Добавляем поисковую строку в ваше приложение с Elasticsearch

Маленький google в вашем приложение

Шестаков Алексей

Инженер-Программист

СКБ Контур

Что будет в этом докладе

- Внутренний поиск
- История и Введение в Elasticsearch
- Примеры использования

Что будет в этом докладе

- Внутренний поиск
- История и Введение в Elasticsearch
- Примеры использования
- Теория
- Как сделать поиск умнее

Чего не будет в этом докладе

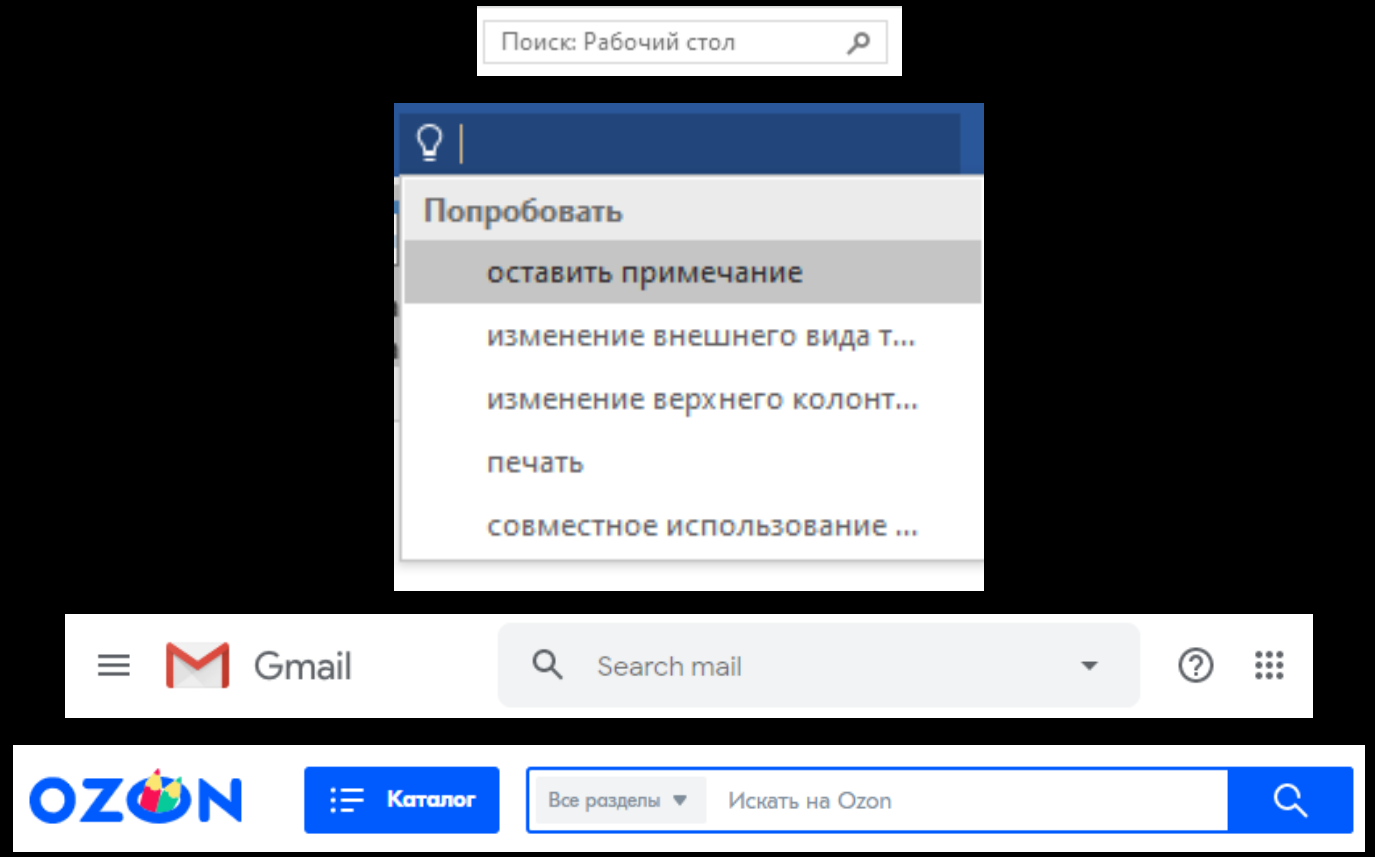
- Эксплуатация ES
- Использование ES для хранения логов
- ML (aka Машинное обучение) в поиске.

Встроенный поиск

Зачем в вашем приложении встроенный поиск?

Встроенный поиск – Примеры

- Windows
- MS Office
- GMail
- Ozon
- Inteliji Idea
- И ещё много-много примеров….



Встроенный поиск – Профит

- Единая точка навигации
- Быстрая навигация

Встроенный поиск – Когда

- Признаки того, что вам он нужен
  - Много контента
  - Много типов контента
  - Много разных функции (фичей)
  - Сложная навигация меню

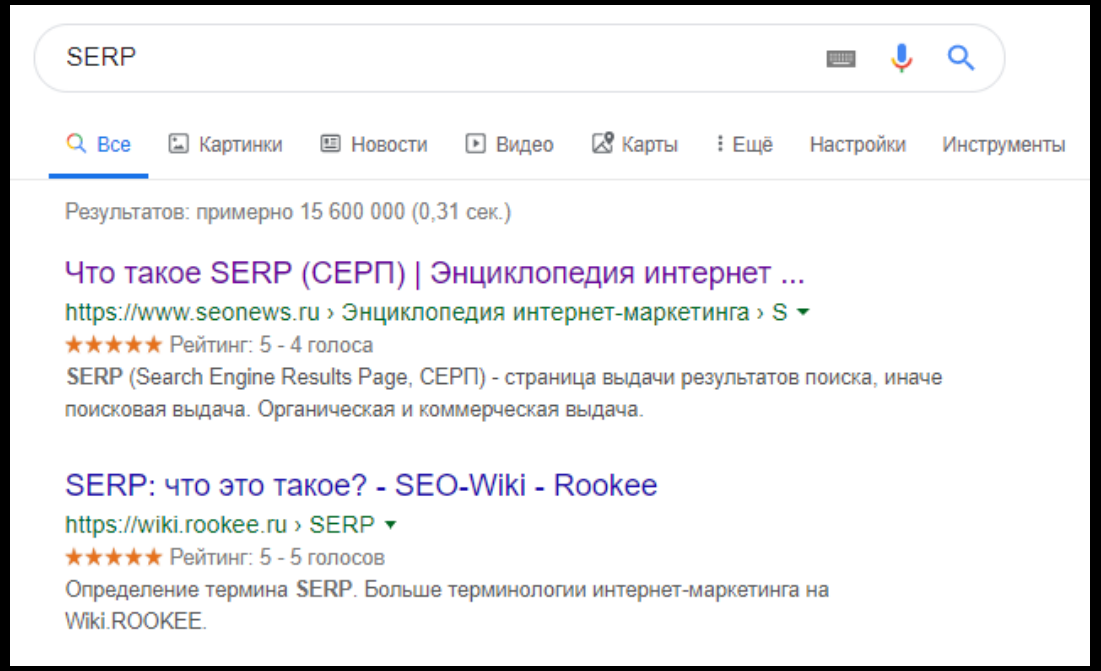
Основные виды поиска

- SERP - Search Engine Result Page

- Поисковая подсказка

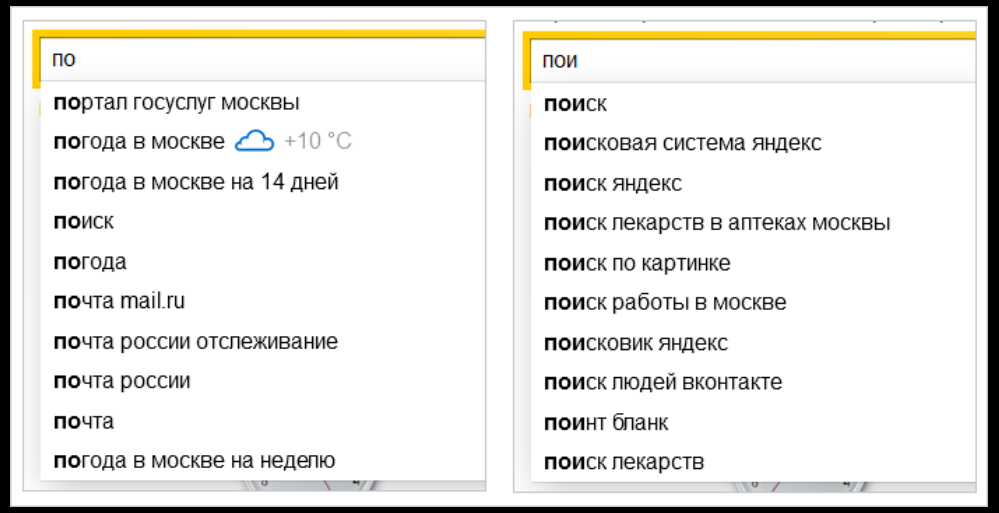
SERP – Search Engine Result Page

- По факту что у Google/Yandex
- Снипет – часть документа, важная в контексте запроса
- Это поиск по **целым словам(токены)**



Поисковые подсказки

- Search-as-you-type
- Помогает сформировать поисковый запрос
- Переход в SERP или на сразу на результат
- Это **префиксный** поиск



Elastic Search - Введение

ES - Движок полнотекстового поиска

- Движки – специализированное ПО
  - Умеют искать документы по запросу
  - Умеют ранжировать результаты поиска

Почему реляционные БД не подходят ?

- Реляционные БД не подходят для поиска слов по многим таблицам
- Не умеют ранжировать
- Не умеют обрабатывать текст
- Некоторые БД включают поиск как доп. фичу
  - MSSQL и PostgreSQL
  - Функционал все равно далек от специального решения

Почему документно-ориентированные БД не подходят ?

- Не умеют ранжировать
- Не умеют обрабатывать текст

Что такое Elasticsearch?

- Реплицированный
- Распределенный
- Горизонтально масштабируемый

Что такое Elasticsearch?

- Реплицированный
- Распределенный
- Горизонтально масштабируемый
- Самодостаточный
- Быстрый
- …. движок поиска

Что такое Elasticsearch?

- REST сервер
- Как одиночный сервер
- Как кластер

Почему Я рассказываю про ES?

- Мы выбрали ES так-как:
  - Opensource
  - Самое популярное решение
  - Просто развертывается

Конкуренты Elasticsearch

- Sphinx
- Solr
- Имеют свои плюсы и заслуживают отдельных докладов

Elasticsearch – История

История Elasticsearch

- Opensource
- Автор Shay Banon
- Начал проект Compass в 2004 г.,
- В 2010 г. как результат большого переписывания появился Elasticsearch
- В качестве основы используется **Lucene**

История Lucene

- Opensource
- Автор Douglass Cutting
- Начал проект в 1997 г.
- В 1999 г. выложил на SourceForge.net
- В 2001 г. вошел в состав проектов фонда Apache

Почему Elasticsearch > Lucene

- Lucene – Java библиотека
- Elasticsearch – REST сервер
- Elasticsearch добавляет поверх Lucene
  - Масштабируемость
  - Реплицируемость
  - Аналитические инструменты

Elasticsearch - 🚀

Начнем с простого примера

Индексируем

POST localhost:9200/information/person/1

```
{
 “name” : “Paul”,
 “lastname” : “Smith”,
 “job_description” : “Business Analyst”
}
```

Поиск

curl -X GET "localhost:9200/*information*/_search?pretty" -H 'Content-Type: application/json' -d'

```
{
    "query" : {
        "term" : { "user" : "kimchy" }
    }
}
```

Еще варианты:

GET localhost:9200/_search?q=kimchy

GET localhost:9200/_search?q=user:kimchy

Ищем по префиксу

```
curl -X GET "localhost:9200/_search?pretty" -H 'Content-Type: application/json' -d'
{
    "query": {
        "prefix": {
            "user": {
                "value": "ki"
            }
        }
    }
}
```

Теория чтобы искать умнее

Elasticsearch – не черная коробка

- Elasticsearch использует классическую теорию поиска

- Для того что бы качественно настроить поиск нужно понимать процесс обработки текста

Модель – Bag of Words

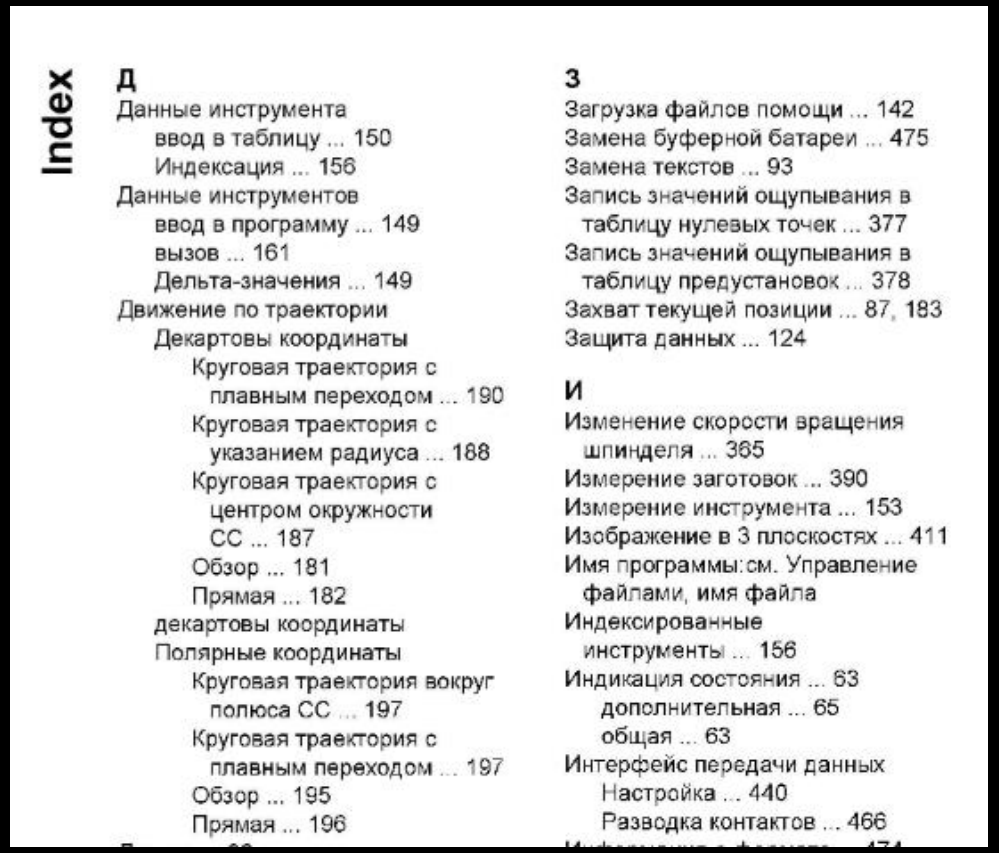
- Документом называется индексируемое текстовое поле
- Каждый документ режется на «слова» или «токены»
- Порядок не учитывается

Что такое «Токен»?

- «Токен» – минимальная единица поиска
- Обычно слово, но не всегда
- Поиск меньших частей «токена» возможен, но более затратен

Поисковый индекс

- ES Строит индекс почти как в книгах
- Запись в индексе:
- Связка токена – и номера документов, где слово встречается
- Еще называется «Инвертированным»



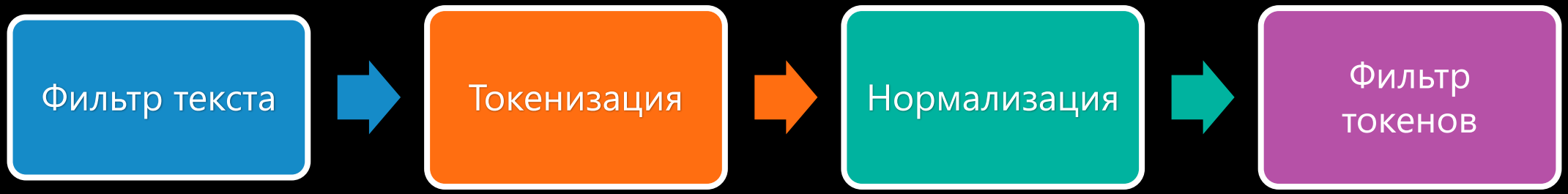
Index

Инвертированный индекс

Слово	Номера документов где слово встречается
все	2,4,7
всегда	2,7
делать	2
дело	3,3

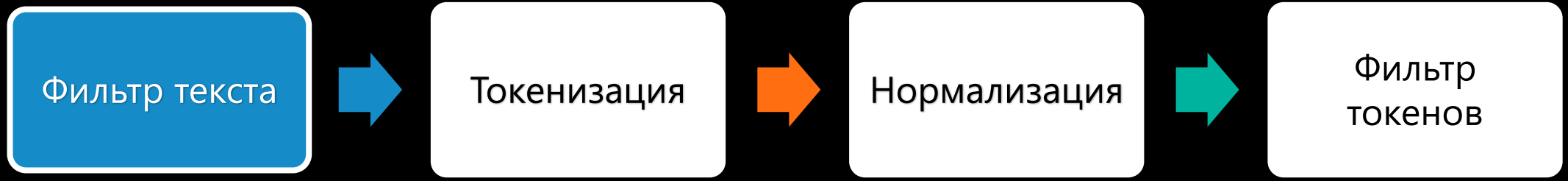
Процесс индексации

- Процесс построения индекса
- Управляя процессом индексации можно
  - Убрать лишнее из выдачи
  - Научить поиск находить больше



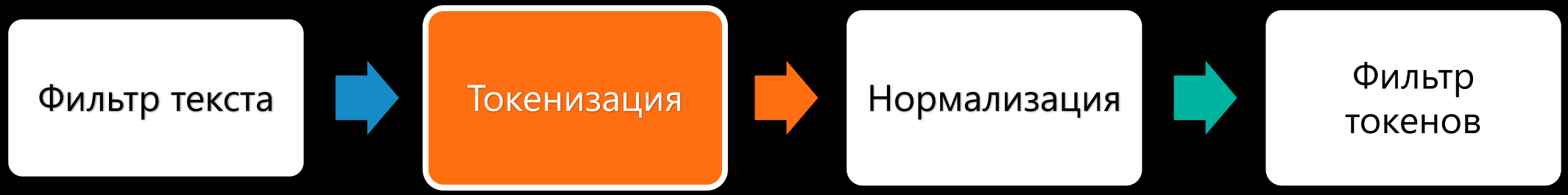
Фильтр текста

- Убираем из текста все что не должно искаться
- В терминах ES – «char_filter»
- Пример – «html_strip» - убирает HTML разметку



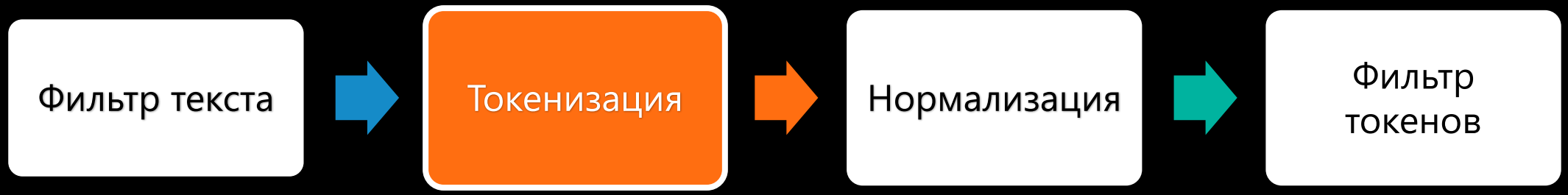
Токенизация

- Управляя токенизацей важно понять что мы хотим считать одним «словом»



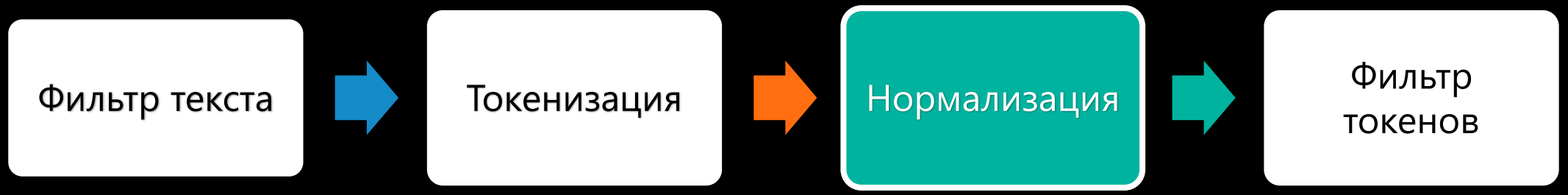
Токенизация

- Мобильные номера
  - + 7 (933) 34344322 –> +7(933)34344322, (933)34344322, 34344322
- От выбора токенов зависит будет ли находиться тот или иной вариант



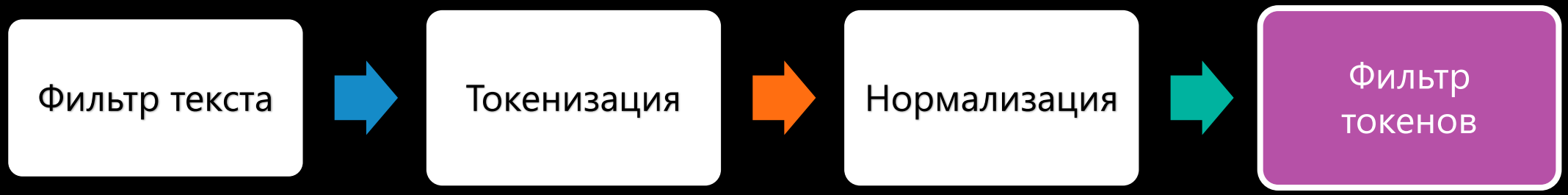
Нормализация

- Приводим схожие токены к одному
  - Хотим ли мы считать слова в разном падеже одним и тем же словом?
  - Мы теряем немного информации, но и будем находить больше



Фильтр токенов

- Не все токены нам интересны
  - Пример: предлоги



Делаем поиск умнее

Настраиваем маппинги

- Маппинг – аналог схемы в ES
- Что нам важно
- Тип поля
- Анализатор

Настраиваем маппинги

- Сущности в ES – иммутабельные
  - Настраивать можно только не индексированные поля
  - Лучше настраивать при создании индексов

Настраиваем маппинги

```
curl -X PUT "localhost:9200/my-index?pretty" -H 'Content-Type: application/json' -d'
{
  "mappings": {
    "properties": {
      "age":    { "type": "integer“ },
      "email":  { "type": " text“ , "analyzer": "std_english"  },
      "name":   { "type": "keyword"  }
    } } }
```

Настраиваем маппинги

- ES Автоматически настраивает поля и для текстовых полей выбирается анализатор “standard”

- Посмотреть настройки маппингов можно так:

```
curl -X GET "localhost:9200/my-index/_mapping?pretty"
```

Настраиваем анализаторы

- Pipeline обработки конкретного поля
- Создаем свой если стандартных мало

Настраиваем анализаторы

```
PUT my_index {
  "settings": { "analysis": {
      "analyzer": {
        "my_custom_analyzer": { "type": "custom", "tokenizer": "standard",
          "char_filter": ["html_strip" ],
          "filter": [
            "lowercase",
] } } } } }
```

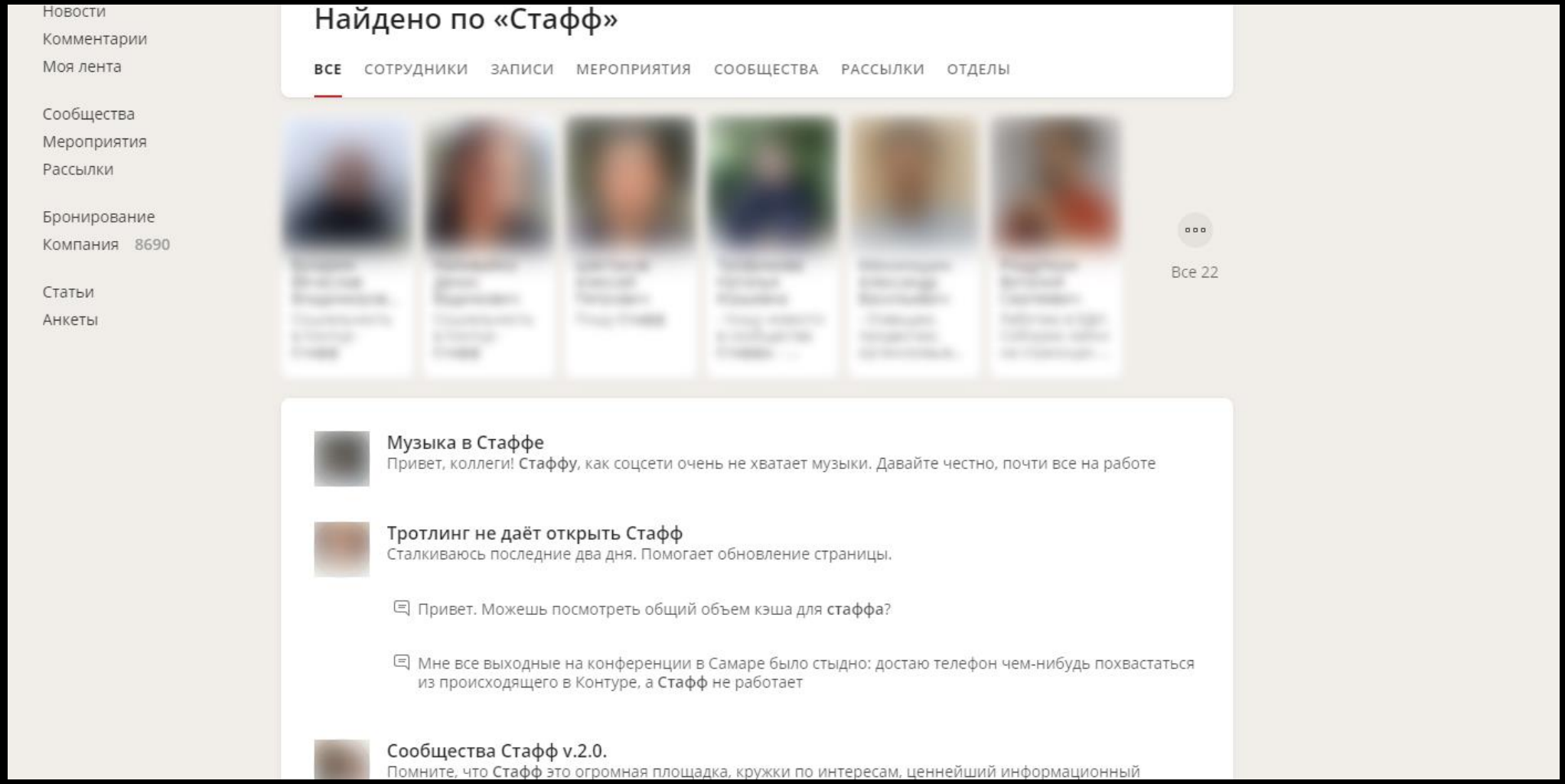
Пример использования – Контур.Стафф

Контур.Стафф

- Справочник сотрудников
- Социальная сеть
- Использует ES для Поиска
- Много разного контента
- ~4k обращений к SERP за день

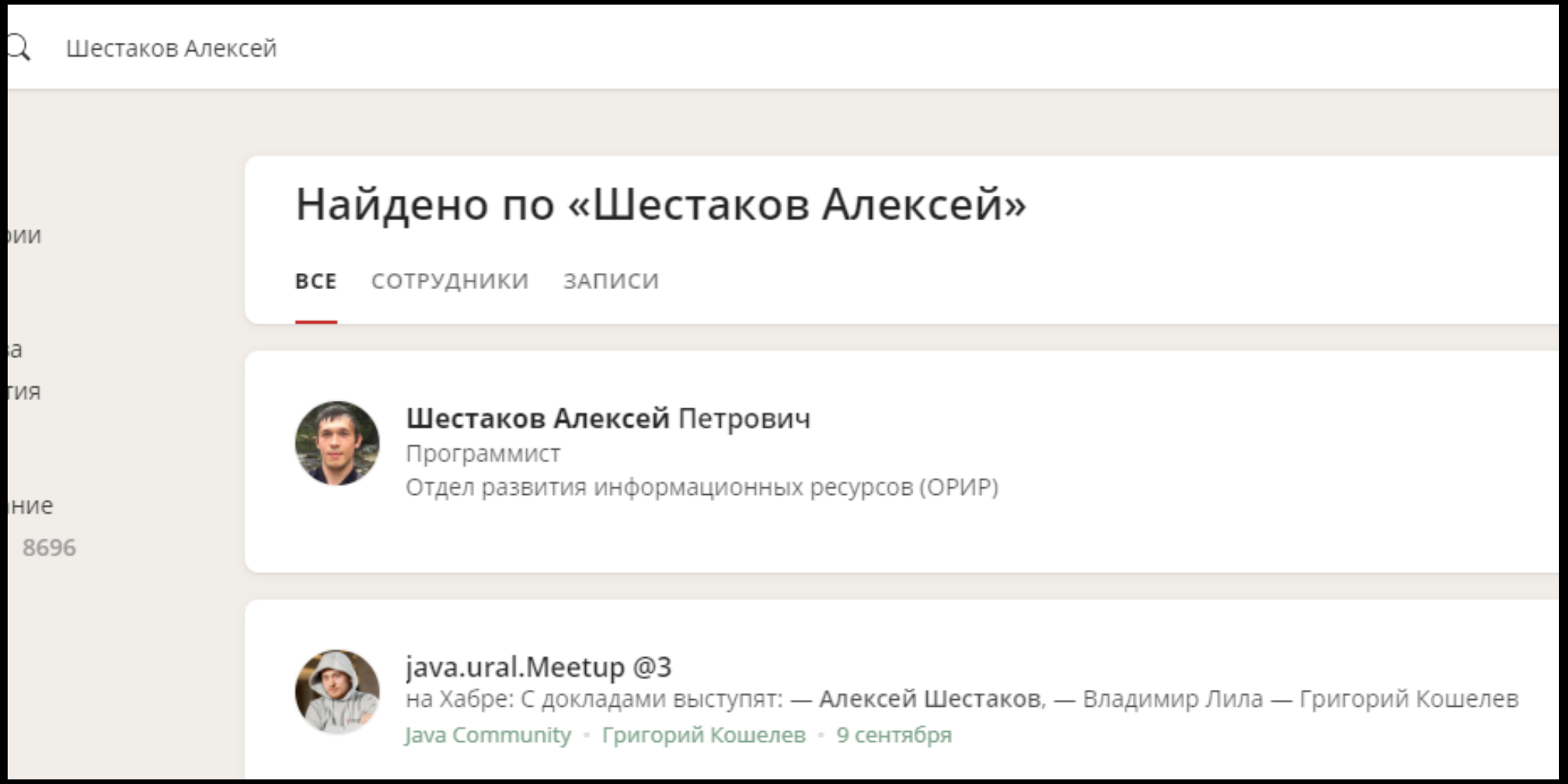
Контур.Стафф

- Использует продвинутые фишки ES
  - Скрипты ранжирования для управления результатом
  - Вложенные объекты
  - Релевантностью полей в запросе
- Настроены фильтры управления видимости
- Агрегирование



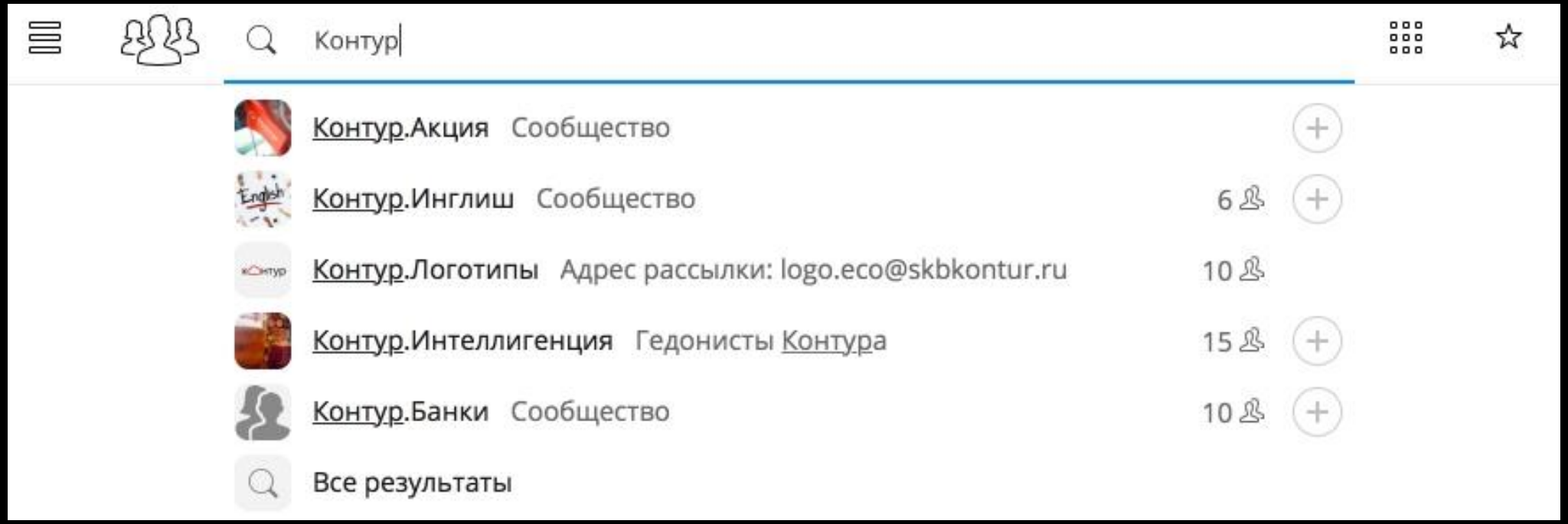
Поиск в Стафф

Это и есть SERP Стафф

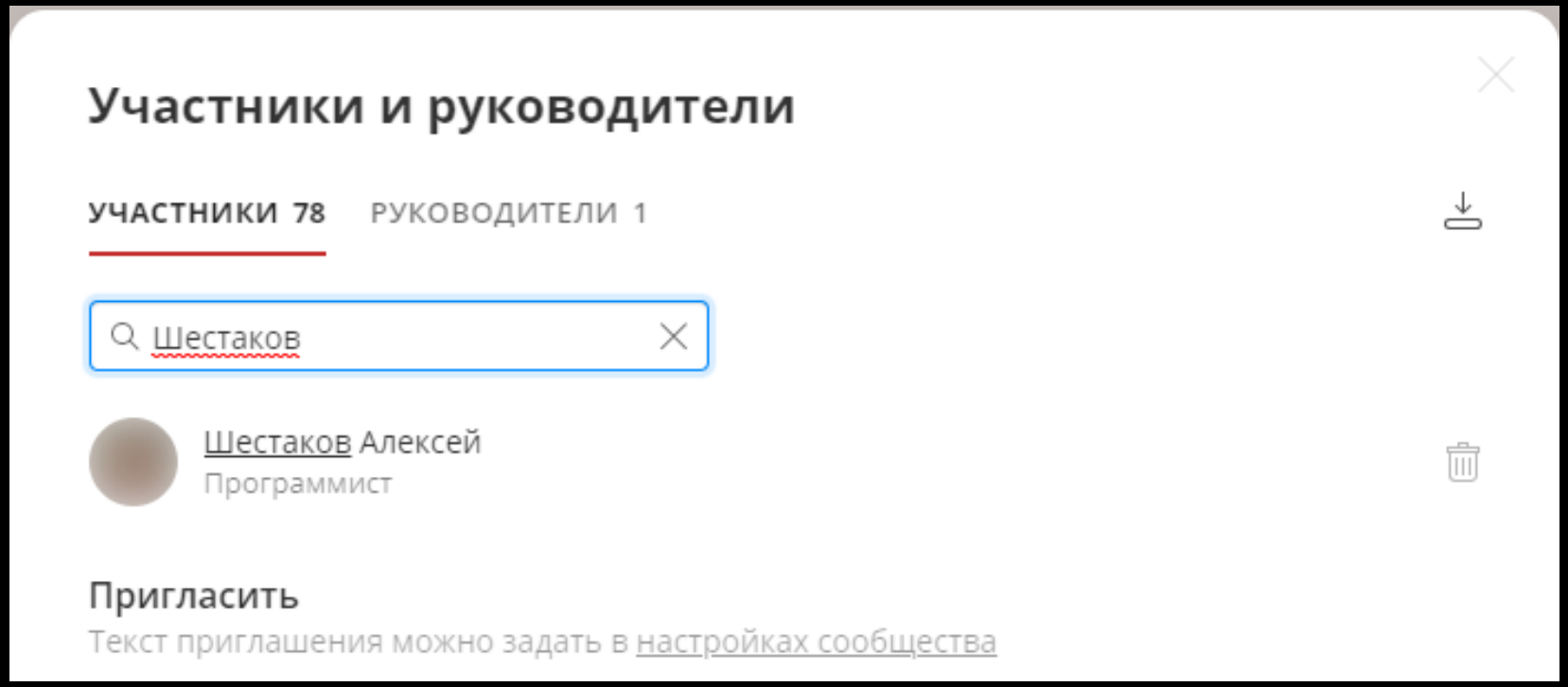


Поиск в Стафф

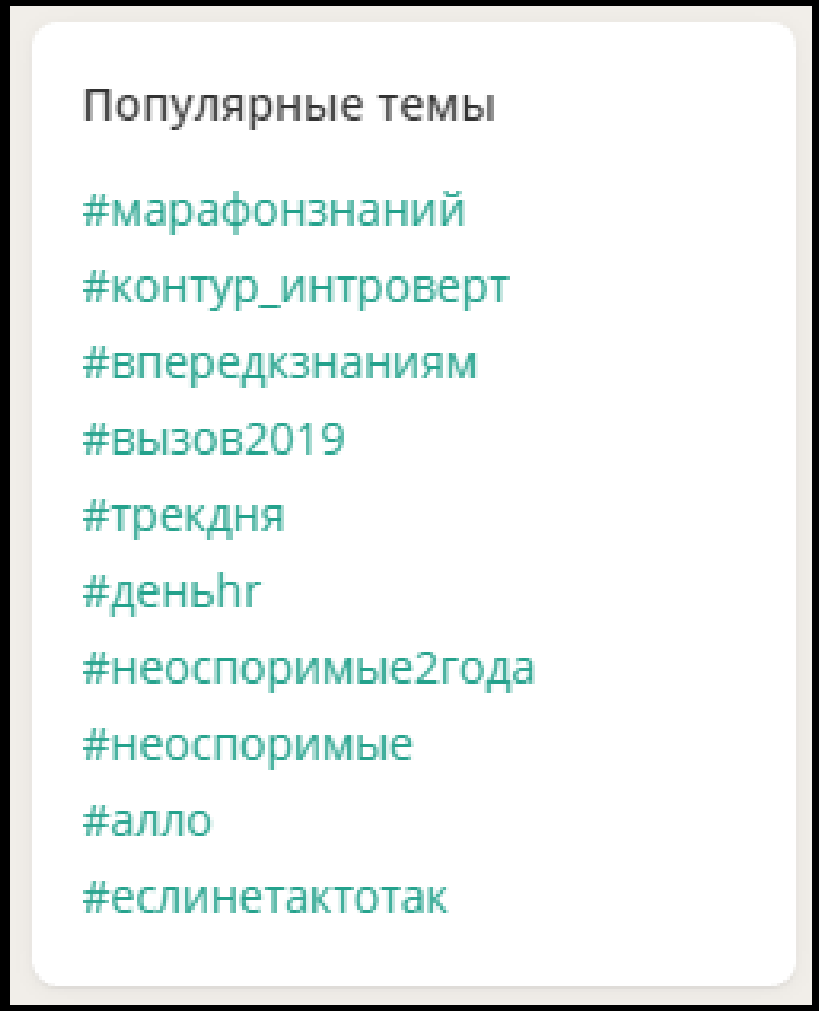
Это и есть SERP Стафф



Поисковая подсказка в Стафф



Это тоже подсказки



Популярные теги за последнее время

Что дальше

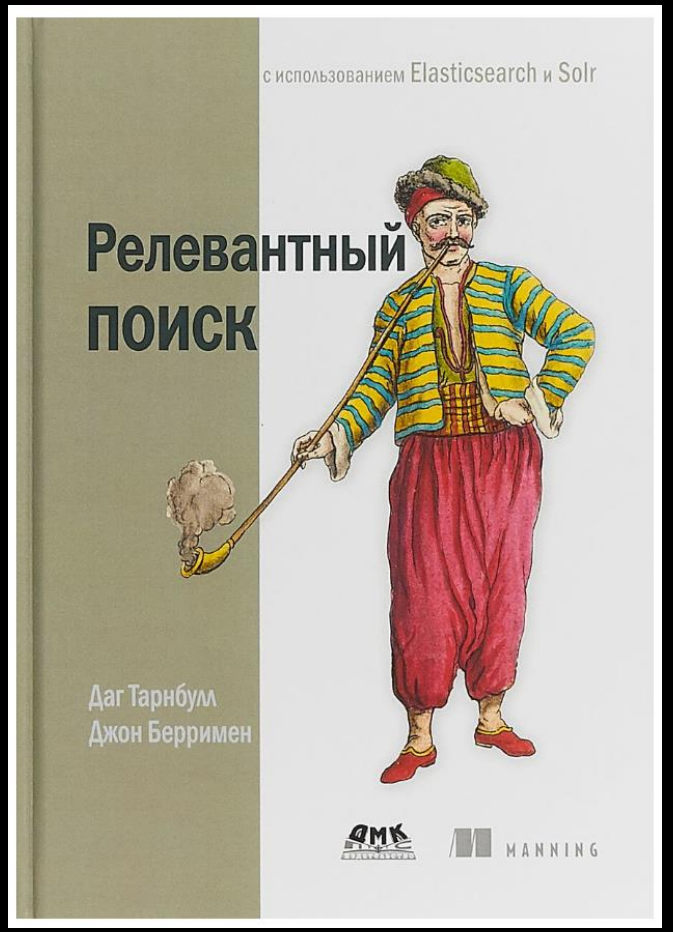
Литература

- Хорошее введение в теорию информационного поиска
- Идеально если вы хотите глубже понимать ES либо написать свой аналог
- Новой редакции нет с 2008 г.
- Устарела в ML части.

https://www.ozon.ru/context/detail/id/5497130/

Литература

- Конкретные рецепты
- Практика использования ES
- Ориентация на ручное управление ранжированием без ML



https://www.ozon.ru/context/detail/id/144631193/

Вопросы?

Алексей Шестаков

Инженер-Программист

shestakovap@skbkontur.ru

kontur.ru