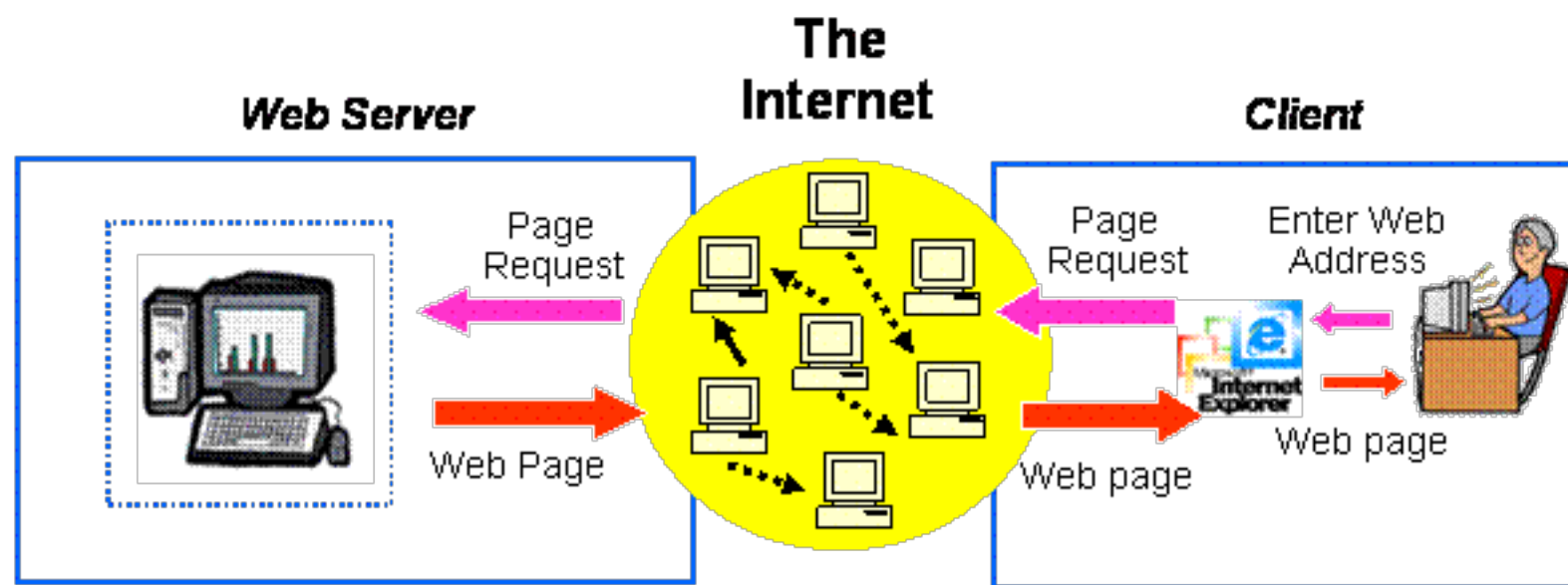
Who am I?

- Former Engineer from Nokia
- Working on mobile since 1999
- Currently CEO of Realm, building a modern mobile database
- Very biased :-)

Realm 

Problem

Mobile Apps are still being build as if they were web apps from 1999.



Mobile apps are treated as dumb clients, with all the data and logic residing on the server.

Why?

- This is how (web) apps have been developed in the last 10 years. So this is what developers are used to.
- May already have existing API's, designed with this mindset, that the mobile app has to conform to.
- Makes it easier to build both mobile and web versions of the app.

Drawbacks:

User Experience

Connectivity

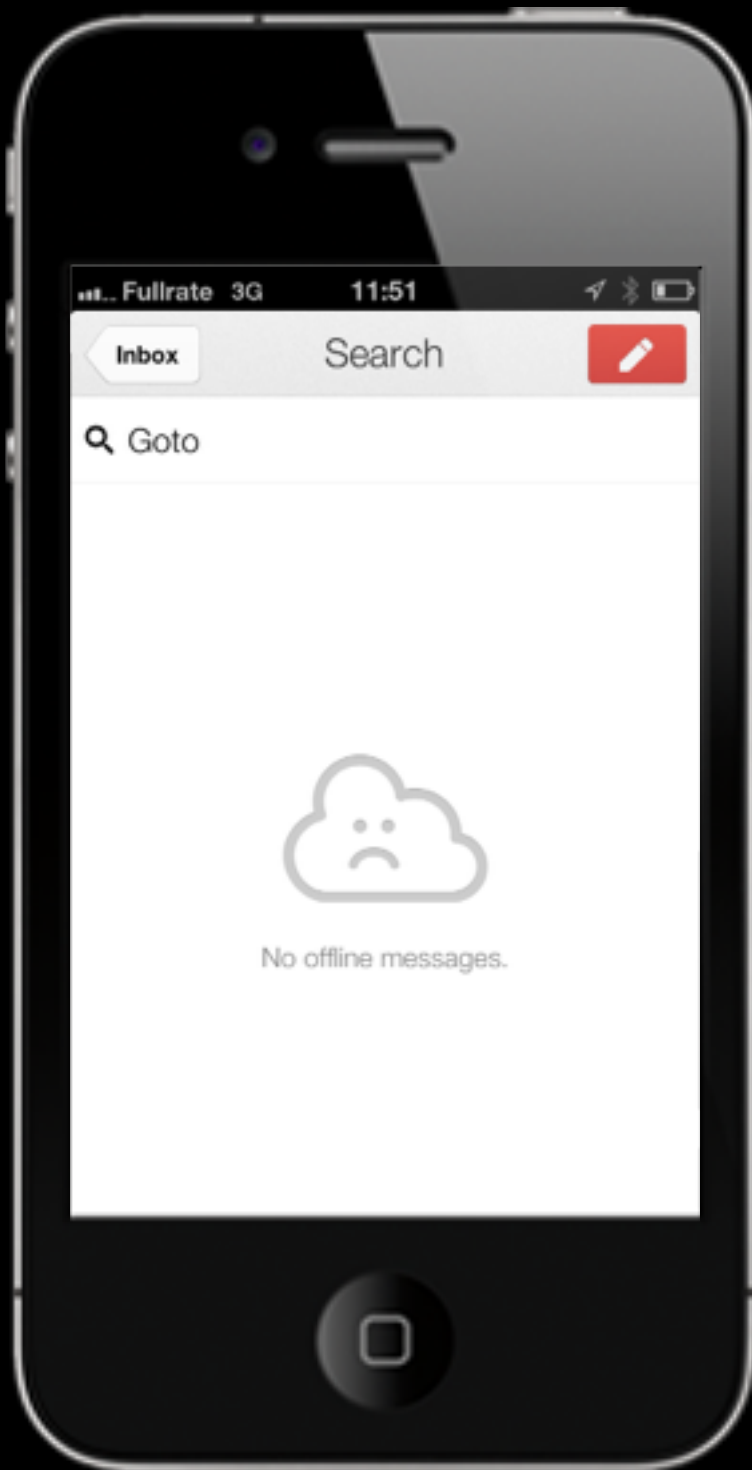
Server Cost

User Experience



- Users have much higher expectations of native apps
- Apps gets more advanced and capable and as such need more complex data models
- No latency or wait times accepted

Connectivity



- Users expect apps to always work
- The failed promise of ubiquitous connectivity
 - Even in SF connectivity is spotty
 - Go to the countryside, still many places with no connectivity
 - Crises or huge events often takes down connectivity
- Developing countries
 - Next big market
 - Low connectivity
 - Expensive data plans

Escalating Server Cost



- #1 Killer of startups
- App may need millions of users before being able to monetize
- Server load scale with number of users
- Amazon is expensive!

Solution?

Modern phone hardware



iPhone 6

1.4 Ghz Dual-core CPU

1 GB memory

16-128 GB storage

Quad-core GPU

Use it!

- Primary store on phone
- User has all his data on the device
 - Always available
 - Always instant
- Backend is only for backup, coordination and statistics

Mobile Driven Design

Change of mindset for developers

Start from device, building the best possible experience for the users

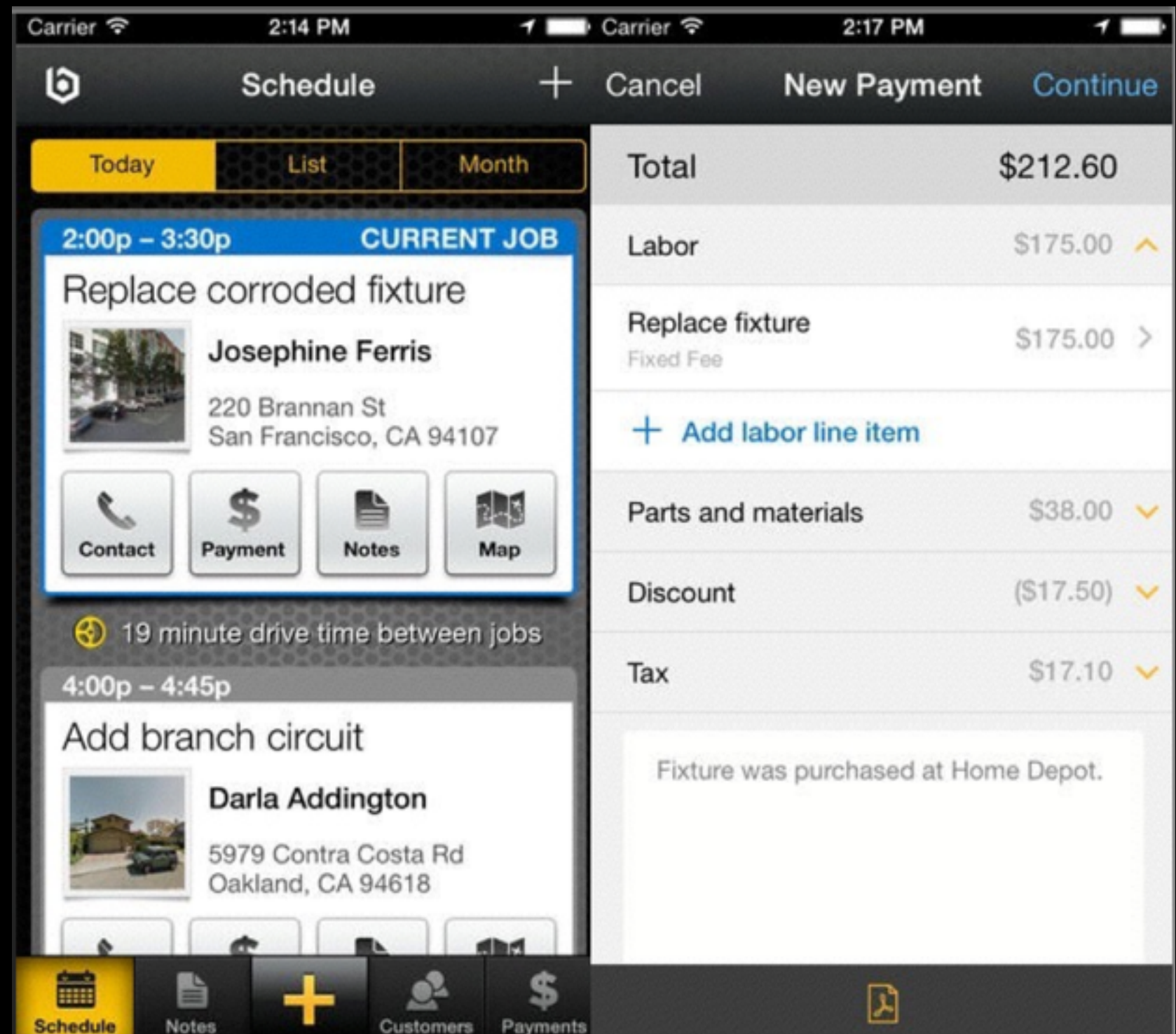
Then let that shape the backend, rather than the other way around

Case study

BreezeWorks

“Run your entire service business from your phone.”

Uses Realm to embed & query large databases of tax rates & service codes.



Is this always the right choice?

Primarily for apps that are user-centric (which most are).

Not for apps that just browse other peoples data

(Youtube would obviously never keep all its data on-device, neither would Tinder)

But even then, they could often benefit from keeping the non-browsing parts local

The Next Frontier

- Live Collaboration
- Internet of Things
- Wearables
- Drones
- VR

Q&A