

Talisman

Платформа для построения предметно-ориентированных систем
поиска и извлечения информации

Денис Турдаков

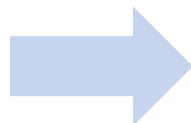
Технологии анализа данных

Неструктурированные данные

- Тексты, изображения, видео...
- Развивается с 90х годов
- Информационный поиск, модель MapReduce и ее развитие, аналитика по метаданным
- Неструктурированных данных на порядок больше, чем структурированных
- Машинное обучение для структуризации данных



Извлечение
информации



Information
Extraction

Структурированные данные

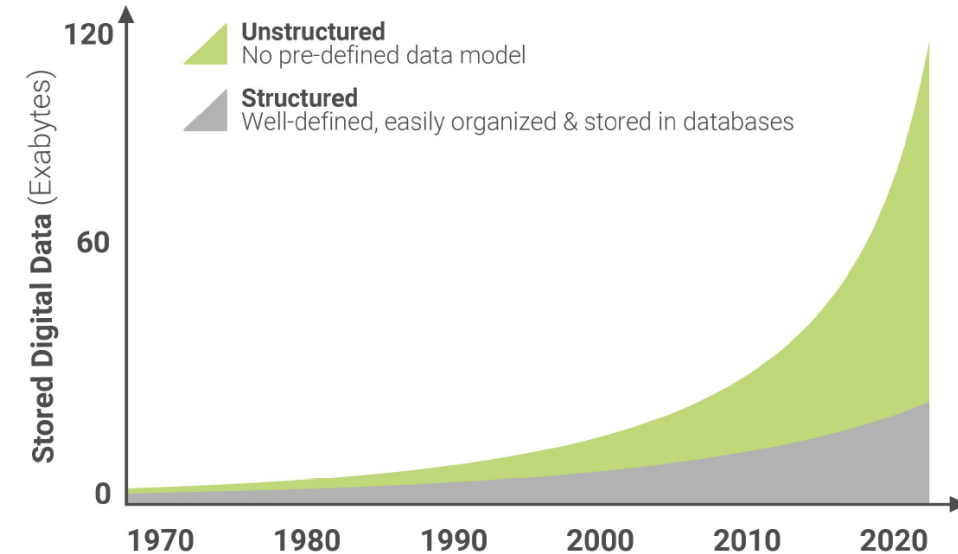
- Реляционная модель
- Развивается с 60х годов
- OLTP, OLAP, интеграция данных, BI-аналитика
- Лучше подходят для автоматического анализа:
 - больше информации можно извлечь
 - зрелые технологии
 - больше специалистов, проще обучать



Извлечение информации

- Задача состоит в извлечении структурированной информации из неструктурированных данных
 - Развитие технологий **больших данных** и методов **машинного обучения** открыло новые перспективы для внедрения технологий извлечения информации
- Проблемы
 - Универсальные методы извлечения информации имеют низкое качество
 - Точечное внедрение технологий → нет общепринятых интерфейсов, сложнее объяснить пользователям

Решение: разработка и внедрение предметно-ориентированных систем



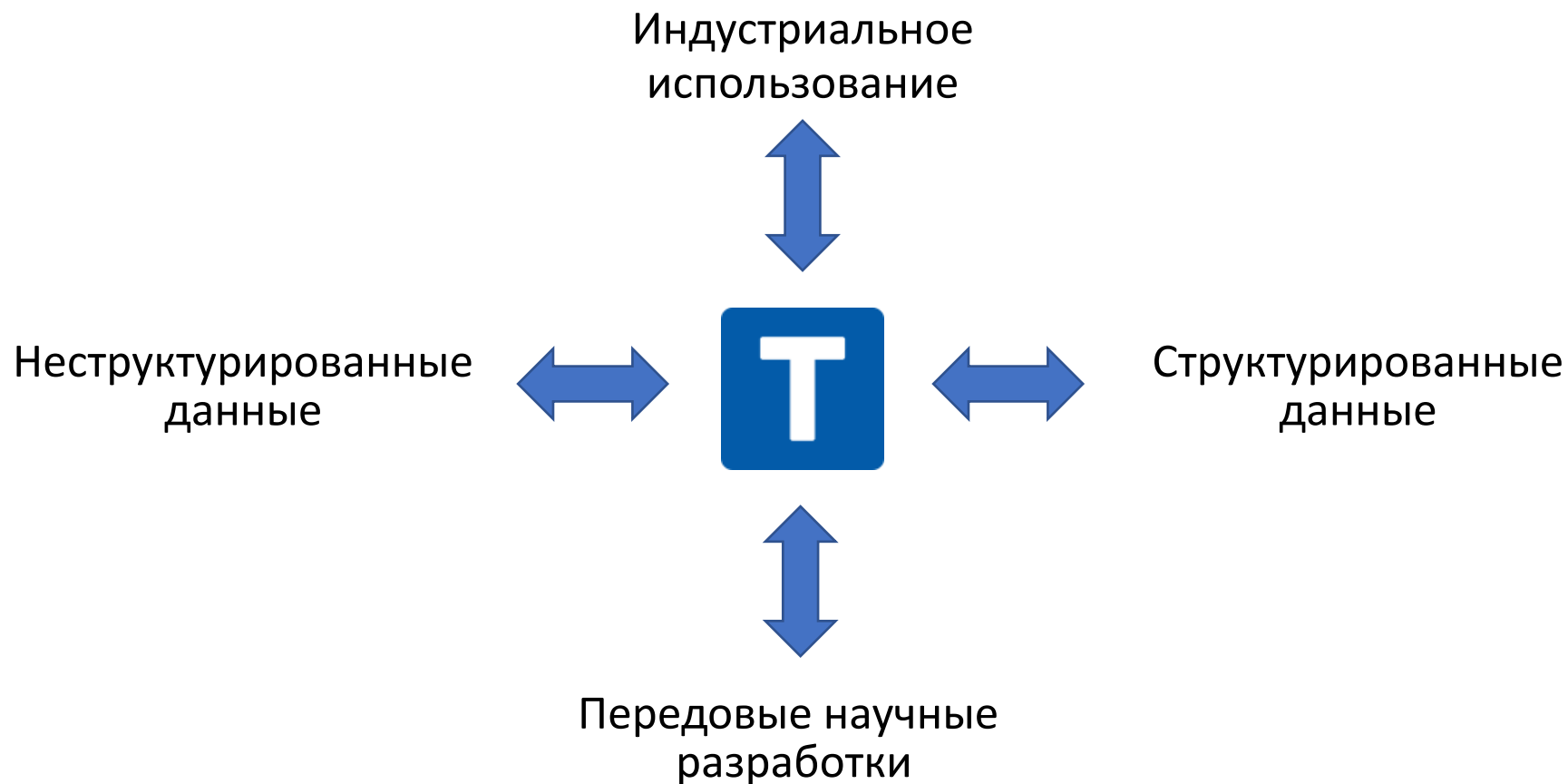
Внедрение технологий извлечения информации

- Область с наибольшим числом инноваций
 - Огромное количество академических открытий (область data science)
 - Скорость внедрения (time-to-market) определяет конкурентоспособность
- Проблема – разрыв между наукой и индустрией
 - ученый сделал алгоритм, но чтобы его внедрить нужна инфраструктура на несколько порядков более сложная, чем для экспериментов
 - принципиально новые подходы сложно внедрять, так как это требует переобучение персонала и перестроение существующих процессов

Решение: платформа, отвечающая индустриальным требованиям, сочетающая классические инструменты работы с данными, понятные бизнесу, и позволяющая быстро внедрять новые алгоритмы, разработанные академическим сообществом

Talisman

Платформа для построения предметно-ориентированных систем поиска и извлечения информации

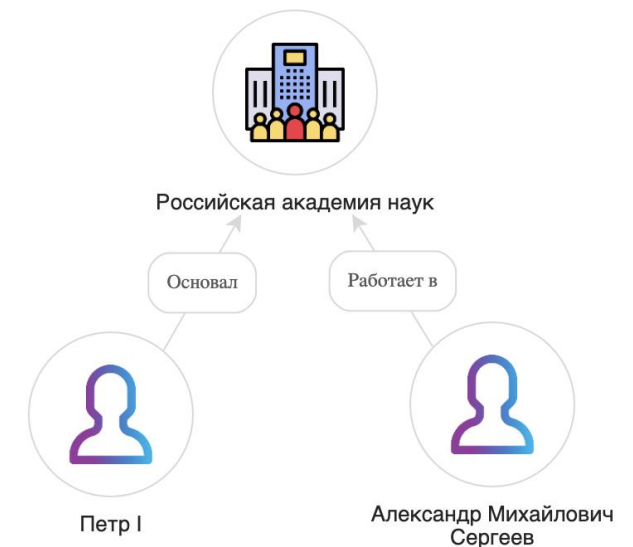


Talisman

Принципы работы

База знаний

- Центральная часть систем извлечения информации
 - Основная цель работы – автоматизация наполнения базы знаний
- Логическая модель: типизированный граф с атрибутами (typed label-property graph)
 - концепты (объекты, события)
 - характеристики (атрибуты) концептов
 - связи между концептами (бинарные)
 - характеристики связей
- универсальная модель знаний
 - эквивалентна моделям semantic web (RDF)
 - выбор модели обусловлен интуитивной понятностью



База знаний: примеры применения

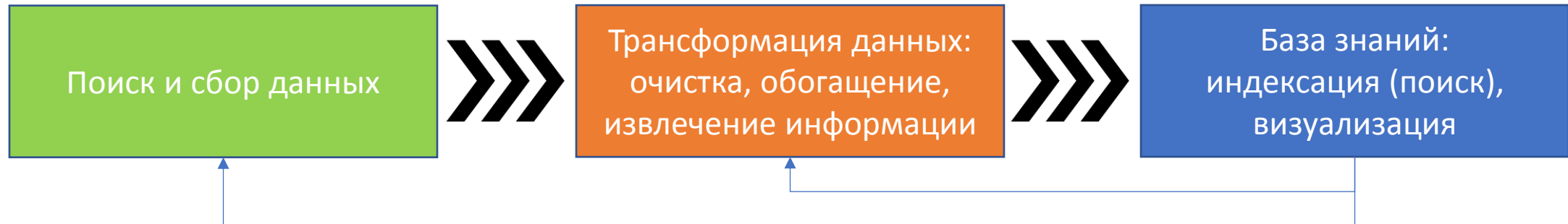
- Системы мониторинга, ситуационные центры
 - Быстрое реагирование при чрезвычайных ситуациях
 - Мониторинг утечек и нарушения корпоративной этики сотрудниками
 - Выявление жалоб на товары/услуги
 - Расчет потребительских цен (Росстат)
 - Выявление антиконкурентных соглашений (ФАС)
- Агрегация информации
 - Помощь в покупке дорогостоящих товаров
 - В какую школу отдать ребенка
- Расширение возможностей поиска
 - Поддержка коллективных научных исследований
 - Поиск серийных преступлений (МВД)
 - Вычислительное право (анализ правоприменения, помощь в экспертизе НПА)
- Системы разведки по открытым данным (OSINT)
 - Конкурентная разведка
 - Оценка контрагентов
- Основа для других систем: вопросно-ответных, диалоговых, систем поддержки принятия решений
 - Повышение продуктивности работы врачей



Нефункциональные требования

- Доверие к результату
 - Предсказуемая **полнота** данных, используемых для принятия решения.
 - Доверие к источникам информации, подтверждение из разных источников
 - Доверие к результатам анализа: алгоритмам или лицу, верифицирующему результат
 - Возможность самостоятельно улучшить систему: добавление новых источников, добавление алгоритмов автоматической обработки, уточнение области интереса
- Продуктивность
 - Скорость выполнения работы в системе должна быть выше, чем без нее (целесообразность использования)
 - **Точность** результатов автоматизации
 - Простота освоения системы: привычные и интуитивно понятные интерфейсы, обучение использованию
 - Систематизация знаний (возможность учесть больше факторов, чем можно удержать в голове)
 - Сохранение результатов работы (в т. ч. других пользователей) и автоматическое использование при решении новых задач
 - Интерактивная визуализация данных для поиска скрытых закономерностей
 - «Самообучение» системы на основе анализа действий пользователей
- Эффективность: обеспечение приемлемой **актуальности** результатов
 - Пакетная VS потоковая обработка
 - Автоматическое масштабирование ключевых сервисов
 - Минимизация необходимых для работы ресурсов
- Безопасность
 - Защита результатов работы от порчи и утечки
 - Защита корпоративных, персональных, грифованных данных
 - Защита от атак на модели машинного обучения

Общая схема извлечения информации

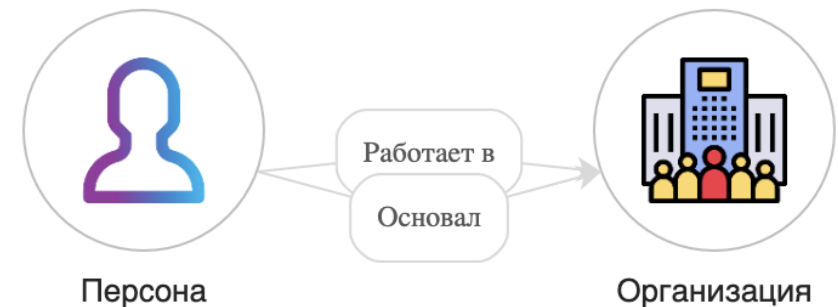


Процесс извлечения знаний

1. Определить область интереса
2. Определить источники информации
3. Настроить постоянный процесс сбора данных, извлечения и загрузки интересующей информации в базу знаний
 - Инструменты извлечения информации на каждом этапе
 - Накопление знаний должно улучшать качество извлечения (обратные связи)

Предметная область (область интереса)

- Предметная область определяет **схему базу** знаний
- Типизированный граф с атрибутами, как база знаний, но определяет **типы**
 - Концептов
 - Характеристик концептов
 - Связей
 - Характеристик связей
- Возможно модифицировать предметную область в процессе эксплуатации системы



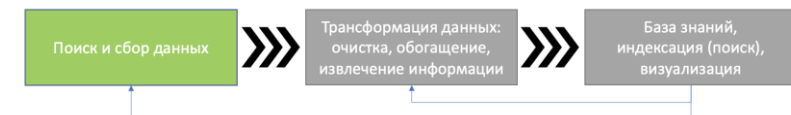
Источники информации

- Интернет
 - **Веб-сайты:** СМИ, журналы и издательства, конференции, тендерные площадки, патентные ведомства, персональные страницы
 - **Аккаунты** людей и организаций в соцсетях, мессенджерах, форумах
 - Агрегаторы информации: общие и специализированные поисковые сервисы, новостные агрегаторы
- Локальные файловые хранилища и СУБД
- Компании - поставщики данных



Talisman: средства сбора данных

- Универсальные сборщики
 - Сборщики отдельных страниц и обход сайтов + первичная очистка
 - Сборщики RSS (извлекают основной текст по ссылке и метаданные: название, аннотация, время публикации)
 - Plugin для браузеров: сохранение текущей страницы в систему (со скриншотом)
 - Направление Autoextract – автоматическое извлечение информации об области интереса с сайта
- Создание карт обхода с помощью визуальной разметки сайтов
 - Plugin для браузеров. Свободное ПО на github ИСП РАН
 - Прозрачная интеграция с логической моделью Talisman и специфичной предметной областью
- Сборщики социальных сетей (через API и веб-интерфейс)
- Сбор результатов запроса к поисковым системам
- *Создание собственных сборщиков на Python + Scrapy и добавление в систему в виде модулей (egg) через веб-интерфейс*



Talisman: подсистема сбора

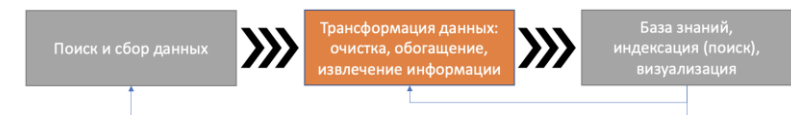
- Веб-интерфейс для
 - Запуска сбора из источника
 - Мониторинга состояния сбора
 - Запуска сбора по расписанию
 - Управления проху и аккаунтами
 - Обход блокировок (цепочки прокси)
 - Сбор от имени зарегистрированного пользователя
 - Импорта данных из файловых хранилищ
- Расширения для браузеров
 - Сохранения страниц и их скриншотов
 - Создания карт сайтов
- Полученные **сообщения** имеют структуру key-value в формате json
- Осуществляется первичная дедупликация
- Складываются в
 - Постоянное хранилище (PG),
 - Очереди брокера сообщений Kafka (каждый сборщик в свои)
 - Приложенные файлы складываются в S3- хранилище (minio, serph)



Talisman.Поток: обработка сообщений

- Система обработки входящего потока сообщений и файлов с целью извлечения информации и загрузки ее в базу знаний
- Логическое устройство:
 - **сообщения** преобразуются в универсальное представление **документа (TDM)**, проходя через *настраиваемую последовательность обработчиков*
 - выбор и настройки обработчиков зависит от входных данных и предметной области
- Обработчики
 - Работа с сообщениями в формате json key-value
 - Работа с сообщениями в формате TDM (формат для хранения иерархического документа с фактами)

- Дедубликация
dedup
- Извлечение файлов
messageunpacker
- Построение модели документа
docreader
- Переименование ключей
key-translator
- Извлечение метаданных
metadata-extractor
- Извлечение фактов
facts-builder
- Семантический анализатор
ie
- Нормализация значений
value-normalizer
- Загрузка в базу знаний
docs-loader

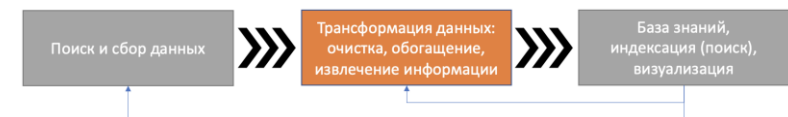
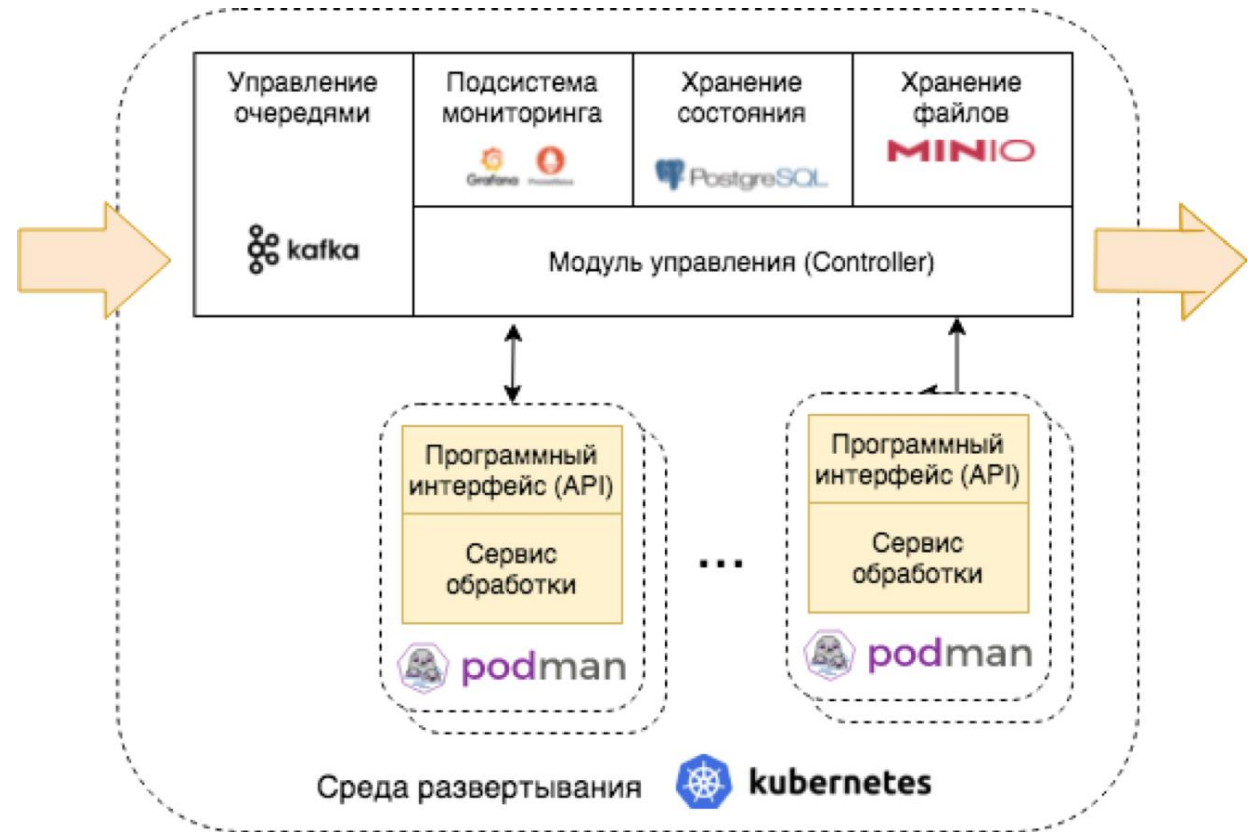


Talisman Document Model

- Универсальное представление документа, в которое преобразуются все сообщения, поступающие в Talisman (json, docx, xlsx, pdf, jpg, ...)
- Документ состоит из трех связанных частей
 - Содержимое (текст со структурой)
 - Метаданные (размер, дата создания, ...)
 - Список **фактов**, извлеченных из документа
- Извлеченная информация представляется фактами

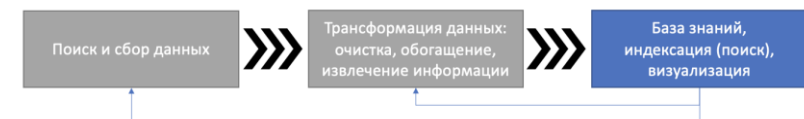
Talisman.Поток: принципы работы

- Принципы работы
 - Поточковая обработка данных
 - Расширяемость (микросервисная архитектура)
 - Эластичная масштабируемость (средствами k8s)
- Сервисы
 - получают на вход сообщения и конфигурацию (json)
 - Возвращают обогащенное сообщение

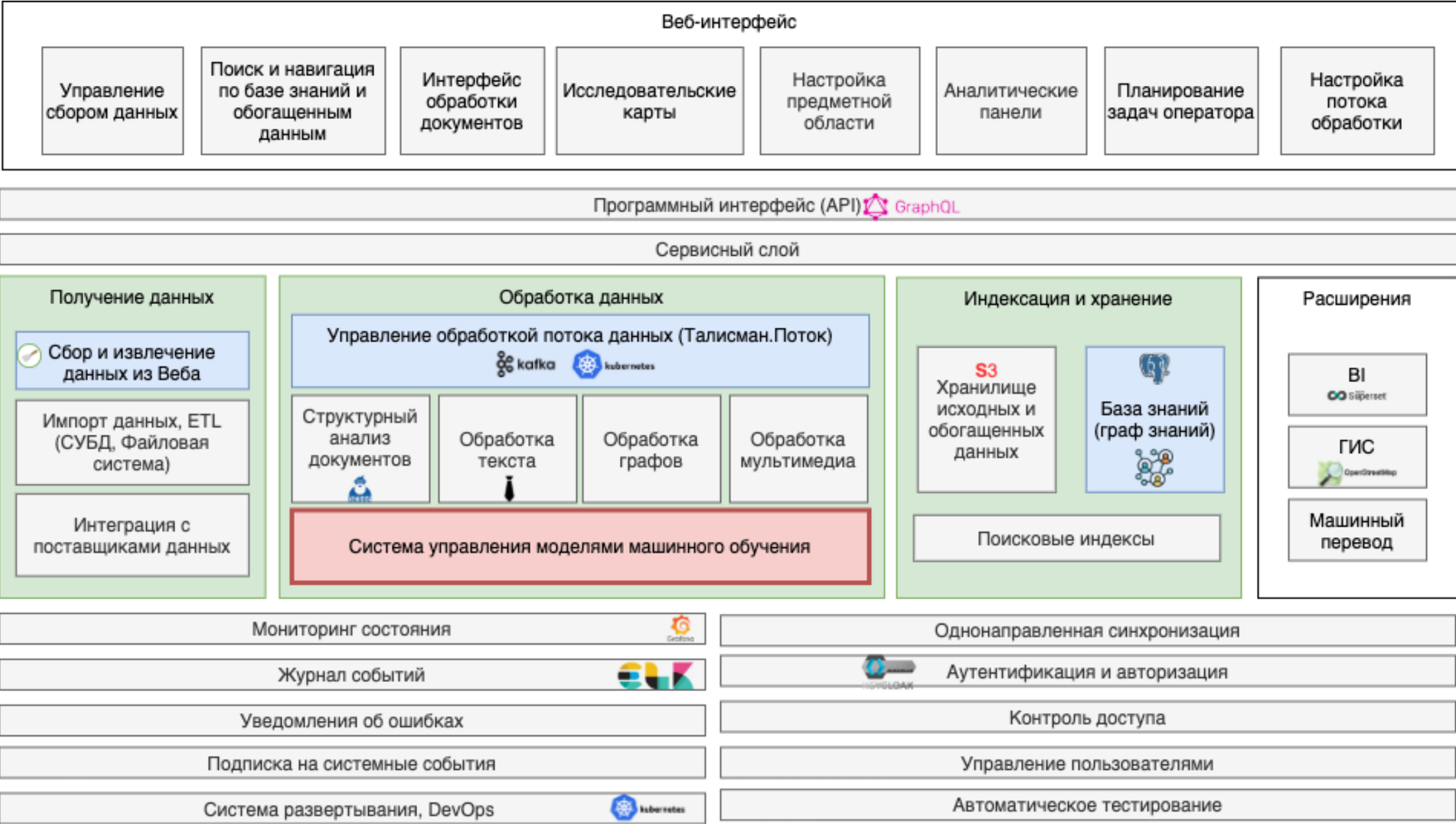


Talisman: управление знаниями

- Каждый факт в базе знаний подтверждается документом
 - Хранилища исходных сообщений, файлов и полученных документов
- Каждый факт может быть верифицирован человеком
 - Система разметки документов
 - Обучение ML-моделей на действиях оператора
- Работа в нескольких контурах
 - Интеграция данных из открытых источников и чувствительных корпоративных данных
- Средства поиска и фильтрации собранных данных
- Аналитические инструменты
 - **Исследовательские карты:** инструмент для интерактивной работы с графом знаний
 - Дашборды с возможностью кастомизации (Apache Superset)
- Импорт
 - Генерация текстового отчета в pdf или docx для дальнейших правок
 - Выгрузка информации из БЗ в машиночитаемом формате
 - Автоматическая загрузка информации в другие системы



Общая архитектура



Разработка систем на основе Talisman

Спиральная модель: MVP (прототип), далее наращивание функционала

Создание прототипа

1. Описать сценарии использования (user story)
2. Определить глоссарий (используя типовой)
3. Сформулировать требования к системе (используя типовое ТЗ)
4. Описать предметную область
5. Определить источники данных
 - Уже поддерживаемые будут в системе сразу
 - Другие можно добавлять в процессе доработки
6. Задать процесс обработки данных
 - Отображение из схемы источника в схему базы знаний
7. Реализовать базовые реализации (baseline) отсутствующих обработчиков
8. Развернуть стенд

Разработка новых версий

1. Расширение пользовательской функциональности
 1. Разработка макетов интерфейсов
 2. Реализация Backend – API – Frontend
 - Модульный Backend на Scala (можно делать отдельные сервисы на произвольных технологиях)
 - Интеграция с системой на уровне API (GraphQL, Mesh)
 - Веб-интерфейс (Typescript – React – Ant Design)
2. Создание собственных обработчиков данных
 - Контейнеры Podman и типовыми интерфейсами
3. Добавление сборщиков
 - Задание точки входа для **универсальных сборщиков**
 - Создание карт обхода с помощью визуальной разметки
 - Написание собственных сборщиков (Scrapy)

Демонстрация

Эластичное масштабирование

