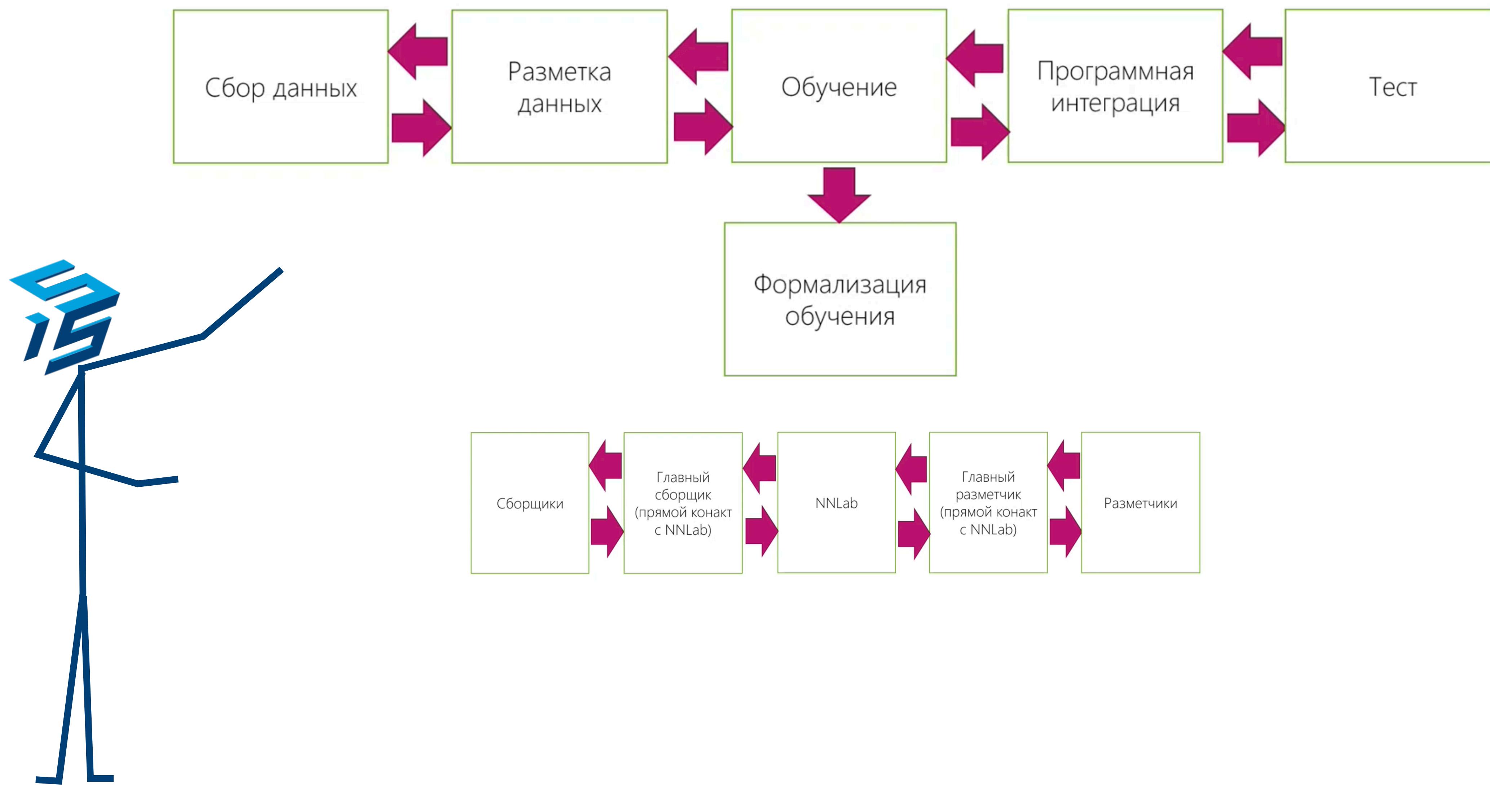


Data Collection Tool

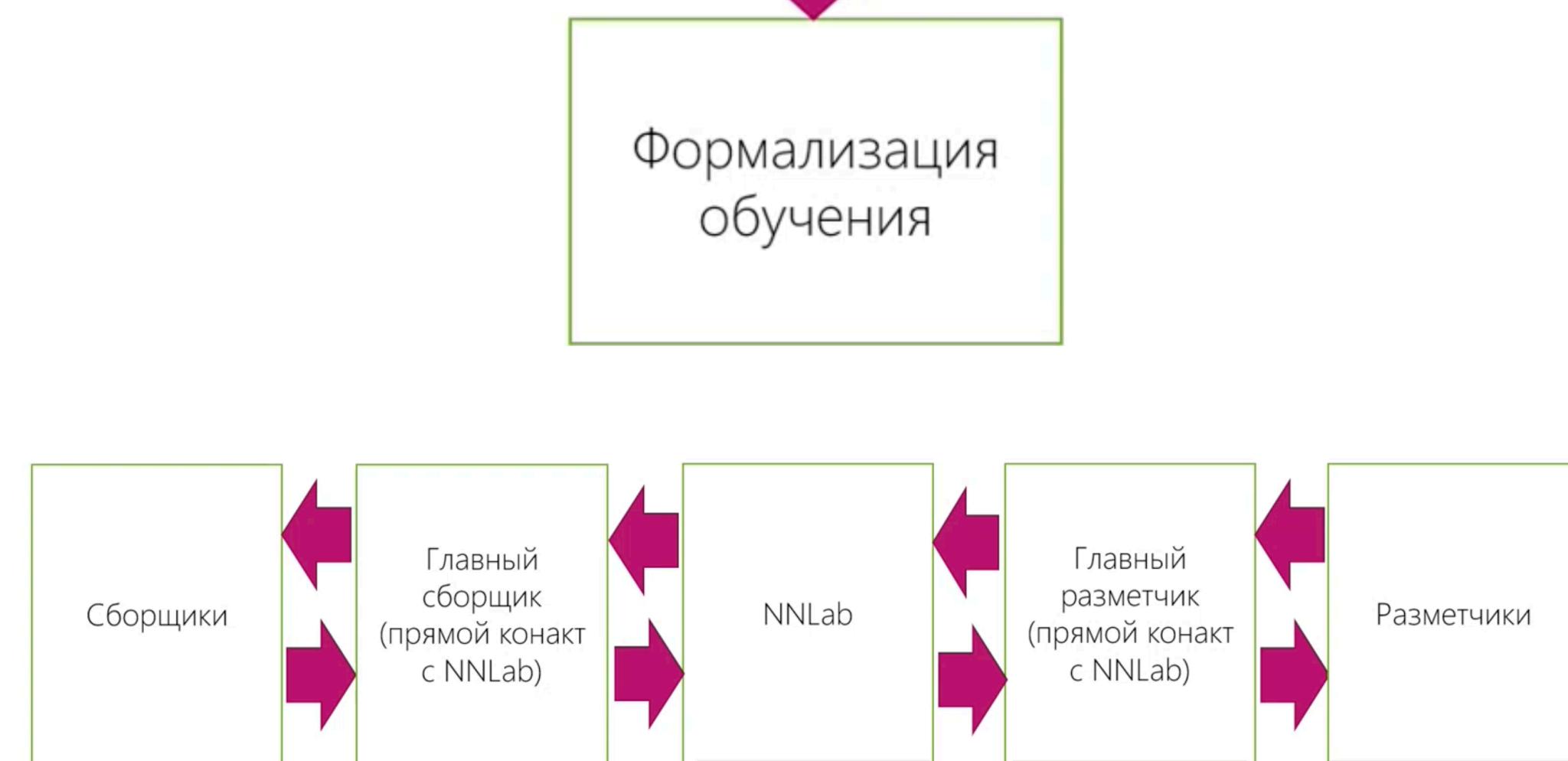
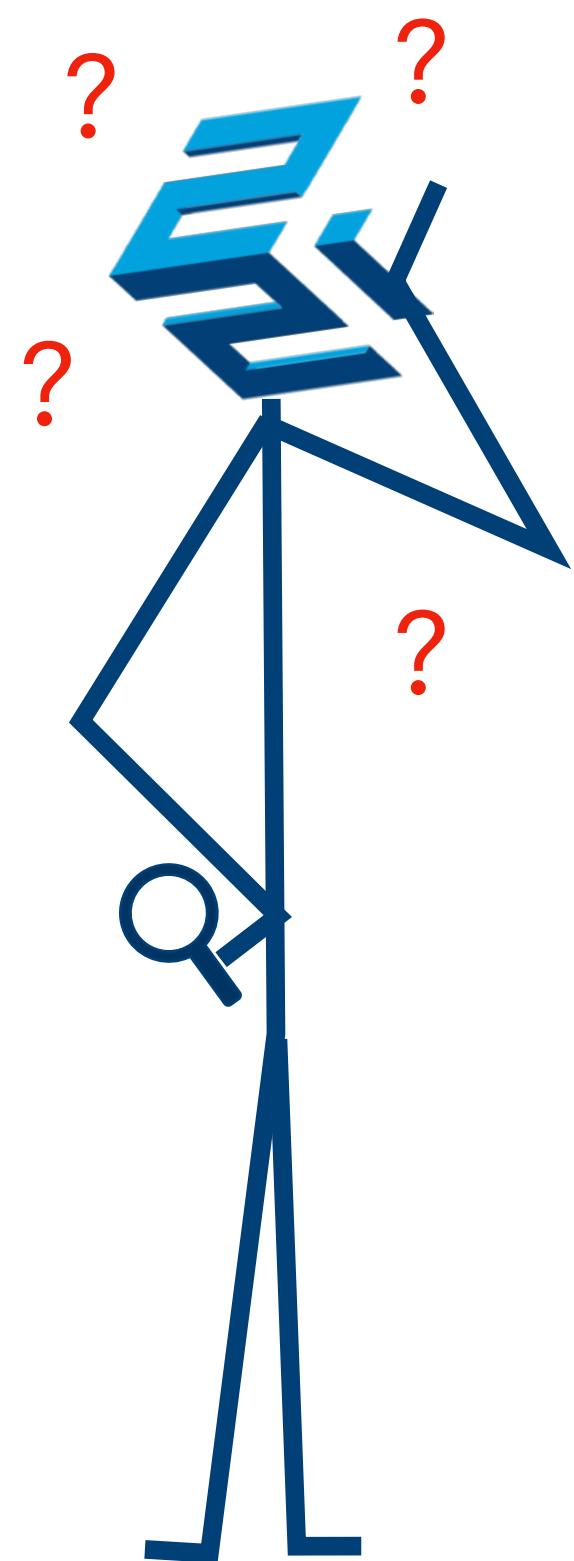
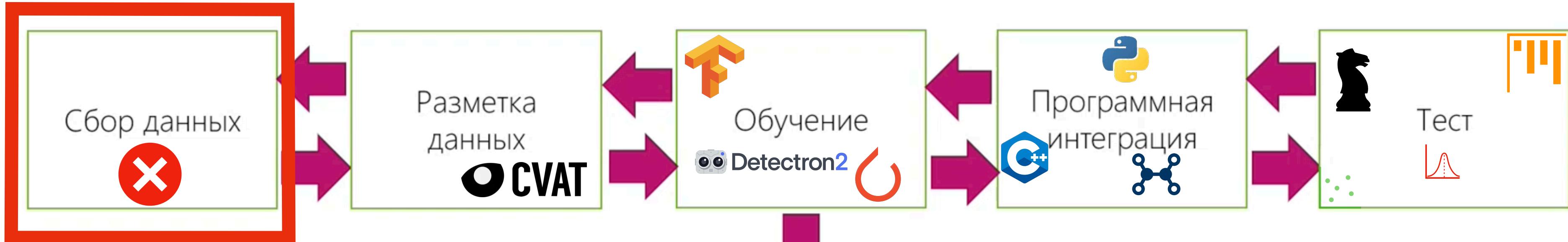
Приложение для сбора данных

Проблема_1

Разработка



Технологии



Data Collection Tool

Решение!



Projects

[all projects](#) [create project](#)

Saudi Arabia

Saudi Arabia LPR

2023-04-27T09:56:08.908458Z

Qatar

Qatar LPR

MMR

Make Model Recognition

2023-05-16T10:54:29.456870Z

Fight

MMA fight videos
резких движений
полный ролик...

Lying people

Videos with lying people

2023-07-14T10:53:48.476943Z

Test Project

desc

Data Collection Tool

- Создание проектов
- Добавление атрибутов
- Загрузка данных
- Валидация данных
- Просмотр статистики
- Выгрузка данных
- Настройка доступов
- И многое другое... (потом)

Система сбора данных с гибкой настройкой, валидацией, просмотром, и сбром статистики

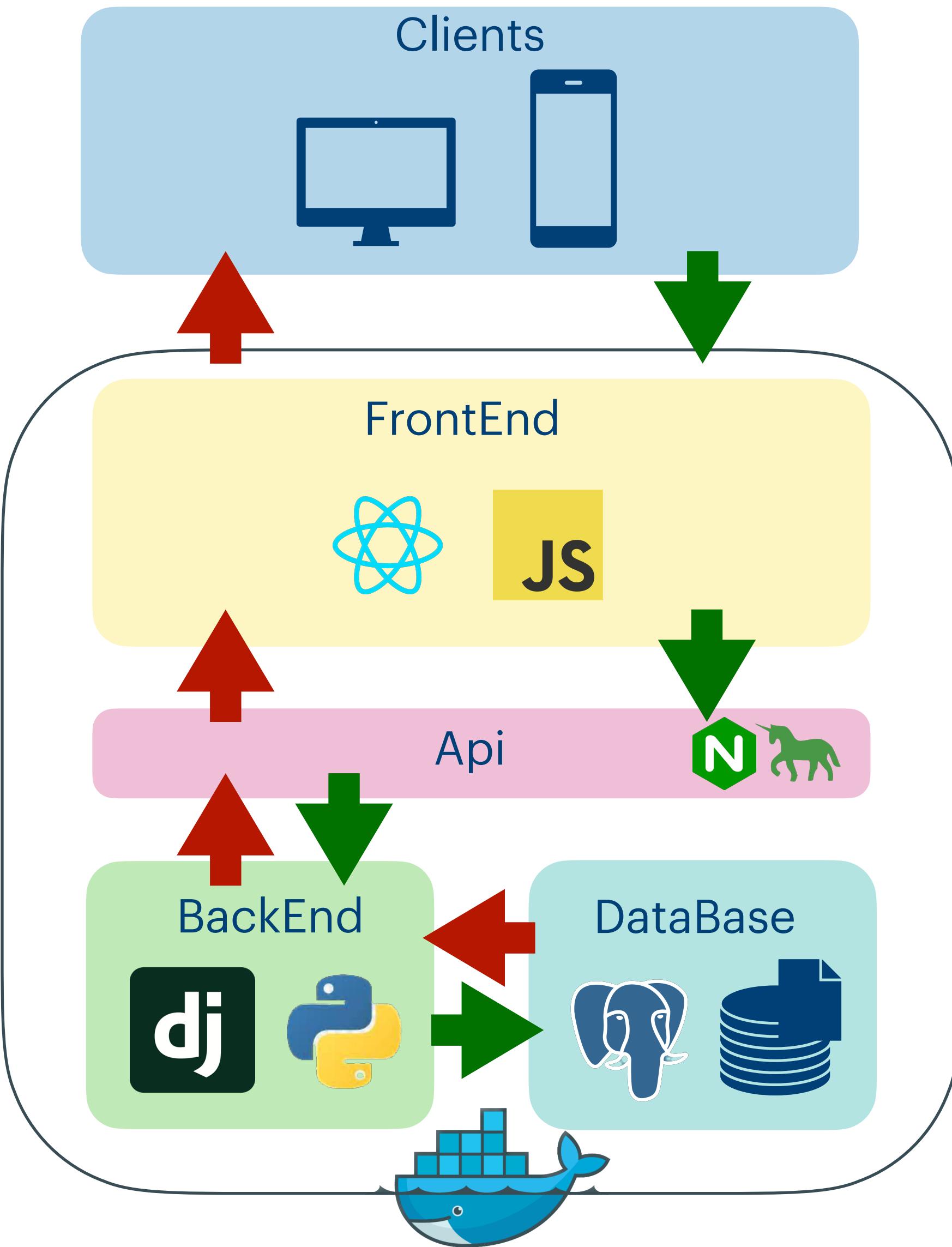
The interface consists of several interconnected panels:

- Attribute Management:** Shows a hierarchical tree of vehicle attributes like Make, Model, Color, Body style, View, and Source.
- Validation:** Displays a table of validation results for various attributes, with columns for Accepted, Declined, and total counts.
- Projects:** A dashboard showing active projects such as Saudi Arabia LPR, Qatar LPR, Fight (MMA fight videos), Lying people, Test Project, and Fire_Smoke.
- File Download:** A list of files to download, each with a checkbox to select manually or via option.
- MMR (Make Model Recognition):** A detailed view of a specific project with filters for file type and a preview of a car image.
- Statistics:** A table showing user permissions for different actions.
- Fight Project:** A specific project for MMA fight videos, showing a preview of a video frame and a list of uploaded files.

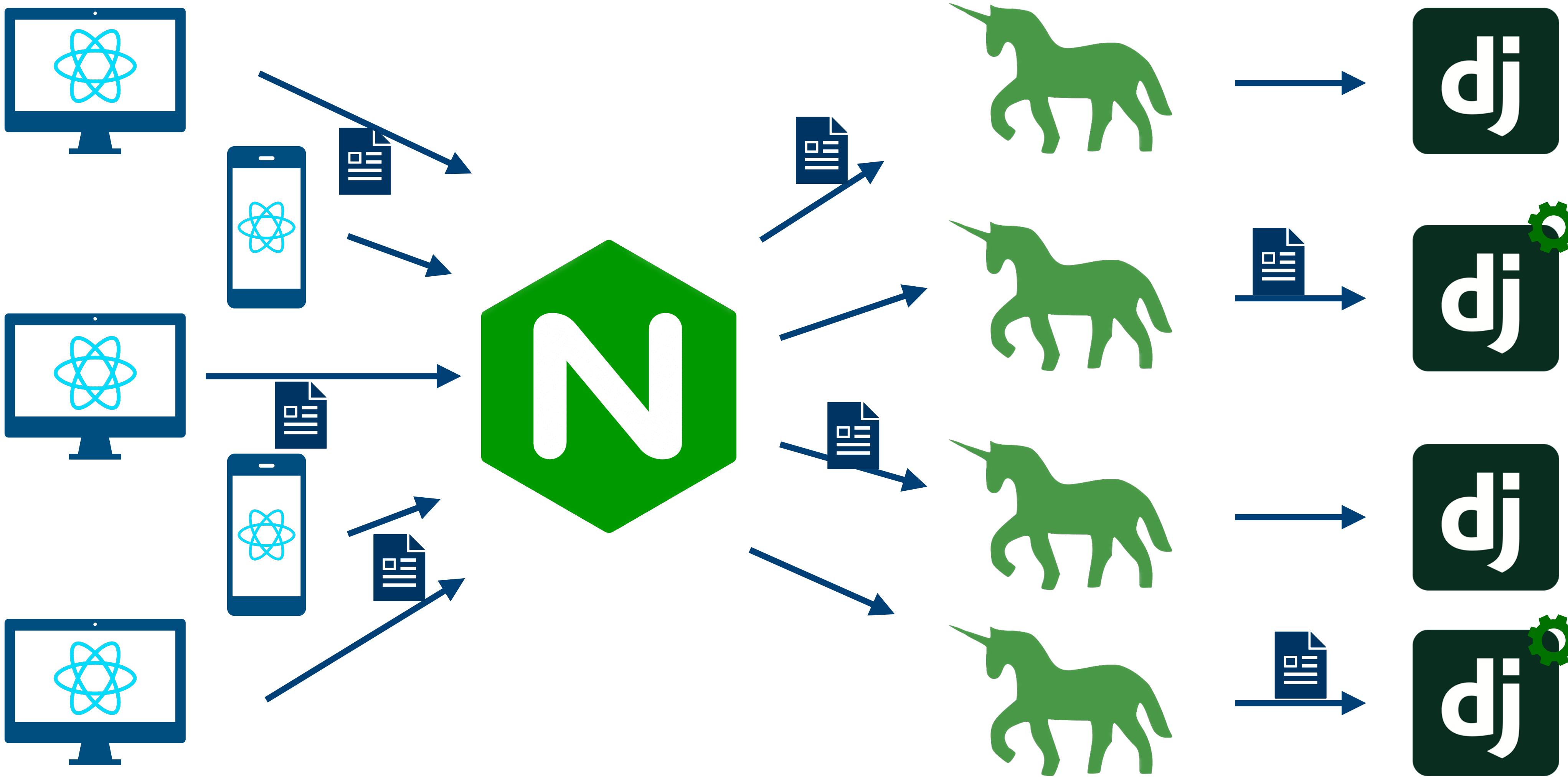
Проблема_2

Проблемы архитектуры

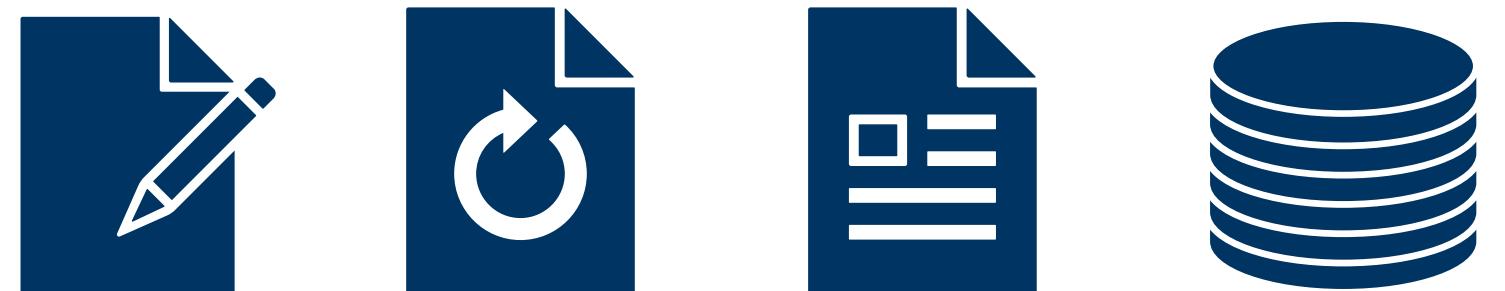
- Передача файла на сервер
- Получение файла с сервера
- Скачивание проекта
- Хранение объектов



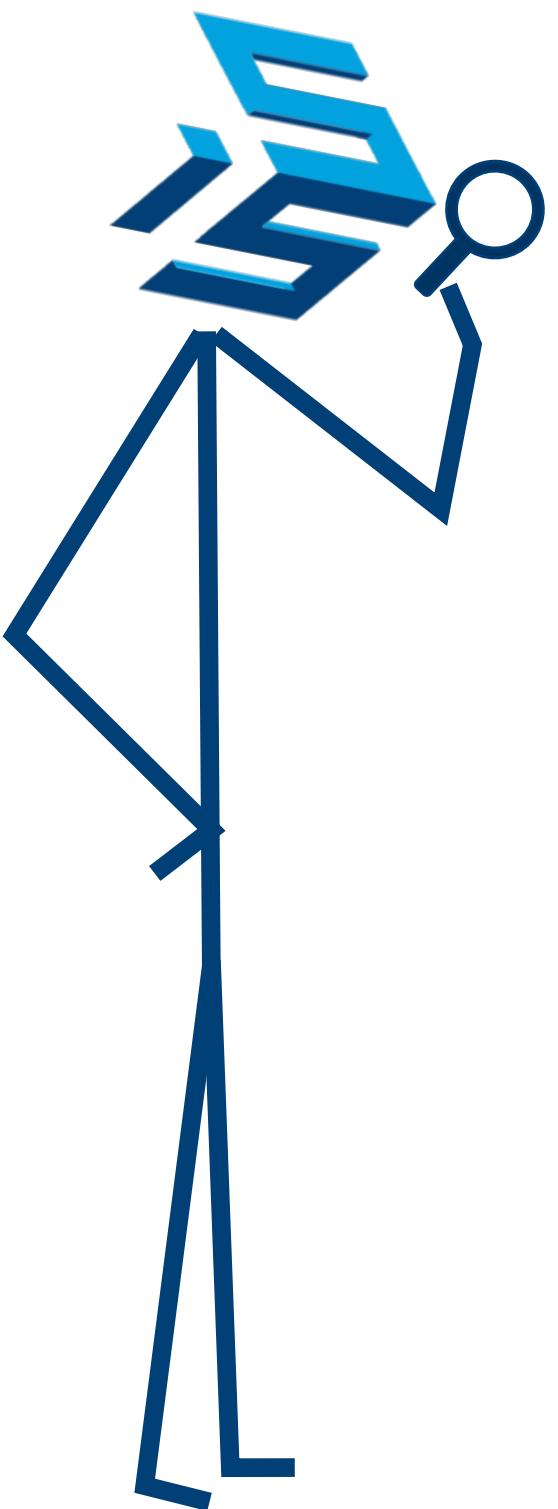
Выполнение запросов



- Зависание сервера
- Ограничения файловой системы
- Бэкап
- Версионирование



Варианты решения



- Objects Storages
- Distributed File Systems
- DataBases

Object Storages

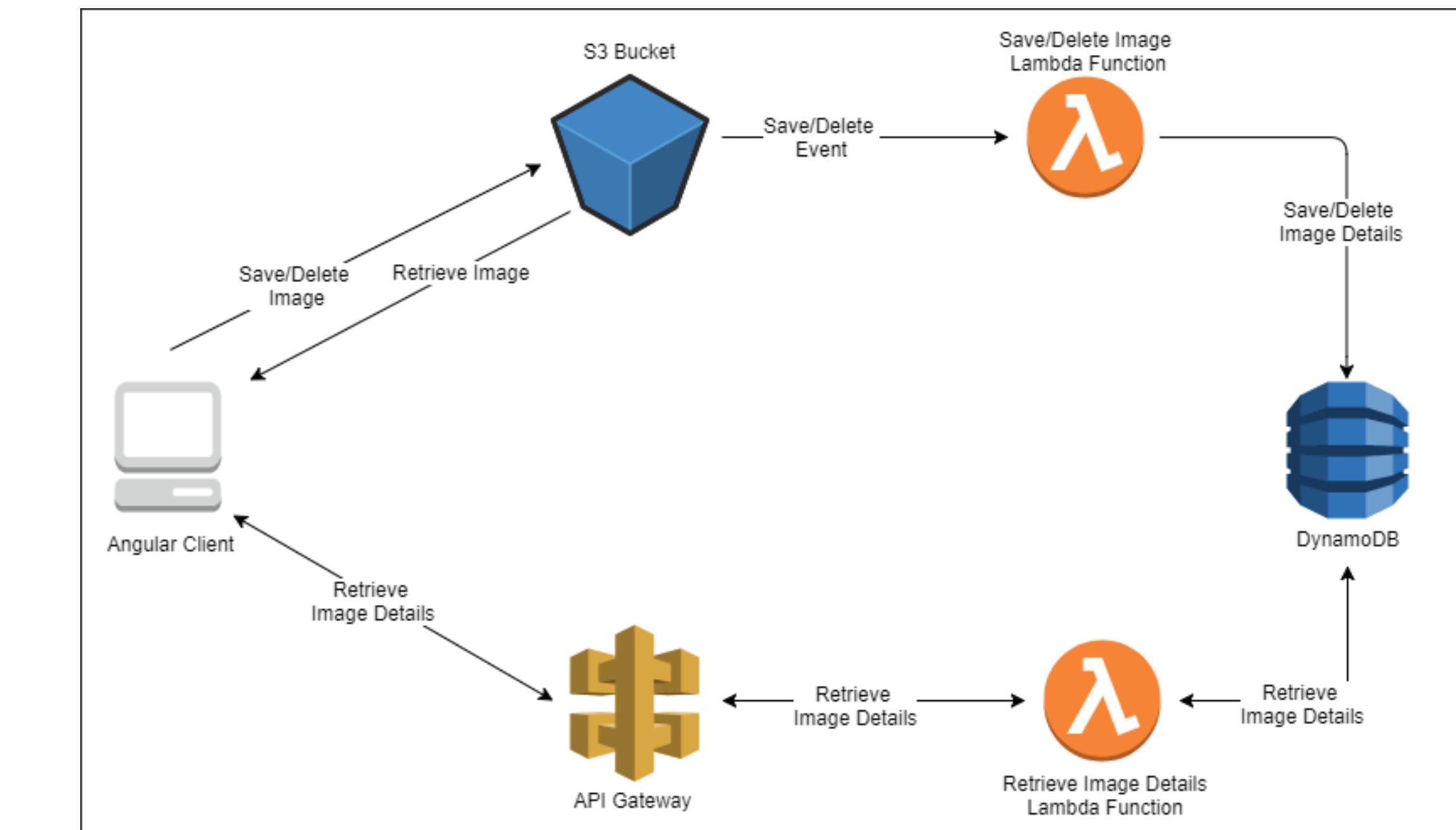
Объектное хранилище – технология хранения и управления данными в неструктурированном формате.

В объектных хранилищах объекты хранятся в одной корзине, а не в виде файлов в папках.

Объектное хранилище обеспечивает архитектуру для хранения массивных объемов данных в виде отдельных объектов, метаданные которых предоставляют им уникальный идентификатор для простого доступа.

Основными преимуществами объектных хранилищ являются практически неограниченные возможности масштабирования и низкая стоимость хранения больших объемов данных.

- Minio
- Cloudian
- G2
- Amazon S3
- Google Cloud
- Microsoft Azure



За/против

- Возможность масштабировать хранение горизонтально практически бесконечно, без значительных изменений в инфраструктуре.
- Реплицирование на разные узлы или центры данных, обеспечивая надежность и доступность.
- Поддержка sdk/api
- Версионирование данных



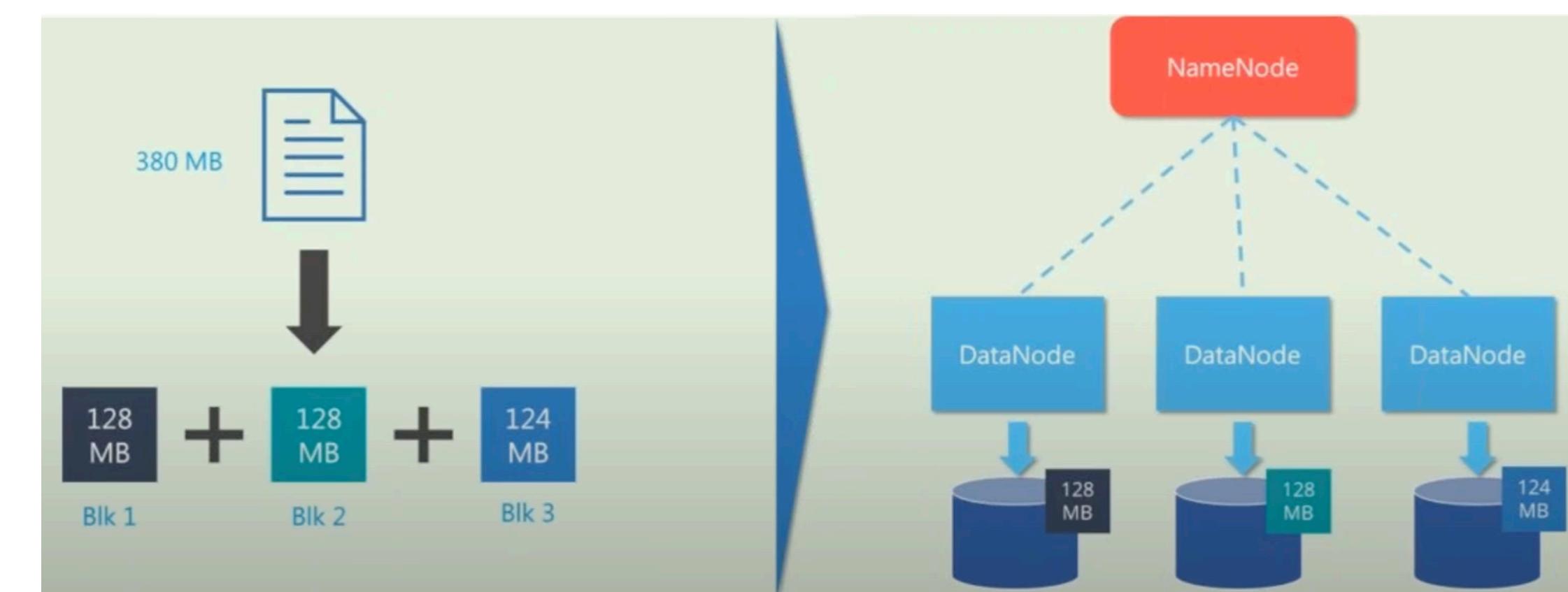
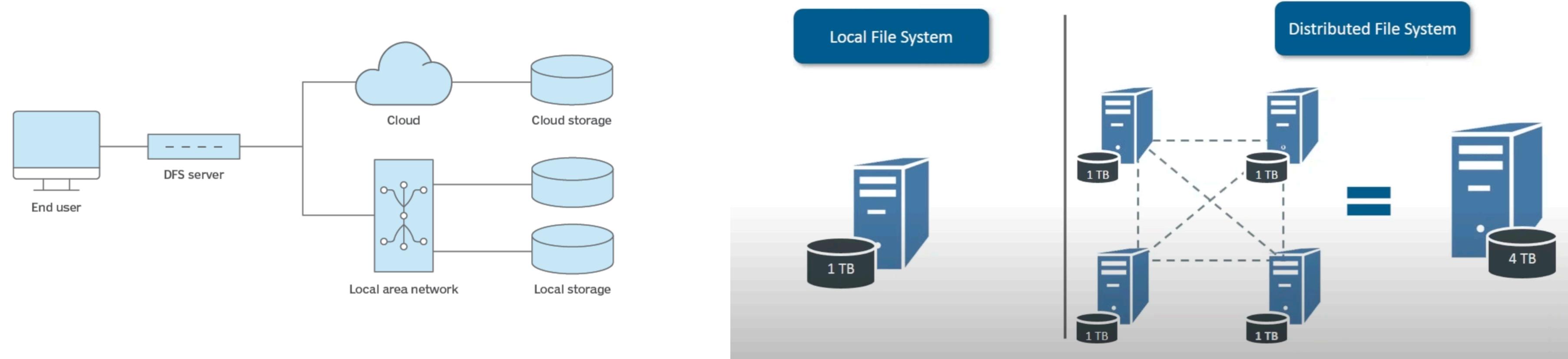
- Менее высокая скорость доступа к данным в случае облачного хранилища.
- Ограничено взаимодействие
- Чувствительность данных
- Закрытость системы

Distributed File Systems

Distributed File System – распределенная файловая система для хранения файлов с возможностью потокового доступа.

Как и любая файловая система – это иерархия каталогов с вложенными в них подкаталогами и файлами.

- Hadoop HDFS
- GlusterFS
- Ceph



За/против

- Горизонтальное масштабирование, добавляя новые узлы или серверы по мере необходимости.
- Репликация данных и автоматического восстановление.
- Поддержка sdk/api
- Поддержка эко системы



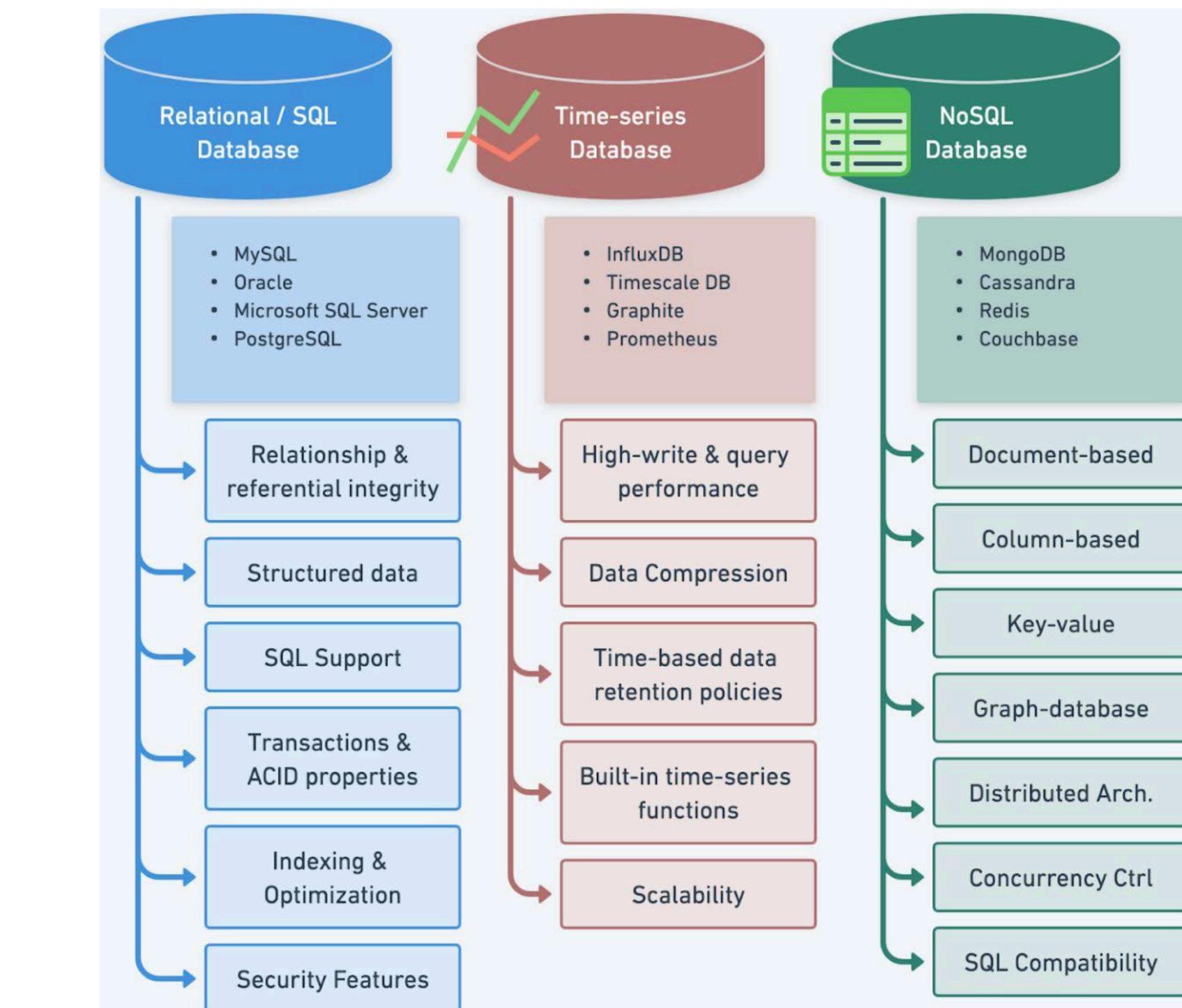
- Закрытость системы
- Ограничено взаимодействие
- Запросы к данным

DataBases

База данных — представленная в объективной форме совокупность самостоятельных материалов, систематизированных таким образом, чтобы эти материалы могли быть найдены и обработаны с помощью электронной вычислительной машины.

База данных — совокупность данных, организованных в соответствии с концептуальной структурой, описывающей характеристики этих данных и взаимоотношения между ними, которая поддерживает одну или более областей применения.

- PostgreSQL
- MySQL
- Oracle
- MongoDB
- DynamoDB
- Redis



SQL vs NoSQL

SQL (Structured Query Language) и NoSQL (Not Only SQL) - два различных подхода к хранению и управлению данными в базах данных.

SQL и NoSQL базы данных имеют разные структуры хранения данных и способы выполнения запросов.

SQL ориентирован на жестко структурированные данные и использует язык SQL для выполнения запросов, в то время как NoSQL предоставляет гибкость в хранении неструктурированных данных с использованием разных моделей данных и языков запросов.

Модель данных

Реляционные базы данных хранят данные в виде таблиц с жестко определенными схемами.

Таблицы имеют строки и столбцы, где каждая строка представляет запись, а каждый столбец определяет тип данных.

Реляционные базы данных обязательно имеют схему данных, которая определяет структуру таблиц и связи между ними.

NoSQL базы данных позволяют хранить данные с гибкой или отсутствующей схемой. Это позволяет хранить разнородные данные в рамках одной базы данных.

NoSQL базы данных поддерживают разнообразные модели данных: включая документы, обычно в формате JSON или BSON; парах ключ-значение; столбцов, подобно таблице; как граф, с узлами и связями.

Запросы

Для выполнения запросов к данным в SQL используется язык запросов SQL.

Он предоставляет средства для выполнения запросов, фильтрации, сортировки и агрегации данных.

Запросы к NoSQL базам данных могут выполняться с использованием специализированных языков запросов, таких как MongoDB Query Language для документ-ориентированных баз данных или Gremlin для графовых баз данных.

Свойства

Реляционные базы данных обычно обеспечивают поддержку транзакций, что позволяет обеспечить согласованность данных при изменении. Они стремятся к соблюдению ACID (Атомарность, Согласованность, Изолированность, Долговечность) для обеспечения надежности данных.

NoSQL базы данных могут обеспечивать высокую производительность для определенных видов задач, таких как чтение и запись больших объемов данных.

За/против

- MongoDB имеет встроенную поддержку для хранения бинарных данных
- Гибкую схему данных
- Версионирование
- Поддержка sdk/api
- Бэкап данных
- Это БД

- Нет дозаписи
- Сложнее масштабирование
- Это бд

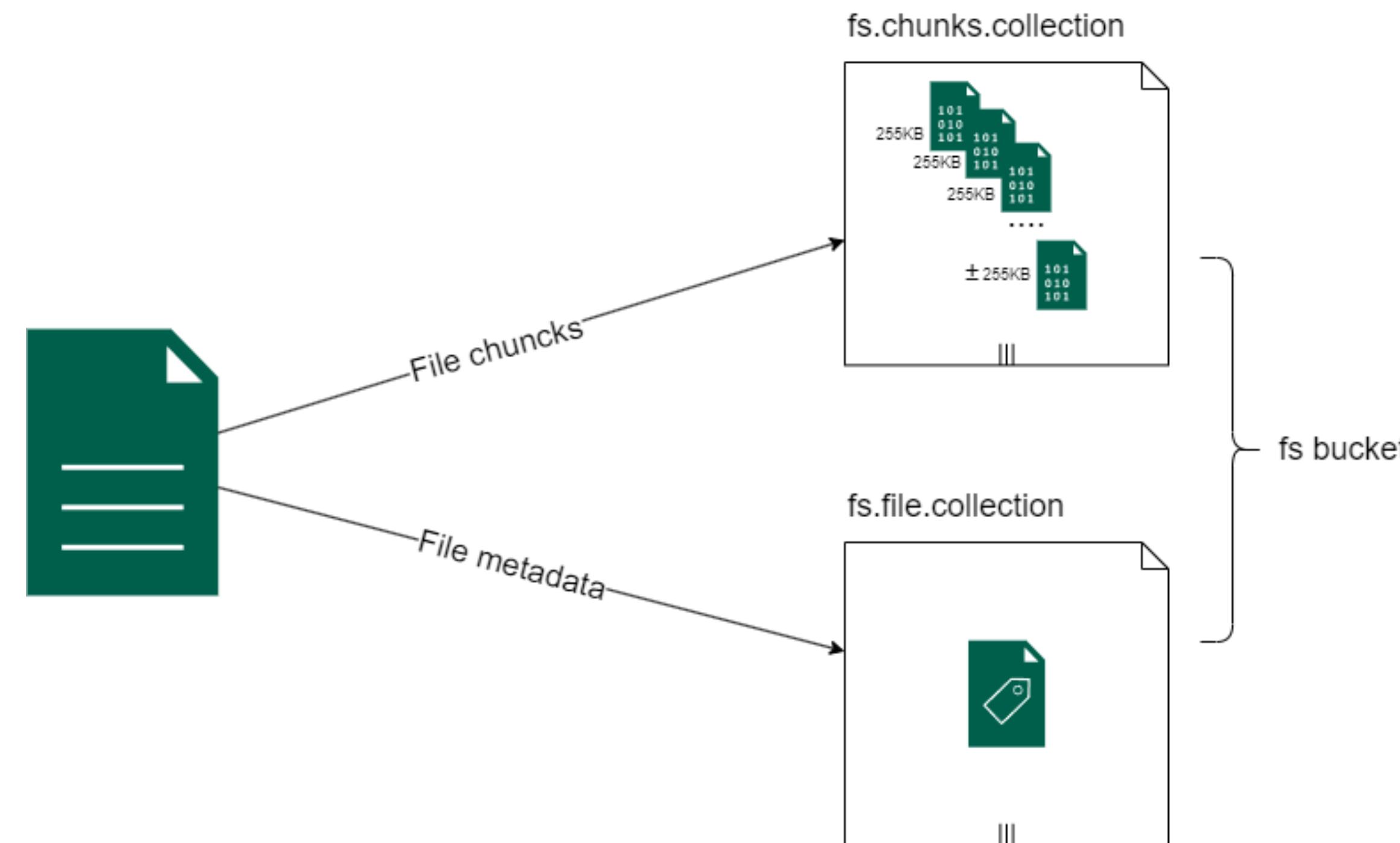


GridFS

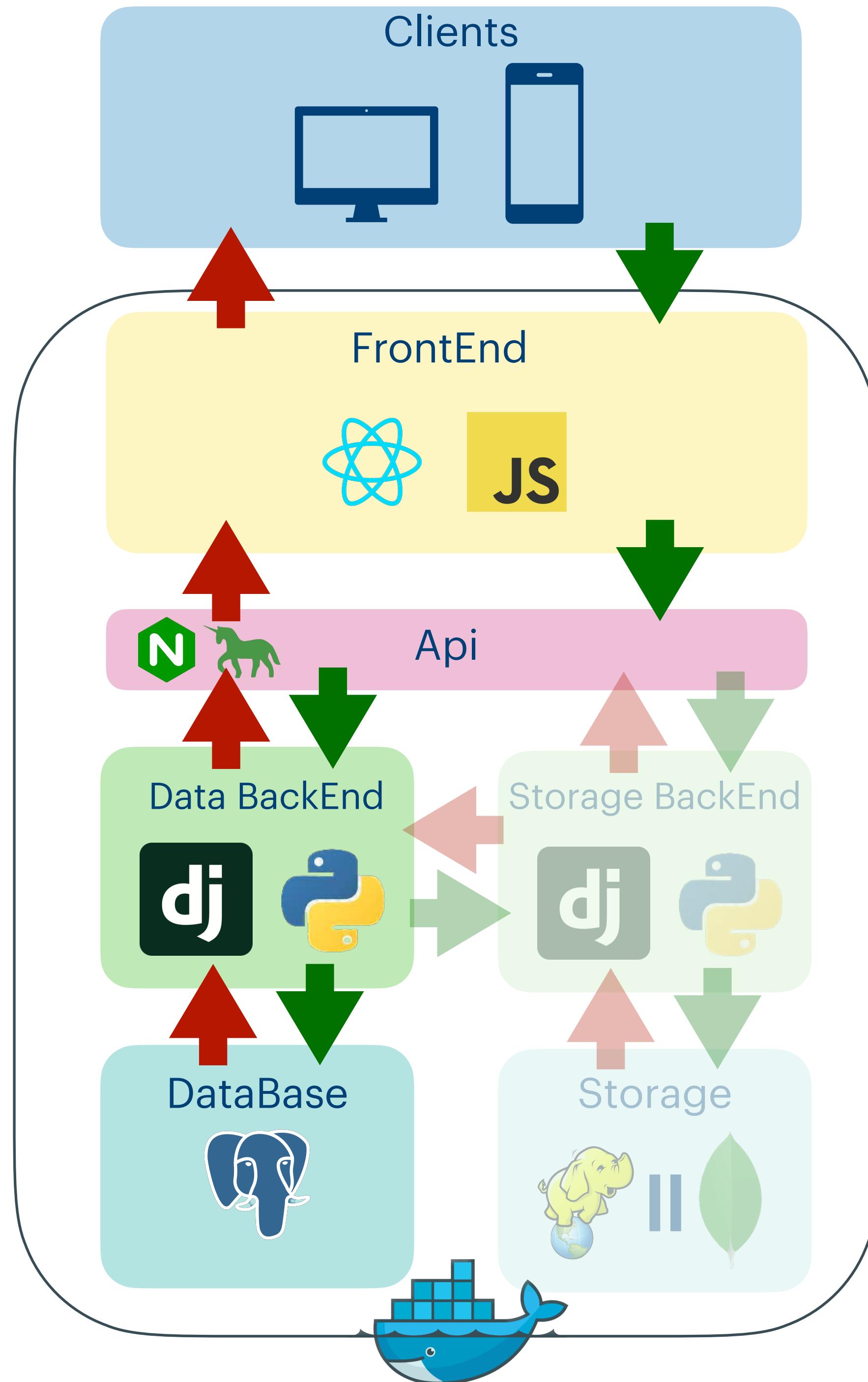
Это спецификация для файлов, размер которых превышает ограничение размера BSON в 16 мб.

Вместо хранения файла в одном документе GridFS делит файл на фрагменты и сохраняет каждый фрагмент как отдельный документ.

GridFS использует две коллекции для хранения файлов. В одной коллекции хранятся фрагменты файлов, а в другой — метаданные файлов.



Новая архитектура



Полезные ссылки

- <https://www.youtube.com/watch?v=4nnOF3geVTI>
-