

ДЕПАРТАМЕНТ ОБРАЗОВАНИЯ И НАУКИ ГОРОДА МОСКВЫ

Государственное автономное образовательное учреждение

Высшего образования города Москвы

«Московский городской педагогический университет»

(ГАОУ ВО МГПУ)

Институт цифрового образования

Лабораторная работа №3

«Проектирование архитектуры хранилища больших
данных для автомобильной компании (Connected Cars) »

По дисциплине «Инструменты для хранения и обработки больших
данных»

Выполнила: Татаринова Екатерина Михайловна

Группа: АДЭУ-221

Преподаватель: доцент Босенко Т.М.

Москва 2025

Задача: создать архитектуру хранилища больших данных для автомобильной компании

Цель: обеспечить сбор и анализ телеметрии с подключенных автомобилей, разработка систем помощи водителю (ADAS), управление обновлениями "по воздуху"

Определение требований

Источники данных и типы:

<u>Структурированные данные</u>	<u>Полуструктурированные данные</u>	<u>Неструктурированные данные</u>
CAN-шина: диагностические коды, статус двигателя, температура, расход топлива, скорость, давление в шинах	Телеметрия в реальном времени с частотой генерации до нескольких килобайт в секунду на один автомобиль	Видео с камер (мультиспектральное видео, тепловизионные потоки)
GPS: координаты, время, высота над уровнем моря	События о состоянии автомобиля (ошибки, предупреждения)	Облачные данные с лидаров (LiDAR point clouds)
OBD-II интерфейс: стандартизированная диагностическая информация	Данные производительности двигателя	Изображения с радаров

Объемы и скорость поступления данных

Для парка из 10 000 подключенных автомобилей с интервалом отправки данных 5-10 секунд:

- Ежесекундный объем: 50-100 Мб (при средней нагрузке)
- Ежедневный объем: 4.3-8.6 Тб

- Пиковые нагрузки при критических событиях (ДТП, отказы систем)

Бизнес-цели:

- Аналитика в реальном времени с задержкой до нескольких сотен миллисекунд для систем безопасности и помощи водителю
- Batch-аналитика для отчетов и предсказательных моделей
- Запуск и интеграция моделей машинного обучения для предсказания состояния автомобиля и поведения на дороге

Компоненты архитектуры

- Слой сбора данных (Ingestion): Apache Kafka
 - Apache Kafka обеспечивает масштабируемость на уровне петабайт в день
 - Поддержка репликации данных из граничных (edge) кластеров в облако через Cluster Linking
 - Встроенная поддержка транзакций и гарантия доставки сообщений
- Слой хранения (Storage): Amazon S3 / Google Cloud Storage:
 - Превосходная производительность при реальной обработке потоков данных (streaming)
 - Встроенная поддержка неструктурированных данных (видео, облака точек)
 - Интеграция с Databricks, которая предоставляет управляемую платформу
- Слой обработки (Processing): Apache Spark, Apache Flink

Flink:

- Агрегирование данных датчиков в окнах 5 секунд
- Обнаружение аномалий в реальном времени (резкое торможение, превышение температуры)

- Трансляция событий в Azure Event Hubs для соединенных систем
- Предикция отказов с использованием встроенного взаимодействия с моделями ML

Spark используется для глубокого анализа исторических данных:

- Трансформация данных от raw к silver и gold слоям
- Вычисление агрегированных статистик по парку
- Обучение и валидация ML моделей
- Слой аналитики и визуализации (Analytics & Visualization): Tableau и Grafana.

Tableau для стратегического анализа

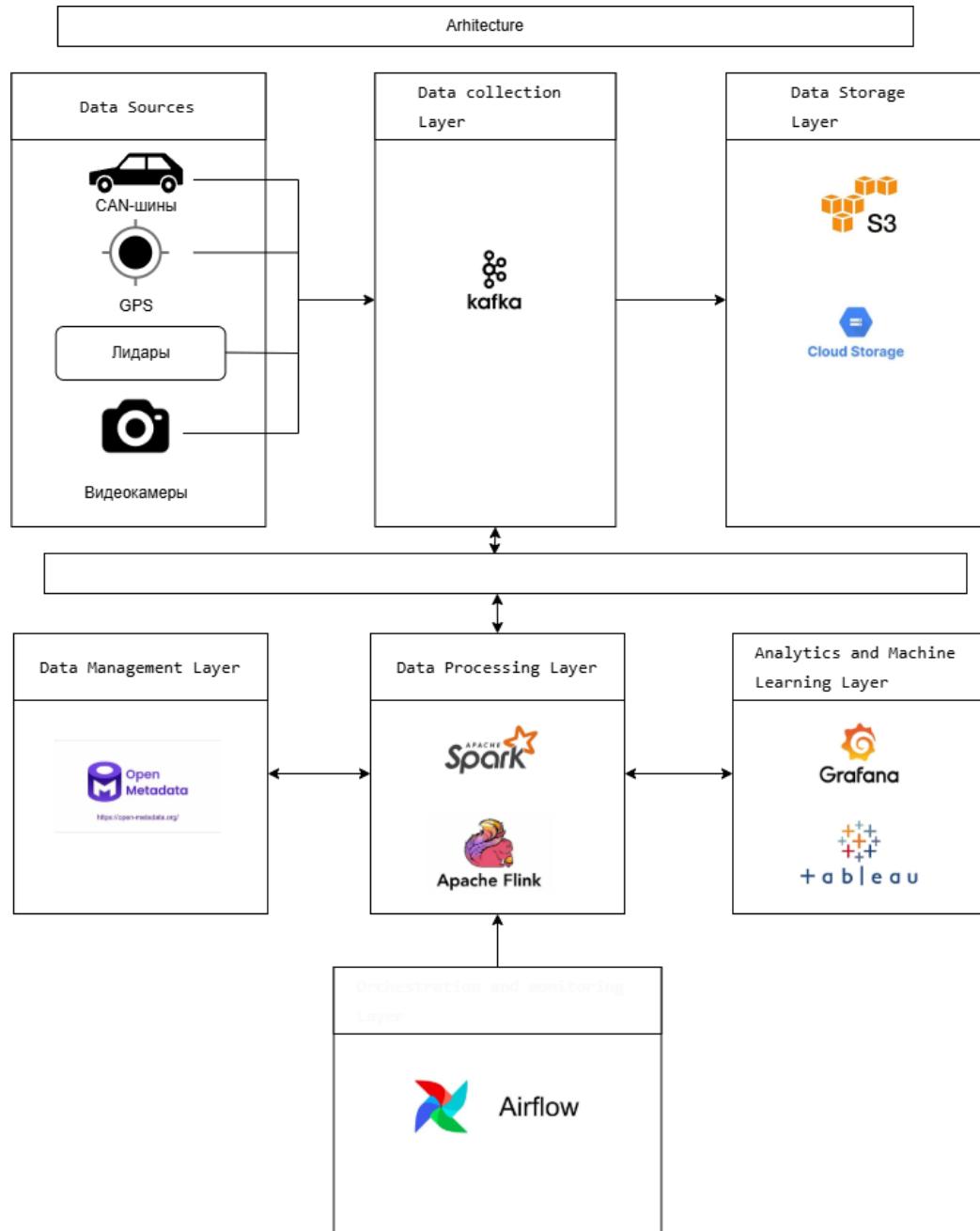
- Создание интерактивных дашбордов для аналитиков и руководства
- Визуализация глобального состояния автопарка
- Поддержка сложных расчетов и параметров

Grafana для операционного мониторинга

- Система показателей в реальном времени для команды операций
- Интеграция с Prometheus для сбора метрик системы
- Пороги оповещений о критических состояниях автомобилей
- Слой оркестрации (Orchestration): Apache Airflow
 - Зрелость экосистемы и широкое распространение в индустрии
 - Встроенная поддержка KubernetesExecutor для масштабирования на тысячи DAG-ов
 - Интеграция с Spark, Flink и облачными сервисами AWS
 - Возможность управления сложными зависимостями между пайплайнами обработки
- Управление данными (Data Governance): OpenMetadata.

- Каталогизация всех таблиц и данных в системе
- Отслеживание родословной данных (lineage) для целей аудита
- Управление качеством данных и документирование схем

Создание диаграммы архитектуры



Потоки данных

Данные с CAN-шины, GPS, лидаров и камер передаются с автомобилей через MQTT по защищенным каналам в AWS IoT Core.

Далее данные направляются в Apache Kafka, где осуществляется масштабируемое хранение и передача сообщений для потребителей.

Потоковая обработка с Apache Flink анализирует данные для обнаружения аномалий, принятия решений ADAS и обогащения потоков.

Одновременно данные сохраняются в Data Lake на базе S3 + Delta Lake с многоуровневой структурой хранения (raw, bronze, silver, gold).

Батч-обработка с Apache Spark очищает и агрегирует данные для последующего анализа и машинного обучения.

Для аналитики используются Google BigQuery и ClickHouse, откуда данные визуализируются в Tableau и Grafana.

Оркестрация всего процесса — Apache Airflow, а мониторинг — Prometheus с визуализацией в Grafana.

Производительность и масштабируемость

Масштабирование Kafka достигается увеличением числа партиций и кластеров.

Flink масштабируется за счет увеличения параллелизма задач обработки.

Использование edge-вычислений на устройствах снижает нагрузку на облачное хранилище.

Репликация и multi-region развертывания для отказоустойчивости.

Горизонтальное масштабирование Airflow через KubernetesExecutor.

Безопасность

Рост стоимости хранения видеоданных решается адаптивным сжатием, tiering-стратегией и выборочным сохранением.

Качество данных обеспечивается валидацией на стороне Flink, использованием quarantine-слоя и data contracts.

Управление сложностью системы решается с помощью Data Mesh подхода, централизованного каталога OpenMetadata и CI/CD для пайплайнов.

Вывод

Предложенная архитектура эффективно решает задачи обработки больших объемов разнообразных данных Connected Cars с акцентом на низкую задержку, масштабируемость и надежность. Использование Data Lakehouse сочетает гибкость и производительность, а современный технологический стек гарантирует готовность платформы к росту числа автомобилей и объема данных. Внимание к мониторингу, оркестрации и управлению данными обеспечивает стабильную и прозрачную эксплуатацию системы.