

Совместное администрирование баз данных и серверов

1. Сравнительная таблица СУБД

СУБД	Преимущества	Недостатки
PostgreSQL	<ul style="list-style-type: none">- Открытый исходный код, бесплатная- Поддержка расширенных типов данных и GIS- Высокая надежность и соответствие стандартам SQL- Расширяемость (модули, расширения)- Современные механизмы репликации и отказоустойчивости	<ul style="list-style-type: none">- Меньше инструментов для администрирования по сравнению с коммерческими СУБД- Более высокая сложность настройки для новичков- Иногда медленнее MySQL на простых задачах
MySQL	<ul style="list-style-type: none">- Быстродействие на простых и средних нагрузках- Простота установки и настройки- Большое сообщество, множество руководств- Бесплатная версия (Community)	<ul style="list-style-type: none">- Ограниченная поддержка расширенных SQL-функций- Меньше возможностей для сложной аналитики- Некоторые функции доступны только в коммерческой версии (Enterprise)
Microsoft SQL Server	<ul style="list-style-type: none">- Высокая производительность и масштабируемость- Гибкая безопасность и управление доступом- Отличная интеграция с продуктами Microsoft- Развитые BI-инструменты и аналитика- Удобные графические инструменты для администрирования	<ul style="list-style-type: none">- Высокая стоимость лицензий- Требуется Windows (хотя есть версии для Linux)- Ресурсоемкость, особенно для больших внедрений- Vendor lock-in (зависимость от экосистемы Microsoft)

MariaDB	<ul style="list-style-type: none"> - Открытый исходный код, бесплатная - Высокая совместимость с MySQL - Активное развитие и поддержка сообществом - Улучшенная производительность и расширенные механизмы хранения данных 	<ul style="list-style-type: none"> - Меньше корпоративной поддержки по сравнению с MySQL - Некоторые расширения несовместимы с MySQL - Меньше инструментов для крупных BI-задач
---------	--	--

2. Анализ предметной области: Автоматизированная система обнаружения трещин в бетонных конструкциях

Краткое описание

Автоматизированная система обнаружения трещин предназначена для мониторинга состояния бетонных конструкций (мосты, здания, тоннели и др.) с целью своевременного выявления дефектов и предотвращения аварийных ситуаций. Современные решения используют компьютерное зрение, нейросети и базы данных для хранения результатов контроля.

Особенности предметной области

Типы данных: Изображения бетонных поверхностей, аннотированные данные о трещинах (координаты, ширина, длина), метаданные (дата, место, оборудование).

Объемы данных: Для обучения моделей и хранения результатов требуется хранить десятки тысяч изображений и связанных с ними метаданных.

Требования к точности: Высокая точность обнаружения и классификации трещин (до 99% по современным моделям).

Автоматизация: Минимизация человеческого фактора, сокращение времени на инспекции и обработку данных.

Интеграция с ИИ: Использование нейросетей (YOLO, CNN, Ridgelet NN) для анализа изображений и автоматической классификации дефектов.

```

CREATE TABLE Объект (
    id INT PRIMARY KEY IDENTITY,
    тип VARCHAR(100) NOT NULL,
    адрес VARCHAR(255) NOT NULL,
    дата_постройки DATE
);

CREATE TABLE Оборудование (
    id INT PRIMARY KEY IDENTITY,
    тип VARCHAR(100) NOT NULL,
    серийный_номер VARCHAR(100) UNIQUE NOT NULL
);

CREATE TABLE Пользователь (
    id INT PRIMARY KEY IDENTITY,
    имя VARCHAR(100) NOT NULL,
    роль VARCHAR(50) NOT NULL,
    логин VARCHAR(100) UNIQUE NOT NULL,
    пароль VARCHAR(100) NOT NULL
);

CREATE TABLE Обследование (
    id INT PRIMARY KEY IDENTITY,
    объект_id INT NOT NULL FOREIGN KEY REFERENCES Объект(id),
    дата DATETIME NOT NULL,
    оператор_id INT NOT NULL FOREIGN KEY REFERENCES Пользователь(id),
    оборудование_id INT NOT NULL FOREIGN KEY REFERENCES Оборудование(id)
);

CREATE TABLE Изображение (
    id INT PRIMARY KEY IDENTITY,
    обследование_id INT NOT NULL FOREIGN KEY REFERENCES Обследование(id),
    путь_к_файлу VARCHAR(255) NOT NULL,
    дата DATETIME NOT NULL
);

CREATE TABLE Трещина (
    id INT PRIMARY KEY IDENTITY,
    изображение_id INT NOT NULL FOREIGN KEY REFERENCES Изображение(id),
    координаты VARCHAR(255) NOT NULL,
    ширина FLOAT CHECK (ширина > 0),
    длина FLOAT CHECK (длина > 0),
    тип VARCHAR(50),
    вероятность FLOAT CHECK (вероятность >= 0 AND вероятность <= 1)
);

```

Безопасн

ость:

Необходимость защищать данные от несанкционированного доступа и потерь.

3. Техническое задание (ТЗ) на разработку БД для автоматизированной системы обнаружения трещин в бетонных конструкциях

3.1. Описание целей и задач БД

а.Цель: Создать централизованную базу данных для хранения, обработки и анализа результатов автоматического контроля трещин в бетонных конструкциях.

Задачи:

- Хранение изображений и результатов анализа трещин.
- Ведение истории обследований и дефектов.
- Предоставление данных для аналитики и отчетности.
- Обеспечение безопасности и резервного копирования информации.

б. Требования к функциональности

- Загрузка и хранение изображений обследуемых объектов.
- Сохранение результатов автоматического анализа (координаты, размеры трещин, классификация).
- Ведение справочников объектов и оборудования.
- Формирование отчетов по объектам, дефектам, динамике состояния.
- Управление доступом пользователей (администратор, оператор, аналитик, читатель).
- Интеграция с внешними системами (например, системой визуального анализа).

с. Требования к данным

- Хранимые данные: Изображения, параметры трещин (координаты, ширина, длина, тип), информация об объекте (тип, адрес, дата постройки), результаты анализа, пользователи, оборудование.
- Объемы: Ожидается хранение десятков тысяч изображений и связанных с ними записей.
- Источники данных: Автоматизированные системы контроля, ручной ввод, внешние системы.

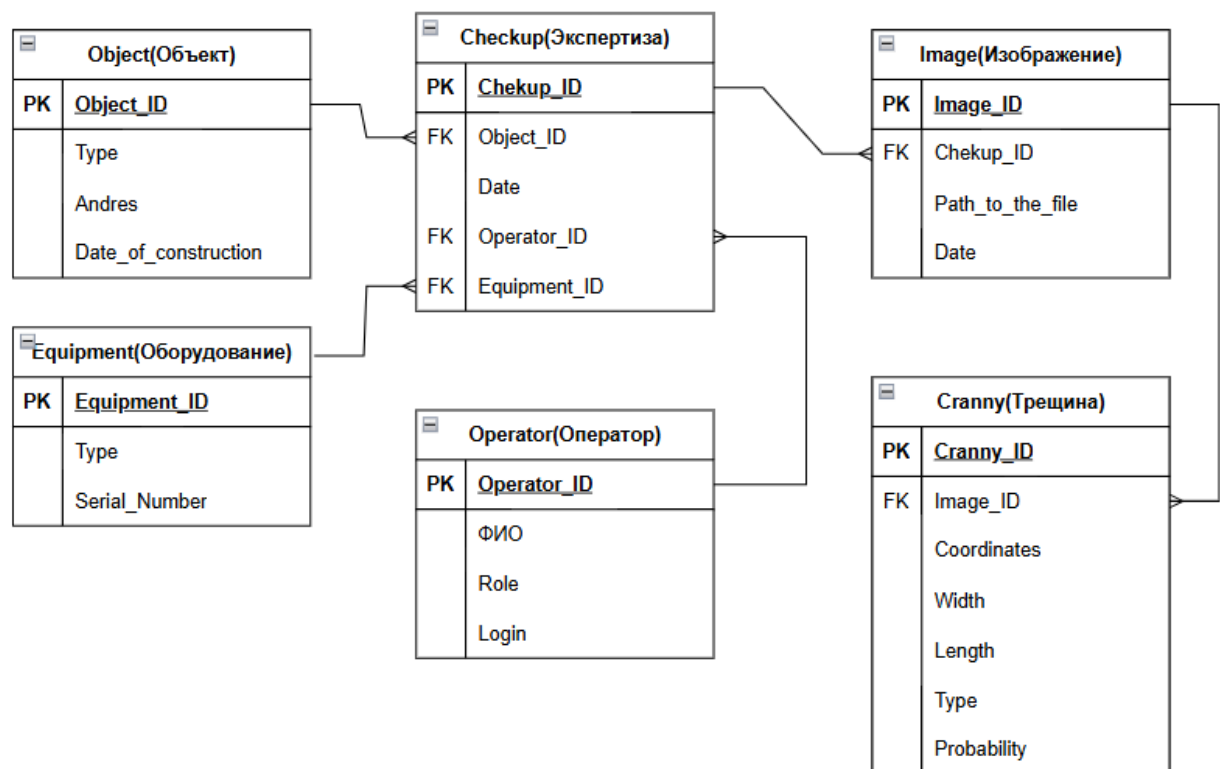
д. Ограничения

- Производительность: Быстрая обработка запросов и загрузка изображений.
- Безопасность: Шифрование данных, разграничение прав доступа.
- Масштабируемость: Возможность расширения объема хранимых данных и числа пользователей.

е. Сроки и этапы разработки

- Анализ требований и проектирование: 2 дня
- Разработка концептуальной и логической моделей: 2 дня
- Реализация физической модели и тестирование: 3 дня
- Внедрение и обучение пользователей: 1 день

3.2. Концептуальная модель (ER-диаграмма)



3.3. Логическая модель данных:

Таблица	Поля (атрибуты)	Тип ключа	Описание поля	Связи (внешние ключи)
Объект	id (int)	РК (первичный ключ)	Уникальный идентификатор объекта	—
	тип (varchar)		Тип бетонной конструкции	—
	адрес (varchar)		Адрес расположения объекта	—
	дата_построй ки (date)		Дата постройки	—
Обследование	id (int)	РК	Уникальный идентификатор обследования	—
	объект_id (int)	FK	Ссылка на объект (Объект.id)	1 Объект — М Обследование
	дата (datetime)		Дата проведения обследования	—
	оператор_id (int)	FK	Ссылка на пользователя (Пользователь.id)	1 Пользователь — М Обследование
	оборудование _id (int)	FK	Ссылка на оборудование (Оборудование.id)	1 Оборудование — М Обследование
Изображение	id (int)	РК	Уникальный идентификатор изображения	—
	обследование _id (int)	FK	Ссылка на обследование (Обследование.id)	1 Обследование — М Изображение
	путь_к_файлу (varchar)		Путь или ссылка на файл изображения	—

	дата (datetime)		Дата съемки изображения	—
Трещина	id (int)	PK	Уникальный идентификатор трещины	—
	изображение_ id (int)	FK	Ссылка на изображение (Изображение.id)	1 Изображение — M Трещина
	координаты (varchar)		Координаты трещины на изображении	—
	ширина (float)		Ширина трещины	—
	длина (float)		Длина трещины	—
	тип (varchar)		Тип трещины (например, поверхностная, глубокая)	—
	вероятность (float)		Вероятность корректного определения трещины (0..1)	—
Оборудование	id (int)	PK	Уникальный идентификатор оборудования	—
	тип (varchar)		Тип оборудования	—
	серийный_но мер (varchar)	UNIQUE	Серийный номер оборудования	—
Пользователь	id (int)	PK	Уникальный идентификатор пользователя	—
	имя (varchar)		Имя пользователя	—

	роль (varchar)		Роль пользователя (админ, оператор, аналитик)	—
	логин (varchar)	UNIQUE	Логин для входа	—
	пароль (varchar)		Хэшированный пароль	—

3.4. Физическая модель (пример для Microsoft SQL Server)

PP07

Database Diagrams

Tables

System Tables

FileTables

External Tables

Graph Tables

dbo.Изображение

dbo.Оборудование

dbo.Обследование

dbo.Объект

dbo.Пользователь

dbo.Трещина

Dropped Ledger Tables

```
CREATE TABLE Объект (
    id INT PRIMARY KEY IDENTITY,
    тип VARCHAR(100) NOT NULL,
    адрес VARCHAR(255) NOT NULL,
    дата_постройки DATE
);

CREATE TABLE Оборудование (
    id INT PRIMARY KEY IDENTITY,
    тип VARCHAR(100) NOT NULL,
    серийный_номер VARCHAR(100) UNIQUE NOT NULL
);

CREATE TABLE Пользователь (
    id INT PRIMARY KEY IDENTITY,
    имя VARCHAR(100) NOT NULL,
    роль VARCHAR(50) NOT NULL,
    логин VARCHAR(100) UNIQUE NOT NULL,
    пароль VARCHAR(100) NOT NULL
);

CREATE TABLE Обследование (
    id INT PRIMARY KEY IDENTITY,
    объект_id INT NOT NULL FOREIGN KEY REFERENCES Объект(id),
    дата DATETIME NOT NULL,
    оператор_id INT NOT NULL FOREIGN KEY REFERENCES Пользователь(id),
    оборудование_id INT NOT NULL FOREIGN KEY REFERENCES Оборудование(id)
);

CREATE TABLE Изображение (
    id INT PRIMARY KEY IDENTITY,
    обследование_id INT NOT NULL FOREIGN KEY REFERENCES Обследование(id),
    путь_к_файлу VARCHAR(255) NOT NULL,
    дата DATETIME NOT NULL
);

CREATE TABLE Трещина (
    id INT PRIMARY KEY IDENTITY,
    изображение_id INT NOT NULL FOREIGN KEY REFERENCES Изображение(id),
    координаты VARCHAR(255) NOT NULL,
    ширина FLOAT CHECK (ширина > 0),
    длина FLOAT CHECK (длина > 0),
    тип VARCHAR(50),
    вероятность FLOAT CHECK (вероятность >= 0 AND вероятность <= 1)
);
```

3.5. Бизнес-правила и ограничения

- Целостность данных: Все FK с ON DELETE CASCADE для автоматического удаления зависимых записей.
- CHECK: Ограничения на диапазоны числовых полей.
- UNIQUE: На логины пользователей и серийные номера оборудования.
- NOT NULL: Для обязательных полей.
- Триггеры/процедуры: Например, автоматическое создание записи о дефекте при загрузке нового изображения.

3.6. Регламент доступа и безопасности

- Роли: Администратор (полный доступ), Оператор (добавление данных), Аналитик (только чтение и отчеты), Читатель (только просмотр).
- Права доступа: Использование GRANT/REVOKE для разграничения доступа.
- Шифрование: Хранение паролей в зашифрованном виде.
- Резервное копирование: Автоматизированные ежедневные бэкапы.

3.7. Руководство администратора

- Резервное копирование: Использовать встроенные средства SQL Server для создания ежедневных бэкапов.
- Мониторинг: Настроить оповещения о сбоях, отслеживать логи ошибок.
- Восстановление: Инструкция по восстановлению из резервной копии.
- Устранение сбоев: Проверка целостности БД, восстановление индексов, анализ журналов транзакций.

3.8. Словарь данных

Таблица	Поле	Описание
Объект	id	Уникальный идентификатор объекта
	тип	Тип конструкции
	адрес	Адрес расположения
	дата_постройки	Дата постройки
Оборудование	id	Уникальный идентификатор оборудования
	тип	Тип оборудования
	серийный_номер	Серийный номер
Пользователь	id	Уникальный идентификатор пользователя
	имя	Имя пользователя
	роль	Роль пользователя
	логин	Логин
	пароль	Пароль (зашифрованный)
Обследование	id	Уникальный идентификатор обследования
	объект_id	Ссылка на объект
	дата	Дата обследования

	оператор_id	Ссылка на пользователя
	оборудование_id	Ссылка на оборудование
Изображение	id	Уникальный идентификатор изображения
	обследование_id	Ссылка на обследование
	путь_к_файлу	Путь к файлу изображения
	дата	Дата съемки
Трещина	id	Уникальный идентификатор трещины
	изображение_id	Ссылка на изображение
	координаты	Координаты трещины на изображении
	ширина	Ширина трещины
	длина	Длина трещины
	тип	Тип трещины
	вероятность	Вероятность корректного определения