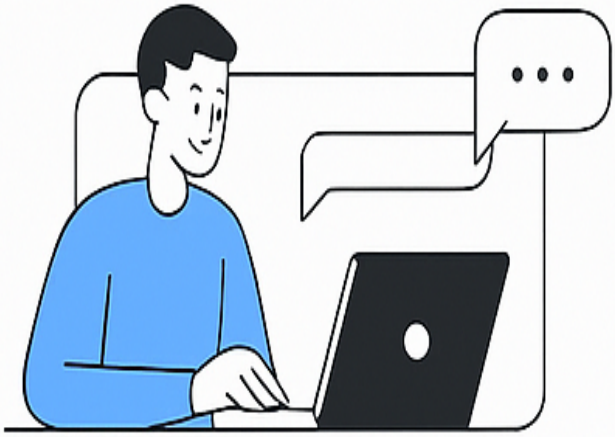
Зачем это нужно

Большой объём данных в Hadoop требует знания Spark/SQL для анализа.
Аналитикам/бизнес-пользователям сложно взаимодействовать с данными напрямую



Архитектура MVP / Как работает агент

1. Пользователь вводит запрос на естественном языке
2. LLM получает схему таблицы и формирует PySpark-код
3. Код исполняется через Spark
4. Пользователю возвращается результат с пояснением

Пример запроса и ответа

Покажи среднюю сумму транзакций по странам за последние 3 месяца

Зачем это нужно?

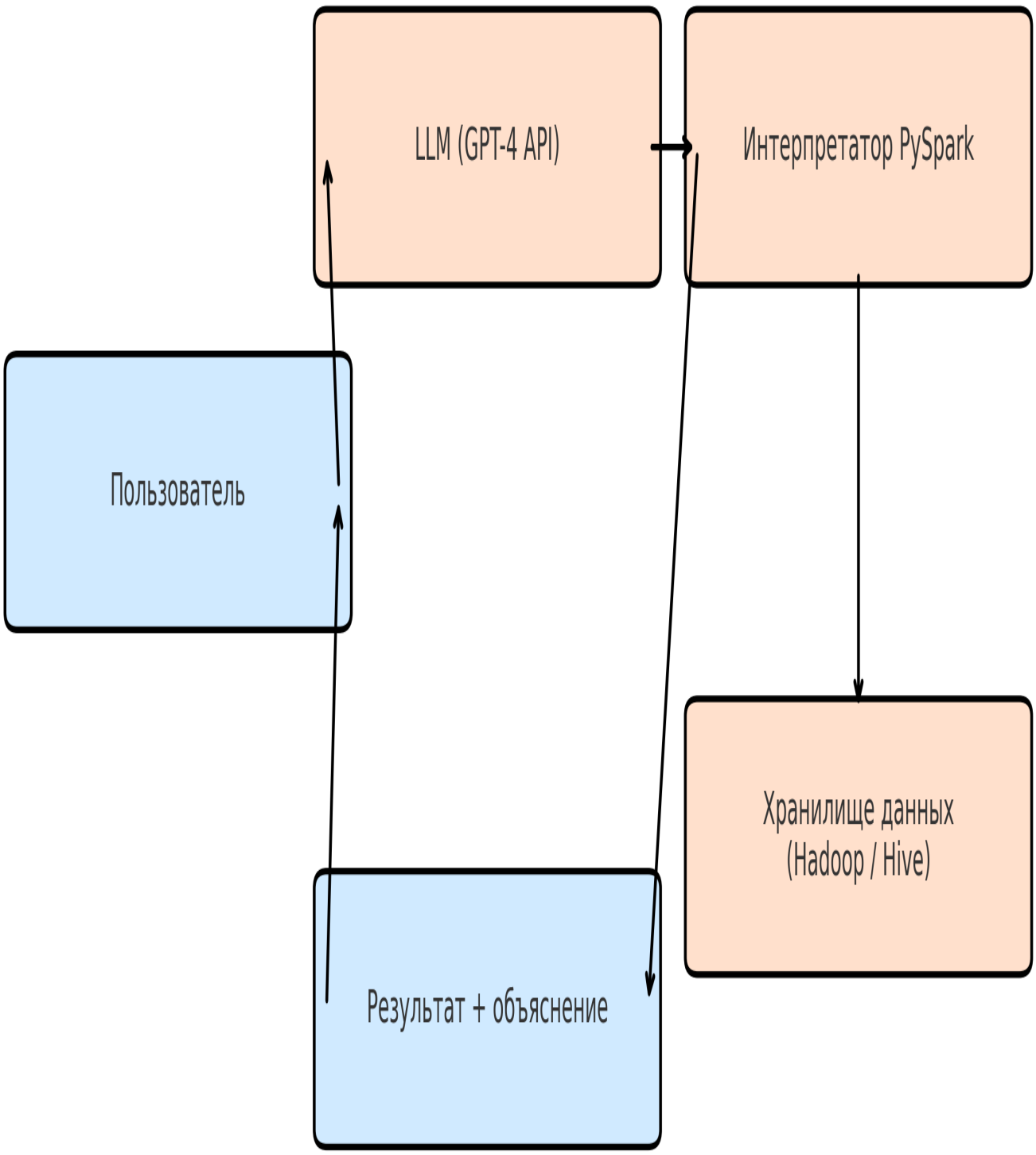
Проблема:
- Большой объём данных в Hadoop требует знания Spark/SQL для анализа.
- Аналитикам/бизнес-пользователям сложно взаимодействовать с данными напрямую

Решение:
- LLM-агент, который по запросу на естественном языке:
  - Понимает задачу,
  - Генерирует PySpark-код,
  - Выполняет его,
  - Возвращает результат с объяснением.

□ Архитектура MVP

[Пользователь] → [LLM (OpenAI GPT-4)] → [PySpark Executor] ↔ [Hadoop/Hive (данные)]
→ [Ответ + Объяснение]

□ Интеграции:
- OpenAI API,
- SparkSession,
- HDFS/Hive через spark.read


LLM (GPT-4 API)
Интерпретатор PySpark
Пользователь
Хранилище данных
(Hadoop / Hive)
Результат + объяснение

⚙ Как работает агент

1. Пользователь вводит запрос (на обычном языке).
2. LLM получает схему таблиц и формулирует PySpark-код.
3. Код исполняется через Spark.
4. Результат + объяснение возвращаются пользователю.

□ Пример запроса и ответа

Запрос:
“Покажи среднюю сумму транзакций по странам за последние 3 месяца”

Сгенерированный код:
```
df = spark.read.parquet("hdfs://.../transactions")
df_filtered = df.filter(df["date"] >= "2024-12-31")
df_grouped = df_filtered.groupBy("country").agg(F.avg("amount").alias("avg_amount"))
df_grouped.show()
```

Ответ:
“Отфильтрованы транзакции за последние 3 месяца и рассчитано среднее значение

🔒 Контроль и безопасность

- 🔒 Ограничение на опасные команды (drop/write)
- 🔒 Логи аудита
- 🔒 Возможность ручного подтверждения перед выполнением

□ План развития

- Поддержка нескольких таблиц и объединений (joins)
- Диалоговая история
- Генерация графиков (через matplotlib/plotly)
- Автоматическая подгрузка схем
- Визуальный интерфейс