Тексовая расшифровка видео:

DATA LAKE И DATA WAREHOUSE

План:

- Что такое Data Lake;
- Варианты решений;
- Структурируем в Data Warehouse;
- Денормализованные витрины;
- Итоговый выбор варианта.

Что такое Data Lake

Основные положения:

- Все данные компании **в сыром виде** агрегируются в **одно общее место**;
- В некоторых компаниях в качестве Data Lake используют структурированные БД, но это скорее исключение;
- Обычно под Data Lake понимается **объектное хранилище**, которое хранит данные просто в виде **бинарных блобов**;
- С помощью использования специализированных форматов (**Parquet, ORC** etc.) можно получить **ускорение запросов** поверх данных в озере.

Варианты решений

Существуют несколько вариантов технических решений:

Hadoop

- Обычно хорошо интегрируется с существующими инструментами Big Data;

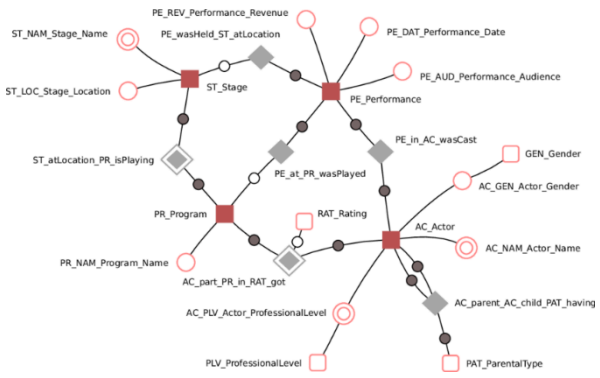
- Проверенное временем решение;
- Может быть не очень простым в поддержке.

Minio

- Полностью совместим с API S3;
- Довольно прост в поддержке и устройстве;
- Относительно новый проект, иногда с мелкими проблемами.

Структурируем в Data Warehouse

Data Warehouse в первую очередь подразумевает под собой уже что-то более структурированное:



Как мы помним, **основная задача** – собрать бизнес-объекты в виде структур в хранилище, чтобы аналитикам было удобно работать с ними. Есть два подхода, чтобы это сделать:

Базовый. Берем серьезную БД/OLAP storage и начинаем проектировать большую и сложную структуру. Это может быть **Vertica, Exasol, Teradata, SingleStore** и т.д. На сегодняшний день чаще используется **Greenplum**.

Денормализованные витрины. Когда данные достаточно простые, гомогенные или, когда мы используем подход по типу Event sourcing, мы можем о них думать, как о денормализованных витринах, когда данные сами по себе представляют широкие таблички с большим количеством полей, куда добавляем новые куски. Поверх этого делаем предагрегацию и простую аналитику.

Можно использовать **PostgreSQL, MongoDB**, однако они больше подходят для конечных представлений. Когда мы пишем большой поток данных логами в хранилище, подойдут **ClickHouse**, **Druid** (в системе Hadoop).

Итоговый выбор варианта

Условно у нас есть два базовых верхнеуровневых варианта:

Берем серьезное и тяжелое решение

- Много разных источников;
- Кросс-доменная аналитика;
- Сложные структурные запросы;
- Долгое хранение.

Достаточно простая доменная область (достаточно гомогенные данные)

- Достаточно простые данные на входе (например, clickstream/event sourcing);
- Статистический анализ с небольшой задержкой;
- Нет нужды в сложных JOIN’ах;
- Быстро устаревающие данные.