

Маленький Data Science для большого фронтенда

Роман Дворнов
Avito

Москва, 2018



- Работаю в [Avito](#)
- Open source:
[csstree](#), [CSSO](#), [hitext](#),
[jora](#), [basis.js](#), [rempl](#),
[component-inspector](#)
и [другие](#)

github.com/lahmatiy

План

-  Что такое Data Science
-  Data Science во фронтенде
-  Велосипеды
-  Взгляд в будущее
-  Выдыхаем, рефлекслируем

Data Science

Data Science

Раздел информатики,
использующий научные методы, процессы,
алгоритмы и системы
для извлечения знаний из данных

en.wikipedia.org/wiki/Data_science

Computer Science

The diagram consists of two concentric circles. The outer circle is blue and contains the text 'Computer Science'. The inner circle is green and contains the text 'Data Science'. A horizontal dashed line extends from the right side of the green circle, with the text 'Последние 20 лет' positioned above it.

Data Science

Последние 20 лет

“...the term "data science" became a **buzzword**. It is now often *used interchangeably* with earlier concepts like **business analytics**, **business intelligence**, **predictive modeling**, and **statistics**.”

en.wikipedia.org/wiki/Data_science

Data Science

Business Analytics

Business Intelligence

Predictive Modeling

Statistics

Big Data

Machine learning

...



Скучно и сложно...

Упрощаем

Зачем это всё?

Чтобы что-то **изменить** (улучшить)
нужно совершить **действие**

Простая концепция



Извлечение
Подготовка

...

Представление
Анализ

...

PROFIT!!!11



Data Science

Data Science

Унификация статистики, анализа данных, машинного обучения и связанных с ними методов для **понимания и анализа явлений**

en.wikipedia.org/wiki/Data_science



Data Science во фронтенде,

в разработке, и вообще

Data Science не только про бизнес,
применимо к любым областям, где есть
данные



Где же данные?

(в разработке)

Где данные:

- Информация о коде: история изменений, владение кодом, связь с задачами...
- Перекрестные связи между файлами и абстракциями, типы...
- Результат работы инструментов: сборка, тестирование...
- ...

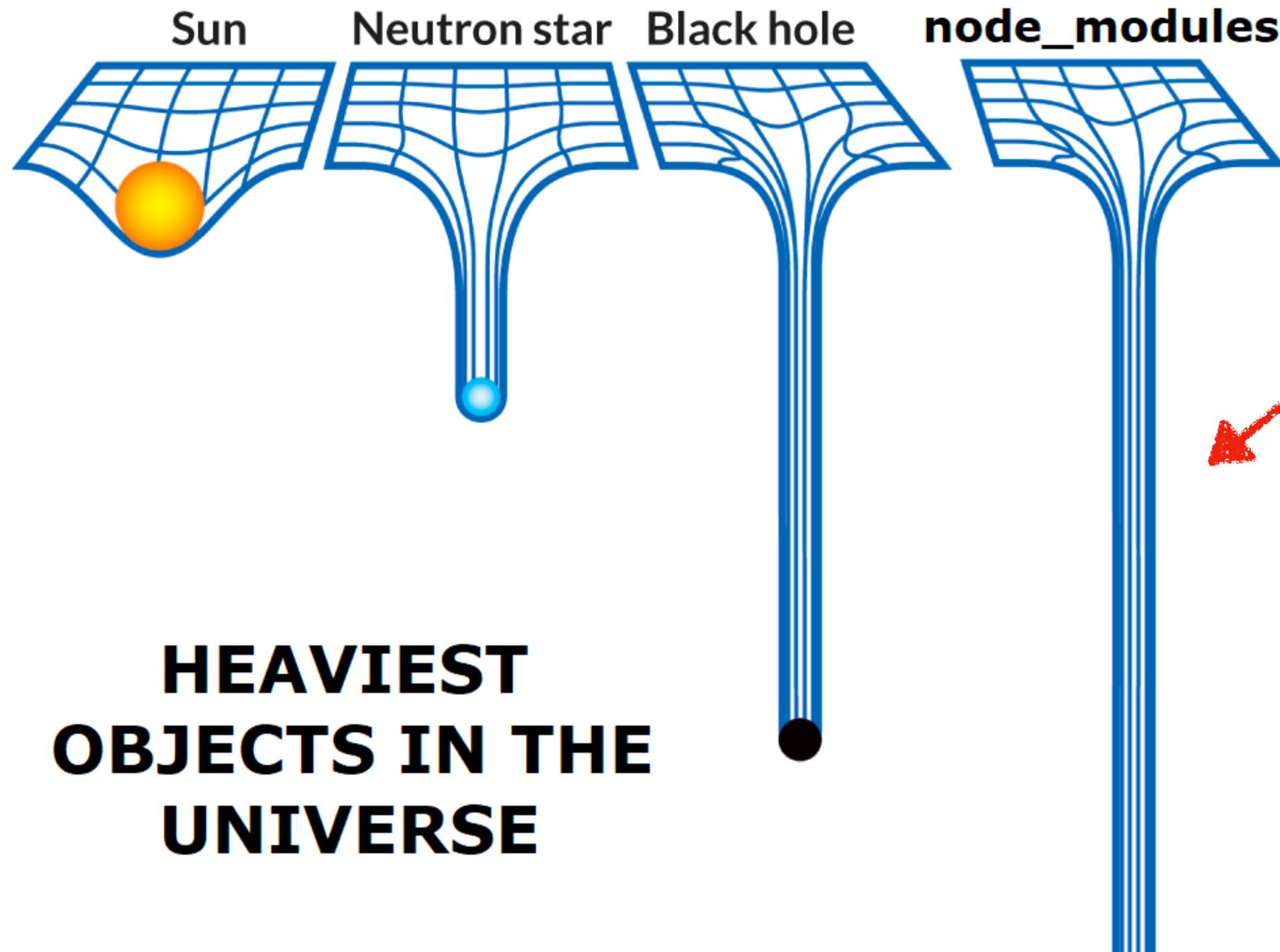


Data... Data everywhere

Это не big data!

Потому «маленький Data Science» ;)

У нас свои проблемы...



Явление

**HEAVIEST
OBJECTS IN THE
UNIVERSE**

Современные вызовы

- Размер кодовой базы
- Сложность кода
- Зависимости (например, `node_modules`)
- Performance
- Оценка трудозатрат
- ...

Методы Data Science полезны
для решения этих проблем

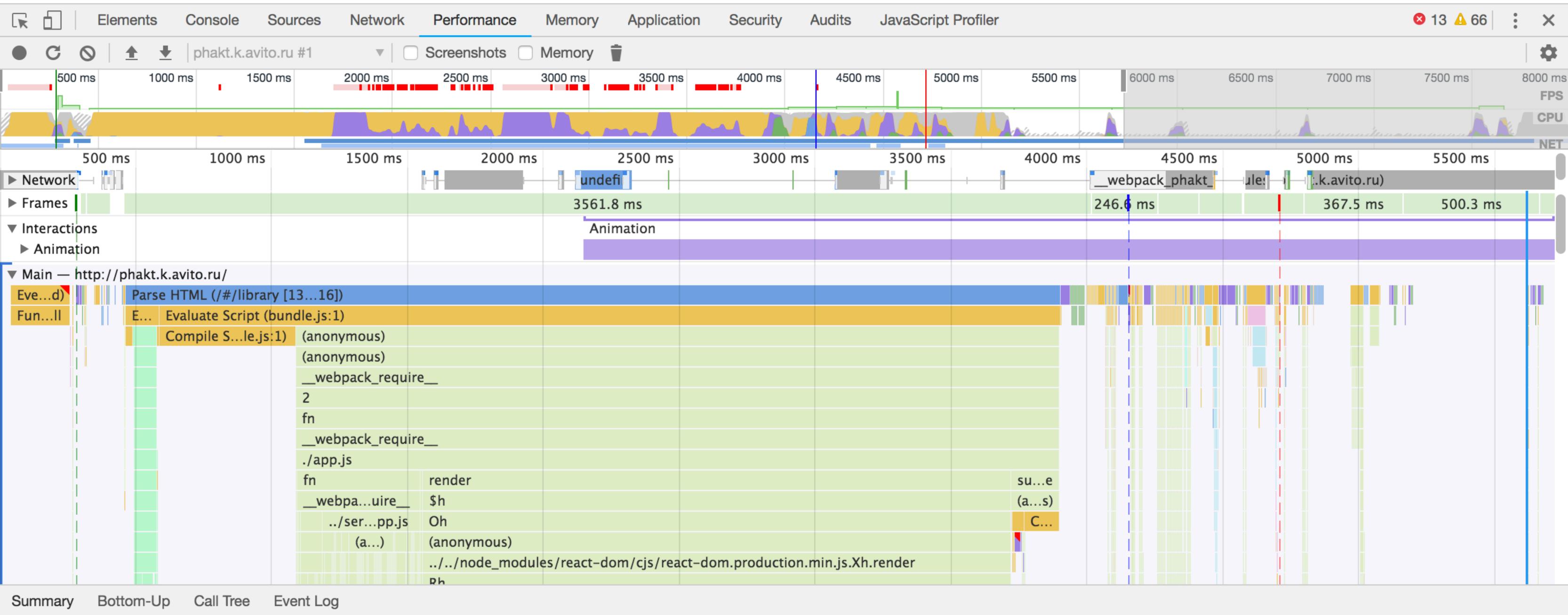
Вероятно вы уже применяете
эти методы, но не знаете об этом



Инструменты

Браузерные DevTools

Chrome DevTools Performance



Под капотом

- Данные в JSON
- Массив из объектов вида

Для предыдущего слайда:
15Mb, 51863 объекта
(для 8 секунд)

```
{  
  pid: 52020,  
  tid: 42499,  
  ts: 1150001714692,  
  ph: "X",  
  cat: "toplevel",  
  name: "MessagePumpLibevent::OnLibeventNotification",  
  args: { src_file: "../..../mojo/core/channel_posix.cc", src_func: "StartOnIOThread" },  
  dur: 1604,  
  tdur: 652,  
  tts: 2085854342  
}
```

Chrome DevTools

Позволяет просто и быстро получить
понимание общей картины,
выявить явления
и предпринять действия



Еще примеры

Webpack

← → ↻ https://webpack
Open Home Modules



NormalModule source size: 190.00 bytes bundled size: 190.00 bytes

./node_modules/react/index.js

Resource

.js node_modules/react/index.js

Links

Dependencies Dependants Chunks Concatenated

- www\.\.app\.\.es6
- .es6 [./var/www/s/avito/apps/auth/app.es6](#)
- ConcatenatedModule [./var/www/s/avito/apps/sendout-banner/app.es6](#)
- .es6 [./var/www/s/avito/apps/shortcuts/app.es6](#)
- ConcatenatedModule [./var/www/s/avito/apps/recommendations/app.es6](#)
- ConcatenatedModule [./var/www/s/avito/apps/web-push-subscribe-banner/app.es6](#)
- ConcatenatedModule [./var/www/s/avito/apps/limited-personal-discount-banner/app.es6](#)
- .es6 [./var/www/s/avito/apps/countdown-timer/app.es6](#)

Show all the rest 2 items...

Messages

Warnings Errors Deoptimizations

Filter (regexp)

ModuleConcatenation bailout: Module is not an ECMAScript module

id	name
0	./lib/index.web.js
1	../cases ^\.\/[^_\/_]+

Webpack

- Обзор инструментов для анализа:
survivejs.com/webpack/optimizing/build-analysis/
- Позволяют оценить результат сборки, выявить явления и предпринять действия

IRHydra

- Визуализирует отладочную информацию от V8
- Отвечает на вопросы:
 - как V8 скомпилировал код, какие оптимизации применены
 - где и почему деоптимизировался код
- Выявление **явлений** и **результат**: понимание какой код требует исправления

Что общего у ЭТИХ инструментов?

Дают понимание общей картины,
возможность выявить явления
и предпринять действия



Применяемые методы

- Множества
- Теория графов
- Кластеризация
- Агрегация, интерполяция
- ...

Визуализация

Визуализация

- Графики, таблицы, графы...
- Dashboards
- Инфографика
- Интерактивные стенды
-

Крайне рекомендую

Bred Victor

Обязательно к просмотру

[Dynamic Animation Creation and Real-Time Coding Feedback](#)

Резюме

Резюме

- В современной разработке **достаточно вызовов**
- Применение методов Data Science может **помочь в проблемах** разработки
- Есть **готовые инструменты** для ряда задач
- Data Science позволяет выявить **явления** и предпринять **действия** в любой области

Данных в разработке достаточно много,
главное это **увидеть**
и начать извлекать из них **пользу**

Велосипеды

Предистория

М – мотивация

Вопросы

- Как распилить большое на части, какие трудозатраты?
- Дублируются ли зависимости (может разных версий)?
- На каких страницах используется ...?
- Что отдается на конкретной странице?
- Как разбивается на бандлы/чанки?
- Кого и как афффектят мои изменения?
- ...

Avito monolith

Запрос на страницу сайта

Route (slug)

Yaml

Router/Controller

PHP

Template

Twig

Script

JavaScript

Style

CSS, Stylus

Images/Fonts/etc

SVG, ...

Как?

- Находим нужные файлы
- Парсим в AST (или reghex), извлекаем определения и ссылки
- Строим граф сущностей и зависимостей
- Делаем анализ
- PROFIT!!!



На JavaScript есть парсеры
практически для любого формата,
остальное – дело техники



Как валидировать, анализировать
и отображать данные?

Вывод результатов

- Консоль

Слишком много данных, мало пространства, нет интерактивности и наглядности, никакой графики

- Excel (Google Docs например)

Нельзя JSON, хорошо для табличных данных (список), плохо для графов, нельзя

- Готовые JSON viewer'ы

Проблема "закидывать" данные во viewer, нельзя циклические ссылки, не позволяют предобрабатывать данные и автоматизировать операции

- Сделать свой велосипед...



Собрали стенд под задачу, но...

Проблемы

- Много усилий для наращивание функциональности
- Значительный рефакторинг при изменении структуры данных
- Много однообразного рутинного кода

Insight

Каким бы хорошим не был инструмент –
хочется большего

(находятся вопросы, на которые не может дать ответ,
хотя данные для этого есть)



Приходят на ваш стенд...

A man wearing a grey short-sleeved shirt and a light-colored flat cap is looking intently at a large, dark bronze statue of a winged horse (Pegasus) rearing up. The statue is highly detailed, showing the horse's musculature and the texture of its wings. To the right, a woman with short blonde hair, wearing a blue blazer over a white collared shirt, is smiling and looking towards the man. The background shows a shop interior with shelves displaying various items, including a purple vase and a silver trophy. A white speech bubble with black text is overlaid on the lower left of the image.

А у вас есть такой же,
без крыльев?

A man wearing a white cap and a grey polo shirt with a red tie is looking towards a woman. The woman is wearing a blue long-sleeved shirt over a white collared shirt and is holding a white garment. They are in a store with many white garments hanging in the background. A white speech bubble is overlaid on the image.

... такой же, но с перламутровыми
пуговицами?



Нафиг такое вообще нужно?!..

В попытке решить эти проблемы
и появилось решение...

Discovery

Hackable JSON discovery tool

Что делает?

Помогает организовать
извлечение данных, подготовку данных,
построение отчетов и их дистрибьюцию

Что НЕ делает?

Не делает магии, нужно настроить
(но для ряда задач есть готовые расширения)

Вернемся в начало



Извлечение
Подготовка

...

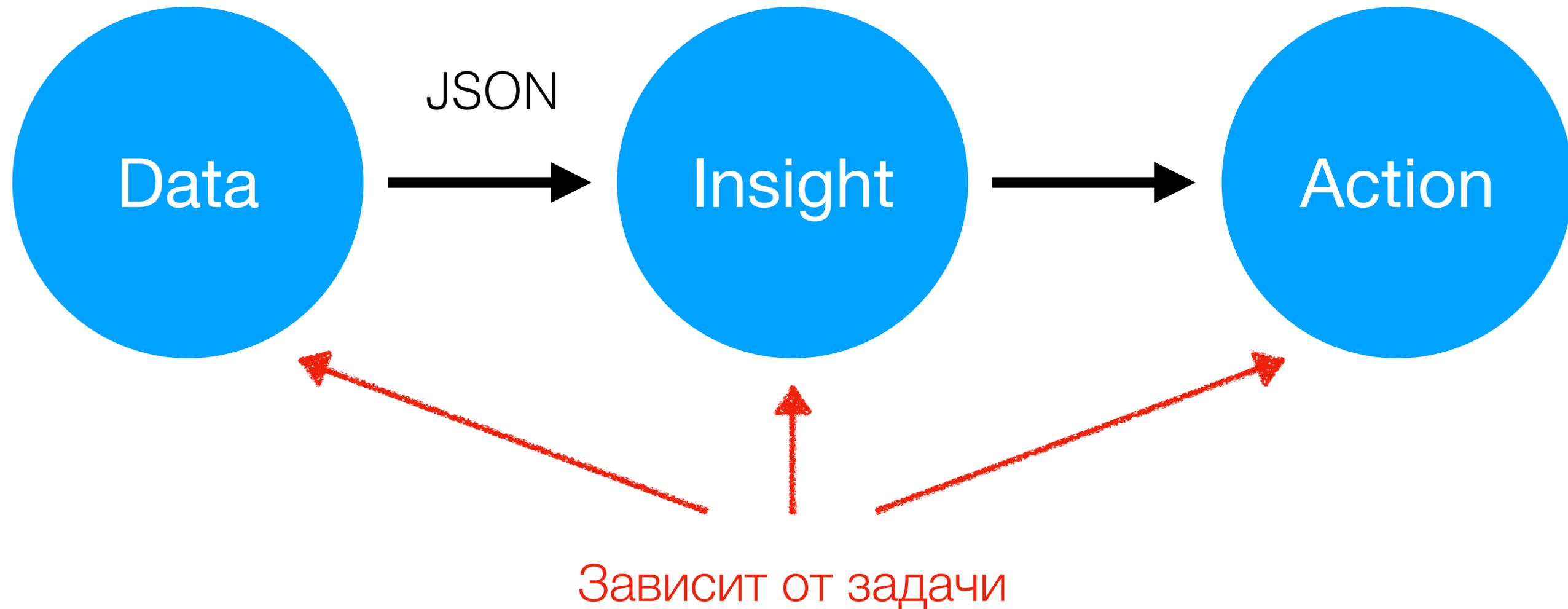
Представление
Анализ

...

PROFIT!!!11



Что можем унифицировать



Модель

ключевое понятие в Discovery
(описывает что мы хотим получить)

Модель



Модель

```
model = {  
  name: 'Model name',  
  data: () => Promise | any,  
  prepare: 'путь к JS модулю',  
  view: {  
    assets: [ ... пути к JS/CSS файлам ... ]  
  }  
}
```

 Всё опционально!

Интерфейс

- Page – готовый отчет, состоит из view
- View – рендерит разметку и/или другие view

DEMO

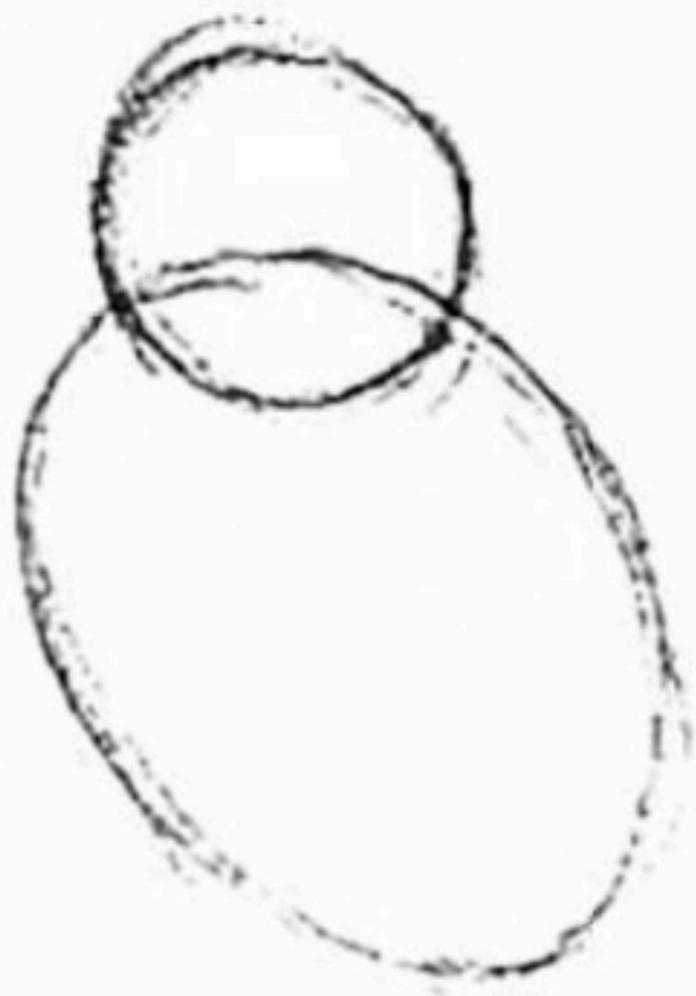
Давайте пощупаем

Об увиденном

- Основная цель – просто создавать стенды анализа данных
- Данные в JSON, структура не важна (нет схемы)
- Просто поддерживать и расширять интерфейс
- Если не хватает заготовок – свободный отчет
- Отчеты можно шарить по ссылке

КАК НАРИСОВАТЬ СОВУ

1.



РИСУЕМ КРУЖОЧКИ

2.



РИСУЕМ ОСТАТОК СОВЫ

Реальные примеры

Avito monolith (frontend)

Получение данных

Структура файлов

Скан FS

Структура node_modules

Скан FS

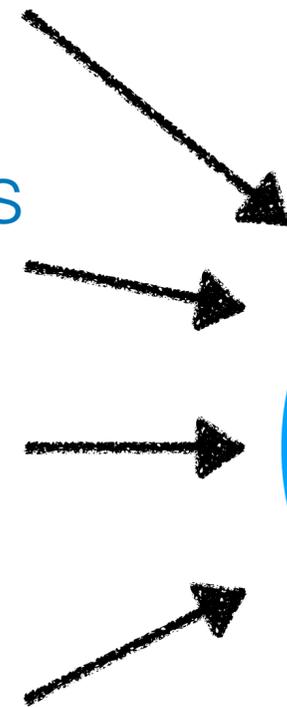
Ownership

Запрос к сервису

Webpack build stat

Запрос к сервису

...



Перекрестные ссылки
Расширение движка запросов

```
{
  repo: {
    name: "avito-site",
    commit: "200ddeb7438f69473385d8c4f5a5b68cee295058",
    commitShort: "200ddeb743",
    branch: "master",
    files: [...],
  },
  files: [
    { type: "php", filename: "lib/ABTestOld.php", size: 4850, defs: [...], ...6 more... },
    { type: "php", filename: "lib/AccountHierarchy.php", size: 3093, defs: [...], ...6 more... },
  ],
  deps: [
    { name: "avito-site", version: null, dev: false, path: ".", ...3 more... },
    { name: "abab", version: "2.0.0", dev: false, path: "node_modules/abab", ...3 more... },
  ],
  build: {
    input: {
      modules: [...],
      assetsBlocks: [...],
      bundlesBlocks: [...],
      files: [...],
    },
    output: {
      bundles: [...],
      chunks: [...],
      files: [...],
    }
  }
}
```

Структура данных,
которая приходит на клиент
~11Mb (1.4Mb gzip)

Avito monolith

← → ↻ https://discovery.local/monolith/ 🔍 🗨️ 👤 ⋮

Make report Switch model Logout

Repo Files Node Modules

Filter (regexp)

Routes 85

- [etc/routes/admin.yml](#)
- [etc/routes/admin/autocatalog.yml](#)
- [etc/routes/admin/billing.yml](#)
- [etc/routes/admin/calltracking.yml](#)
- [etc/routes/admin/content/support.yml](#)
- [etc/routes/admin/helpdesk.yml](#)
- [etc/routes/api.yml](#)
- [etc/routes/base.yml](#)
- [etc/routes/base/account.yml](#)
- [etc/routes/base/auth.yml](#)
- [etc/routes/base/autosearch.yml](#)
- [etc/routes/base/company.yml](#)
- [etc/routes/base/info.yml](#)
- [etc/routes/base/payments.yml](#)
- [etc/routes/base/profile.yml](#)

Show 15 more... Show all the rest 70 items...

Twig 1649 48 errors

Avito Monolith (Frontend)

Data collected at 23/11/2018 12:51:12

17233 Files	7919 Definitions	20794 References	311 Errors	1086 Node modules
----------------	---------------------	---------------------	---------------	----------------------

File stat

- routes 85
 - References by type:
 - controller – 2367
- twig 1649 48 errors
 - References by type:
 - assets – 2793
 - bundle – 31
 - embed – 73 5
 - extends – 836 6

- Definitions:
 - bundle – 27
 - css-assets – 535
 - js-assets – 539

Repository

- Name: [avito-site](#)
- Branch: [master](#)
- Head: [200ddeb743](#)

Example reports

- [File types stat](#)
- [Extended file types stat](#)
- [Asset duplicates in Twig templates](#)
- [SVG images with preview](#)

CSSTree syntax reference

CSSTree syntax reference

← → ↻ https://csstree.github.io/docs/new-syntax/#Property:background



Index

Make report

Switch model

ba

Property 29

[<'-moz-background-clip'>](#)

[<'-ms-scrollbar-3dlight-color'>](#)

[<'-ms-scrollbar-arrow-color'>](#)

[<'-ms-scrollbar-base-color'>](#)

[<'-ms-scrollbar-darkshadow-color'>](#)

[<'-ms-scrollbar-face-color'>](#)

[<'-ms-scrollbar-highlight-color'>](#)

[<'-ms-scrollbar-shadow-color'>](#)

[<'-ms-scrollbar-track-color'>](#)

[<'-webkit-background-clip'>](#)

[<'alignment-baseline'>](#)

[<'backdrop-filter'>](#)

[<'backface-visibility'>](#)

[<'background'>](#)

[<'background-attachment'>](#)

[<'background-blend-mode'>](#)

[<'background-clip'>](#)

[<'background-color'>](#)

[<'background-image'>](#)

[<'background-origin'>](#)

<'background'>

Test a value

url(image.png) 10px no-repeat red|

Match result 150 iteration(s)

url(image.png)	10px	no-repeat	red
<url>			<length>	<repeat-style>	<named-color>
<image>			<length-percentage>		<color>
<bg-image>			<bg-position>		<'background-color'>
			<final-bg-layer>		
			<'background'>		

Formal syntax

[<bg-layer> ,]* <final-bg-layer>

<bg-layer> = <bg-image> || <bg-position> [/ <bg-size>]? || <repeat-style> || <attachment> || <box> || <box>

<bg-image> = none | <image>

DEMO

ИАРы

* Изящные архитектурные решения

ИАР #1 – режимы работы

ИАР #1 – режимы работы

- Model free (без конфига)
JSON viewer + report builder
- Single model
одна модель (стенд)
- Multi model
несколько моделей (стендов) + индексная страница

Model free каждому for free

Плагин к Chrome для работы JSON ответами

github.com/discoveryjs/browser-extension-json-discovery

DEMO

Multi model

```
config = {  
  name: 'Multi mode config',  
  models: {  
    foo: modelConfig1, // http://discovery.host/foo/  
    bar: modelConfig2, // http://discovery.host/bar/  
    ...  
  }  
}
```

ИАР #2 – поставка

ИАР #2 – поставка

- Готовое приложение – [npx discovery](#)
- Как библиотека
 - Класс `Widget` – для интеграции в другие приложения или создания своих (например, плагин для браузера)
 - Хелперы, `views`, `pages`...

Интеграция в приложение

```
import Widget from 'discovery/dist/lib';
```

```
// или
```

```
const Widget = require('discovery/dist/lib.umd.js');
```

```
const instance = new Widget(document.getElementById('...'));
```

```
instance.setPage('foo');
```

```
instance.setData({ ... })
```

ИАР #3 – шаринг

Это просто и удобно:

изменяем отчет или создаем новый
и можем пошарить ссылку на результат

ИАР #4 – гибкие view

ИАР #4 – гибкие view

- Легко добавить новое
`discovery.view.define(name, ...)`
- Встроенные view
 - Построены на DOM, но можно использовать любой фреймворк
 - Минимум настроек, подстраиваются под данные

Объявление view

```
discovery.view.define('my-view', (el, config, data, context) => {  
  el.innerHTML = 'Hello world!';  
  // discovery.view.render(el, ...);  
  // или ReactDOM.render(<Widget ... />, el);  
  // или что угодно  
});
```



Можно использовать **любые** ГОТОВЫЕ
КОМПОНЕНТЫ, обернув в `view.define(...)`

Используем view

```
discovery.view.render(e1, { view: 'my-view' }, data, context);
```

// или в конструкторе отчета

View: **Default** Custom

```
{  
  view: 'my-view'  
}
```

Available views: alert, alert-danger, alert-success, alert-warning, badge, block, chart, checkbox, column, columns, comma-list, content-filter, context, expand, fallback, h1, h2, h3, h4, header, hstack, html, image-preview, indicator, inheritance, inline-list, input, link, list, list-item, my-view, ol, package-tree, packages-table, pill-badge, section, sidebar, source, struct, switch, tab, table, table-cell, table-row, tabs, text, text-match, toc-item, toc-section, ul

build on input

Build

Hello world!

IAP #5 – extendable

ИАР #5 – все переопределяется

- Все переопределяется и/или расширяется:
 - Page
 - View
 - Route
 - Движок запросов
 - Поведение
 - ...

ИАР #6 – сборка в статику

ИАР #6 – сборка в статику

- `npx discovery-build`
- Поддерживаются все режимы
- Результат: 4 файла на модель (html, js, css, json)
- Можно публиковать на любом сервере, например, на [github pages](#)
(так публикуется стенд CSSTree)

Jora

JavaScript object query language

Еще один язык?

Альтернатив не так уж много,
их возможностей не хватило

Вы можете использовать любое решение для запросов к данным, адаптировав для discovery

Jora

- Целится на упрощение работы с графами и стат методами
- Функциональный (по крайней мере пока)
- Толерантен к структуре данных
- Адаптирует результат выборки к ожидаемому
(например, возвращает уникальные значения в выборках, склеивает массивы)
- Расширяемый
- Достаточно быстрый

Простое API

```
const jora = require('jora');

// create a query
const query = jora('foo.bar') ;
// or with custom methods
const query = jora('foo.myMethod()', {
    myMethod(current) { /* do something */ }
});

// perform a query
const result = query(data, context);
```



Главный недостаток – пока молодой

(~4 месяца)

Большие планы

- Доработать и расширить синтаксис
- Оптимизация/упрощение запросов (например, кеш результатов внутри запроса и между запросами)
- Интроспекция выполнения запросов
- Статистика времени выполнения разных частей запроса
- Подсказки при наборе запроса
- ...

github.com/lahmadiyah/jora

```
npm install jora
```

Резюме

Discovery не что-то принципиально
новое, это объединение простых
идей в простой инструмент

Основная цель:

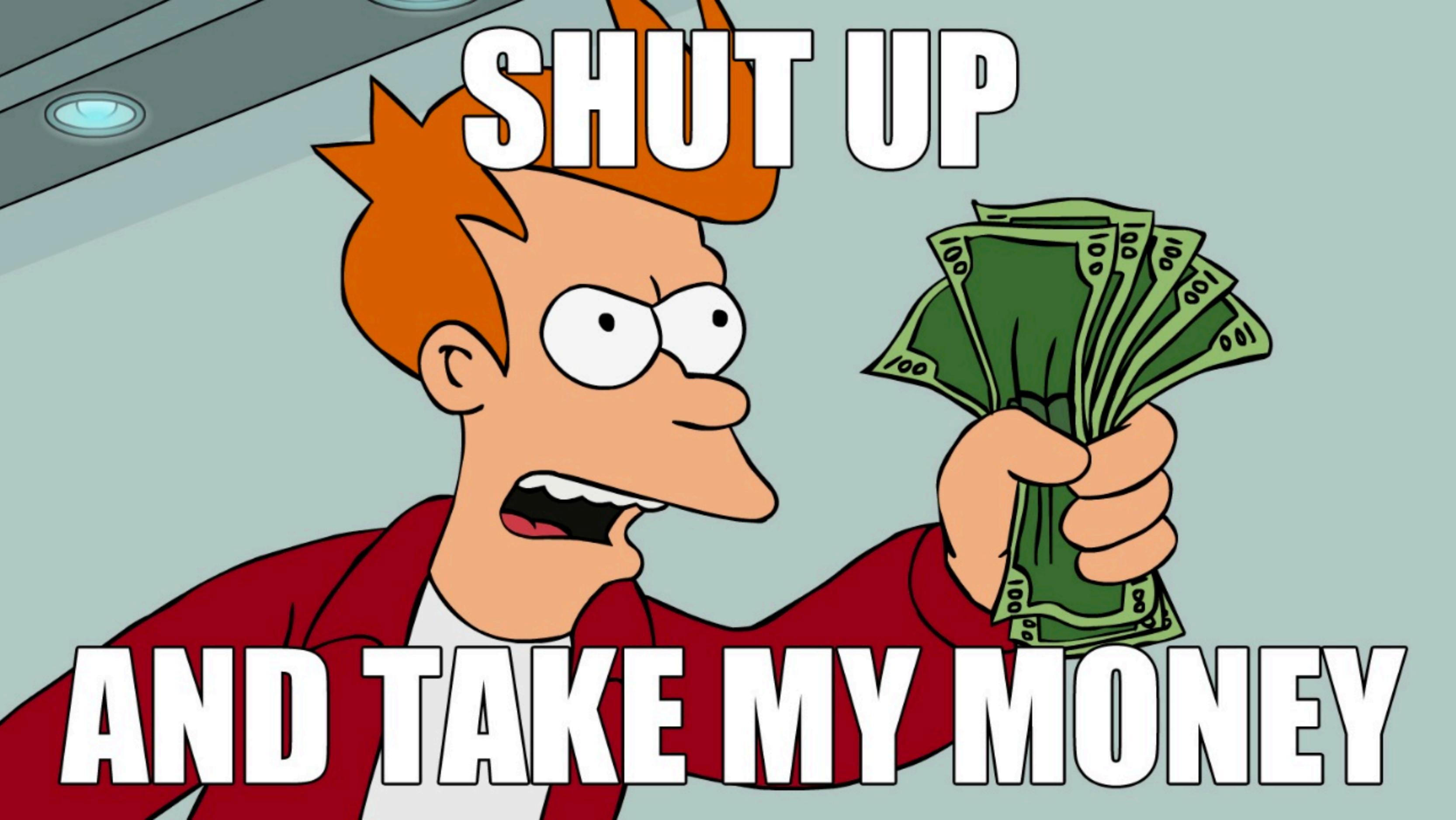
упростить анализ данных для прикладных задач,
СНИЗИТЬ ПОРОГ ВХОДА

Вспомним проблемы

- Много усилий для наращивание функциональности
- Значительный рефакторинг при изменении структуры данных
- Много однообразного рутинного кода
- Всегда хочется большего

Больше не проблемы

- ~~Много усилий~~ для наращивание функциональности
- ~~Значительный рефакторинг~~ при изменении структуры данных
- Много однообразного ~~рутинного кода~~
- Всегда ~~хочется большего~~

A cartoon illustration of a man with spiky orange hair, wide eyes, and an open mouth in a shocked or desperate expression. He is wearing a red jacket and holding a large stack of green 100-dollar bills in his right hand. The background is a light blue wall with a circular light fixture on the left.

SHUT UP

AND TAKE MY MONEY

Где взять?



На данный момент это локальная разработка...

... но давайте это исправим ;)

github.com/discoveryjs/discovery

```
npm install @discoveryjs/discovery
```



Нам еще нужно некоторое время **обкатать**
и выпустить **другие наработки** ОТДЕЛЬНЫМИ
пакетами

Stay tuned!

Взгляд в будущее

Можно сделать процесс разработки
лучше и эффективнее,
применяя методы Data Science

Главное, чтобы **возможность их**
применения была более **доступна**

Решения вроде Discovery должны в этом помочь

Мы планируем **наращивать** набор
решений вокруг Discovery

Некоторые идеи...

- Результат вызова бекенда при открытии в браузере открывать в интерфейсе discovery
- Использовать discovery для скриптов: автоматизация, умный линтинг, мониторинг etc
- ...

Присоединяйтесь!

И тогда мемы вокруг фронтенда
станут пережитками прошлого



В заключении...



Начните изучать
методы Data Science

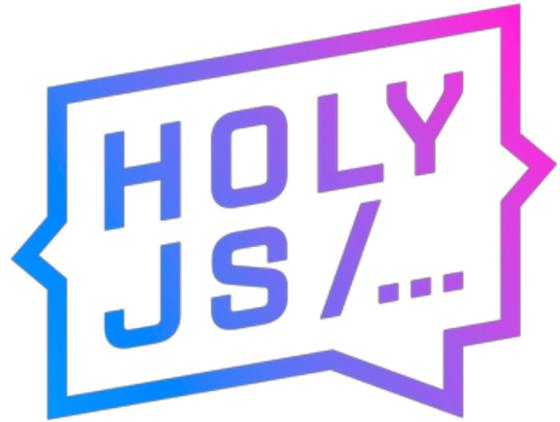
Помните про пользу
визуализаций

Задавайте вопросы,
ищите ответы

Теперь у вас есть

Discovery





Спасибо!

github.com/discoveryjs/discovery

github.com/lahmatiy/jora

Роман Дворнов

[@rdvornov](https://twitter.com/rdvornov)

rdvornov@gmail.com

github.com/lahmatiy