Департамент образования города Москвы

Государственное автономное образовательное учреждение высшего образования города Москвы

«Московский городской педагогический университет»

Институт цифрового образования

Департамент информатики, управления и технологий

Инструменты для хранения и обработки больших данных

Лабораторная работа 3.1

Проектирование архитектуры хранилища больших данных

Выполнила: студентка группы АДЭУ-221

Пришлецова Кристина Сергеевна

Проверил:

доцент департамента информатики, управления и технологий

Босенко Тимур Муртазович

Москва

2025

Вариант 11.

Задача: создать архитектуру хранилища больших данных для компании, занимающейся анализом потребительского поведения клиентов крупной сети отелей.

Цель: обеспечить надежное хранение, эффективную обработку и анализ больших объемов данных, получаемых из различных источников, таких как системы бронирования, данные о гостях (CRM), отзывы с онлайн-платформ.

Определение требований

- **Объем данных:** Ожидаемый объем данных, которые будут храниться.
- **Скорость получения данных:** как часто данные будут поступать в хранилище.
- **Типы данных:** какие типы данных будут храниться (структурированные, неструктурированные, полуструктурированные).
- **Требования к обработке:** как данные будут использоваться (аналитика, машинное обучение, отчетность).
- **Доступность данных:** требования к доступности и времени отклика.
- **Безопасность данных:** требования к защите данных от несанкционированного доступа.

1. Требования к данным для крупной компании в России

1. **Объем данных:**
   - Ожидаемый объем: 150–250 ТБ в год.
   Учитывая большое количество филиалов, клиентов и отзывов с внешних источников).
   - Рост: 40–60% ежегодно.
   (из-за расширения сети, роста клиентской базы и увеличения числа онлайн-бронирования).
2. **Скорость получения данных:**
   - Системы онлайн-бронирования: данные поступают в режиме реального времени, до 3000 событий в секунду (новые бронирования, отмены, изменения).
   - CRM и данные о гостях: ежедневное обновление, синхронизация профилей и историй пребывания.
   - Отзывы с онлайн-платформ (Оттело, Т-Путешествия, Суточно.ру, Booking, соцсети): обновления каждые 15–30 минут.

3. **Типы данных:**

- Структурированные (около 30%)

- Данные о бронированиях, транзакциях, загрузке номеров;
- Тарифные планы, скидки, сезонные коэффициенты.

- Полуструктурированные (около 45%)

- Данные CRM: история проживания, предпочтения гостей, статистика лояльности;
- Логи систем бронирования (в формате JSON);
- Данные API от онлайн-платформ бронирования (XML, JSON);
- Данные IoT-сенсоров (например, датчики присутствия, энергопотребления).

- Неструктурированные (около 25%)

- Текстовые отзывы и комментарии гостей;
- изображения номеров и объектов инфраструктуры;
- публикации из соцсетей.

4. **Требования к обработке:**

- Динамическое ценообразование: расчет и обновление цен в реальном времени на основе спроса, загрузки и внешних факторов (праздники, сезон, события).
- Прогнозирование загрузки: ежедневное и еженедельное обновление моделей, основанных на исторических данных и сезонности.
- Сегментация клиентов: ежемесячно, с учетом новых данных о бронированиях и отзывах.
- Персонализированные предложения: в реальном времени, при взаимодействии клиента с системой бронирования или онлайн-платформой.
- Аналитические отчеты для менеджмента: ежедневно (оперативные) и ежемесячно (стратегические).

5. **Доступность данных:**

- Время отклика для аналитических запросов: 10–30 секунд.
- Доступность системы: 99.95–99.99% (допустимое время простоя – не более 4.4 часов в год.)
- Резервное копирование: каждые 5 часов с хранением копий в отдельном дата-центре.

6. **Безопасность данных**

- Шифрование:
  - данных в состоянии покоя;
  - данных при передаче.

- Доступ:
  - многофакторная аутентификация для сотрудников;
  - разграничение прав доступа по ролям (администратор, аналитик, менеджер).

- Мониторинг и аудит:
  - автоматическое логирование всех операций с персональными данными;
  - ежемесячные проверки безопасности.

- Соответствие требованиям:
  - 152-ФЗ “О персональных данных” (для гостей из РФ);
  - GDPR (для иностранных гостей);
  - Внутренние стандарты информационной безопасности сети отелей.

Выбор модели хранилища данных

- **Data Lake:** хранение необработанных данных в едином репозитории.
- **Data Warehouse:** хранение структурированных данных, оптимизированных для аналитики.
- **Hybrid Data Storage:** сочетание Data Lake и Data Warehouse.

2. Архитектура хранилища больших данных

- **Источники данных:**
  - Системы бронирования (внешние API);
  - CRM-системы (данные о гостях);
  - Онлайн-платформы (отзывы);
  - API платежных систем.
- **Слой сборки данных (Ingestion):**
  - Apache Kafka (или Redpanda, ее современный аналог) – для основного стриминга (потоковые данные)
  - Debezium для CDC (из PMS/CRM)
  - Airbyte для API/batch коннекторов
  - Vector на фронте для логов

- **Слой хранения (Storage):**
  - Yandex Object Storage – облачное хранилище архивной информации и неструктурированных медиа данных
  - Delta Lake - эффективный open-source формат хранения данных поверх Yandex Object Storage, позволяющий эффективно управлять большими наборами данных, обеспечивать консистентность и атомарность изменений, а также значительно ускорять чтение и запись данных.
  - ClickHouse – отечественная СУБД, которая работает со структурированными данными и обладает высокой скоростью выполнения аналитических запросов.
- **Слой обработки (Processing):**
  - Apache Spark – для пакетной обработки
  - Apache Flink – для потоковой обработки в реальном времени
- **Слой аналитики и визуализации (Analytics & Visualization):**
  - JupyterLab - это усовершенствованная версия Jupyter Notebook, в которой можно выполнять очистку данных, предобработку, визуализацию и построение моделей
  - Yandex DataLens для построения дашбордов и отчетов
- **Слой оркестрации (Orchestration):**
  - Apache Airflow - платформа для автоматизации и оркестрации задач обработки данных и обеспечения систематичного и контролируемого процесс выполнения задач (jobs) в определенной последовательности, образующей рабочий процесс (workflow).
- **Управление данными (Data Management):**
  - OpenMetadata - универсальная открытая платформа для управления метаданными в организациях.

3. Проектирование архитектуры

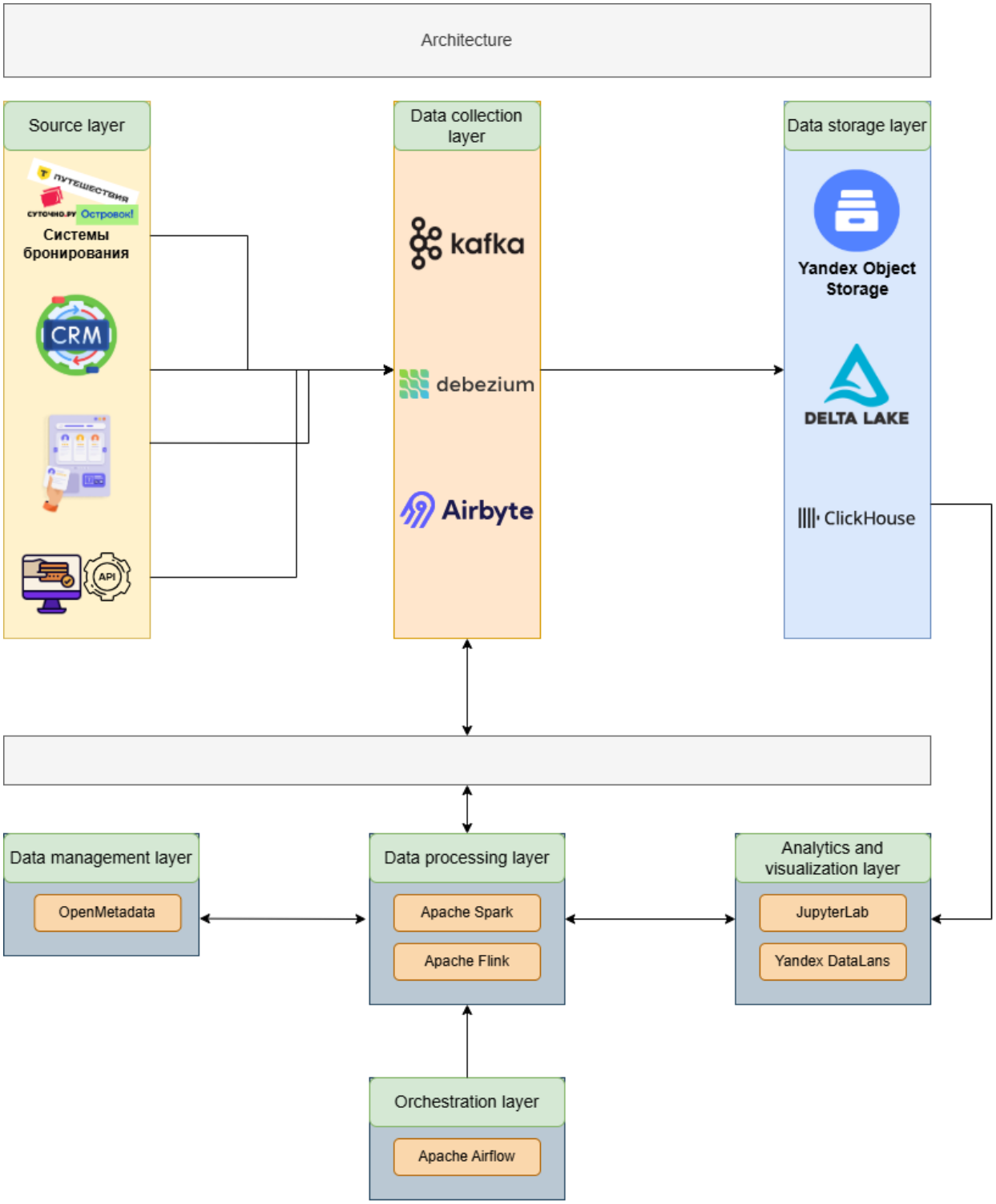

Рисунок 1 Схема архитектуры системы

4. Описание компонентов и обоснование выбора

1. Слой сборки данных (Ingestion)

Apache Kafka (или Redpanda) Роль: Основный брокер сообщений для обработки потоковых данных. Обоснование выбора: Apache Kafka обеспечивает низкую задержку и высокую пропускную способность при обработке больших объемов потоковых данных, таких как события бронирования, изменения в CRM и отзывы. Он отлично сочетается с остальной частью инфраструктуры, позволяя эффективно передавать данные в хранилище.

Debezium Роль: Механизм Change Data Capture (CDC) для инкрементального захвата изменений в системах CRM и PMS. Обоснование выбора: Debezium позволяет аккуратно отслеживать изменения в системах регистрации гостей и бронирования, что критично для построения актуальной картины данных.

Airbyte Роль: Средство для интеграции с внешними API и batch-коннекторами. Обоснование выбора: Airbyte упрощает подключение и синхронизацию данных из внешних источников (например, платежных систем и онлайн-платформ), что позволяет собрать все данные в одном месте.

Vector Роль: Агрегатор и отправитель логов. Обоснование выбора: Используется для сбора и передачи логов и метрик, обеспечивая единый фронт обработки данных и удобство мониторинга.

2. Слой хранения данных (Storage)

Yandex Object Storage Роль: Облачное хранилище для неструктурированной информации и архивных данных. Обоснование выбора: Это решение обеспечивает недорогую и масштабируемую платформу для хранения медиаконтента и иных архивных данных, обеспечивая необходимую доступность и эластичность.

Delta Lake Роль: Open-source формат хранения данных поверх Yandex Object Storage, обеспечивающий консистентность и атомарность изменений. Обоснование выбора: Delta Lake оптимален для больших наборов данных, обеспечивает эффективную работу с аналитическими запросами и хорошую интеграцию с Apache Spark.

ClickHouse Роль: Столбчатая аналитическая СУБД для выполнения быстрых аналитических запросов. Обоснование выбора: Высокая производительность и низкие задержки делают ClickHouse отличным выбором для аналитики в

реальном времени и создания отчетов, что критично для крупных сетей отелей.

3. Слой обработки данных (Processing)

Apache Spark Роль: Инструмент для пакетной обработки данных. Обоснование выбора: Apache Spark позволяет эффективно обрабатывать большие объемы данных, что идеально подходит для аналитических задач и подготовке данных для машинного обучения.

Apache Flink Роль: Инструмент для потоковой обработки данных в реальном времени. Обоснование выбора: Flink обеспечивает ультранизкую задержку и высокую производительность при обработке потоковых данных, что критично для расчетов цен и персонификации предложений.

4. Слой аналитики и визуализации (Analytics & Visualization)

JupyterLab Роль: Платформа для проведения исследований данных, разработки моделей и предобработки данных. Обоснование выбора: JupyterLab предоставляет удобную среду для работы с Python и другими языками, что делает его важным элементом в рабочем процессе аналитики и исследований.

Yandex DataLens Роль: Инструмент для визуализации данных и построения дашбордов. Обоснование выбора: DataLens интегрируется с ClickHouse и позволяет быстро создавать интерактивные отчёты и дашборды, что важно для оперативного принятия решений.

5. Слой оркестрации (Orchestration)

Apache Airflow Роль: Система для автоматизации и оркестрации задач обработки данных. Обоснование выбора: Airflow позволяет автоматизировать и координировать задачи обработки данных, обеспечивая выполнение рабочих процессов (workflows) в правильной последовательности и нужный момент времени.

6. Управление данными (Data Management)

OpenMetadata Роль: Открытой платформа для управления метаданными и обеспечения прозрачности данных. Обоснование выбора: OpenMetadata предоставляет инструменты для управления метаданными, классификацией и инвентаризацией данных, что способствует улучшению контроля и управлению данными в компании.

5. Анализ потенциальных проблем и их решений

1. Проблема: Рост стоимости хранения данных

По мере роста бизнеса и увеличения объема собираемых данных, расходы на хранение могут существенно возрасти. Особенно остро этот вопрос встает при увеличении количества отзывов, изображений и видеоматериалов, накапливаемых системой.

Пути решения:

Tiered storage approach: Внедрение многоуровневого подхода к хранению данных. Например, горячие данные (часто используемые) можно размещать на более дорогих скоростных уровнях (SSD-диски), а холодные данные (редко используемые) — на дешевых устройствах (HDD или объекты в S3). Это поможет уменьшить общие расходы на хранение.

Архивация данных: Периодически перемещать старые данные в архивные хранилища с низкими тарифами, например, в холодный tier Yandex Object Storage.

Удаление ненужных данных: Регулярно удалять данные, потерявшие актуальность или ценность для бизнеса. Например, удалить файлы отзывов старше определенного срока, если они не несут значимой информации.

2. Проблема: Качество данных и их консистентность

В процессе обработки данных из различных источников (CRM, бронь, оплата, отзывы) возникают риски появления некачественных данных, что негативно скажется на результатах аналитики и принятии решений.

Пути решения:

Создание ETL-процессов:Реализация серии процессов для очищения и нормализации данных перед их попаданием в хранилище. Используйте инструменты вроде Apache Spark или Apache Flink для фильтрации плохих данных и приведения их к нужному виду.

Настройка мониторинга качества данных:Постоянный мониторинг здоровья данных с помощью OpenMetadata и ClickHouse, чтобы своевременно обнаружить отклонения и исправить их.

Проведение regular audits:Регулярная проверка данных на предмет правильности и полноты, устранение пробелов и противоречий.