

Бэкенд для фронтендеров

Андрей Мелихов

BF

Backend For Front-End

Давным давно...



Apache

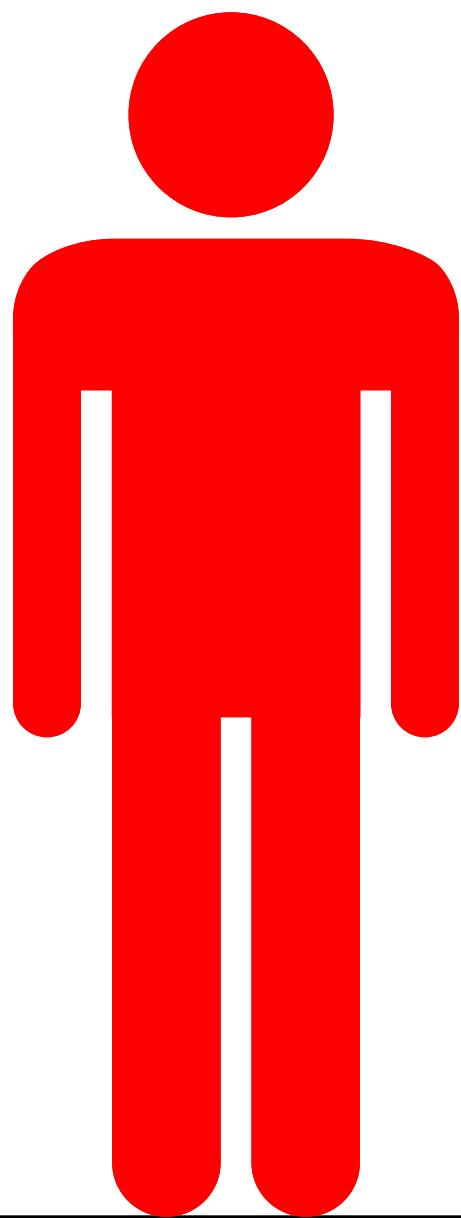
PHP

MySQL

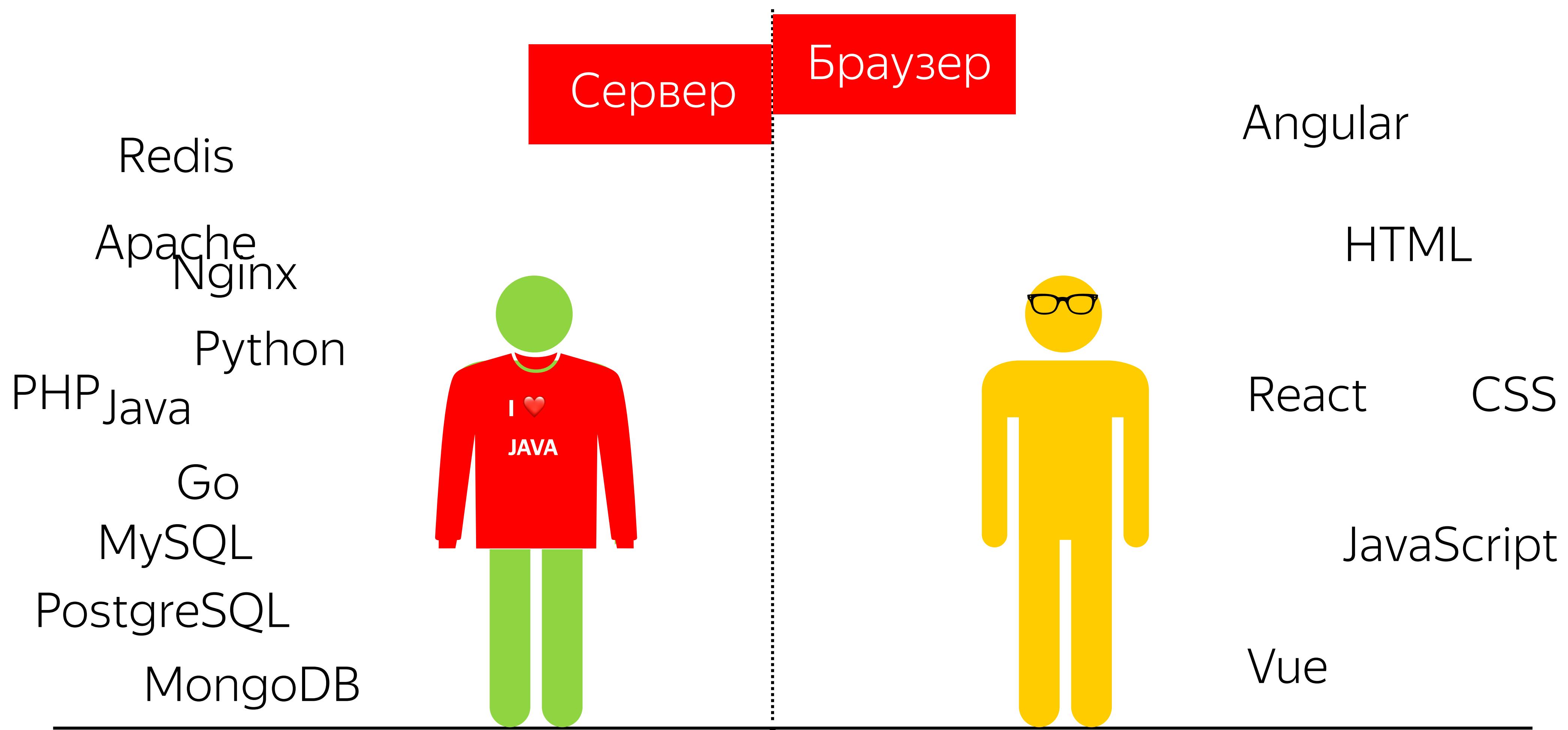
HTML

CSS

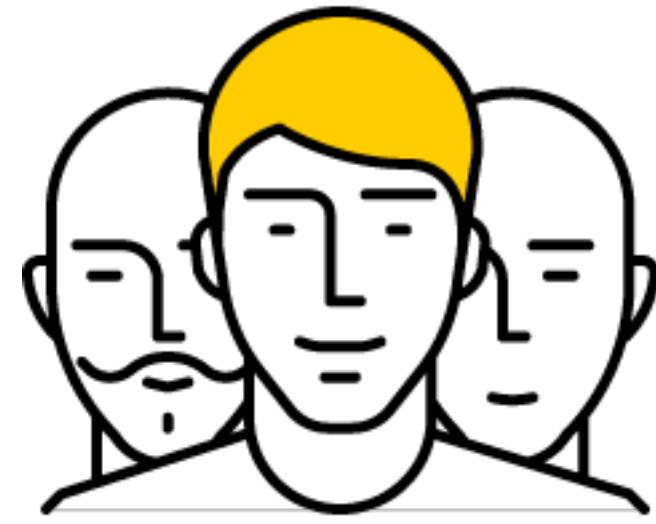
JavaScript



Веб-разработчик



Структура ИТ-организации. Начальный этап



Эксплуатация



Фронтендеры



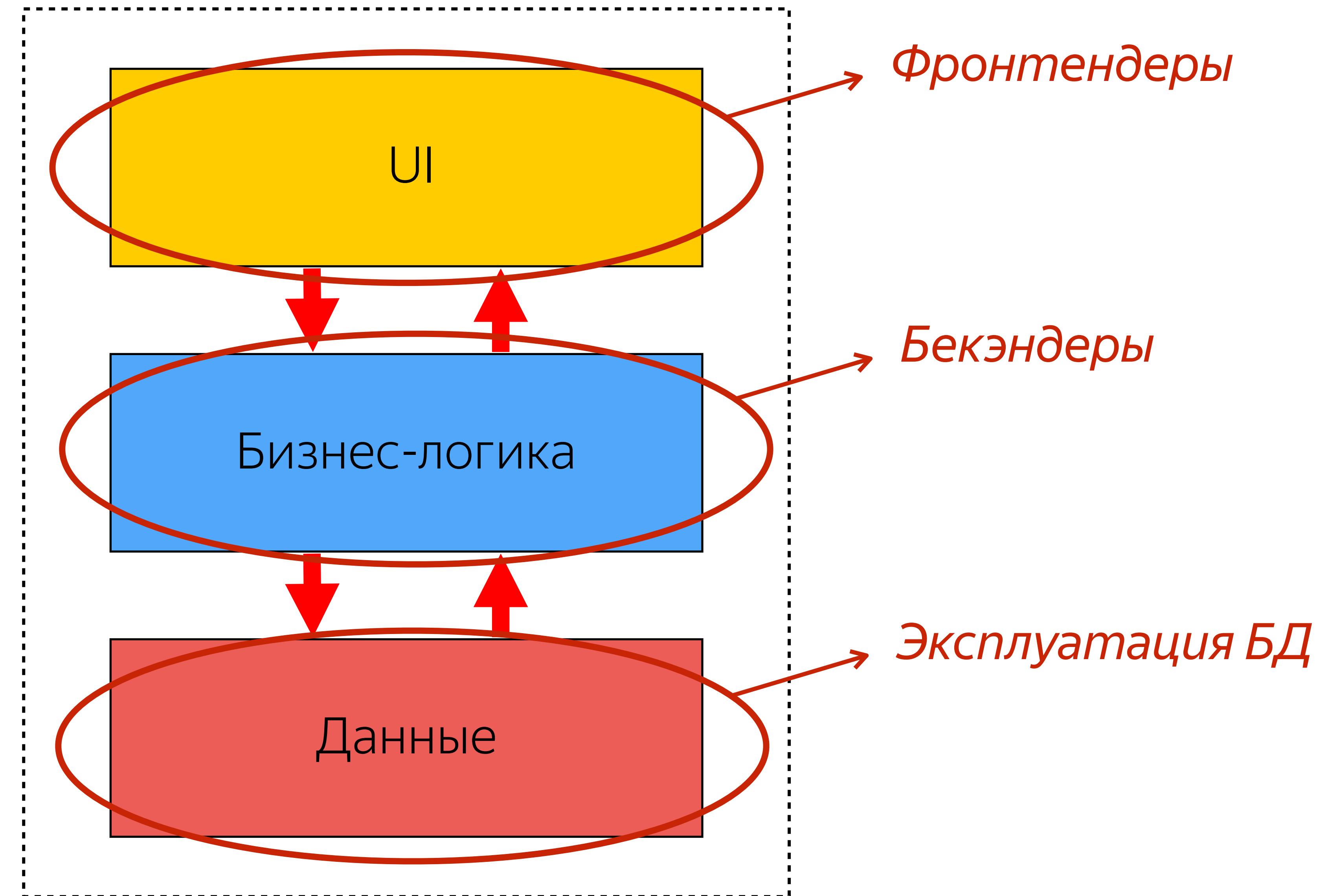
Бекендеры



Организации, проектирующие
системы, ограничены
дизайном, который копирует
структуре коммуникации в
этой организации

Мелвин Конвей

Монолит



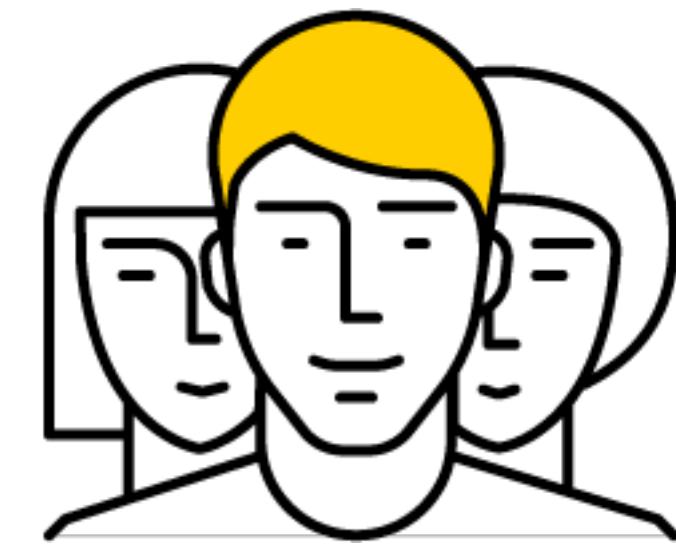
Структура ИТ-организации. Развитие



Команда 1

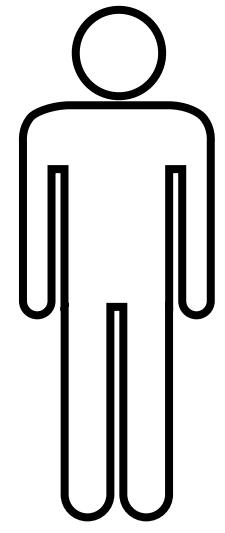


Команда 2

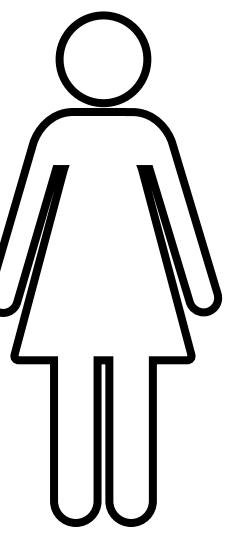


Команда 3

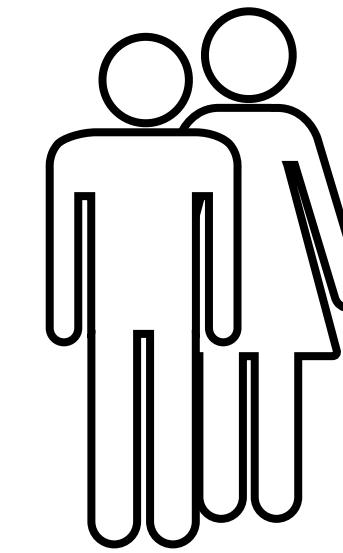
Команда



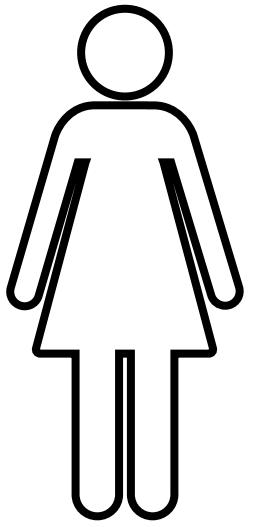
PO



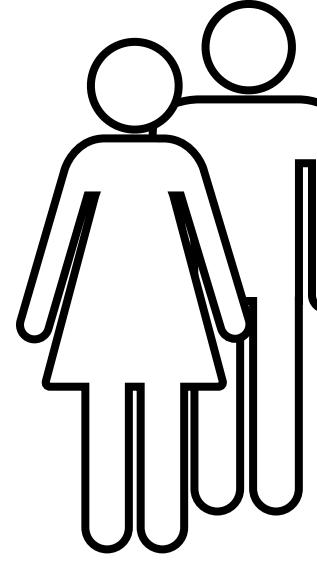
Ops



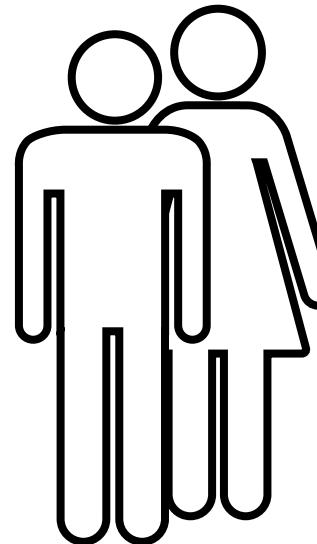
Frontend



PM

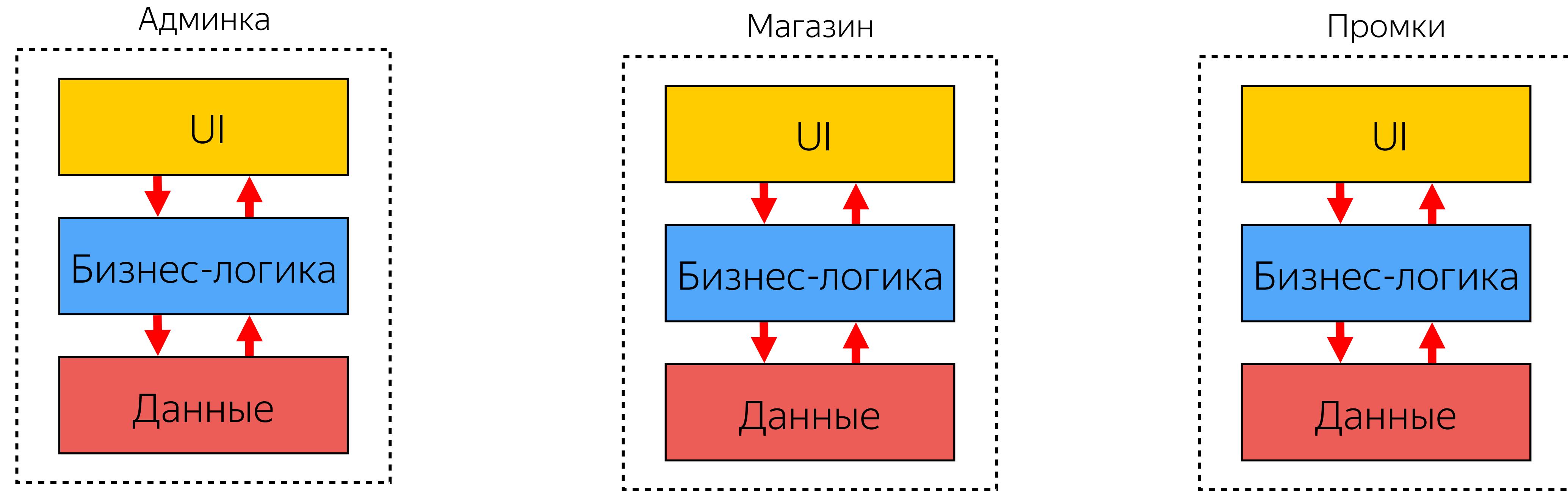


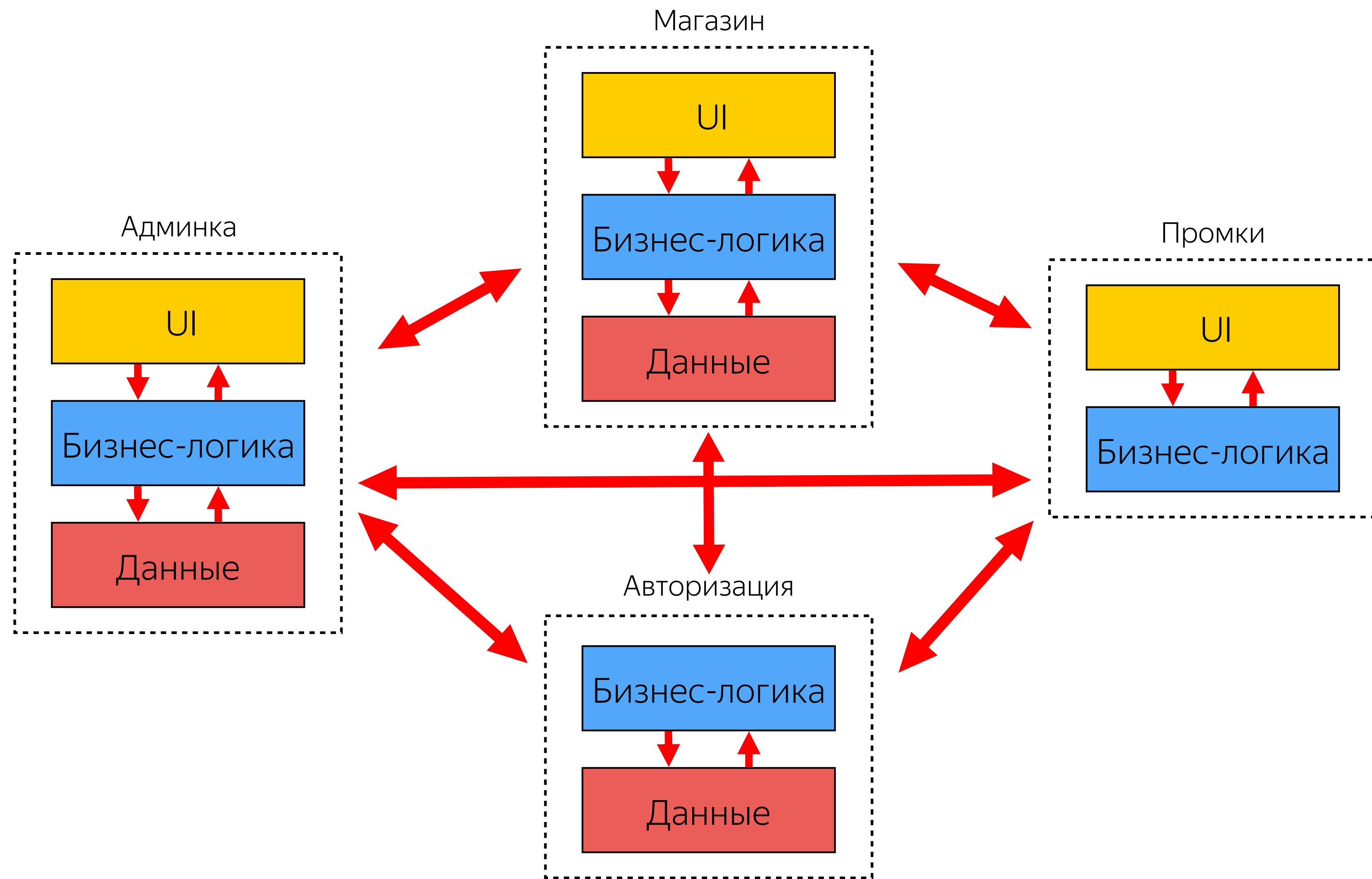
Backend

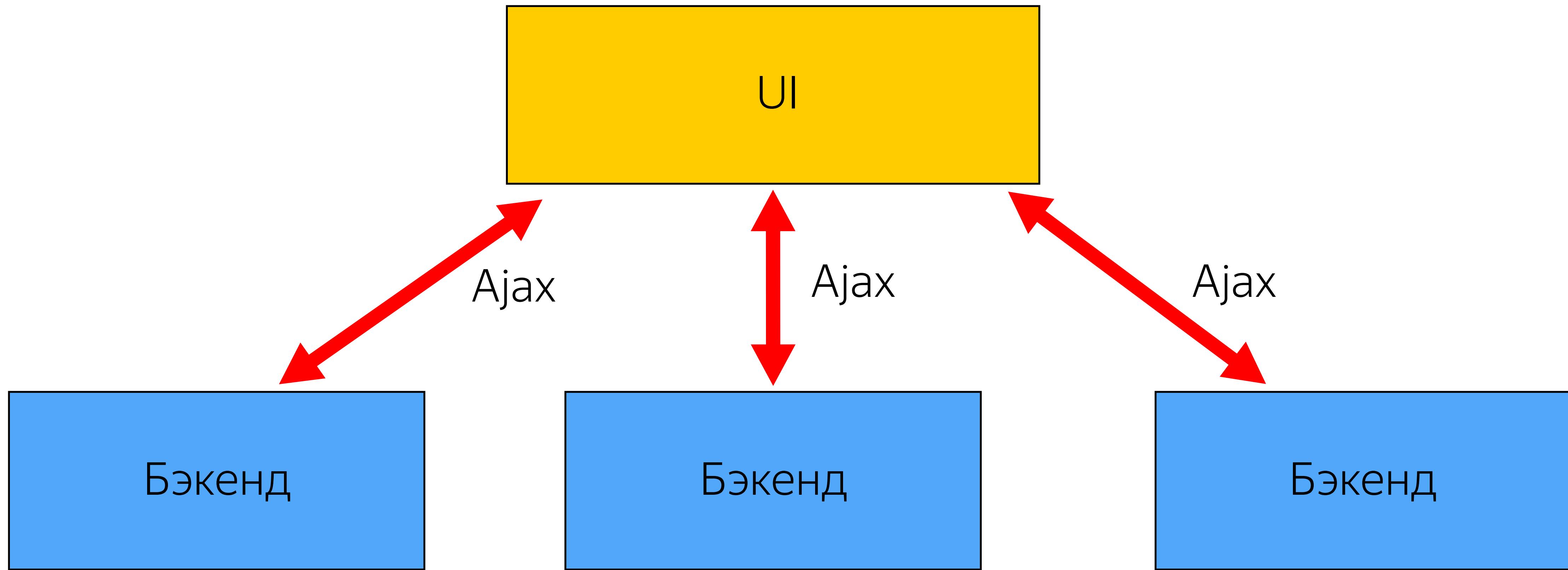


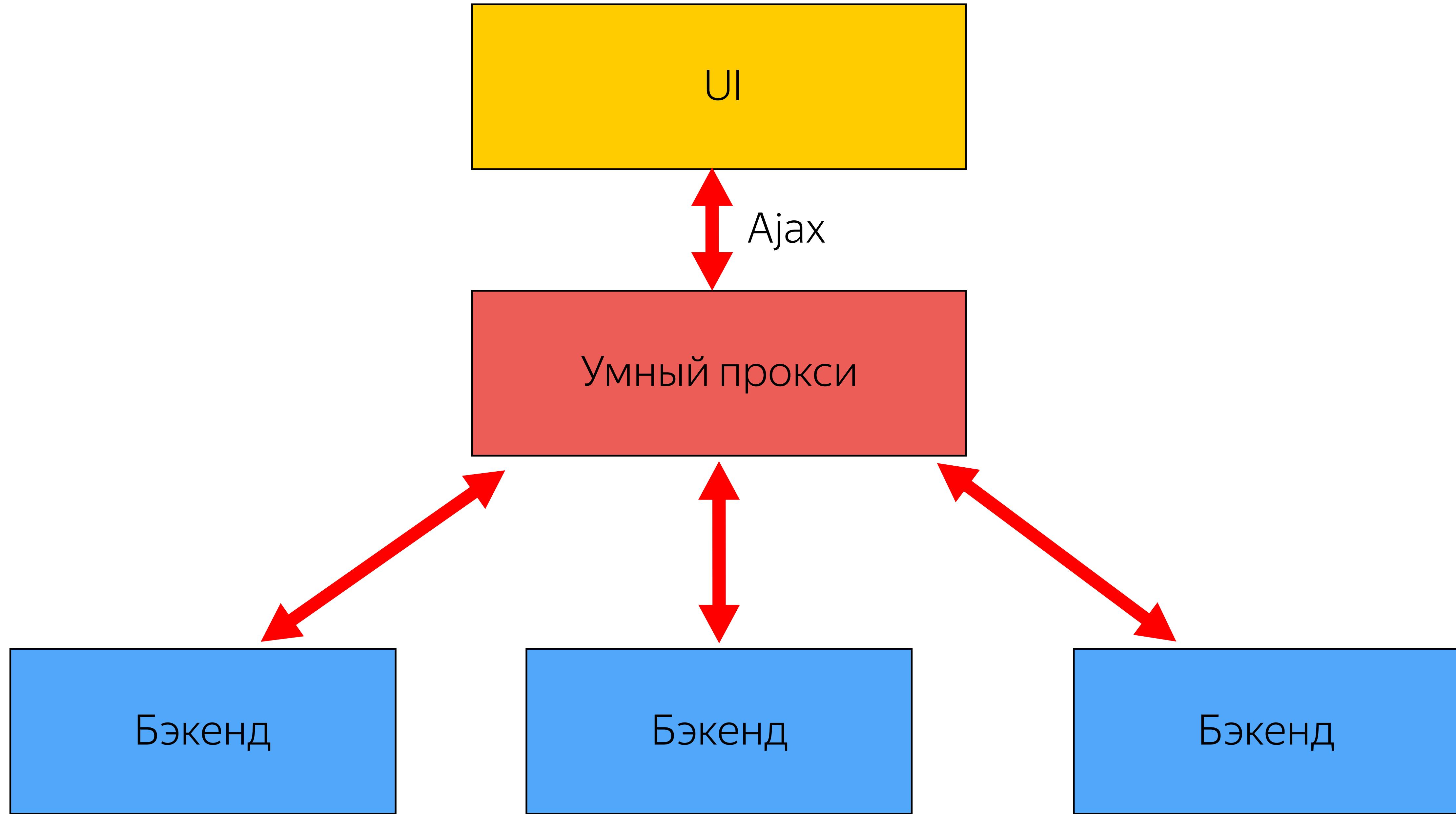
QA

Микросервисы!









Кто будет писать прокси?



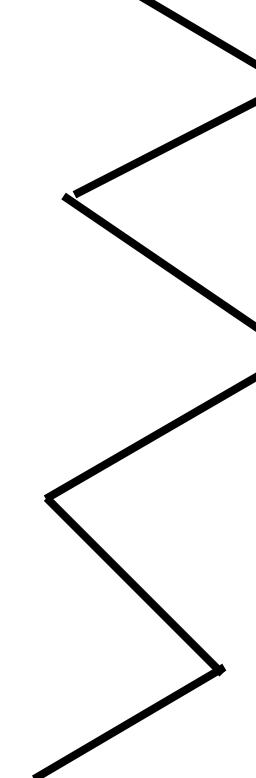
Сервер

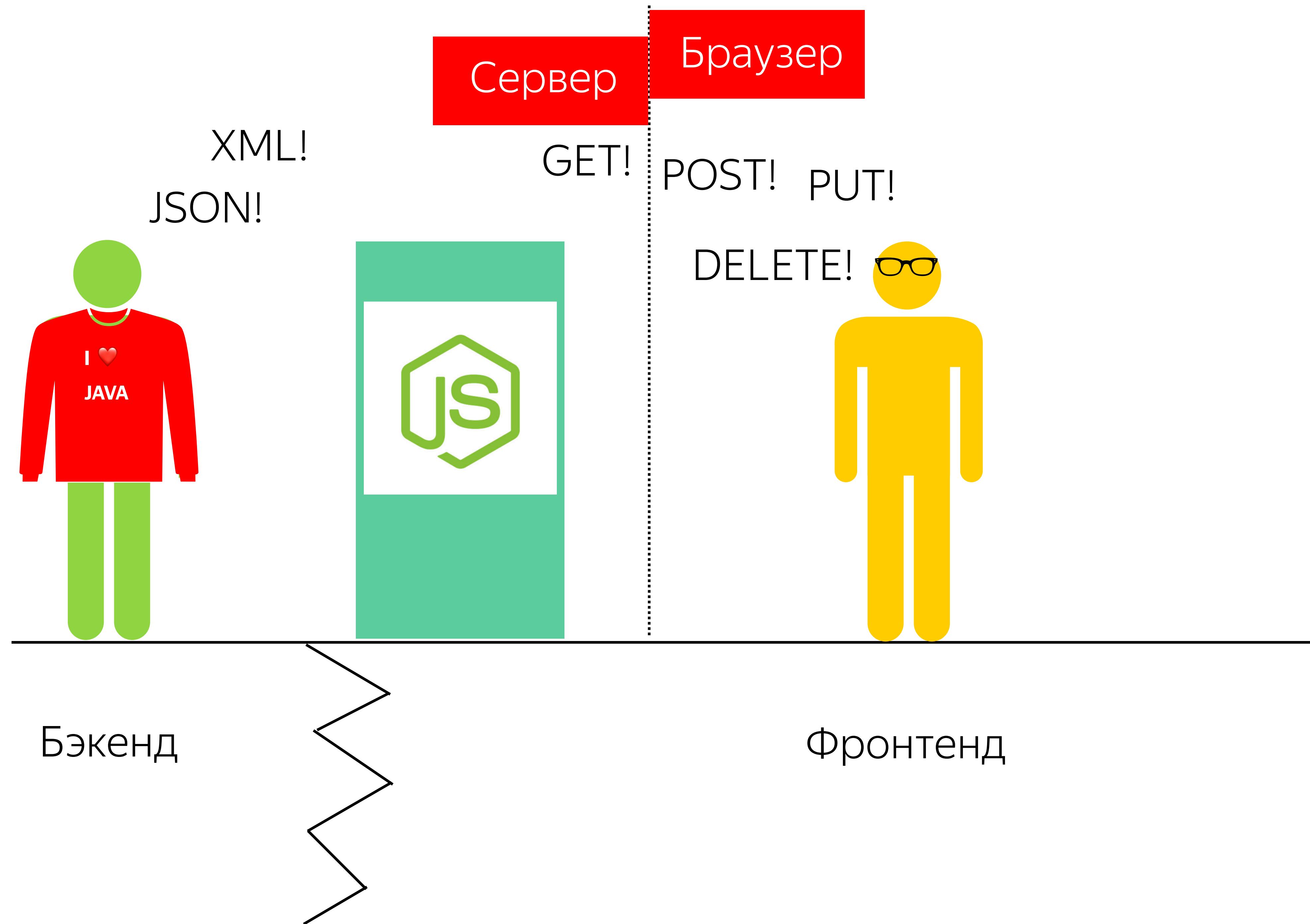
Браузер

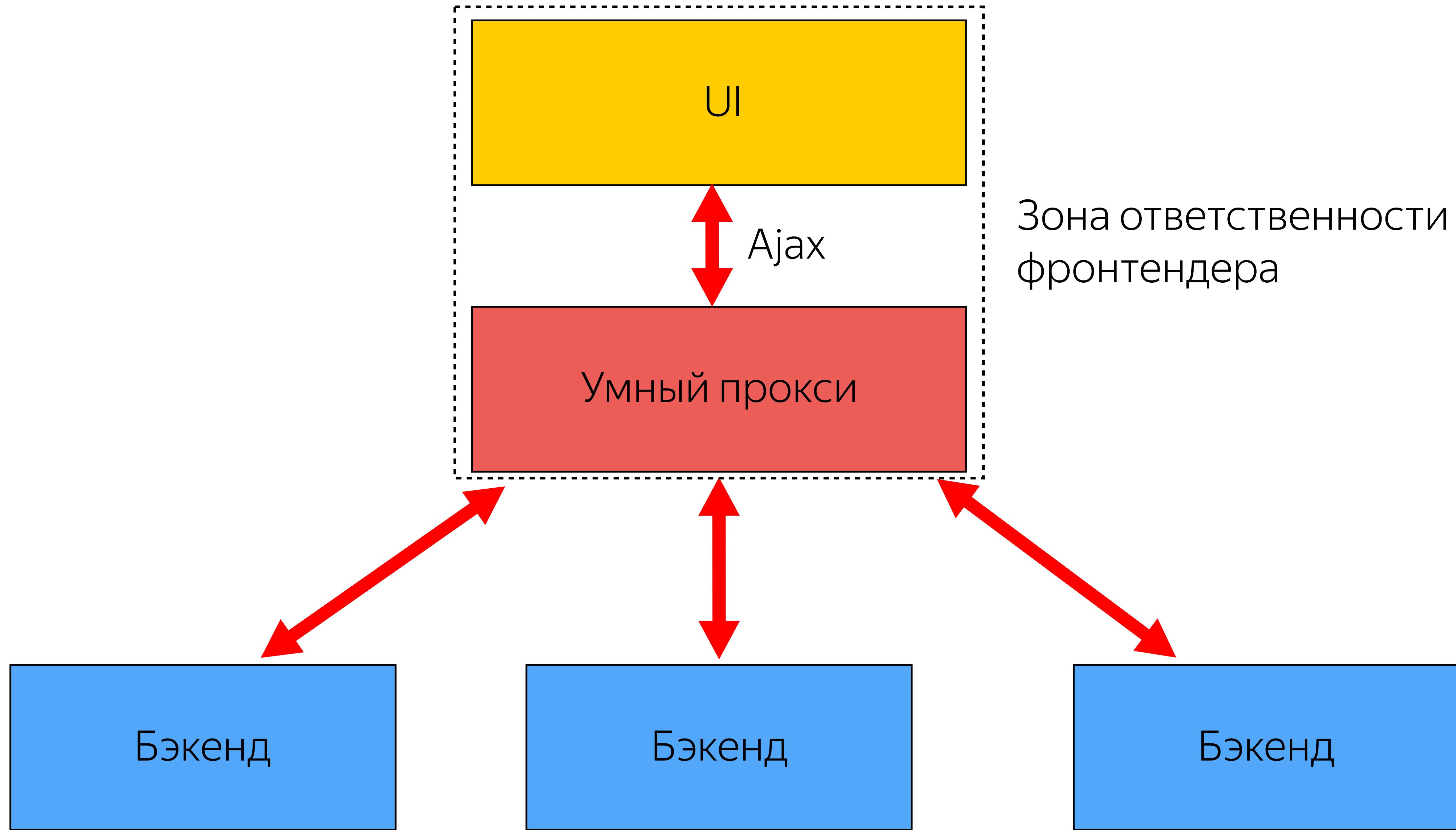


Бэкенд

Фронтенд







Почему нода???





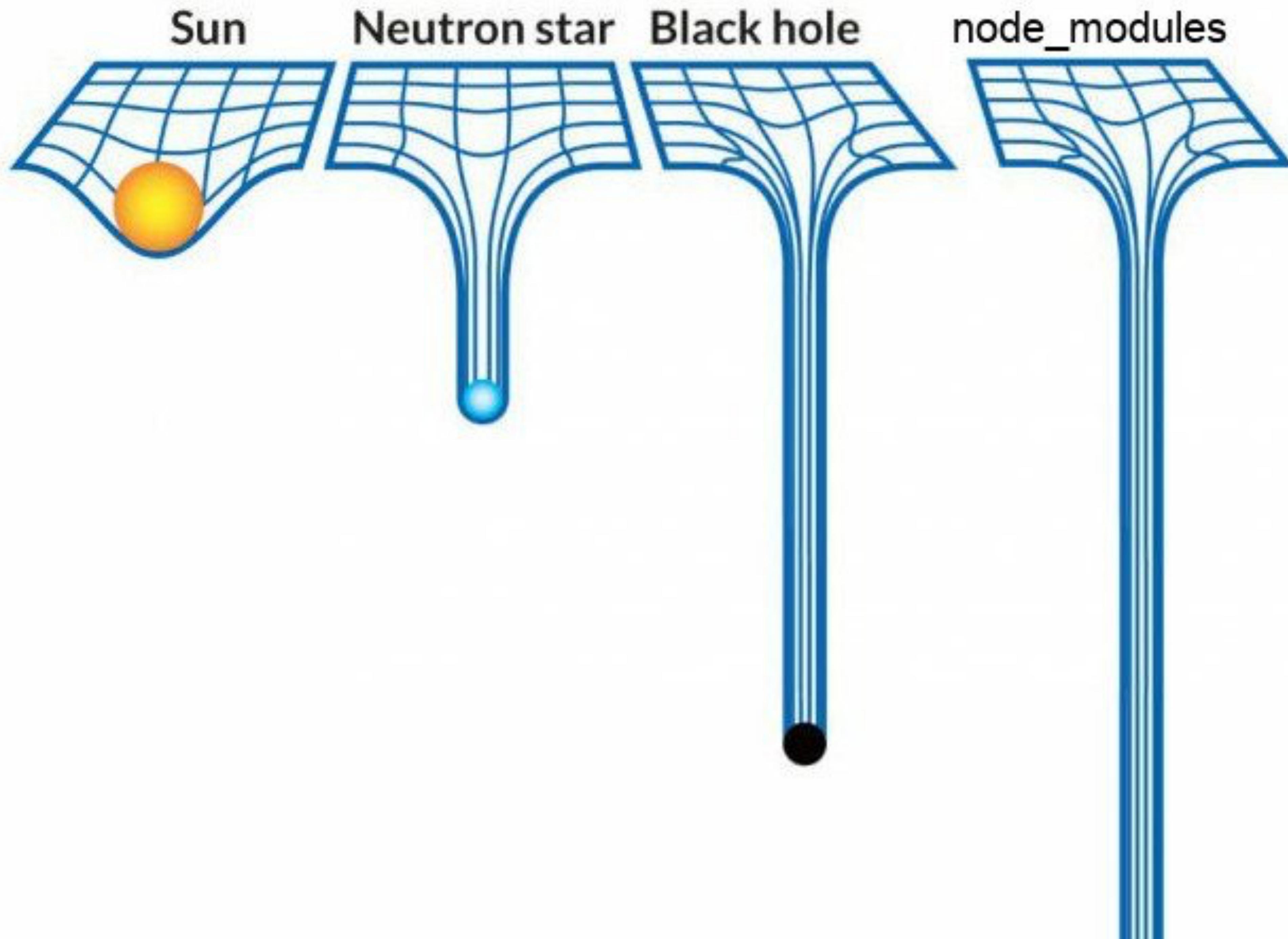
NOOOOOOOOOOOOOOOOO!!

Три причины

- › Быстрая
- › Позволяет не переключать контекст разработчика
- › SSR — Server Side Rendering или рендеринг на сервере

Node.js — познакомимся поближе







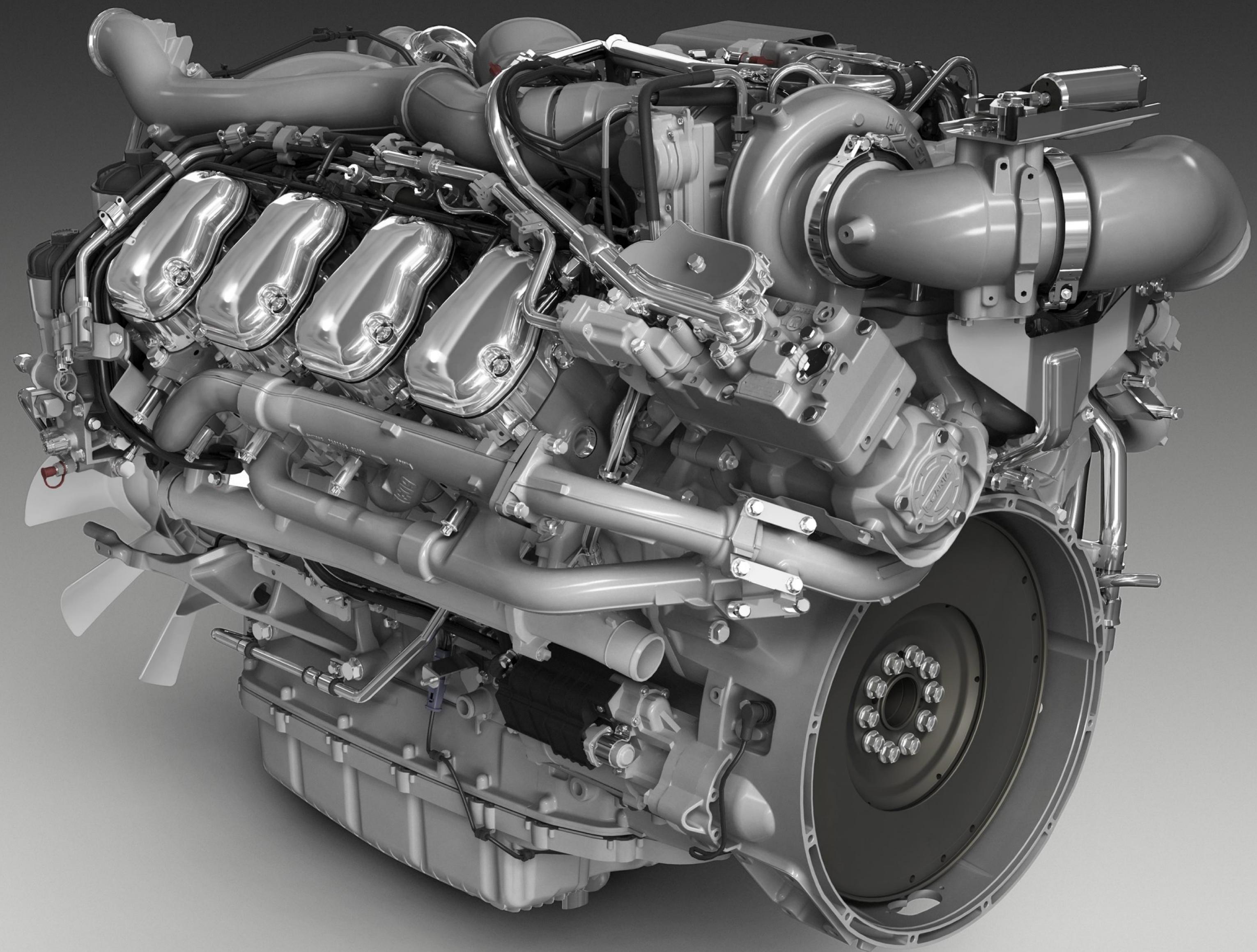
V8

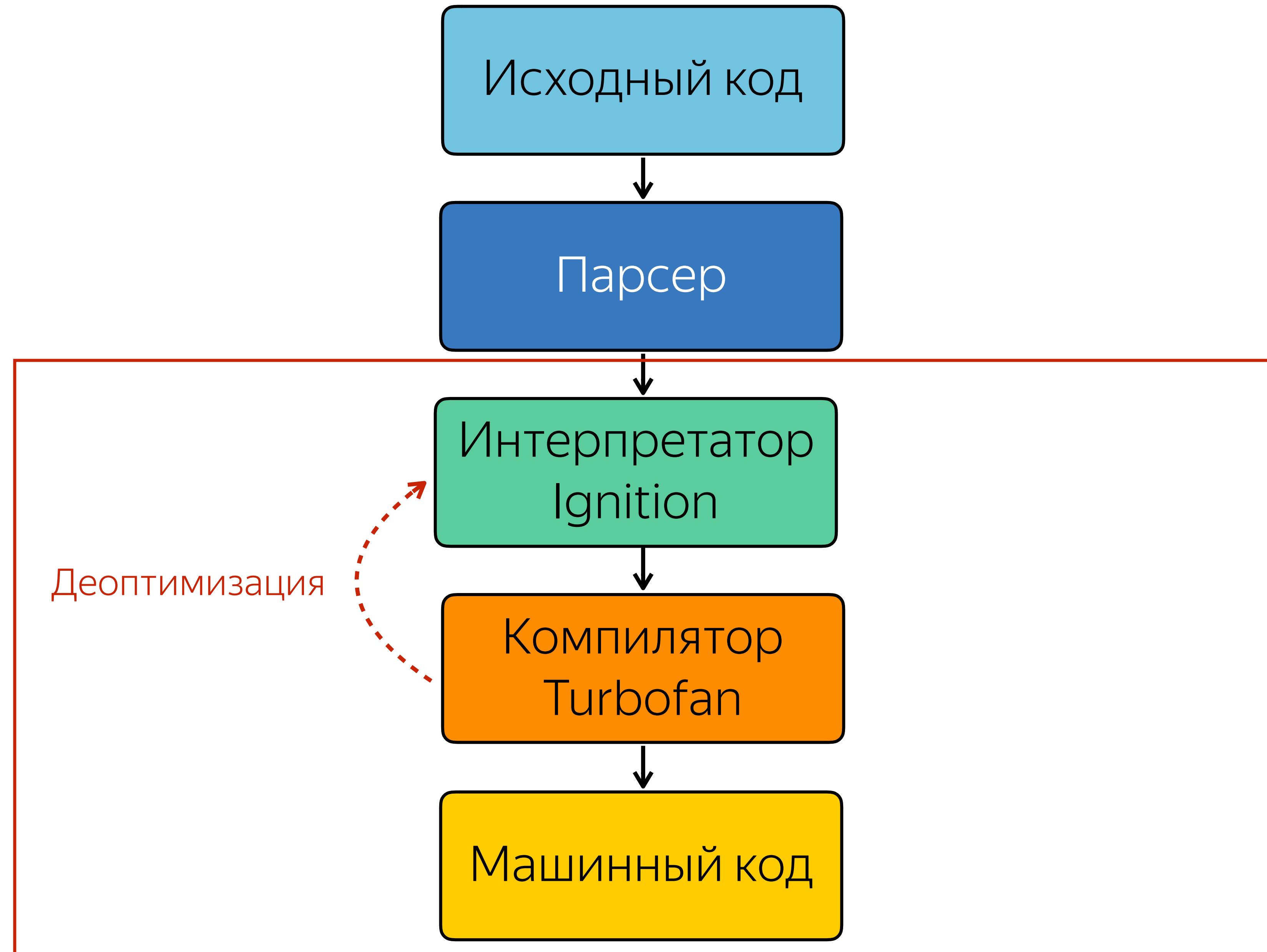
+



LibUV









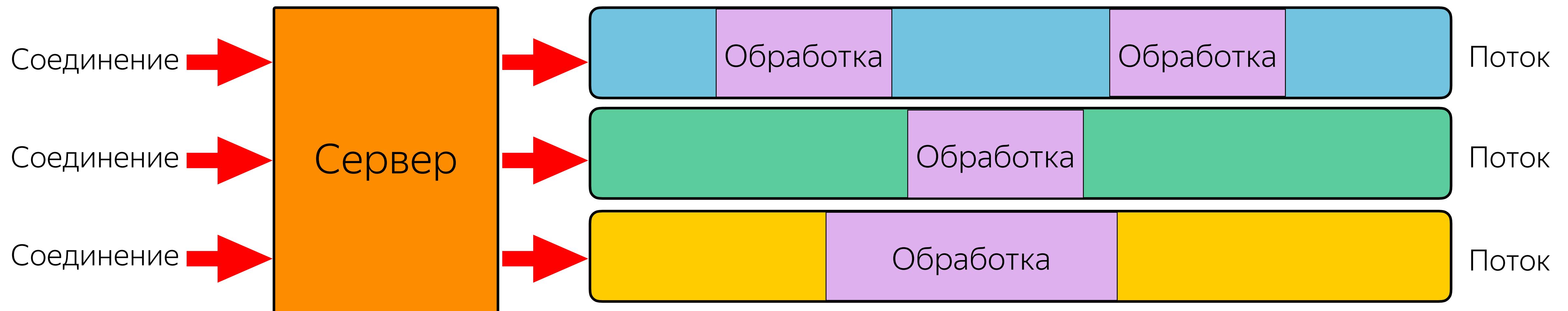
Парадокс: асинхронное
API в синхронном
однопоточном языке



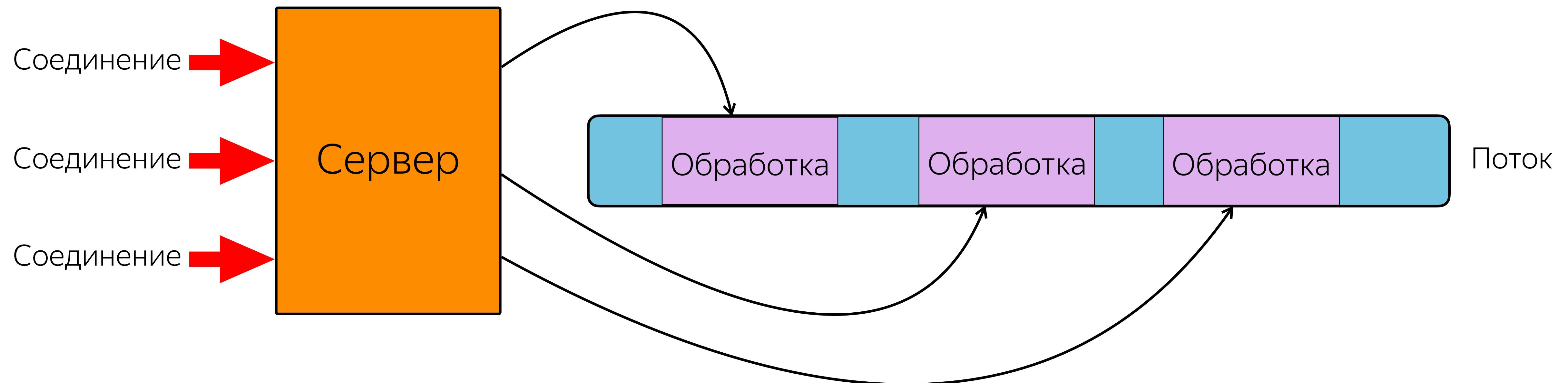


Кроссплатформенные
операции I/O,
таймеры, работа
с файлами и сетью

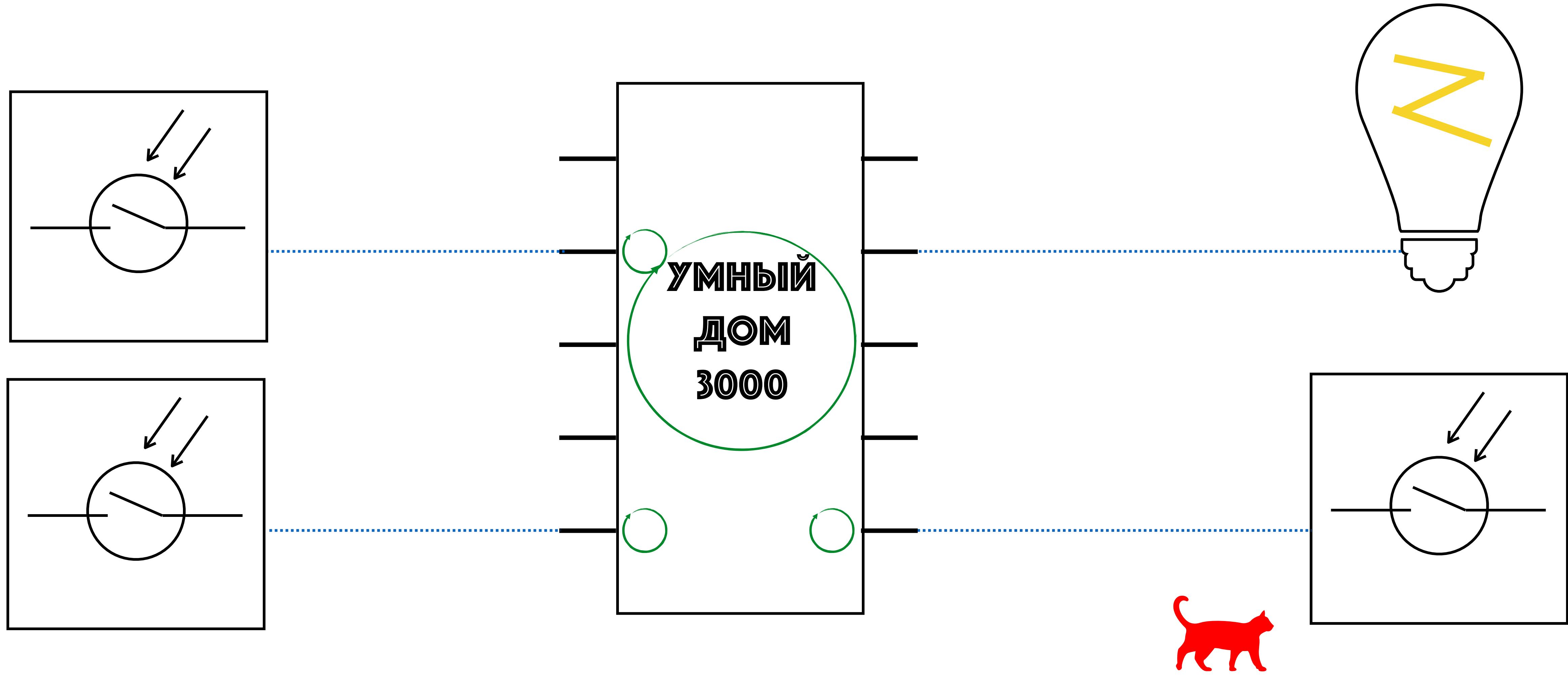
«Классический» сервер



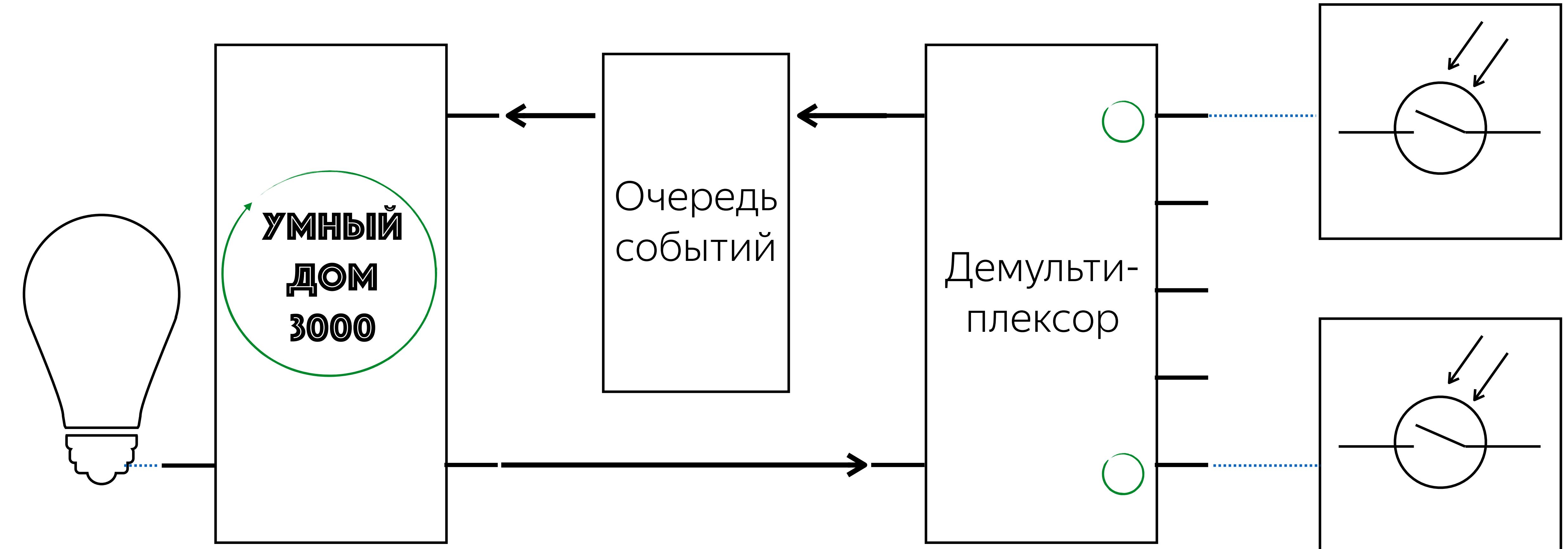
Node.js



Цикл ожидания



Цикл событий (Event loop)



Пользовательский код

Node.js

Core JavaScript API

Нативные биндинги

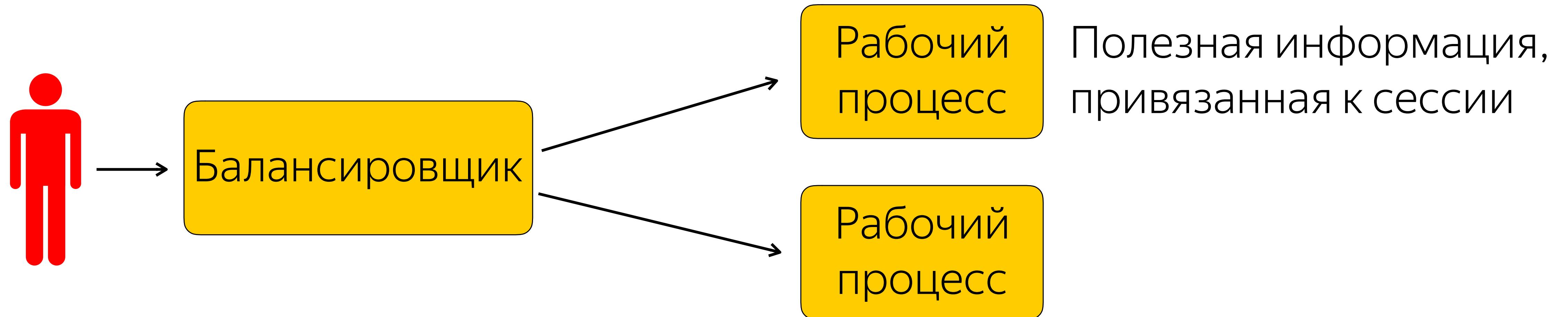
v8

libuv

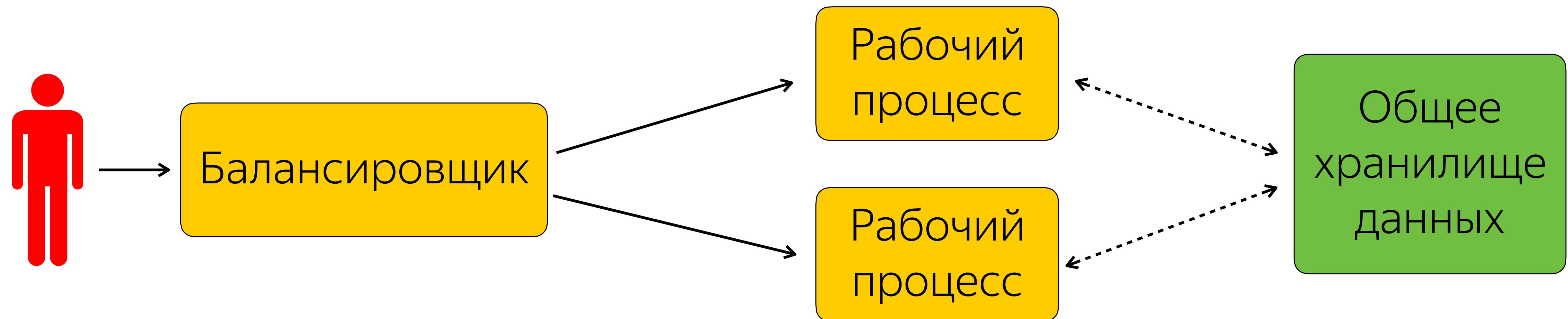
Приложения без состояния (stateless)



Потеря данных



Внешнее хранилище



Что дальше?



Меньше бэкенда!



Я серьёзно!

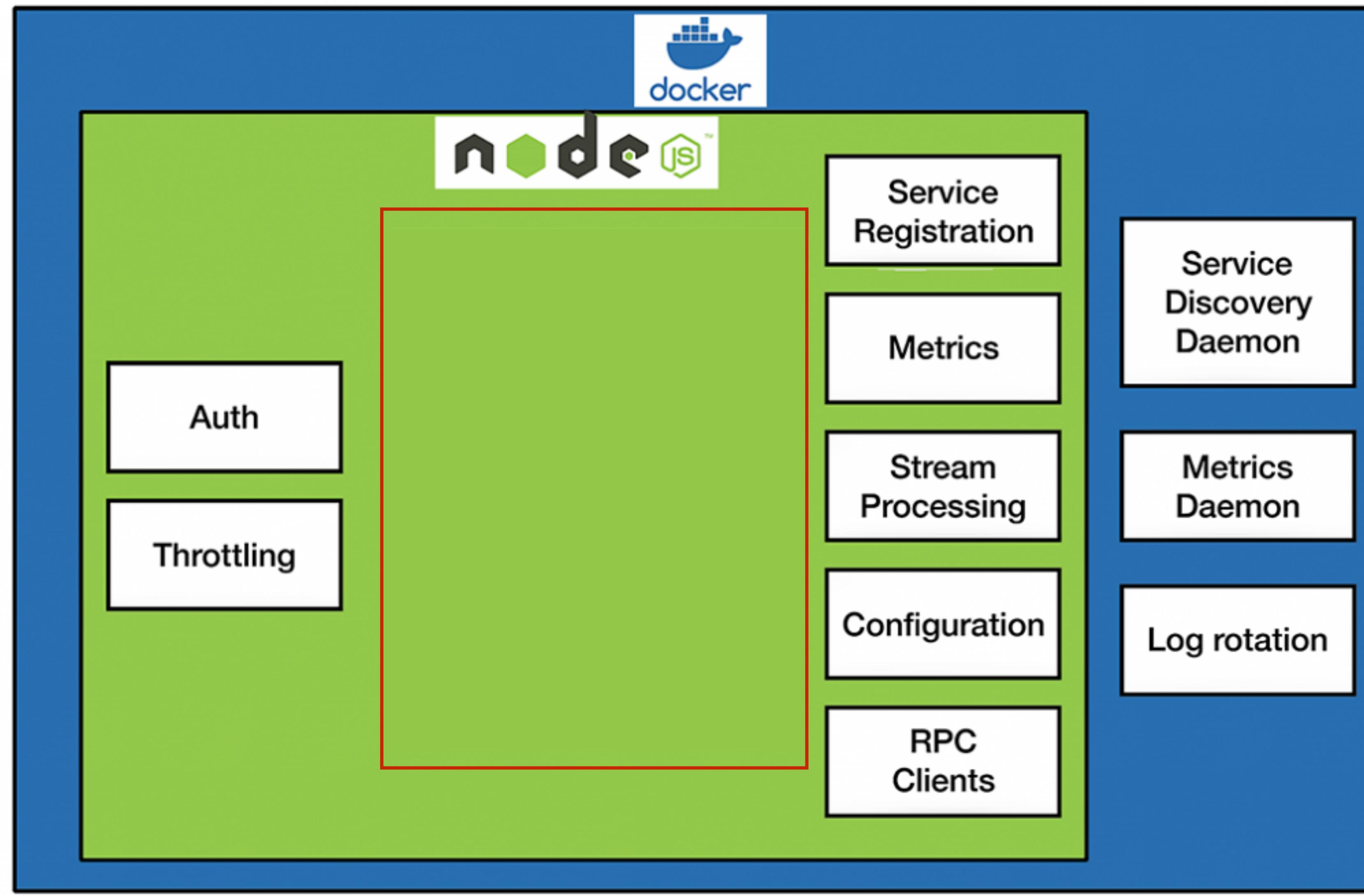


Serverless

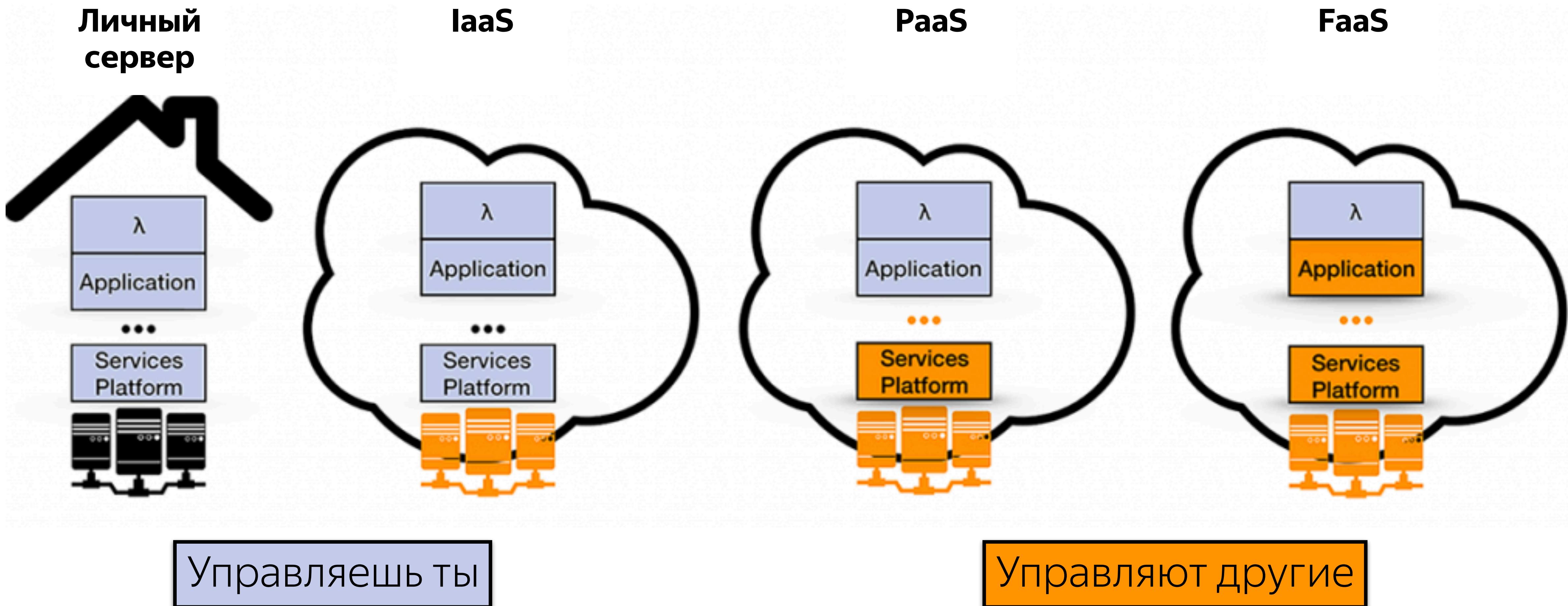
FaaS

Function-as-a-Service

Netflix NodeQuark FaaS platform



Эволюция

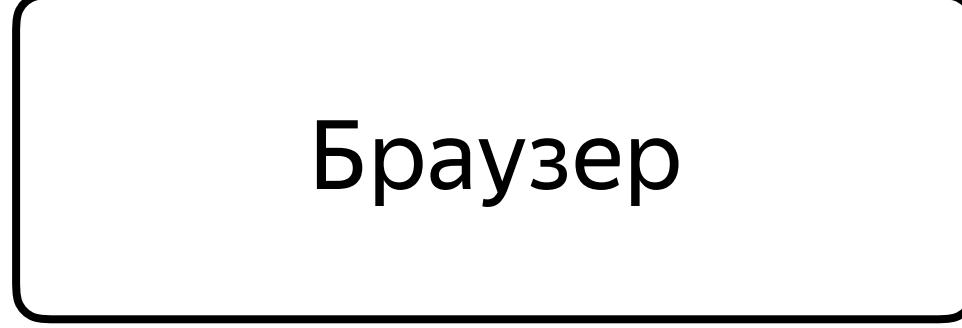


Итого



Где стоит применять Node.js?

Фронтенд



Бэкенд

Рендер HTML
API
Прокси

«Тяжёлая»
бизнес-логика

Развитие Node.js

- › Идёт быстро и стабильно
- › Node.js Fundation
- › First-class citizen в Google



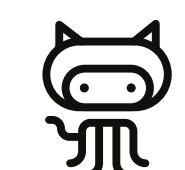
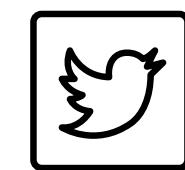
Какую версию Node.js
выбрать?

Версии Node.js

Релиз	Статус	Дата выпуска	Начало LTS	Переход в статус Maintenance	Окончание жизни
6.x	Maintenance LTS	2016-04-26	2016-10-18	2018-04-30	April 2019
8.x	Active LTS	2017-05-30	2017-10-31	April 2019	December 2019
10.x	Active LTS	2018-04-24	2018-10-30	April 2020	April 2021
11.x	Current Release	2018-10-23			June 2019

Спасибо за внимание!

Андрей Мелихов



amel_true

amel-true