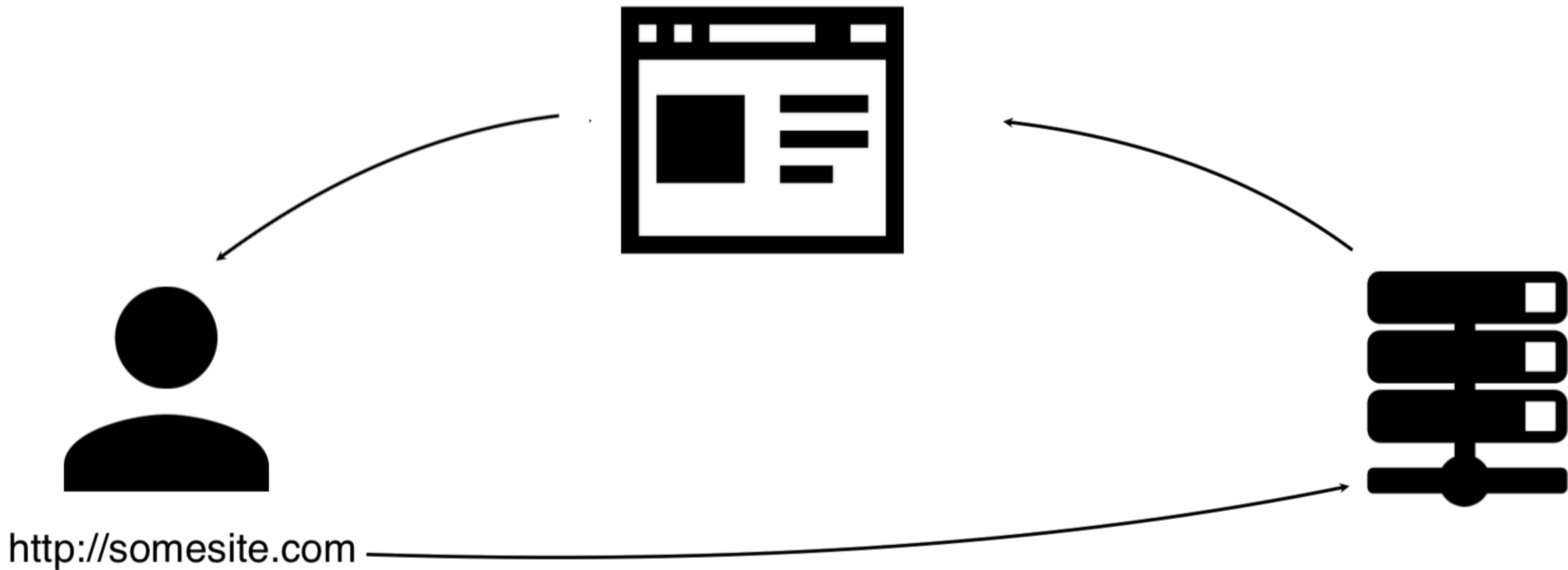


Vue и современные front-end приложения

Часть первая

SPA - что, зачем и почему





Single page application

Преимущества SPA

- Быстродействие
- Разграничение Front-end и back-end
- Возможность строить mobile-app и web-app на одном backend`е
- Выше скорость разработки
- Модульный подход к разработке

Минусы SPA

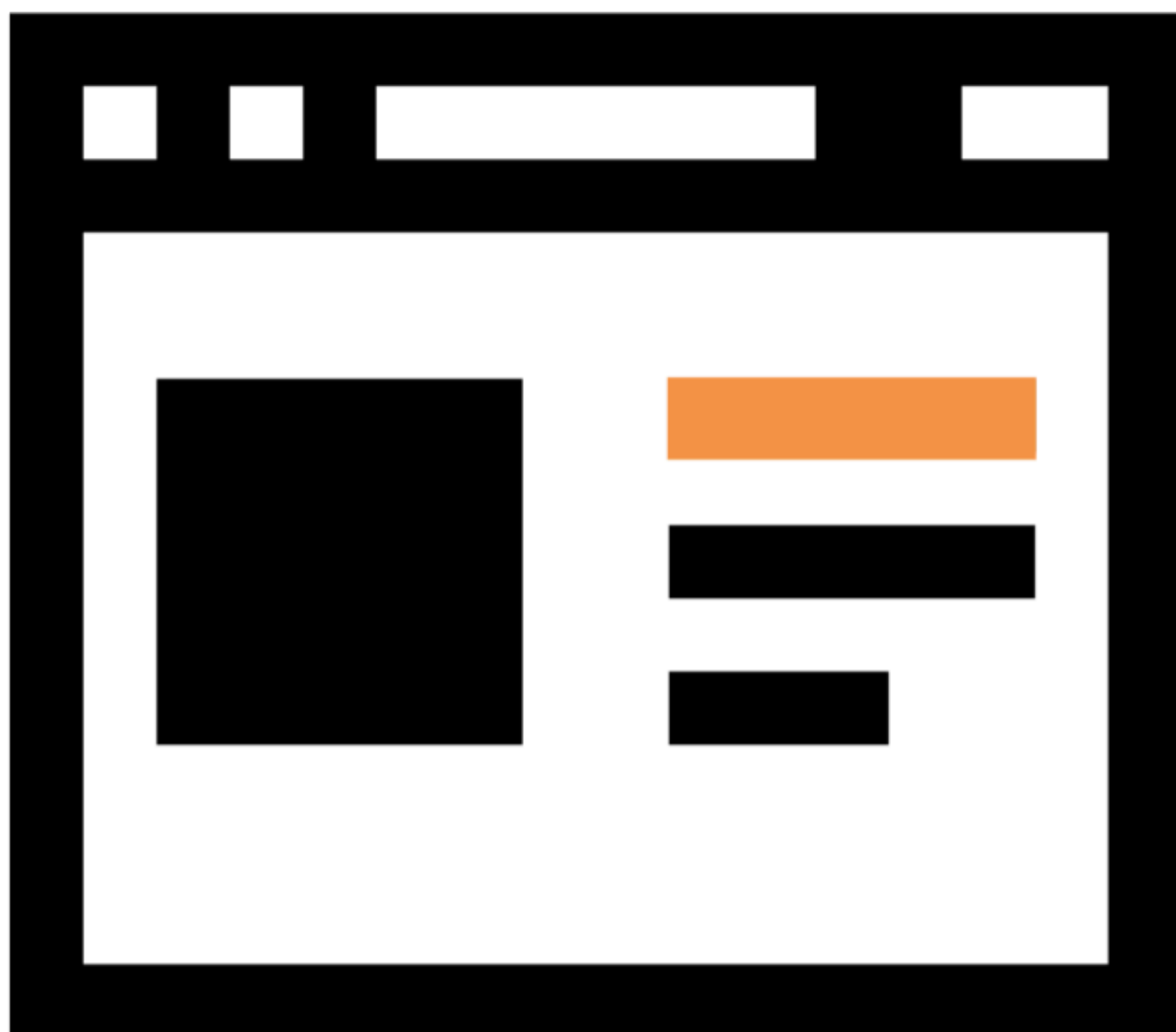
- SEO
- Утечки памяти в JS(Связанные с monkey coding)

Часть вторая

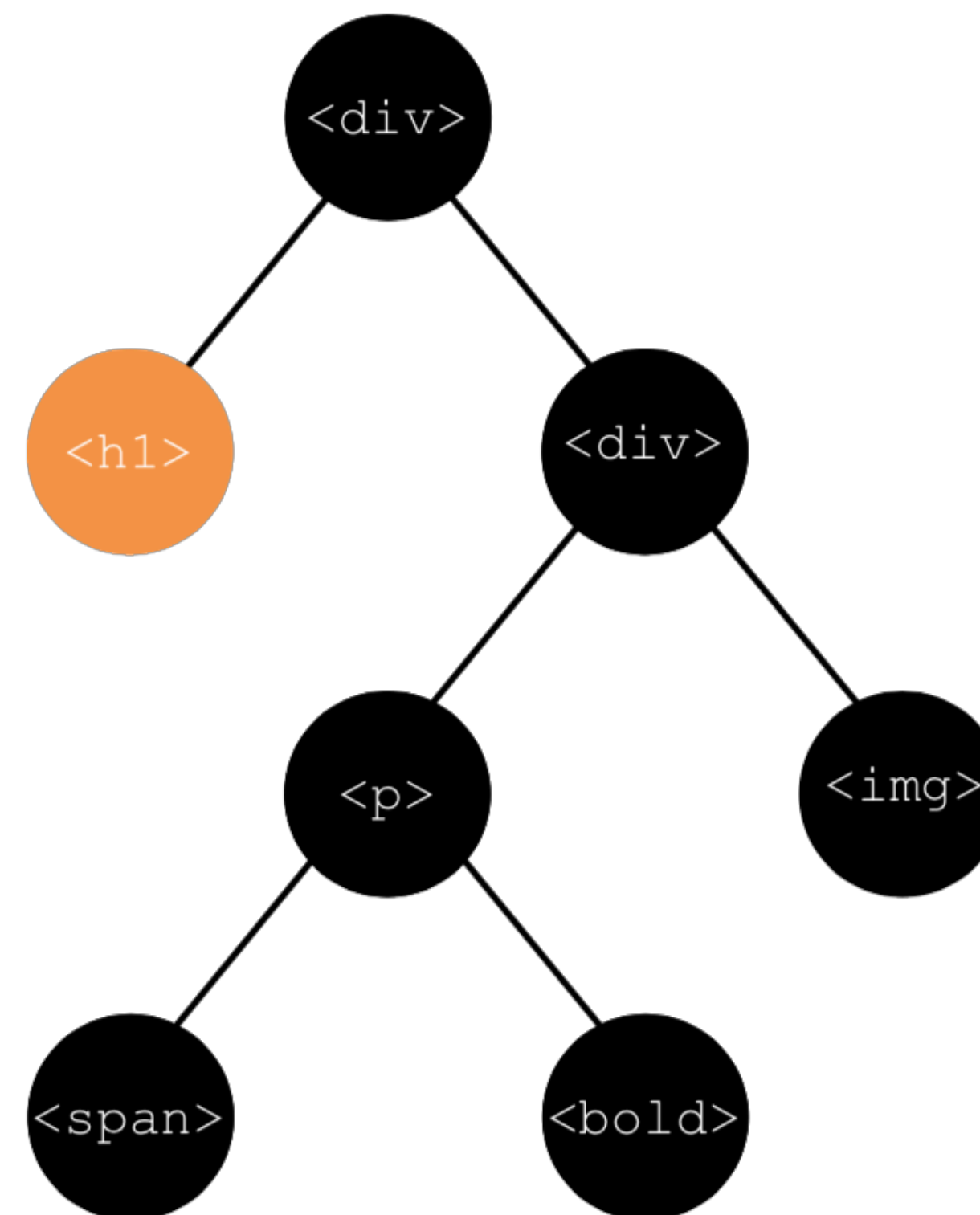
Vue - как это устроено

Document Object Model

DOM (аббревиатура от Document Object Model) — способ представления структурного документа с помощью объектов. Это кроссплатформенное и языко-независимое соглашение для представления и взаимодействия с данными в HTML, XML и т.д.



Страница



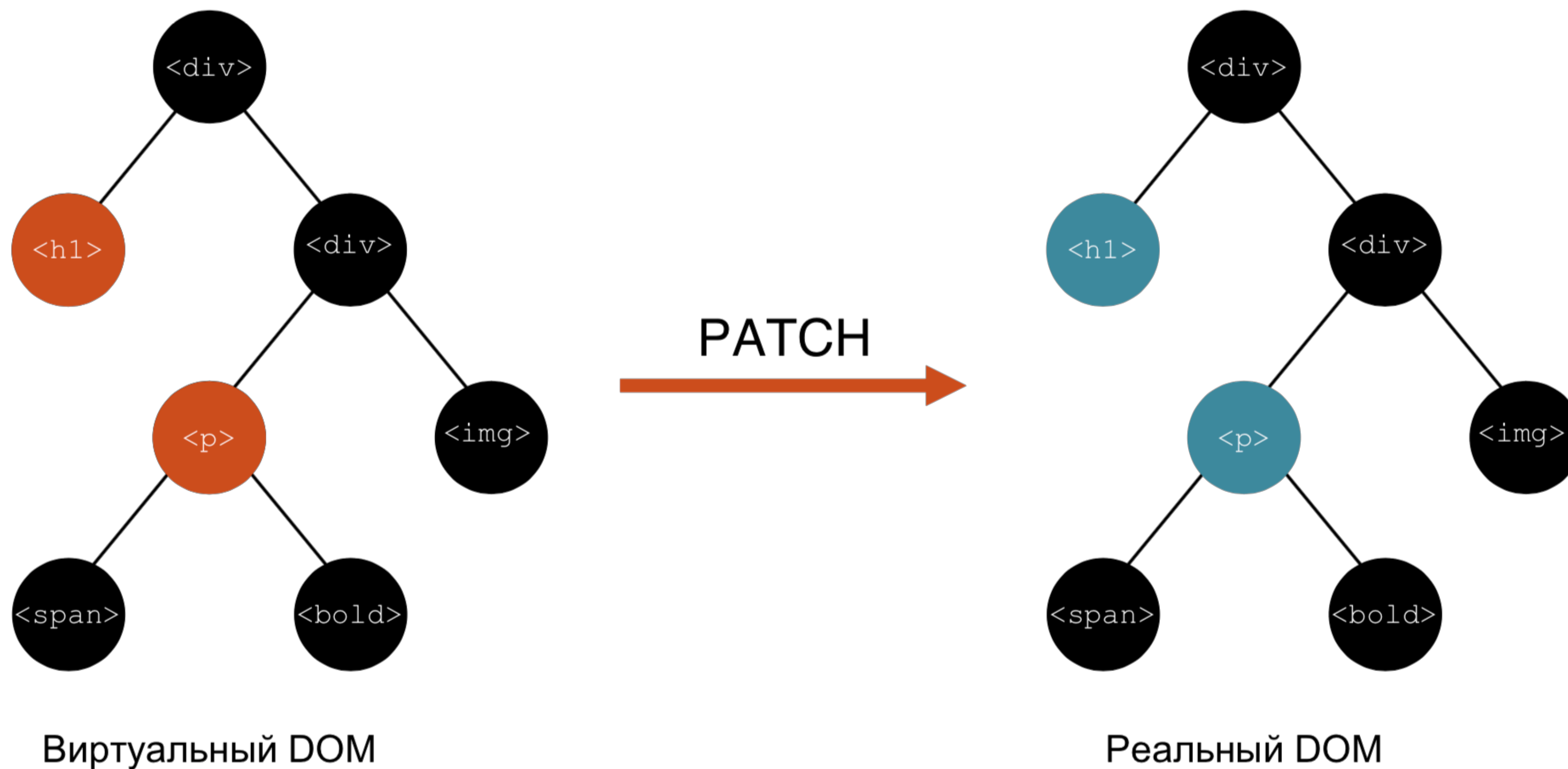
DOM дерево

Вывод

Виртуальный DOM

Вместо того, чтобы взаимодействовать с DOM напрямую, мы работаем с его легковесной копией. Мы можем вносить изменения в копию, исходя из наших потребностей, а после этого применять изменения к реальному DOM.

При этом происходит сравнение DOM-дерева с его виртуальной копией, определяется разница и запускается перерисовка того, что было изменено.



Изменение

Часть третья

Компонентный подход и компоненты в Vue

Компоненты в Vue

Компонент в Vue - представляет собой разграниченный файл, на 3 блока, `template`(html с возможностью вывода данных и директивами), `script`(js собственно логика нашего компонента) и `style`(стили для компонента).

Компоненты подход

Компонентный подход подразумевает - разбиение приложения на мелкие компоненты(Блоки), что даёт возможность разработчику переиспользовать компоненты в нужных местах, иметь небольшие блоки кода в разных файлах, что упростит задачу отладки и внесения изменений в проект.

News list

```
<template lang="html">
  <div class="list-items">
    <news-item v-for='news in newsList'
      :key='news.id'
      :news='news'>
    </news-item>
  </div>
</template>

<script>
import news from '@/store/news.js'

import newsItem from './news-item.vue';

export default {

  components: {
    newsItem
  },

  computed: {

    newsList: () => {
      return news;
    }

  }

}
```

News item

```
<template lang="html">
  <div class="item">
    
    <h2>{{ news.title }}</h2>
    <p>{{ news.lead }}</p>
  </div>
</template>

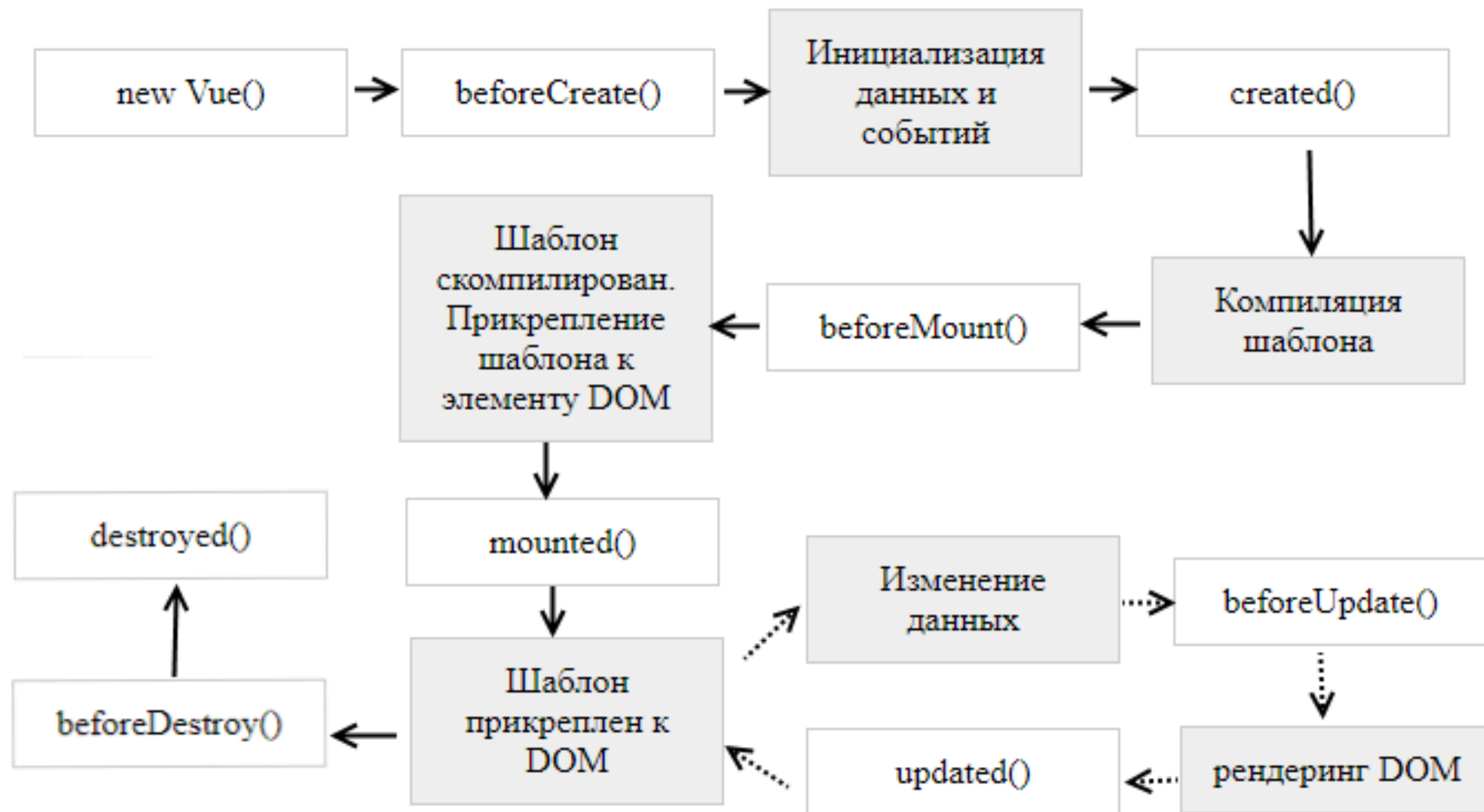
<script>
export default {

  props: {
    news: {
      type: Object
    }
  }

}

</script>

<style lang="css">
</style>
```



Цикл жизни компонента

Часть четвертая

Vue в цифрах и сравнениях

Vue на сегодня

- 724к+ загрузок на npm
- 50к+ сайтов индексированы wappalyzer
- Целостная экосистема(Vuex, Vue-routing, nuxt, etc.)
- 40к+ плагинов для vue
- Один из самых легковесных front-end framework (из большой четверки Vue, React, Ember, Angular самый легковесный).
- Офф партнер laravel(из под коробки уже вшит Vue)
- Лучшая документация из популярных framework`ов
- 9gag написан на vue, а мемасики это важно:)



React vs Vue

- React и Vue очень похожи, Они оба используют Virtual DOM, предоставляют реактивность и компонентную структуру
- Для написания React приложений нужен опыт в JSX в Vue этой проблемы нет
- Vue слегка быстрее из-за более легковесной реализации Virtual DOM.
- React был создан Facebook для своих нужд и «Под себя», Vue создавался людьми и для людей

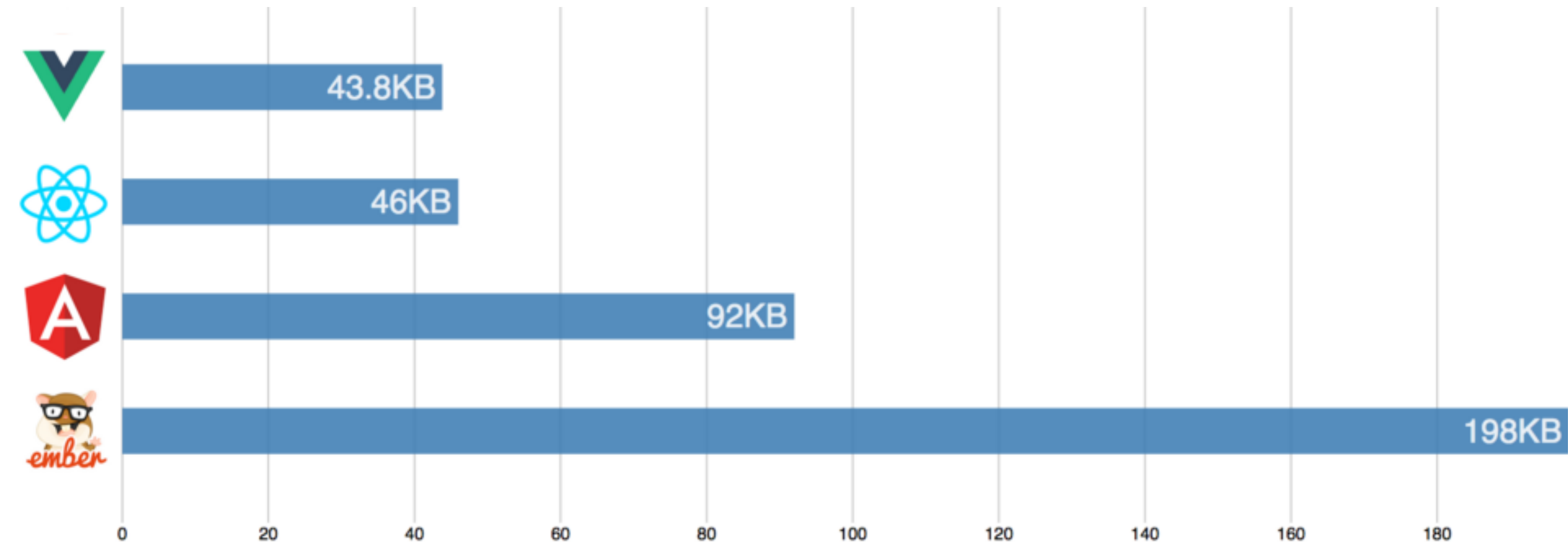
Angular vs Vue

- Синтаксис template очень схож на Angular, но в остальном это кардинально разные вещи, просто откройте доку и попробуйте написать «Hello world».
- Angular имеет жёсткое мнение насчёт структуры вашего приложения, в то время как Vue проявляет гибкость и является более модульным решением.
- Angular после установки в крокеты займет у вас в 5 раз больше места (111kb vs 20.9kb).
- Vue производительней Angular.
- Angular - переизбыток функционала не есть хорошо.

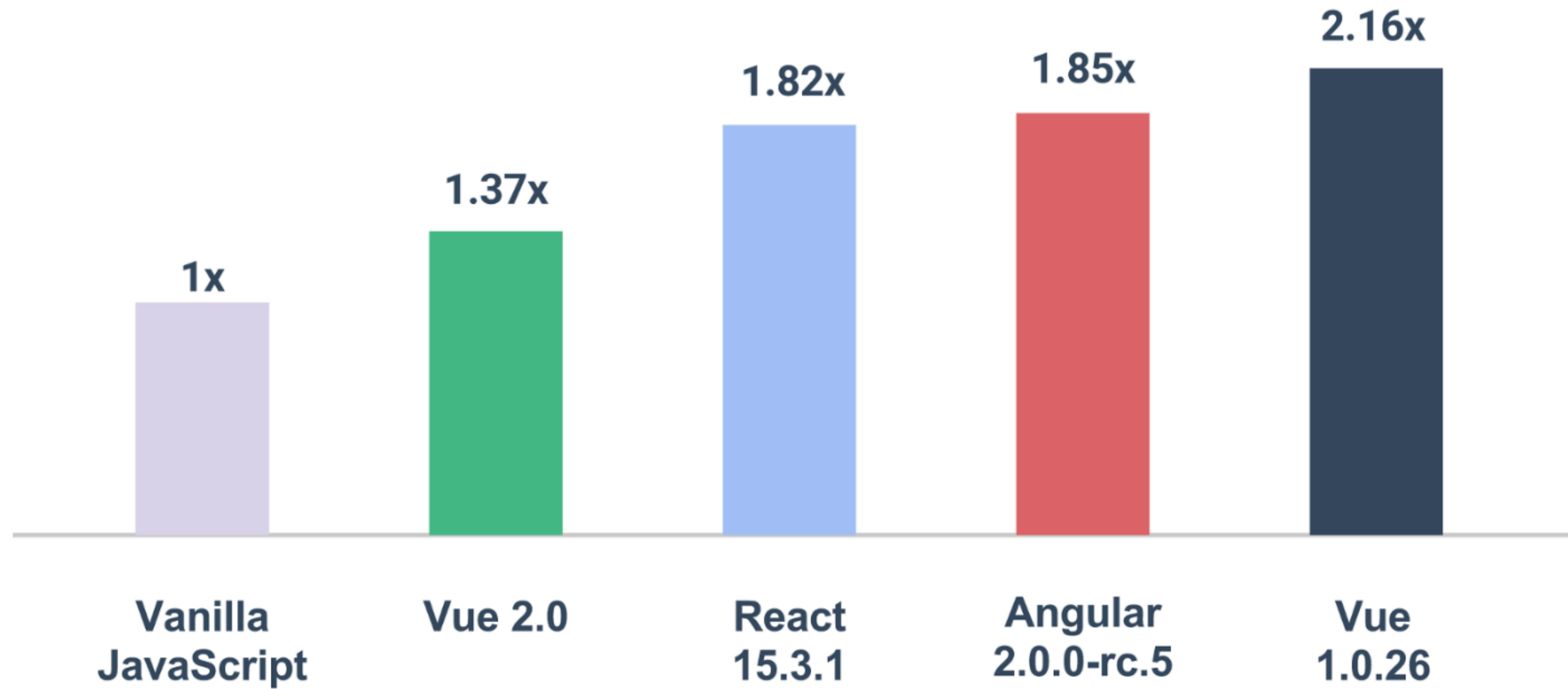
Минусы Vue

- Молодость, он менее популярный, сообщество меньше.
- React разработал и поддерживает Facebook. Angular – Google. А Vue какой-то китаец :) – Evan You.

Сравнение размера



Performance



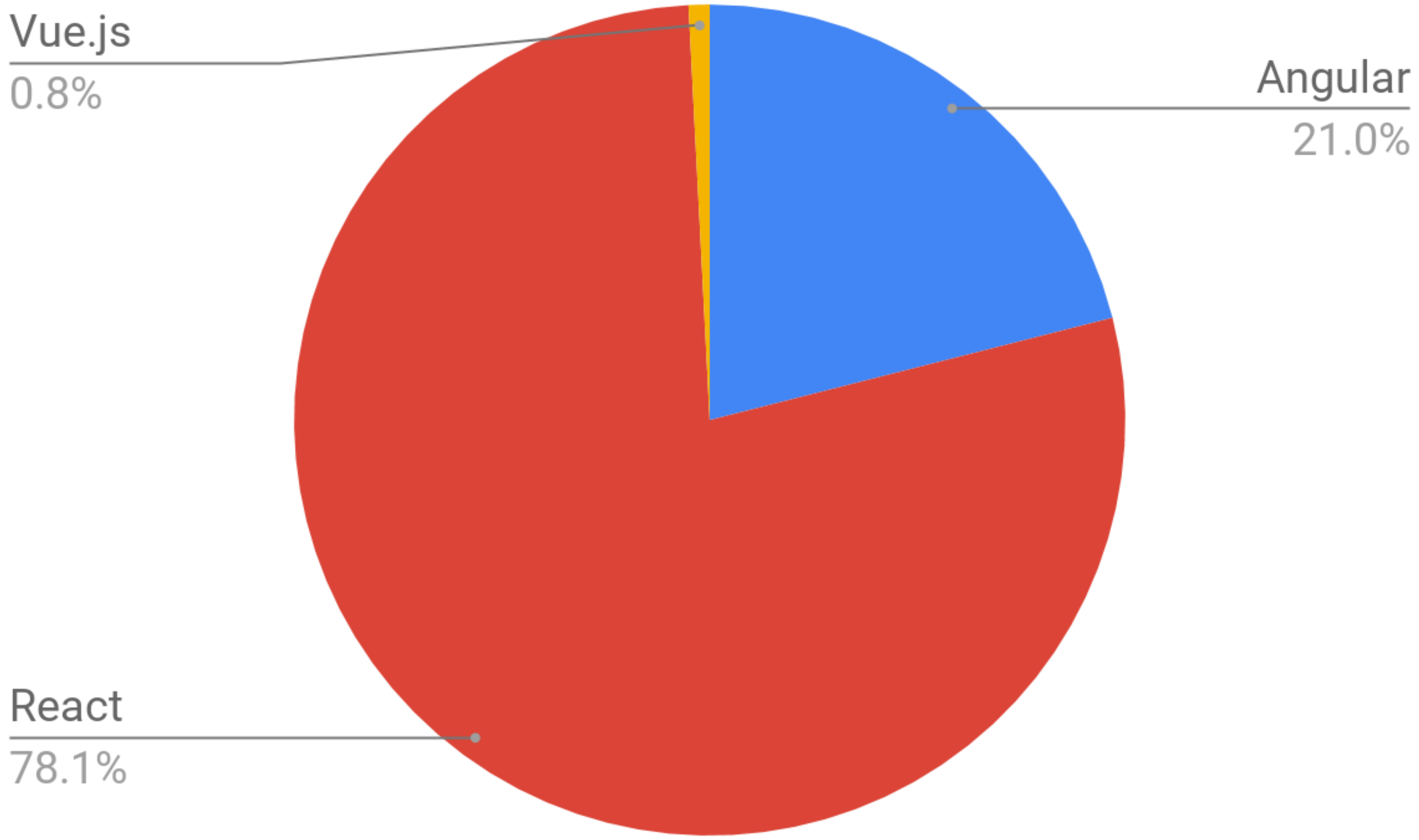
Based on 3rd party benchmark, lower is better

Выводы в сравнениях

- Vue более лояльный в использовании, у него нет перегруженности лишним функционалом, все что нужно можно доставить отдельно. За счет этого в скорости он обгоняет и React и Angular
- В плане синтаксиса у него есть сходства с Angular(темплейт) и с React(в Палне методов объекта).
- Лучшая экосистема, ни у React ни у Angular нет настолько удобной cli, а дока самая обширная.
- Порог входа ниже, не нужен JSX как с React и TypeScript как с Angular, структуру проекта вам не навязывают, а вы выбираете её сами.
- Vue моложе, а соответственно на нем меньше проектов, меньше сообщество, но посмотрим на это с другой стороны, вы можете помочь другим людям написав полезный пакет, чего увы уже не сделать в React или Angular:)

Часть пятая

Популярность и перспективы



react x

vue x

@angular/core x

angular x

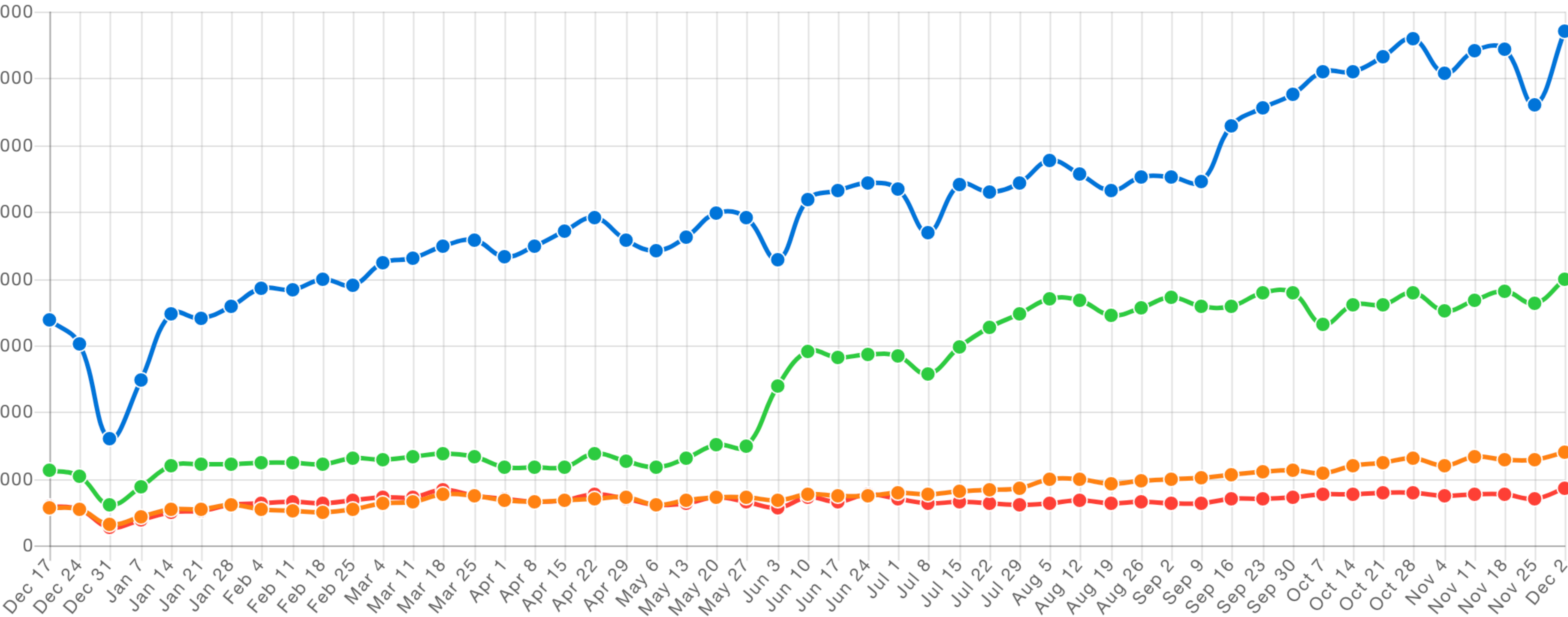
Downloads in past 1 Year

react

vue

@angular/core

angular



Но не все так плохо

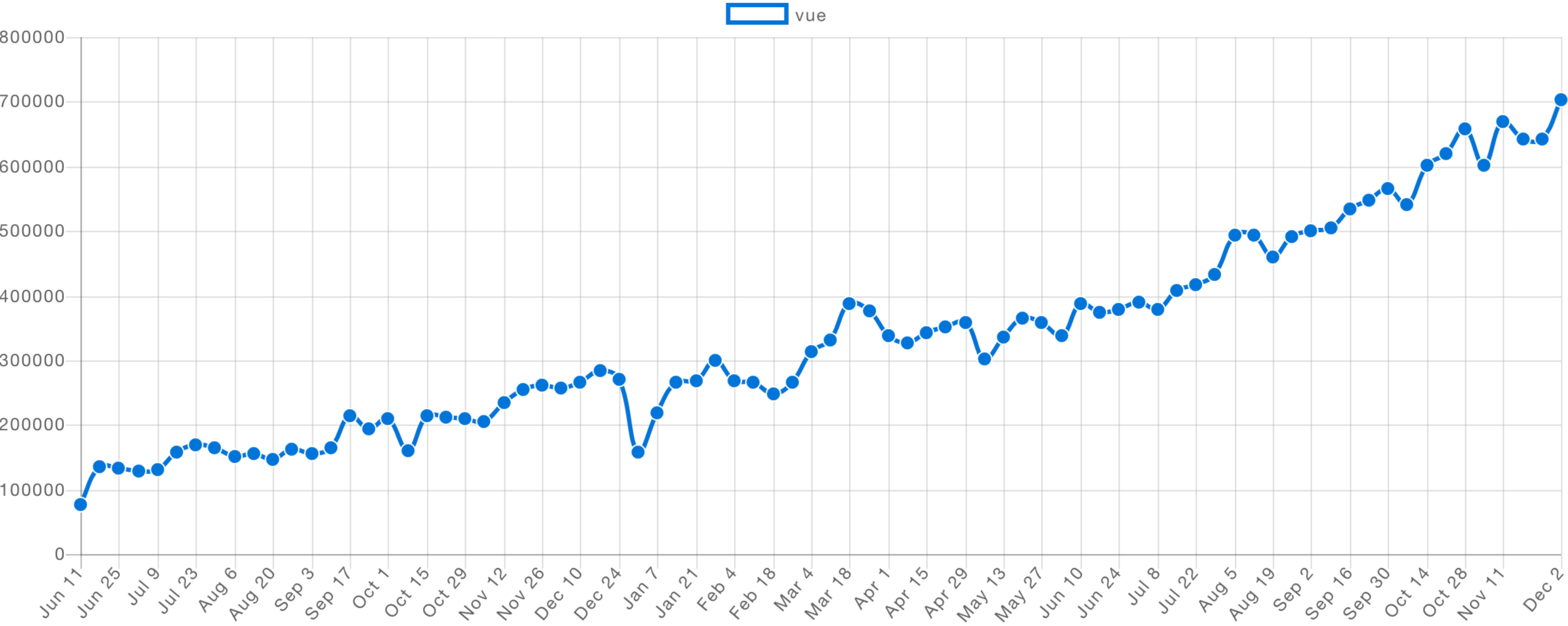
Create a list of 53,000 websites that use Vue.js.

✓ Get an instant quote

Websites using Vue.js

#	HOSTNAME
1	9gag.com
2	laravel.com
3	laracasts.com
4	bilibili.com
5	jianshu.com
6	app.codeship.com
7	vuejs.org
8	ithelp.ithome.com.tw
9	mp.weixin.qq.com
10	affiliate-program.amazon.com

Downloads in past 2 Years



Выводы

Vue - это быстро растущий, прогрессивный и набирающий популярность в последний год framework. Стремительный уровень развития порождает все больше и больше проектов которые используют vue. Он все ещё менее популярный чем его главные конкуренты, но большие проекты уже начинают смотреть в сторону vue.

Часть шестая

Vue экосистема

Экосистема

- Vuex - хранилище данных vue
- Vue-router - роутинг внутри приложения
- Vue-CLI - создание проекта с множеством вшитых плюшек, от локального сервера до чанков и линтера
- Nuxt.js - ssr(Server side rendering)
- Поддержка TypeScript, coffe script, много кастомных плагинов под любой code style

0 курсе

16 лекций на которых мы рассмотрим:

- Разработка на vue
- Vuex - зачем он нужен, как работать с ним, поработаем в паре с vue
- Vue-router - создадим многостраничный проект на Vue + Vuex
- ES6/7 - основные фишки
- HTTP запросы, что это такое, зачем они в проекте
- Тестирование приложений и оптимизация приложений
- Напишем полноценный SPA проект на базе Vue + Vuex + Vue-router

По окончании курса вы будете понимать как работать с целостной экосистемой в которой живет Vue. Сможете самостоятельно написать приложение в этой экосистеме. Научитесь работать с API, настраивать проект, продумывать его архитектуру. Тестить, рефакторить и оптимизировать свой код.