

Яндекс

Особенности архитектуры Single Page Application

Антон Киршанов, разработчик интерфейсов

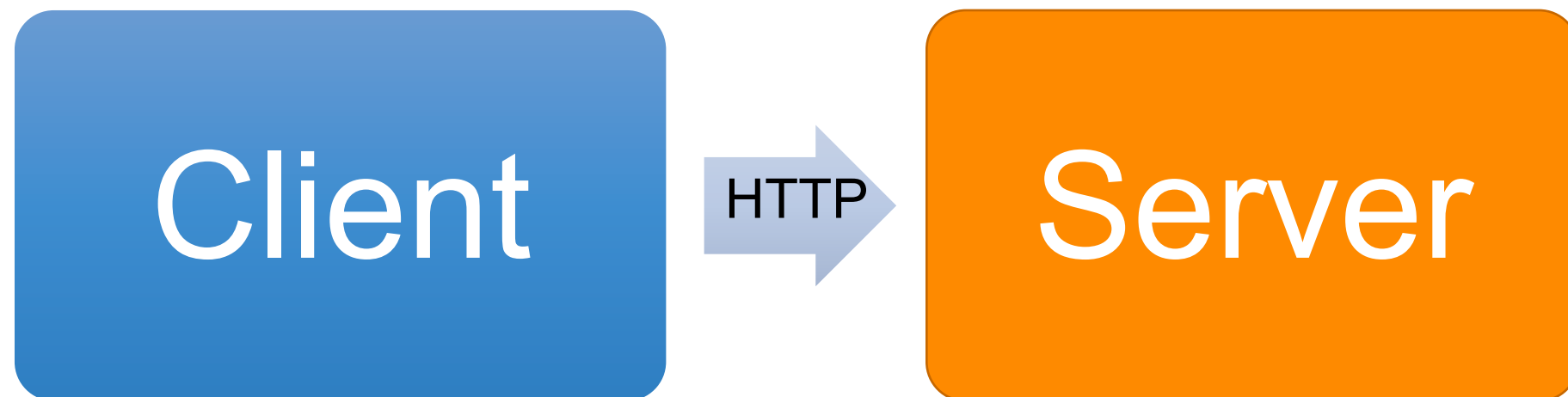
Frontend party, Нижний Новгород, 22 ноября 2014

Куда лошадь запрягать?

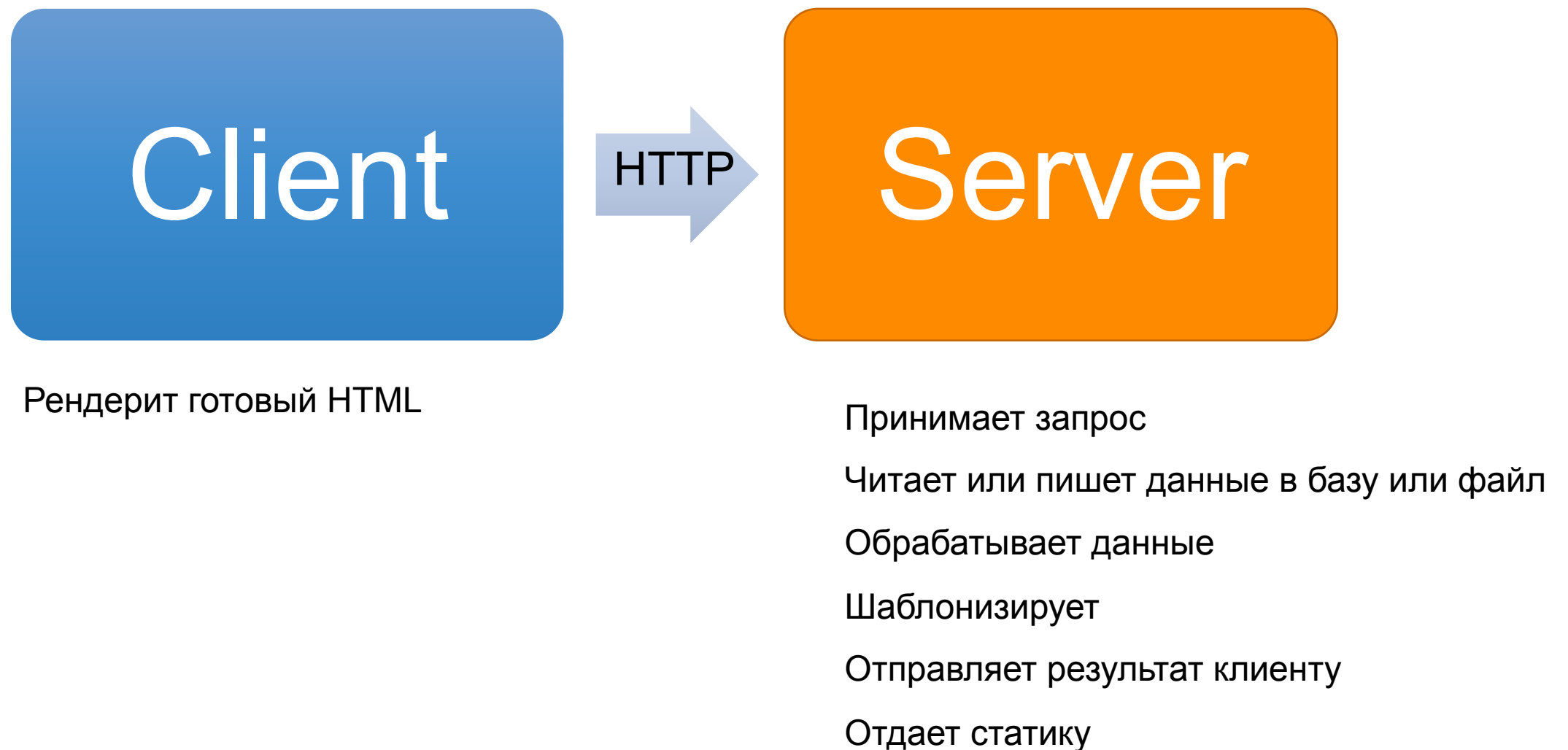




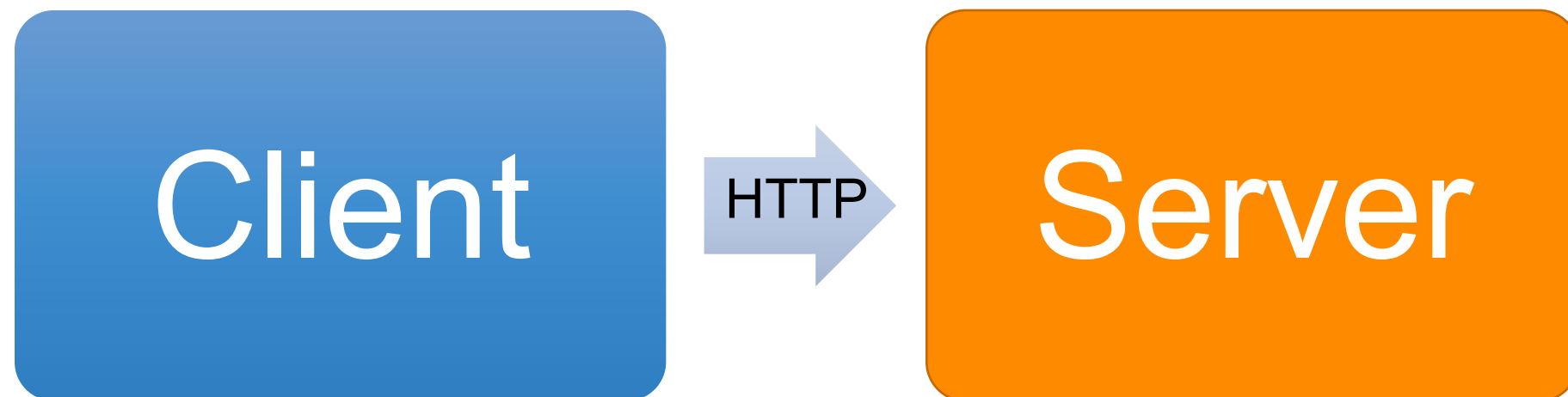
Архитектура



Архитектура



Архитектура



Каждый запрос с клиента приводит к перезагрузке страницы



Все в одном

Трехзвенная архитектура



Трёхзвенная архитектура



Трёхзвенная архитектура



Обработывает и сохраняет данные

Трехзвенная архитектура



Ходит в бекенд за данными
Шаблонизирует
Отдает готовый результат на клиент
Отдает статику

Трёхзвенная архитектура



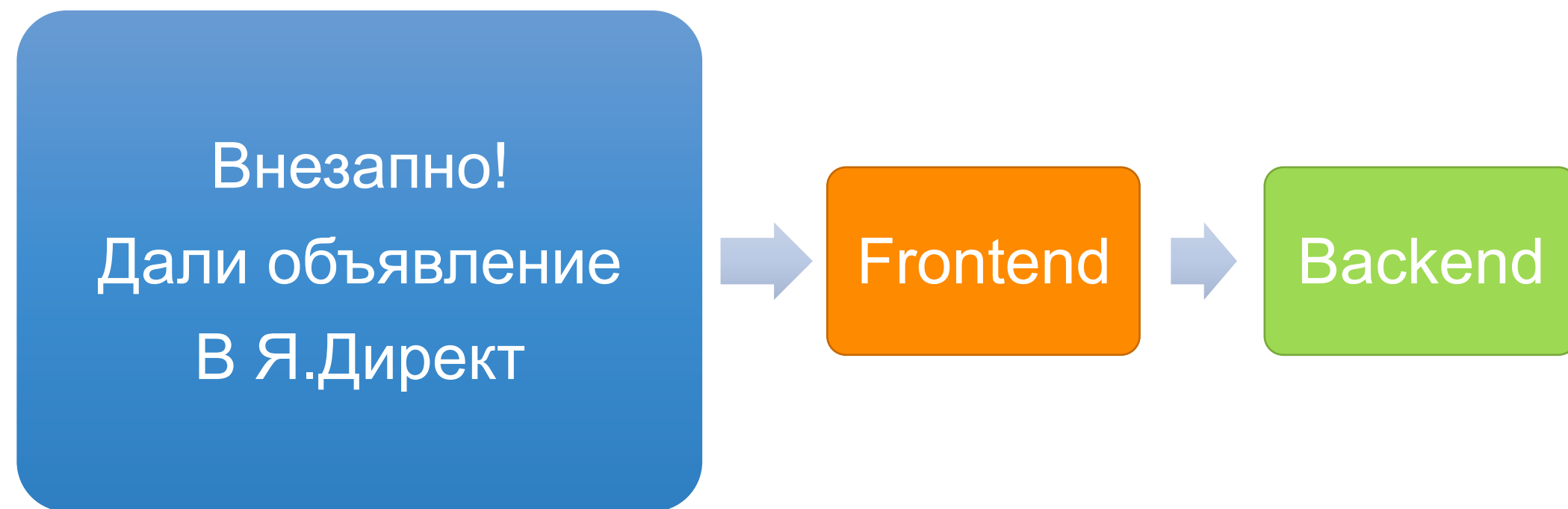
Рендерит готовый HTML

Трехзвенная архитектура

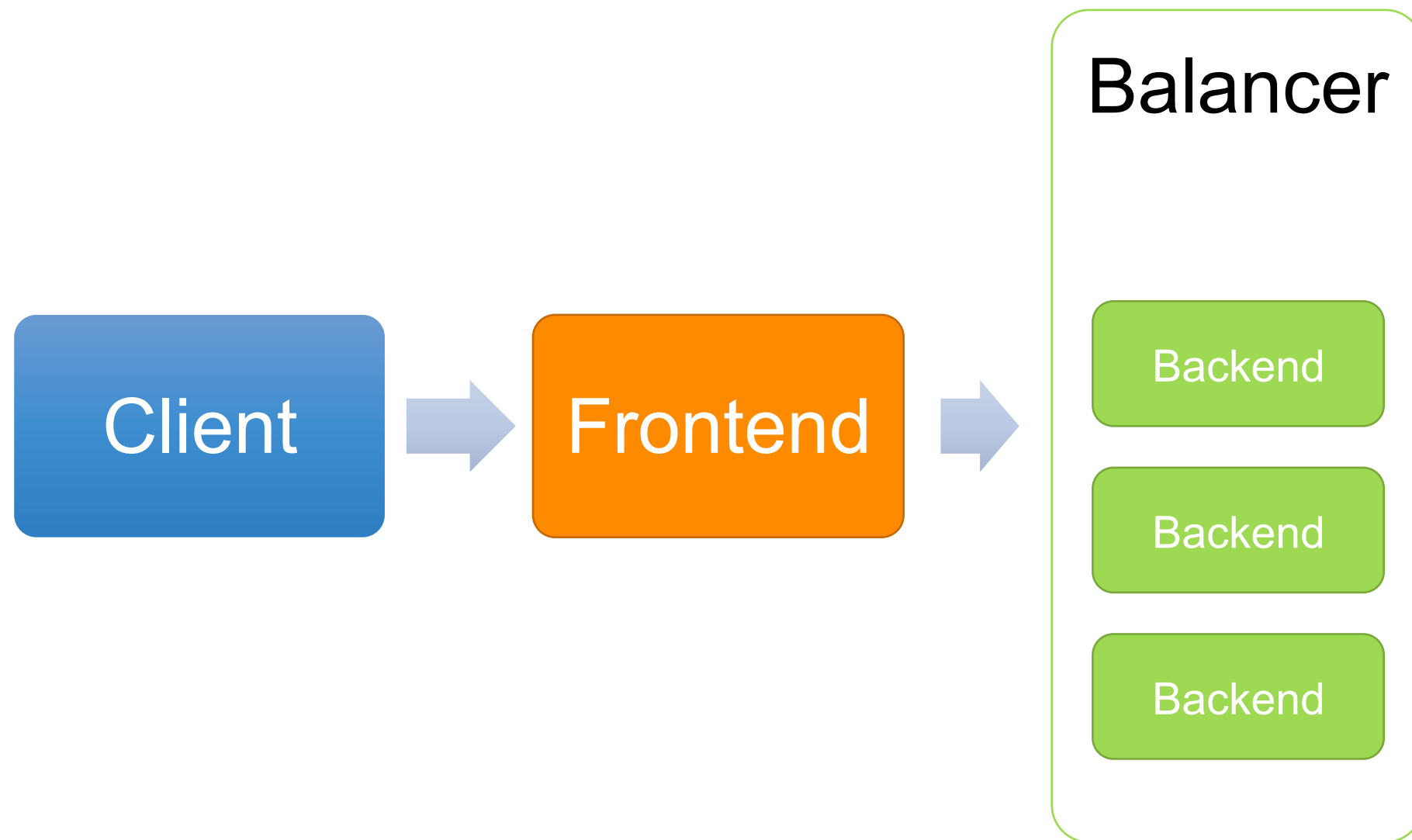


Каждый запрос с клиента приводит к перезагрузке страницы

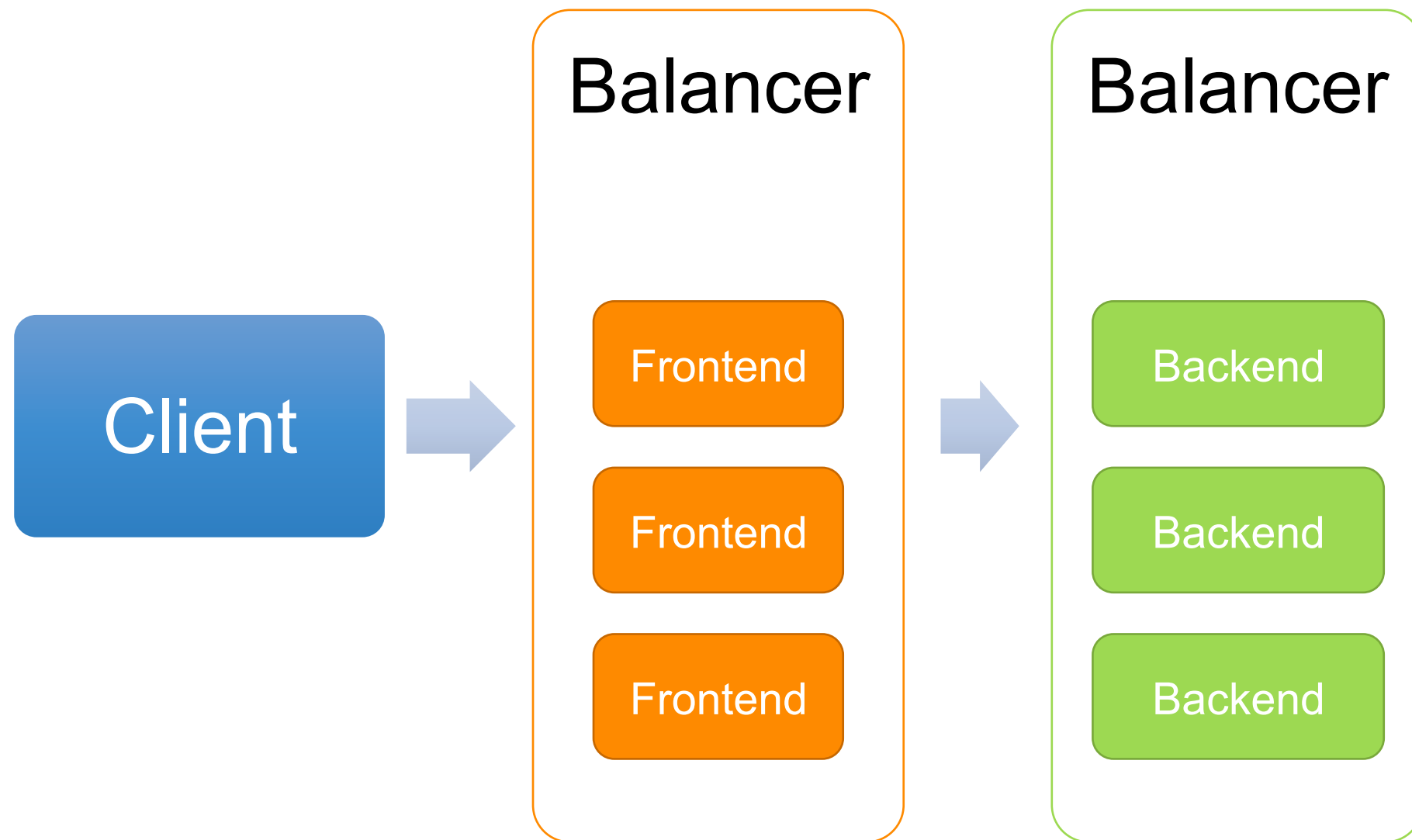
Трехзвенная архитектура



Трёхзвенная архитектура



Трехзвенная архитектура



Трехзвенная архитектура



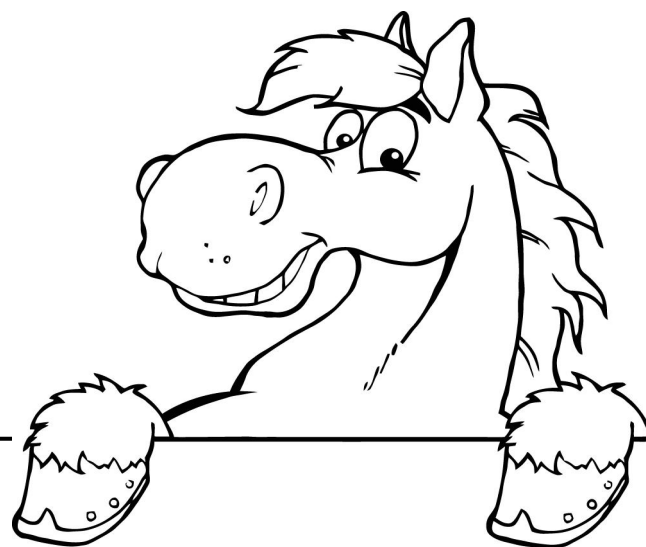
Масштабируемость
Производительность
Отказоустойчивость



Масштабируемость
Производительность
Отказоустойчивость

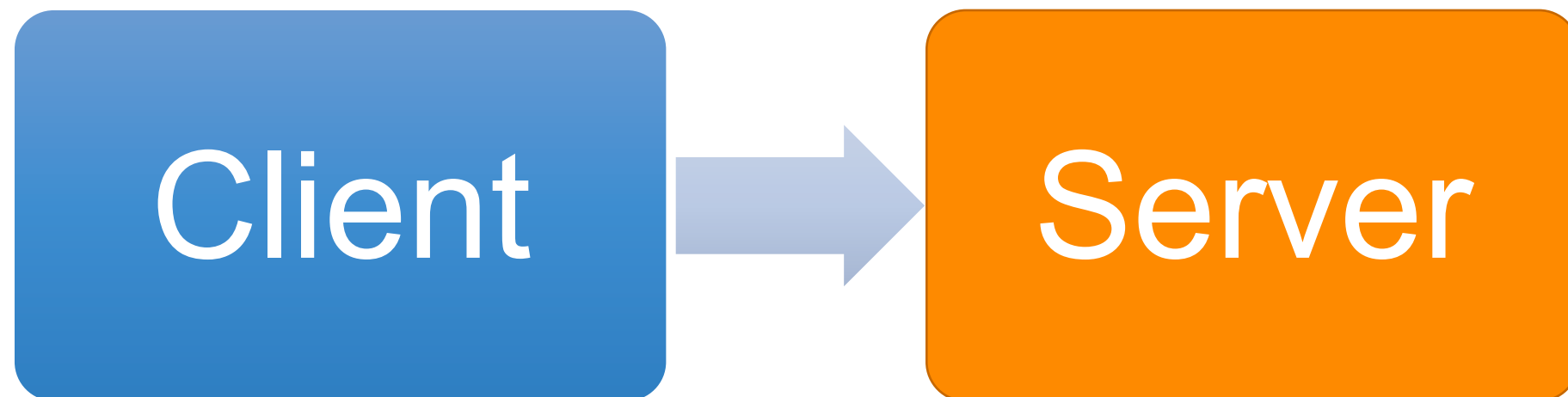


Разделение специализации

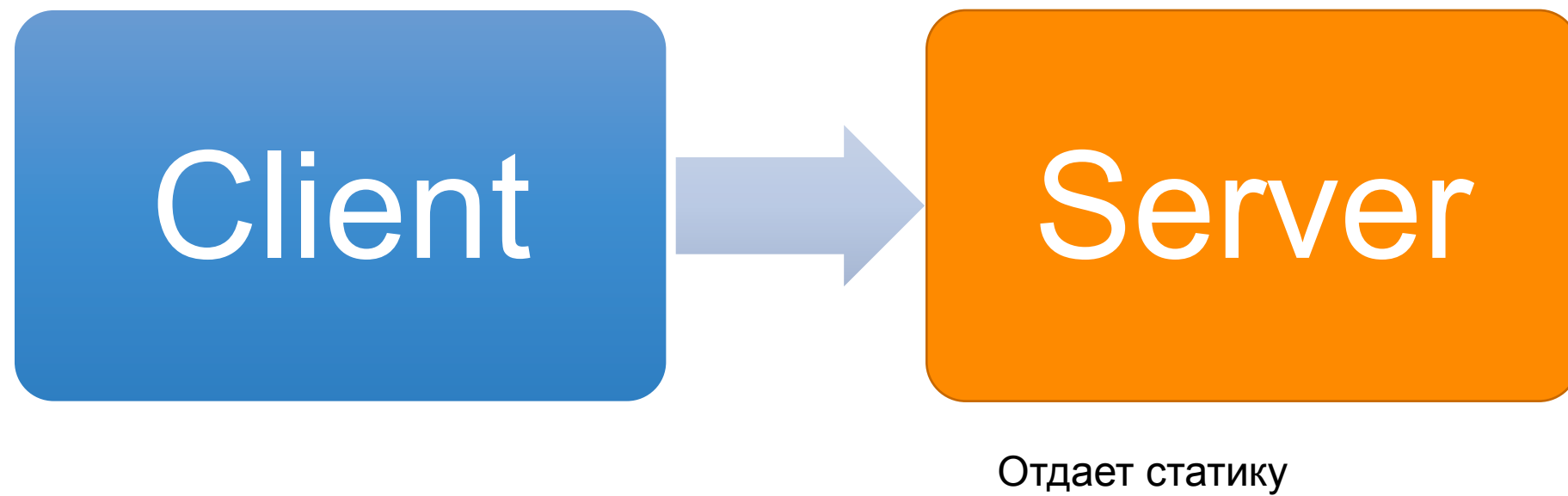


Необычный порошок

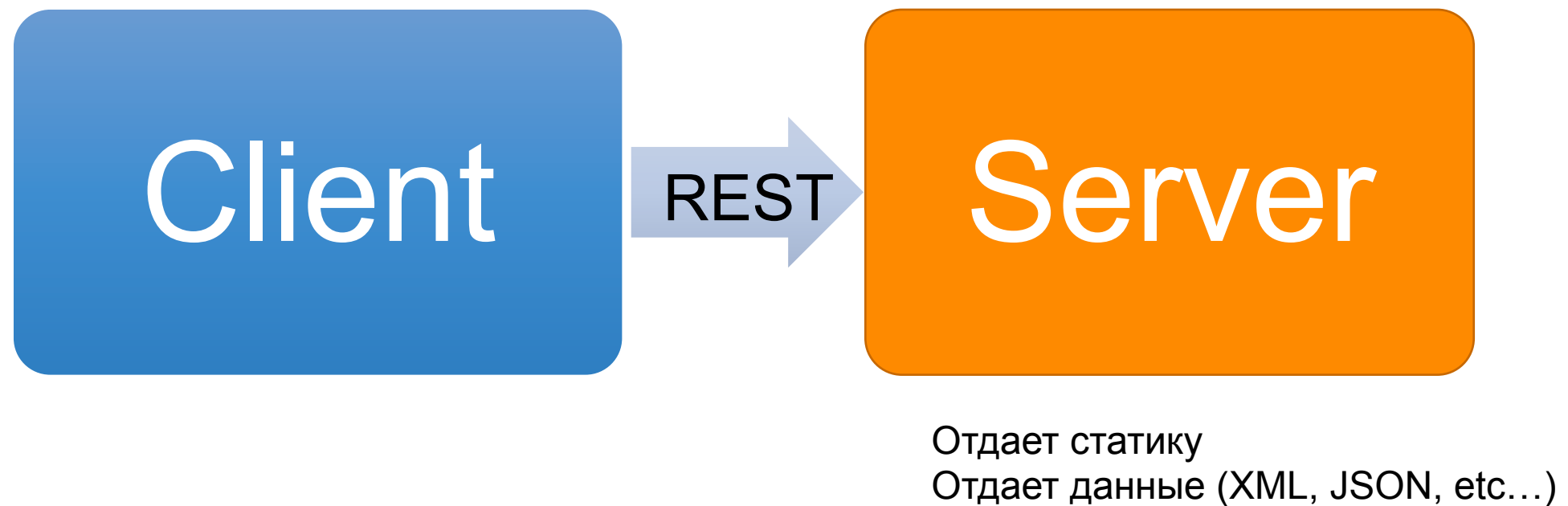
Архитектура SPA



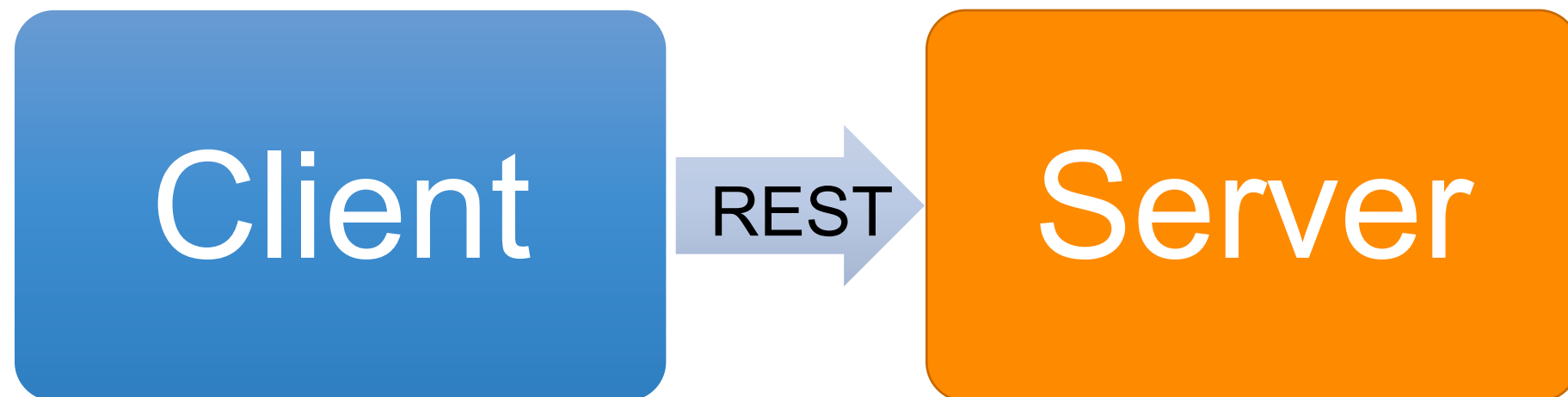
Архитектура SPA



Архитектура SPA

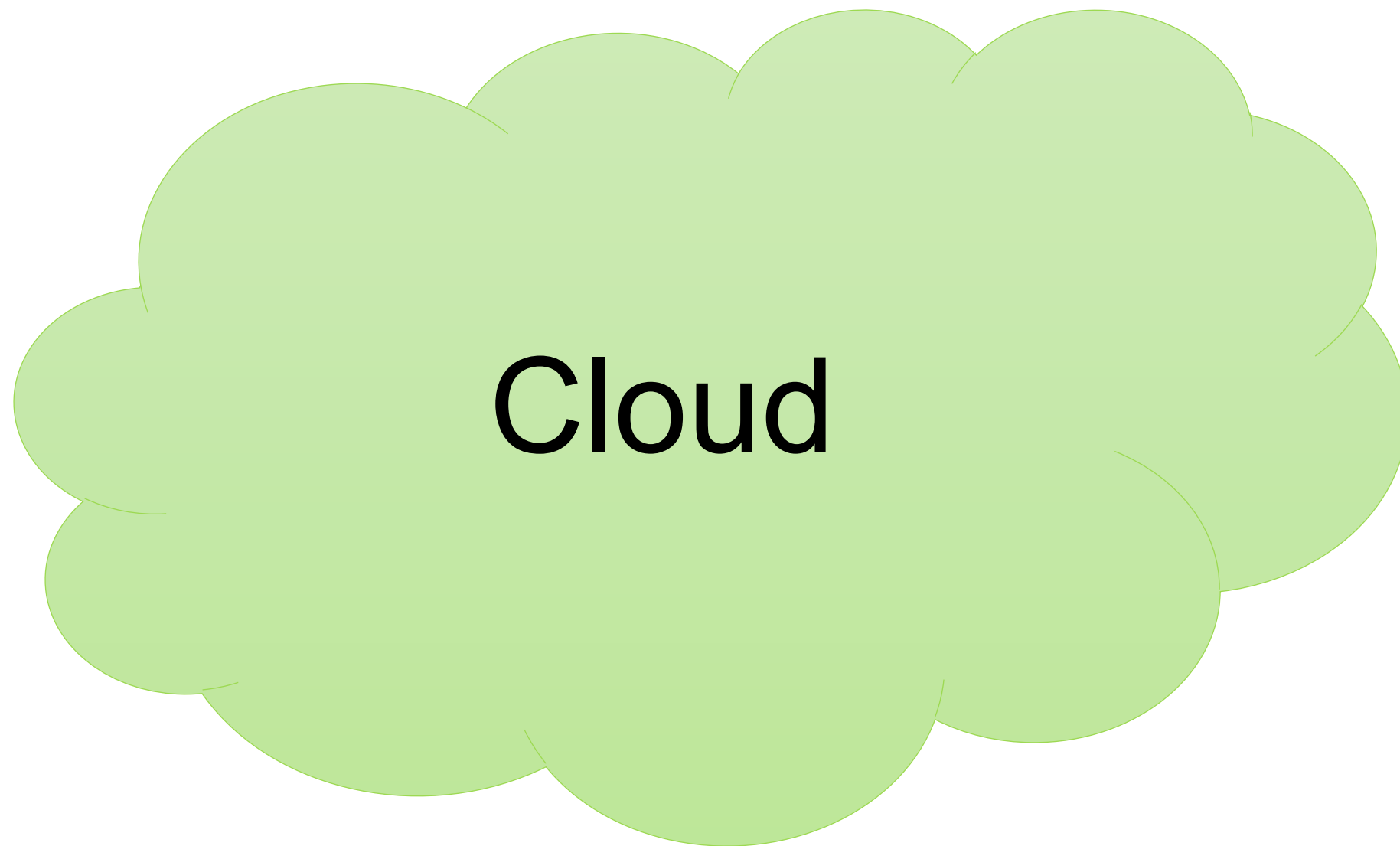


Архитектура SPA



Агрегирует данные
Строит объекты
Шаблонизирует
Создает DOM
Контролирует View

Архитектура SPA

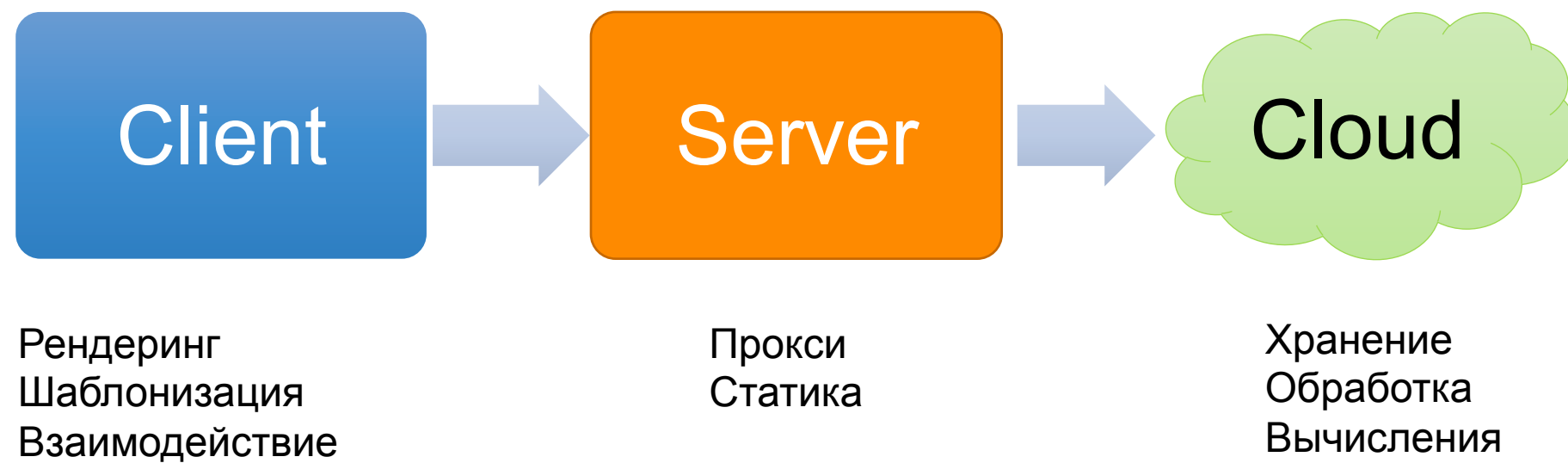


Архитектура SPA



Проксирует запросы

Архитектура SPA





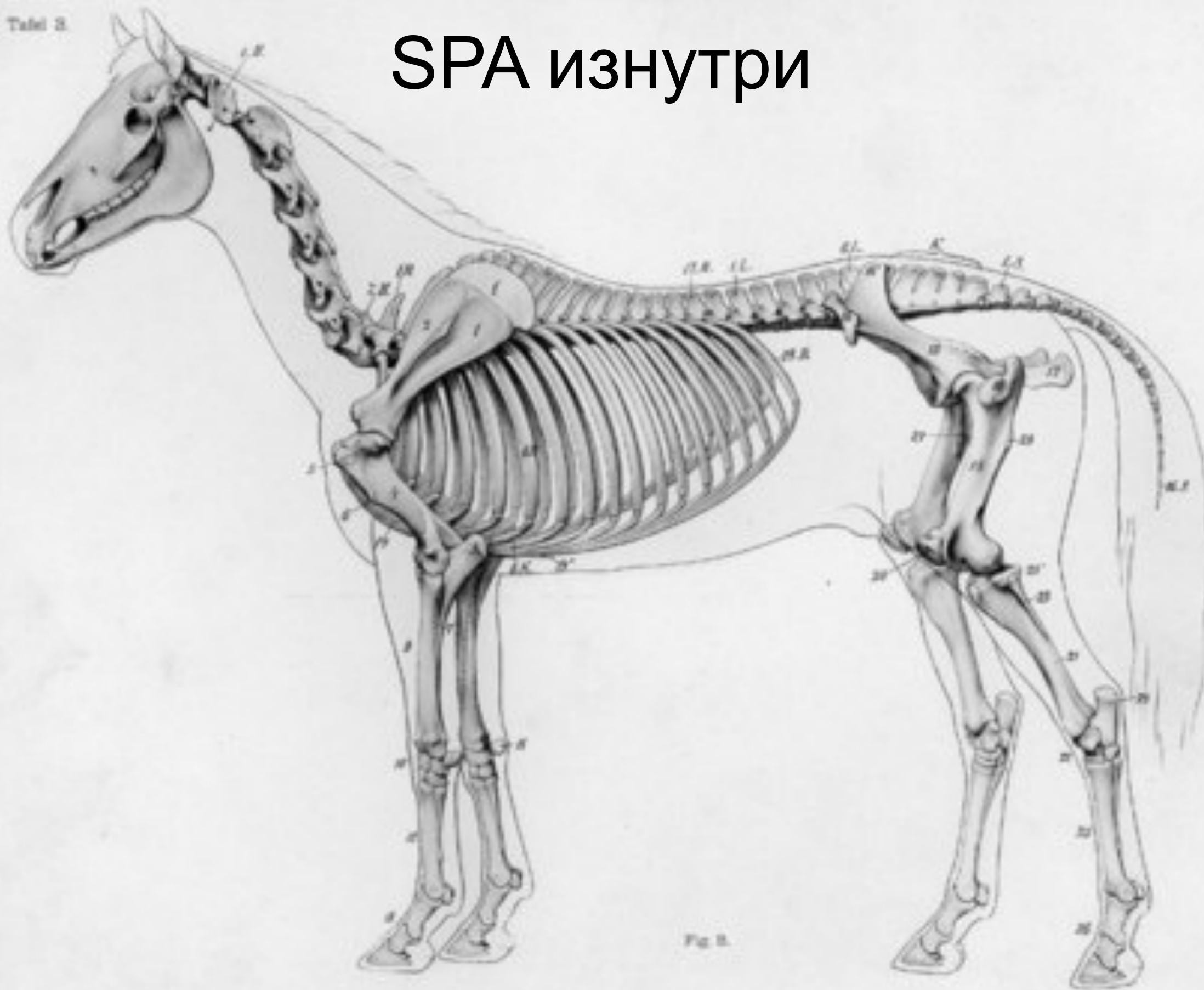


Масштабируемость
Производительность
Отказоустойчивость



Удобство разработки

SPA изнутри



SPA изнутри



Index.html

SPA изнутри



JavaScript

JavaScript

VanillaJS

JavaScript

VanillaJS



JavaScript

VanillaJS

```
positioning.js | hovercard.js* x
217 // Element is already showing, just make sure it
218 this.clearHideTimer();
219 return;
220 }
221 if (nchorElement == this.anchor) {
222 // Hovercard is pending, no need to retrigger.
223 return;
224 }
225
226 // If a previous hovercard was being triggered, ca
227 this.maybeCancelTrigger_();
228
229 // Create a new event for this trigger
230 var triggerEvent = new qooq.ui.HoverCard.TriggerEv
231
Uncaught ReferenceError: nchorElement is not defined (repeated 6 times)
Line 221, Column 7
```

JavaScript

VanillaJS
Jquery

JavaScript

VanillaJS

Jquery

MV*

JavaScript

VanillaJS

Jquery

MV* (Backbone, Angular, Ember ...)

JavaScript

VanillaJS

Jquery

MV* (Backbone, Angular, Ember ...)

React.js

JavaScript

VanillaJS

Jquery

MV* (Backbone, Angular, Ember ...)

React.js

Велосипед

Backbone



Backbone



- Простой
- Гибкий
- Ненавязчивый
- Хорошо документирован
- Куча рецептов, плагинов и готовых модулей

Шаблонизация

Шаблонизация

- Mustache
- Underscore
- Handlebars
- Jade
- Hogan
- Microtemplating
- domjs
- ...

Шаблонизация

Вложенность отступами

Шаблонная логика

DOM-constructor

Поддержка JS

Пространства имен

Декларация зависимостей

Шаблонизация



Marat Abdullin

hoho

 Yandex

 dakota@brokenpipe.ru

<https://github.com/hoho/concat.js>

DOM Builder

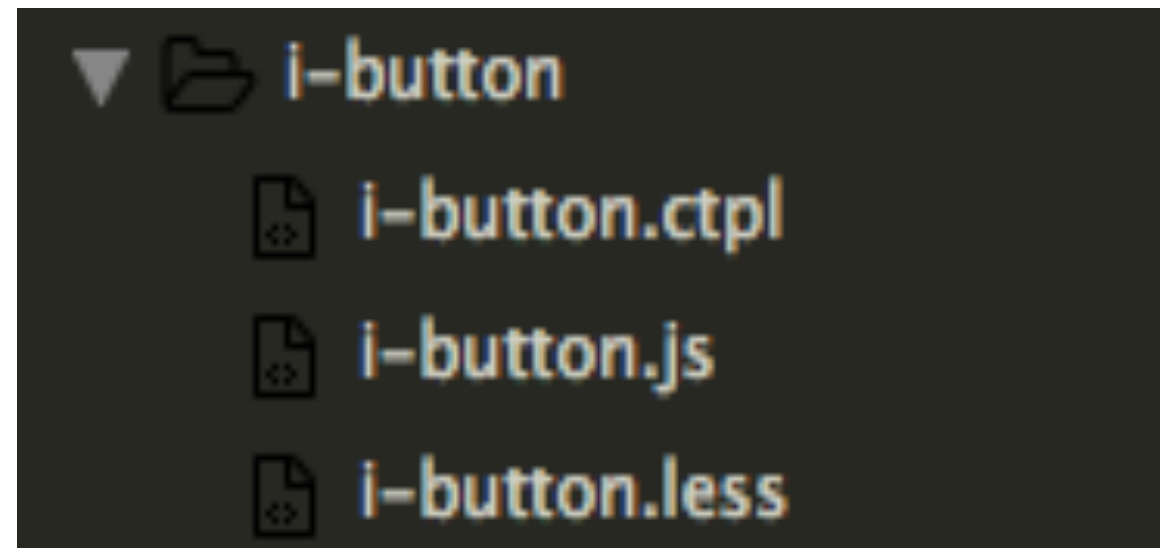
<https://github.com/hoho/conkitty>

Compiler

Блоки

- Модульность
- Переиспользование
- Тестирование
- Версионирование

Блоки



Блок кнопка

```
ns.models.button = Backbone.Model.extend(  
  {  
    'defaults': {  
      'size': 'M',  
      'checked': false,  
      'disabled': false,  
      'loading': false,  
      'badge': null  
    },  
    'isDisabled': function(){  
      return this.get('disabled') || this.get('loading');  
    }  
  }  
);
```

Блок кнопка

```
1  i::button $data
2    &"../i/i.js"
3    &"../i/i.less"
4    &"i-button.js"
5    &"i-button.less"
6    SET $block ("i-button")
7    SET $model ("get" in $data ? $data : new ns.models.button($data))
8    button AS $button
9      @class $block
10     @class +($block + "__size-" + $model.get("size"))
11     TEST ($model.get("loading"))
12       @class +($block+"__loading")
13     TEST ($model.get("pressed"))
14       @class +($block+"__pressed")
15     TEST ($model.get("checked"))
16       @class +($block+"__checked")
17     TEST ($model.get("disabled"))
18       @class +($block+"__disabled")
19     span
20       @class ($block+"__label")
21       span.text AS $label
22         ($model.get("label"))
23   JS
24     new ns.views.button(
25       {
26         'el': $button,
27         'model': $model
28       }
29     );
30
31   EXPOSE $model
32
```

Блок кнопка

i-button

normal

ctpl

```
CALL i::button ({"label":"normal"})
```

loading

ctpl

```
CALL i::button ({"label":"loading", "loading": true})
```

checked

ctpl

```
CALL i::button ({"label":"checked", "checked": true})
```

SPA

- Масштабируемость
- Тестируемость
- Управление зависимостями
- Современные технологии
- Удобство разработки

SPA - это удобно

- Локальный вебсервер
- Разработка через API
- Единый стек
- Быстрая сборка
- Интеграция и развёртывание

Контакты

Антон Киршанов

Разработчик интерфейсов



maetchkin@ya.ru



@maetchkin



Вопросы?