Meteor

- **Изоморфный код**
- **Постоянный обмен данными** (Distributed Data Protocol)
- **Обширная экосистема**: богатая документация и обучающие материалы, большая база плагинов, комьюнити, в т.ч. собственное, Stackoverflow
- **Full-stack реактивность**, позволяет UI работать с реактивными состояниями легко и с минимальными затратами на разработку

Базовые возможности Meteor

ПОСТАВКА

- Node.js, MongoDB, Cordova, ES6, Blaze (Handlebars)
- Свой собственный менеджер пакетов
- Преднастроенный сборщик проекта

АРХИТЕКТУРА

- Абстракция для работы с данными через коллекции (одна модель для сервера и клиента)
- DDP (EJSON) - работает с использованием WebSockets
- Строгая структура проекта и порядок загрузки скриптов
- Можно использовать с Angular или React

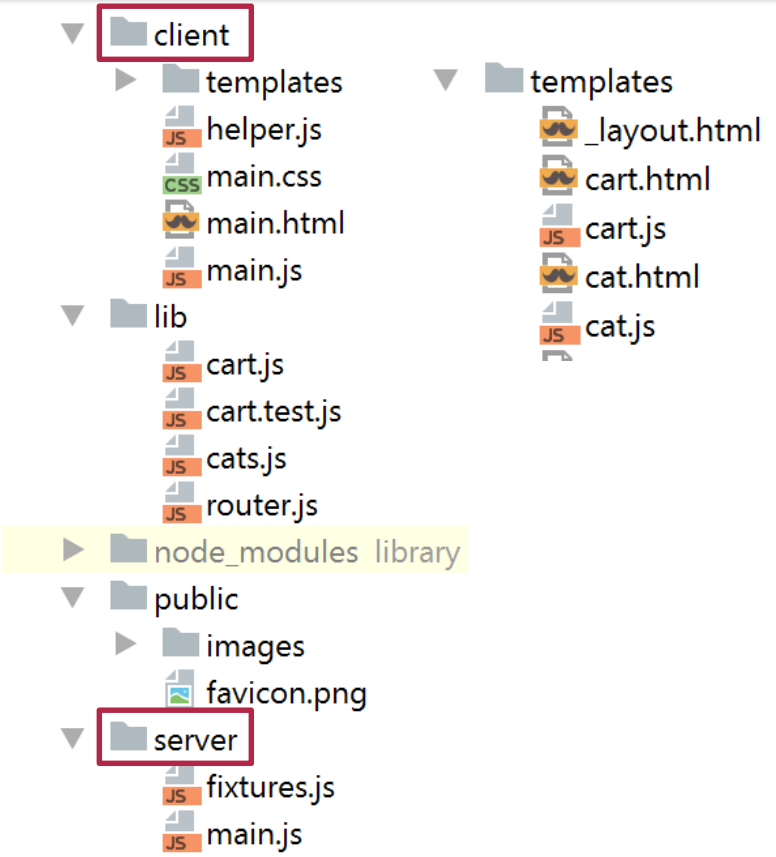
Пример организации приложения

1. Разделение клиентского и серверного кода в структуре папок и условиями вида

```
if(Meteor.isServer) {
  //code for server execution
}
if(Meteor.isClient) {
  //code for server execution
}
```

2. Шаблоны Blaze

```
<template name="Layout">
  {{> Nav}}
  {{> yield}}
</template>
```



Пример компонента

helpers.js

```
Template.registerHelper('FormatMoney', (number) => {
  return '$' + number.toLocaleString();
});
```

cat.html

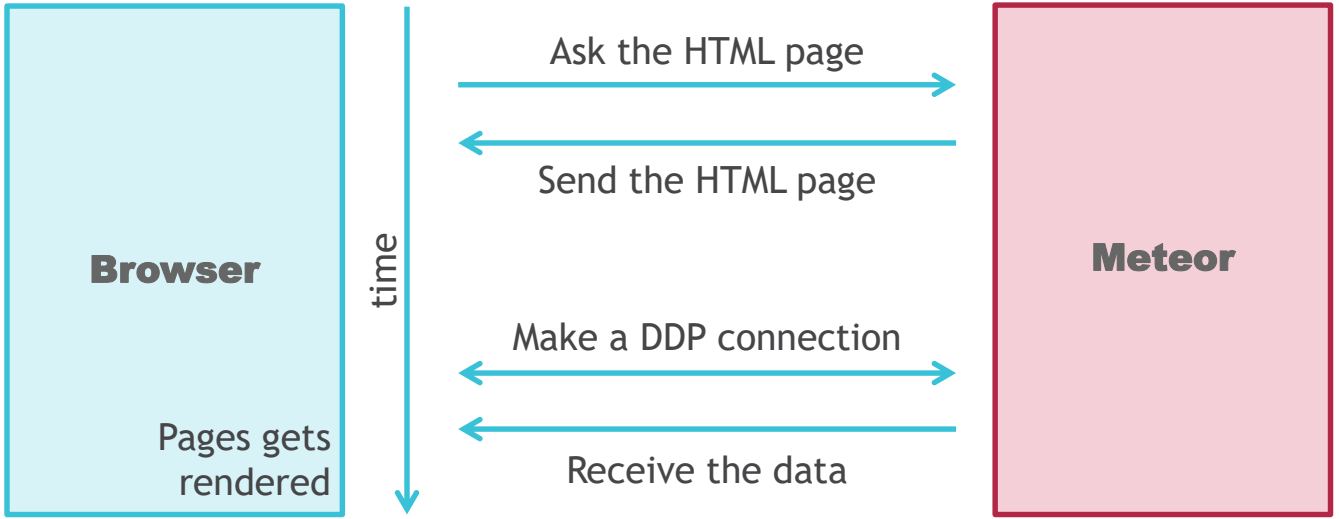
```
<template name="Cat">
 ...
 <p class="orange-text">{{FormatMoney price}}</p>
 {{#if isInCart}}
     <button id="add">Add to cart</button>
 {{else}}
     <a href="/cart">Go to cart</a>
 {{/if}}
 ...
</template>
```

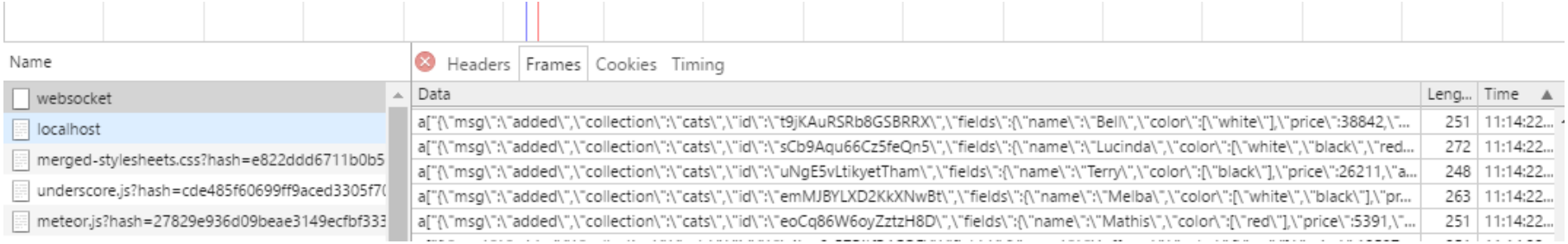
from template data

cat.js

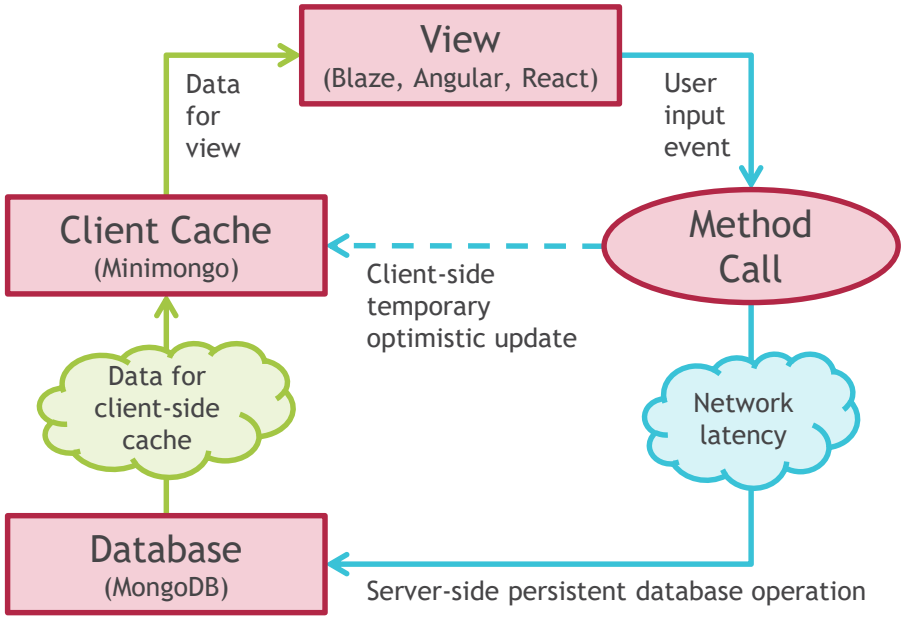
```
import { Template } from 'meteor/templating';
import { Cart } from '../../lib/cart.js';

Template.Cat.helpers({
  isInCart() {
    const instance = Template.instance();
    return !Cart.findOne({id: instance.data._id});
  },
});

Template.Cat.events({
  'click #add'(event, instance) {
    event.preventDefault();
    Cart.insert({id: instance.data._id});
  },
});
```

DDP


Browser
Pages gets rendered
time
Ask the HTML page
Send the HTML page
Make a DDP connection
Receive the data
Meteor


Name
websocket
localhost
merged-stylesheets.css?hash=e822ddd6711b0b5
underscore.js?hash=cde485f60699ff9aced3305f7(
meteor.js?hash=27829e936d09beae3149ecfbf333
Headers
Frames
Cookies
Timing
Data
Leng...
Time
a["{\"msg\":\"added\",\"collection\":\"cats\",\"id\":\"t9jKAuRSRb8GSBRRX\",\"fields\":{\"name\":\"Bell\",\"color\":[\"white\"],\"price\":38842,\"... 251 11:14:22...
a["{\"msg\":\"added\",\"collection\":\"cats\",\"id\":\"sCb9Aqu66Cz5feQn5\",\"fields\":{\"name\":\"Lucinda\",\"color\":[\"white\",\"black\",\"red... 272 11:14:22...
a["{\"msg\":\"added\",\"collection\":\"cats\",\"id\":\"uNgE5vLtikyetTham\",\"fields\":{\"name\":\"Terry\",\"color\":[\"black\"],\"price\":26211,\"a... 248 11:14:22...
a["{\"msg\":\"added\",\"collection\":\"cats\",\"id\":\"emMJBYLXD2KkXNwBt\",\"fields\":{\"name\":\"Melba\",\"color\":[\"white\",\"black\"],\"pr... 263 11:14:22...
a["{\"msg\":\"added\",\"collection\":\"cats\",\"id\":\"eoCq86W6oyZztzH8D\",\"fields\":{\"name\":\"Mathis\",\"color\":[\"red\"],\"price\":5391,\"... 251 11:14:22...

Data flow



Создание коллекции

```
Cart = new Mongo.Collection('cart');
```

Публикация и подписка

```
Meteor.publish('cats', function() {
  return Cats.find();
});

Meteor.subscribe('cats');
```

Удалённый вызов процедур (RPC)

```
Meteor.methods({
  'cart.clear'() {
    return Cart.remove({});
  },
});

Meteor.call('cart.clear');
```

Реактивность на клиенте

filters.js

```
import { Template } from 'meteor/templating';
import { ReactiveDict } from 'meteor/reactive-dict';

export const state = new ReactiveDict();

Template.Filters.onCreated(function() {
  state.clear();
});

Template.Filters.events({
   //....
  'input #agefrom'(event, instance) {
    state.set('ageFrom', event.target.value);
  }
  //....
});
```

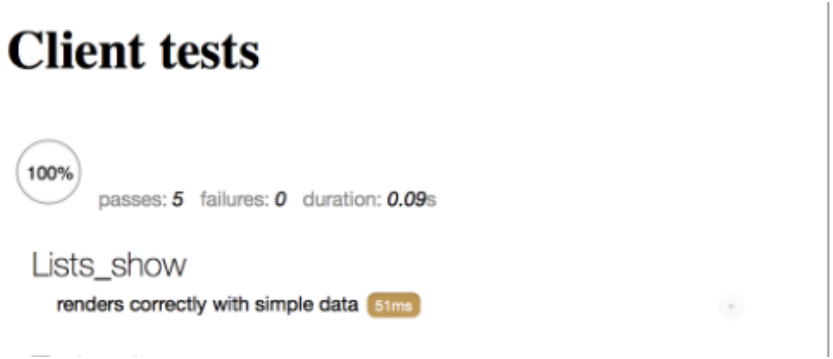
home.js

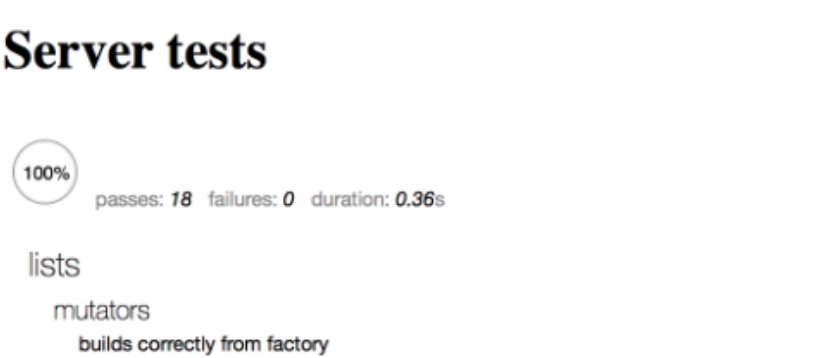
```
import { Template } from 'meteor/templating';
import { Cats } from '../../lib/cats.js';

import { state } from './filters.js';

Template.Home.helpers({
  cats() {
    if (!state) return [];
    let filters = state.all();
    let params = {};
    //......
    if (filters.colors && filters.colors.length > 0) {
      params.color = { $all: filters.colors };
    }
    //......
    return Cats.find(params).fetch();
  },
});
```

Тестирование

1. Mocha, Chai
2. Плагины для мок-данных, random и прочее
3. Kadira, Meteor DevTools (Chrome plugin) для профилирования DDP соединения

```
describe('Cart', () => {
  describe('methods', () => {
    beforeEach(() => {
      Cart.remove({});
      Cart.insert({ id: "123456" });
    });

    it('can clear the cart', () => {
      Meteor.server.method_handlers['cart.clear']();
      assert.equal(Cart.find().count(), 0);
    });
  });
});
```





Применение и подводные камни

1. Удобство деплоя приложения под разные платформы
2. Простой код, отличная документация и примеры использования
3. Быстрый старт

1. Высокий порог входа из-за большого количества технологий
2. Мало Meteor-разработчиков
3. Неполная поддержка Windows