

[methods] Example Design Process

< *in teams* >

Social Computing Systems

Walter S. Lasecki

EECS 498, Winter 2017

Logistics

- Assignment for tomorrow: still due tomorrow!
 - You should have feedback on the last part now

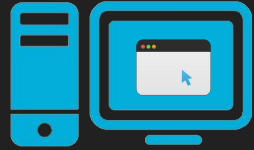
- Assignment for NEXT Friday: **Meteor!**

A quick primer on system design

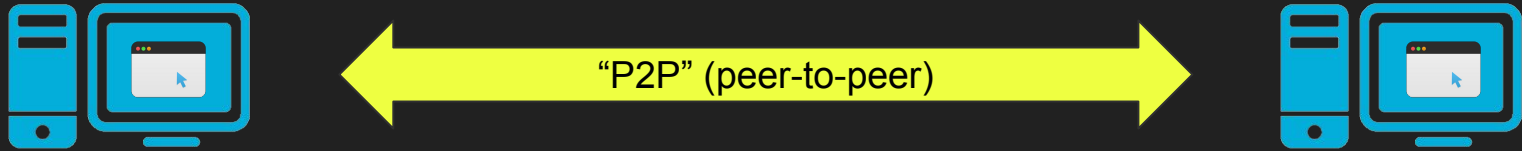
What do we need for social apps?

- Communication
- Collaboration
- Shared state (model)
- Shared view
- Shared control

What do we need for communicative/sync apps?

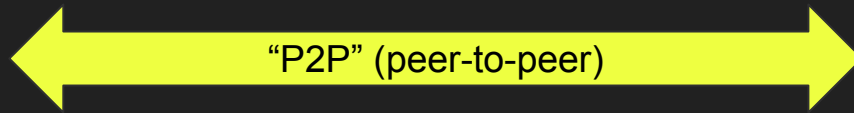


What do we need for communicative/sync apps?

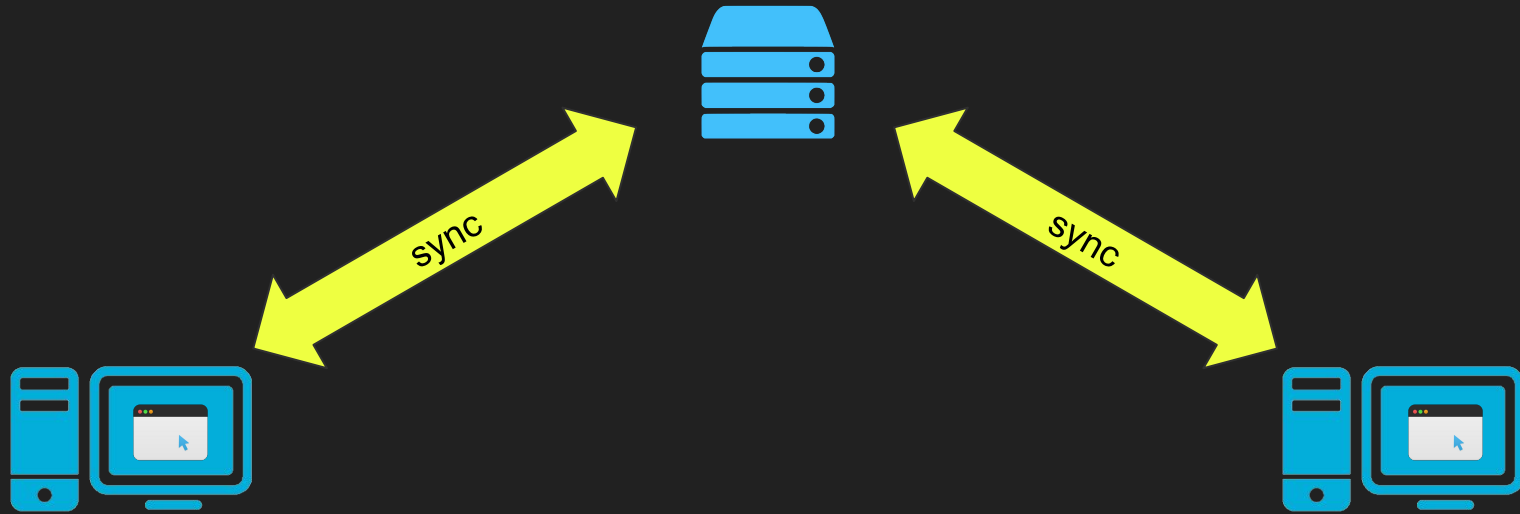


What do we need for communicative/sync apps?

**Good solution?
Why / Why Not?**



What do we need for communicative/sync apps?



(client-server model)

What do we need for communicative/sync apps?



(client-server model)

Intro to MeteorJS

What is Meteor?

Javascript framework (“what is a framework?”)

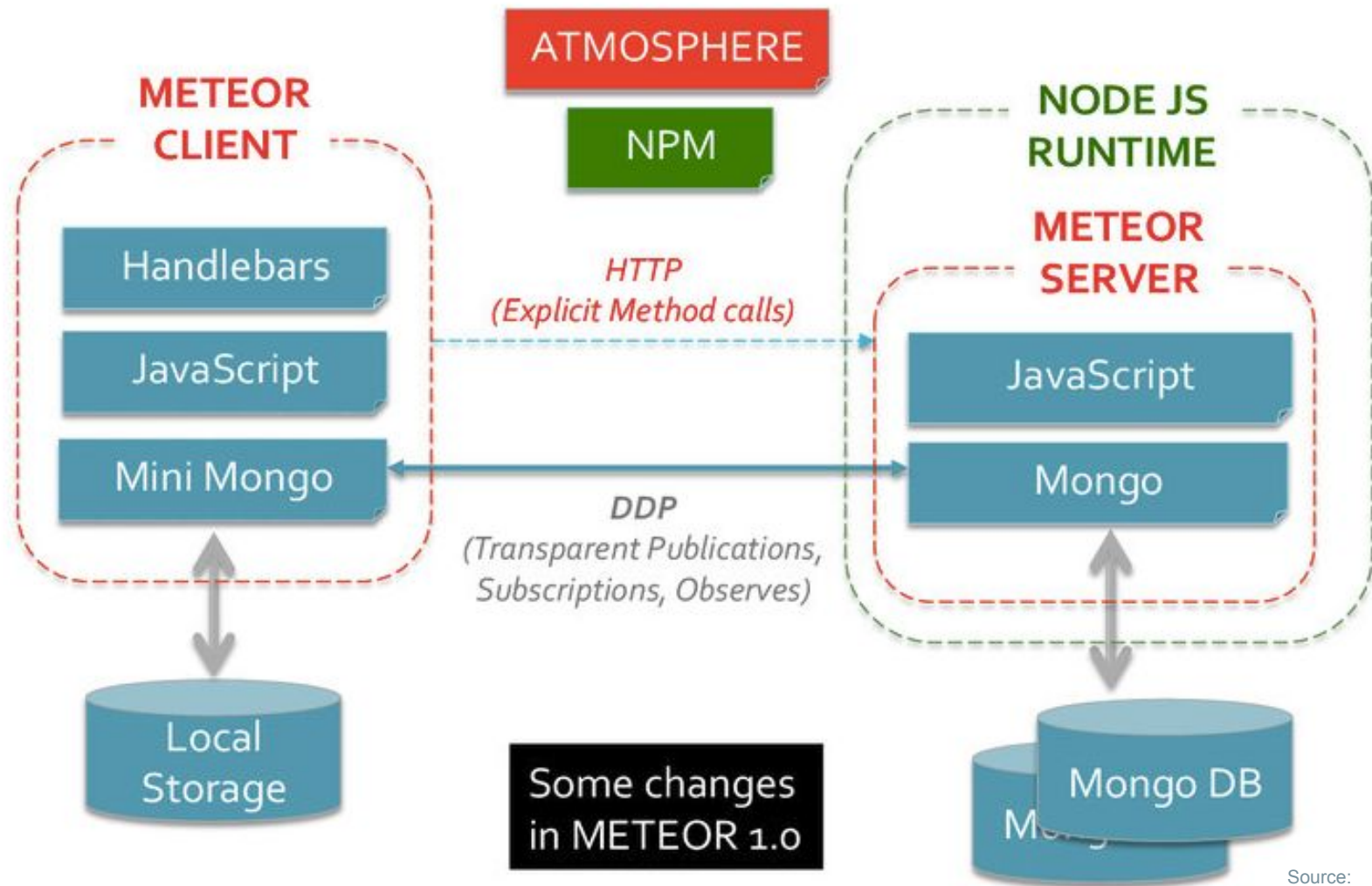
Front-end + back-end support

Front End:

- HTML + handlebars → connects HTML to JS, allows for variables/functions

Back End:

- Server-side JS. Wraps up node.js and mongoDB and abstracts them away



[example]