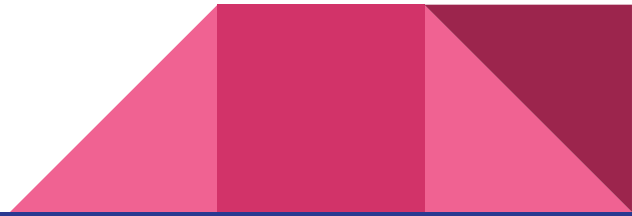
Работа с СУБД SQL

Введение. Организация работы с данными.

Проверить, идет ли запись

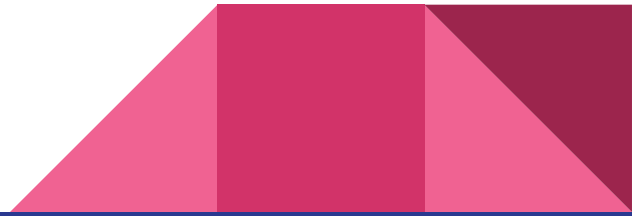
Знакомство

- настройка микрофона и аудио
- проверка работы чата
- напишите, пожалуйста, в чат кратко про свой опыт работы с SQL (0..10)



План занятия

- Архитектура данных
- Примеры архитектуры данных. Data Lake. Lakehouse.
- Цели и задачи Data warehouse
- Примеры архитектур
- Выбор технологий
- OLAP vs OLTP



СМОТРИ В БУДУЩЕЕ. ИНВЕСТИРУЙ В ЗНАНИЯ.

Архитектура данных

Архитектура данных

Архитектура данных — определение потребностей организации в данных (безотносительно к структуре), а также разработка и сопровождение основных рабочих описаний решений по их обеспечению.

Использование основных рабочих описаний в качестве руководящих материалов при осуществлении интеграции данных и контроля информационных активов, а также при согласовании инвестиций в области данных с бизнес-стратегией.

Ценность архитектуры данных

Ценность достигается за счет оптимизации требуемых ресурсов, операционной и проектной эффективности, а также расширения возможностей организации по использованию данных.

Чтобы этого добиться, требуются *качественное проектирование* и *планирование*, равно как и *способность обеспечить* эффективную *реализацию* проектов и планов.

Цели архитектуры данных

- Определение требований к хранению и обработке данных
- Разработка структур и планов, направленных на обеспечение текущих и долгосрочных потребностей организации в данных
- Обеспечение стратегической готовности организации к быстрому развитию своих продуктов, услуг и данных с целью получения преимуществ от использования возможностей, заложенных в новейших технологиях

Достижение целей

Для достижения целей архитекторы данных определяют и поддерживают **спецификации**, которые:

- определяют текущее состояние данных в организации;
- предоставляют стандартный бизнес-словарь для данных и компонентов;
- обеспечивают согласованность архитектуры данных с корпоративной стратегией и бизнес-архитектурой;
- отражают стратегические требования к данным;
- очерчивают высокоуровневые интегрированные проектные решения, призванные обеспечить выполнение этих требований;
- обеспечивают интеграцию разрабатываемых решений с дорожной картой реализации общей корпоративной архитектуры организации.

Основные артефакты архитектуры данных

Описания архитектуры данных должны включать:

- Корпоративные модели данных (Enterprise data model, EDM)
- Описания потоков данных

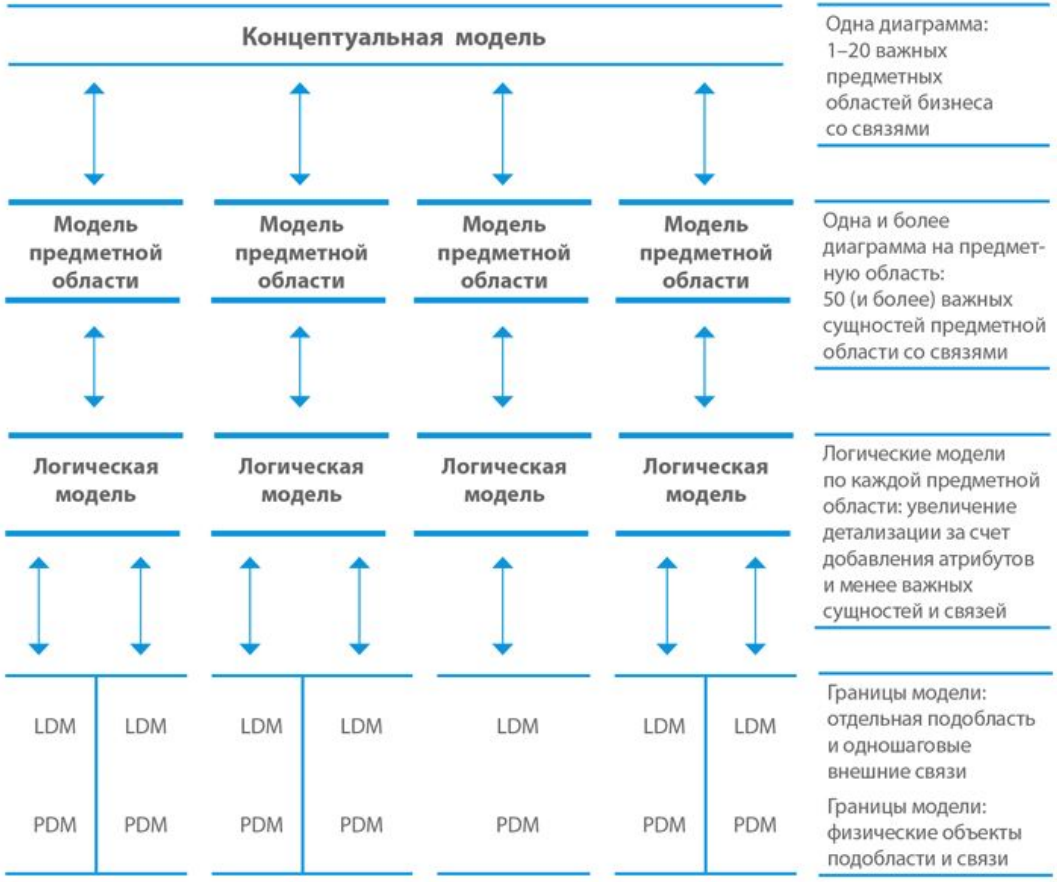
Модель и потоки данных отображаются в трёх состояниях:

- текущем
- целевом (архитектурная перспектива)
- переходном (проектная перспектива)

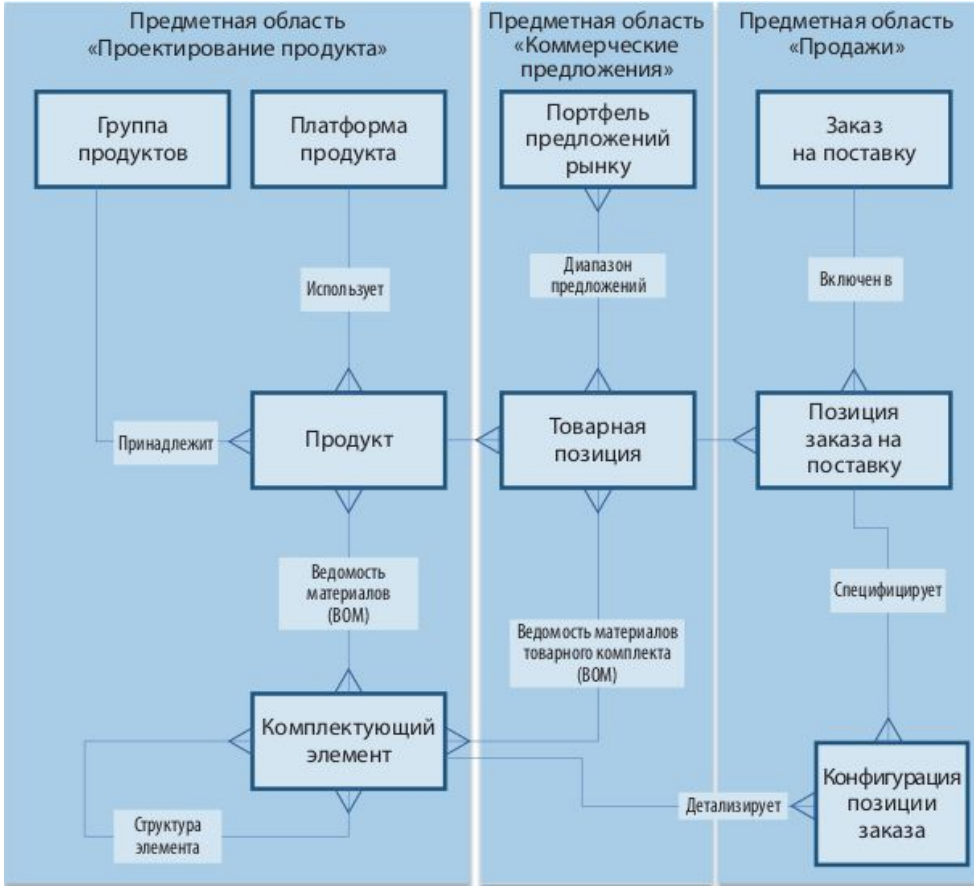
Корпоративная модель данных

Корпоративная модель данных (Enterprise Data Model, EDM) представляет собой целостную, не зависящую от технических средств реализации концептуальную или логическую модель данных, отражающую единый согласованный взгляд на данные в масштабах всей организации.

Корпоративная модель данных



Пример диаграмм предметных областей

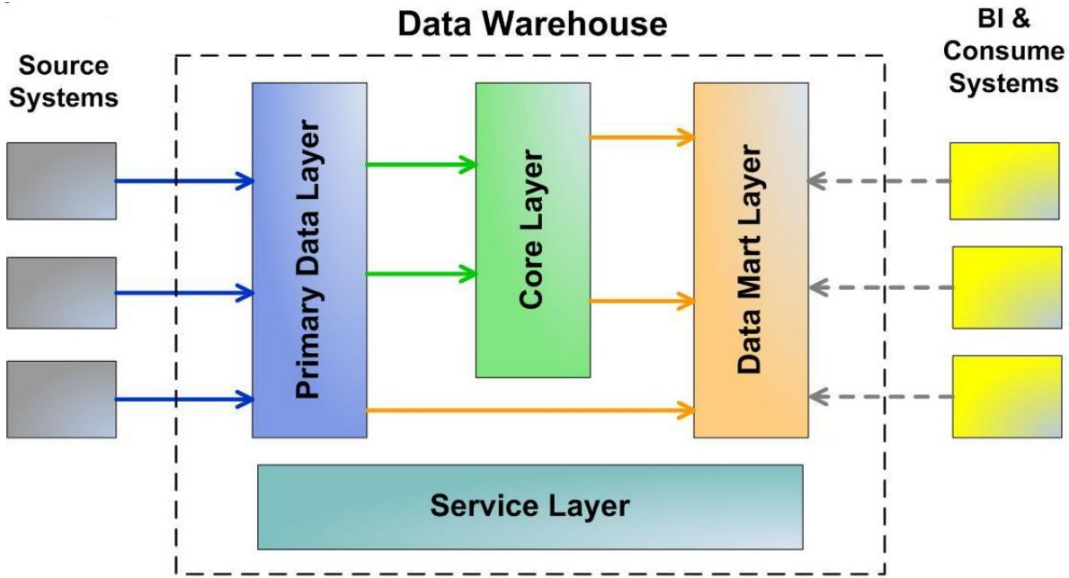


СМОТРИ В БУДУЩЕЕ. ИНВЕСТИРУЙ В ЗНАНИЯ.

Примеры архитектуры данных

Слои данных

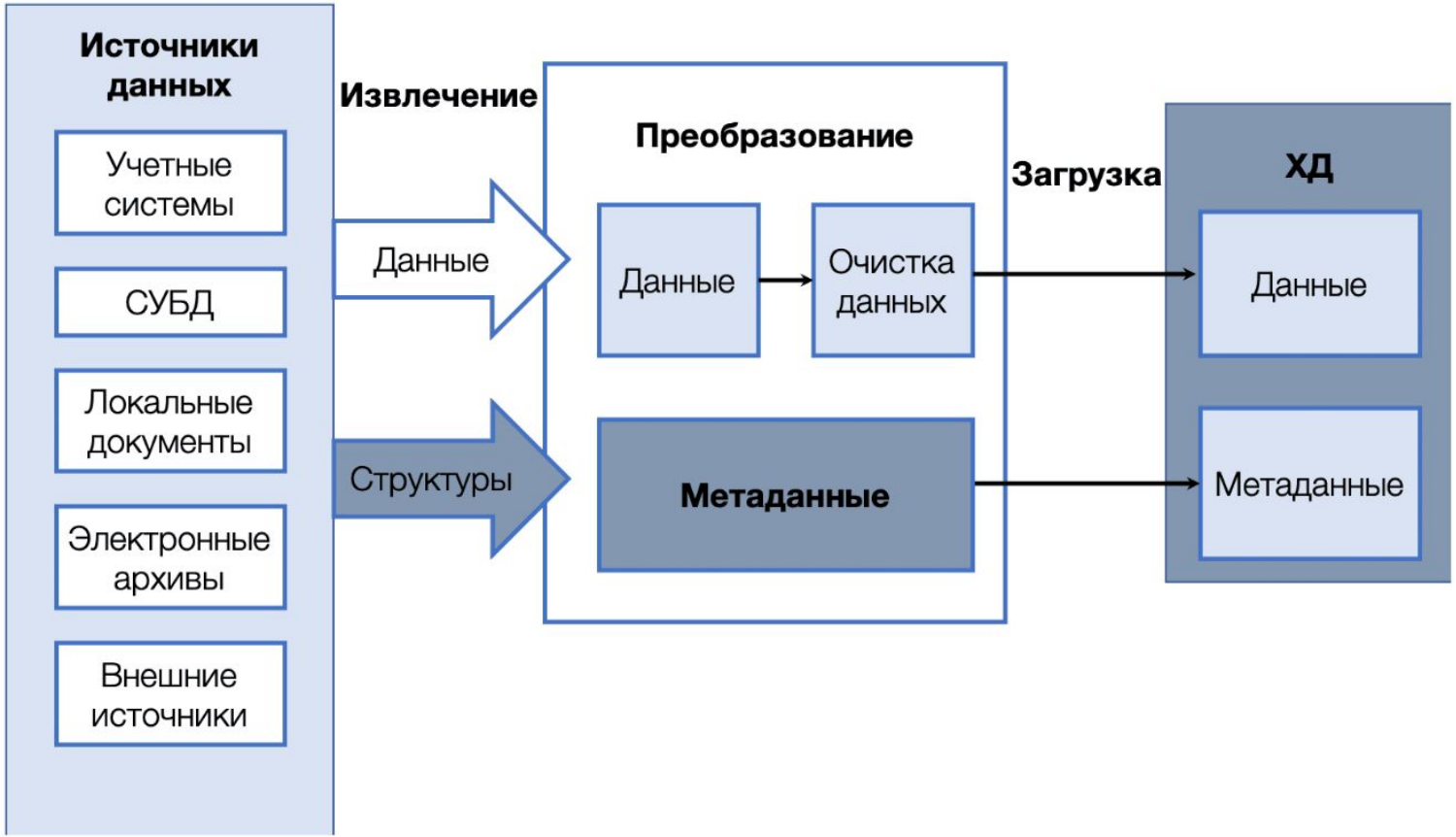
Уровневая архитектура – это средство борьбы со сложностью системы. Каждый последующий уровень абстрагирован от сложностей внутренней реализации предыдущего



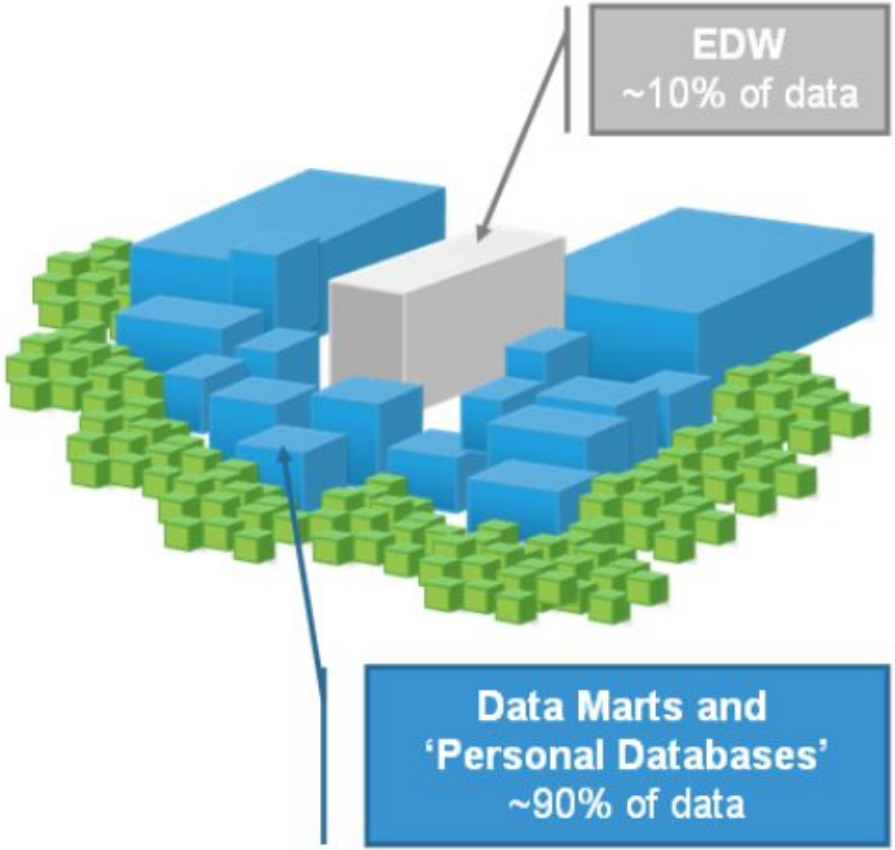
Слои данных

- **Операционный слой первичных данных** (Primary Data Layer, Raw или Staging) — загрузка информации из систем-источников в исходном качестве и сохранением полной истории изменений.
- **Ядро хранилища** (Core Data Layer) — центральный компонент, консолидация данных из разных источников, приведение их к единым структурам и ключам.
- **Аналитические витрины** (Data Mart Layer) — данные преобразуются к структурам, удобным для анализа и использования в BI-дашбордах или других системах-потребителях.

ETL / ELT


Источники данных
Учетные системы
СУБД
Локальные документы
Электронные архивы
Внешние источники
Извлечение
Данные
Структуры
Преобразование
Данные
Очистка данных
Метаданные
Загрузка
ХД
Данные
Метаданные

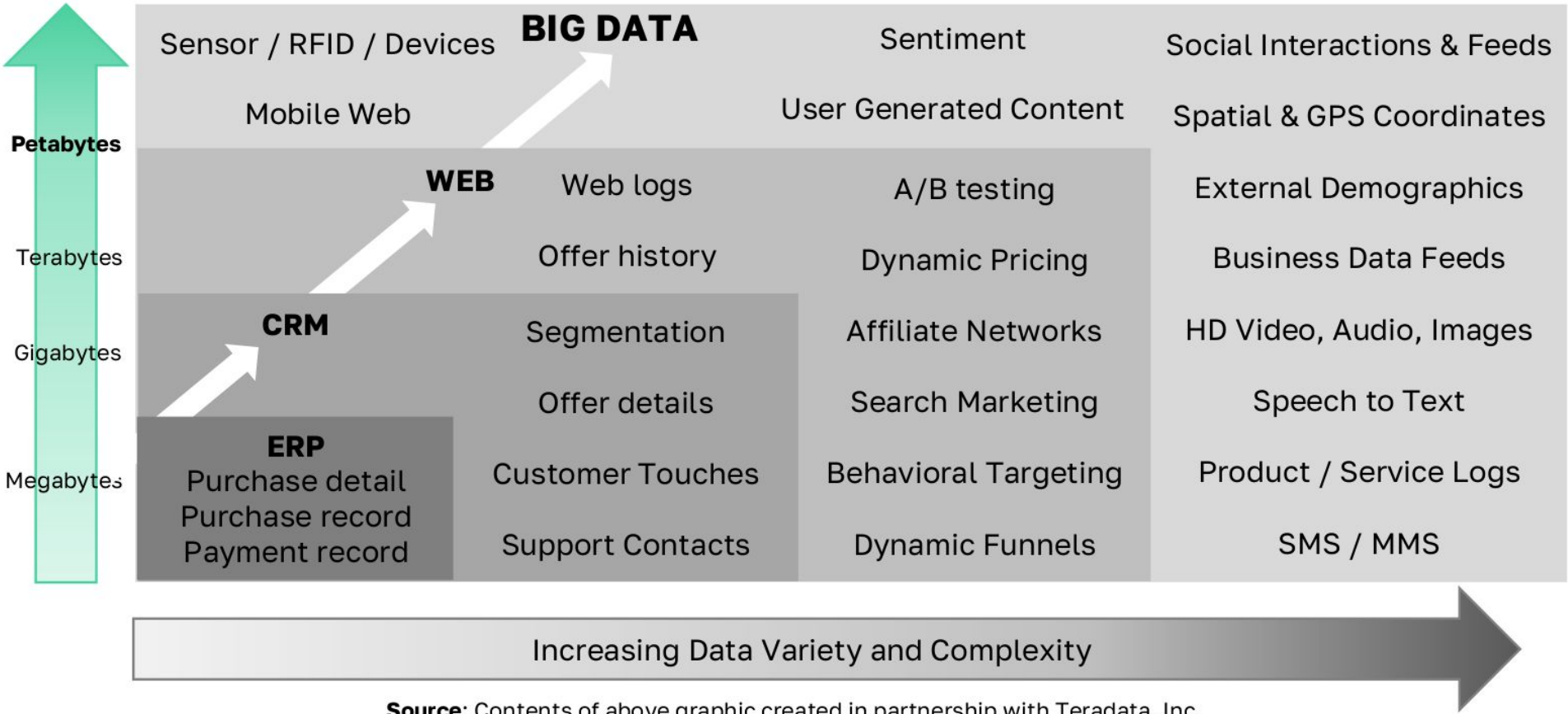
Данные в компаниях



Hadoop и Big Data

- **Hadoop** "удешевил" и упростил процесс анализа петабайтов доступной информации.
- Зародилась эра «**Больших Данных**» в которой заново преобразилась аналитическая структура.
- Термин **Big Data** впервые был озвучен в 2008 году на страницах спецвыпуска журнала Nature в статье главного редактора Клиффорда Линча. Этот номер издания был посвящен взрывному росту глобальных объемов данных и их роли в науке.

Большие данные

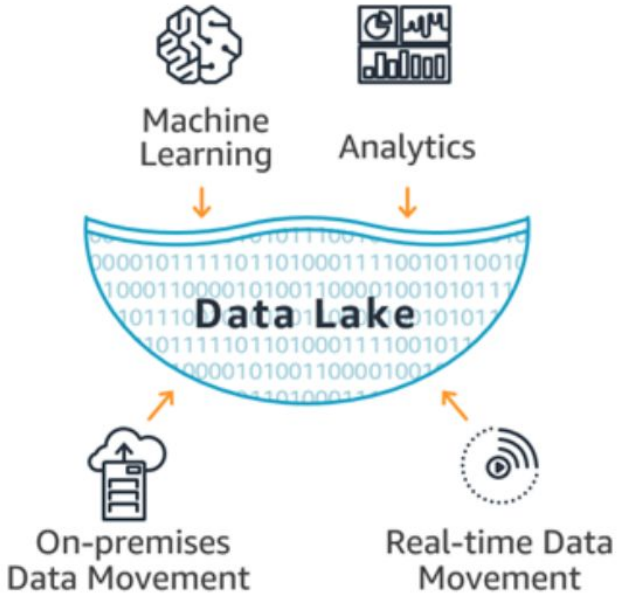


Source: Contents of above graphic created in partnership with Teradata, Inc.

Data Lake

“**Озеро данных**” — это система или хранилище данных, хранящихся в естественных, “сырых” форматах.

Озера Данных позволяют пользователям работать со всеми данными в компании без привлечения ИТ-специалистов



Data Lake

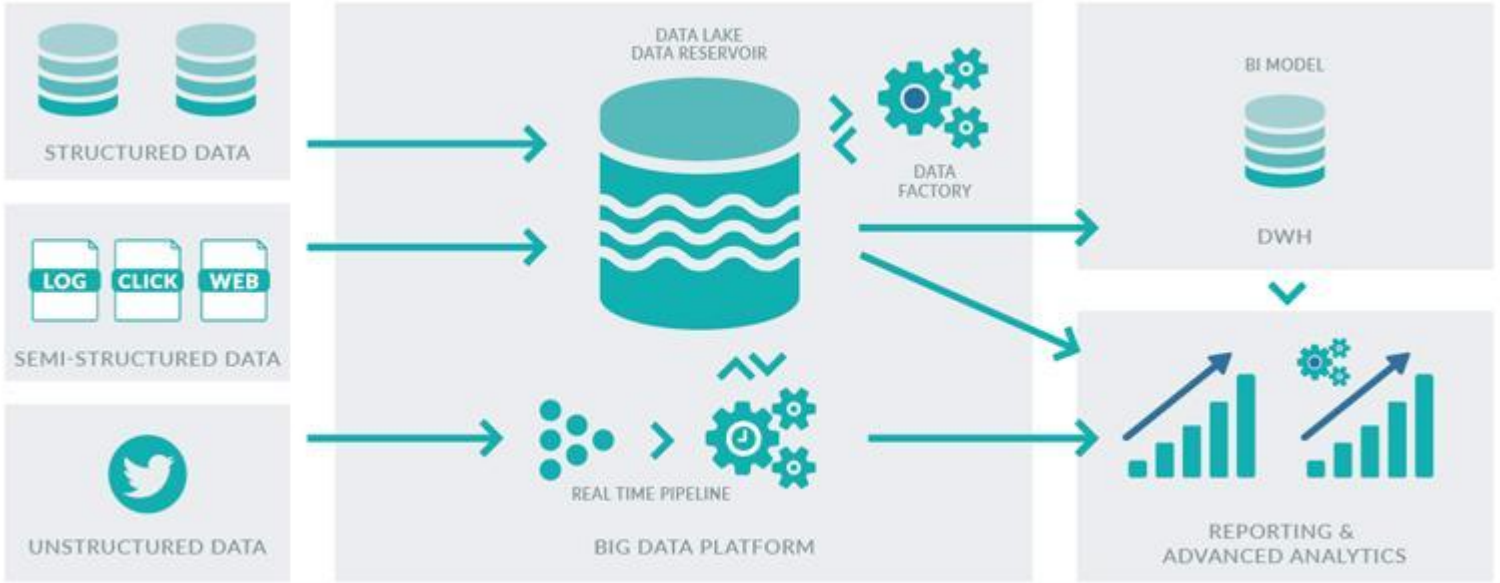
Единое хранилище данных:

- Необработанные копии исходных данных
- Преобразованные данные
- Структурированные данные из СУБД
- Полуструктурированные данные (CSV, JSON, XML)
- Неструктурированные данные (письма, документы)
- Двоичные данные (изображения, аудио, видео)

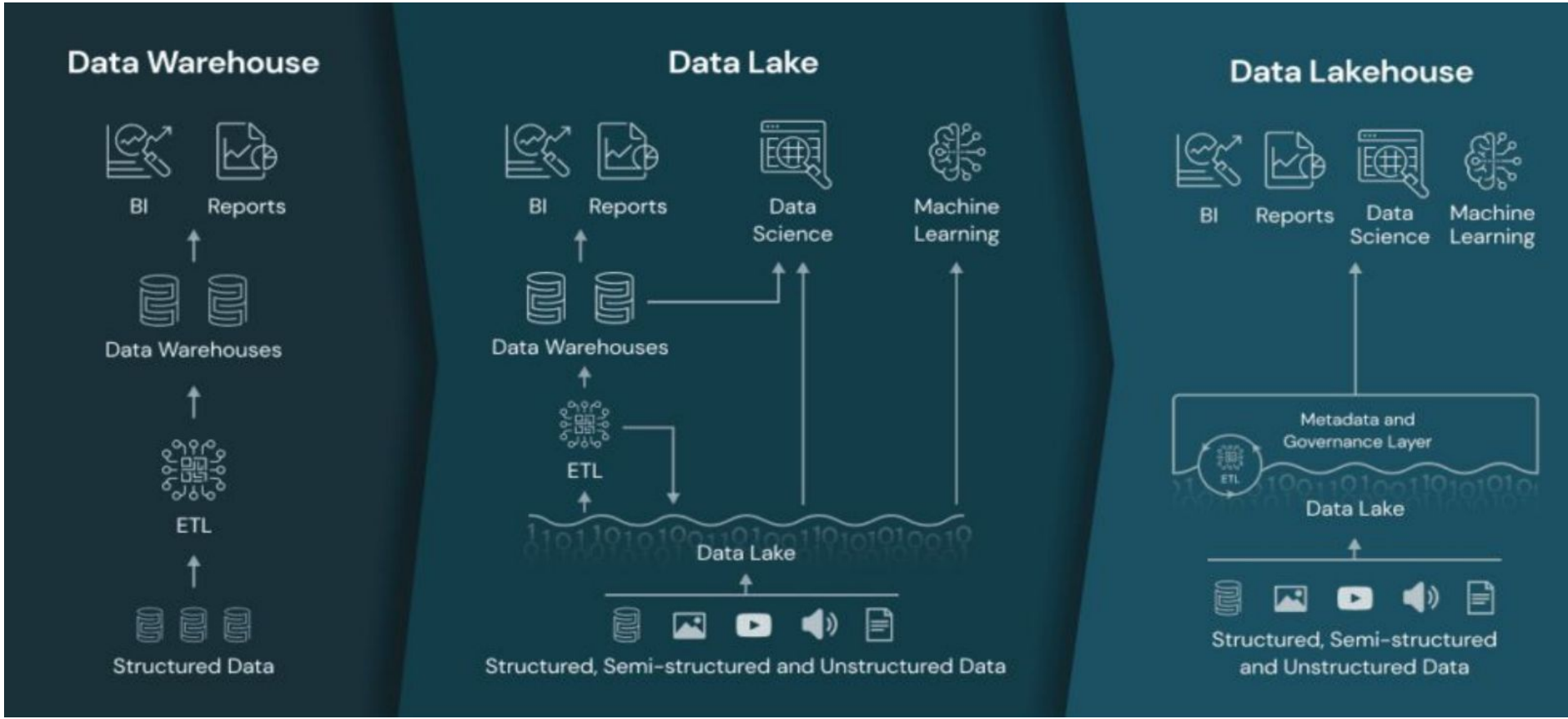
Data Lake vs DWH

	Data Lake	Data Warehouse
Data	Нереляционные и реляционные	Реляционные
Schema	Schema-on-read	Schema-on-write
Price / Performance	Становится быстрее Недорогие системы хранения	Быстро Дорогие системы хранения
Data Quality	Любые данные	Высокое качество данных
Users	Data Scientists, Data Developers, Бизнес аналитики	Бизнес аналитики
Analytics	Машинное обучение, прогнозная аналитика, разведочный анализ	Отчёты, BI, визуализация

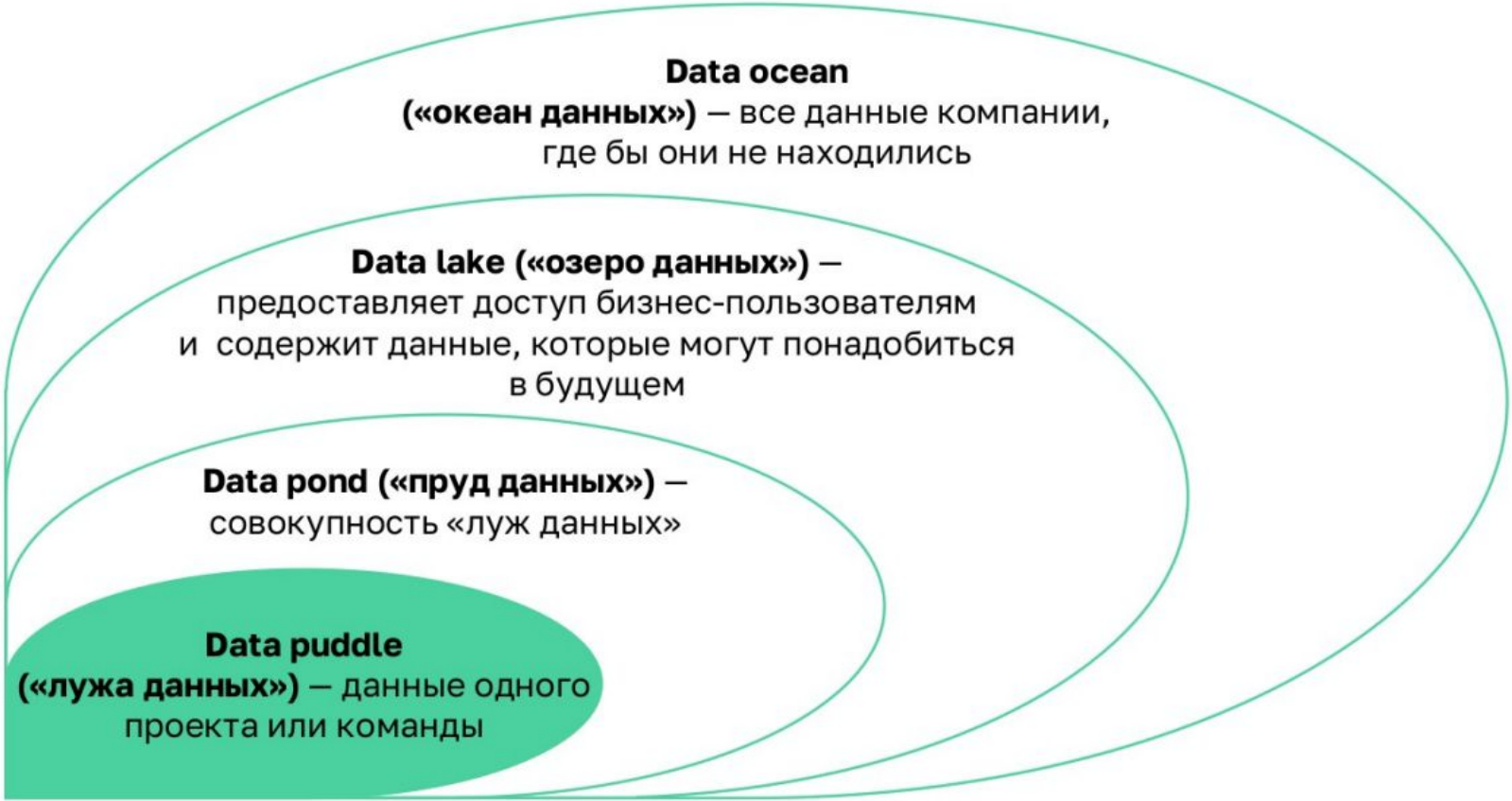
LakeHouse = Data Lake + DWH



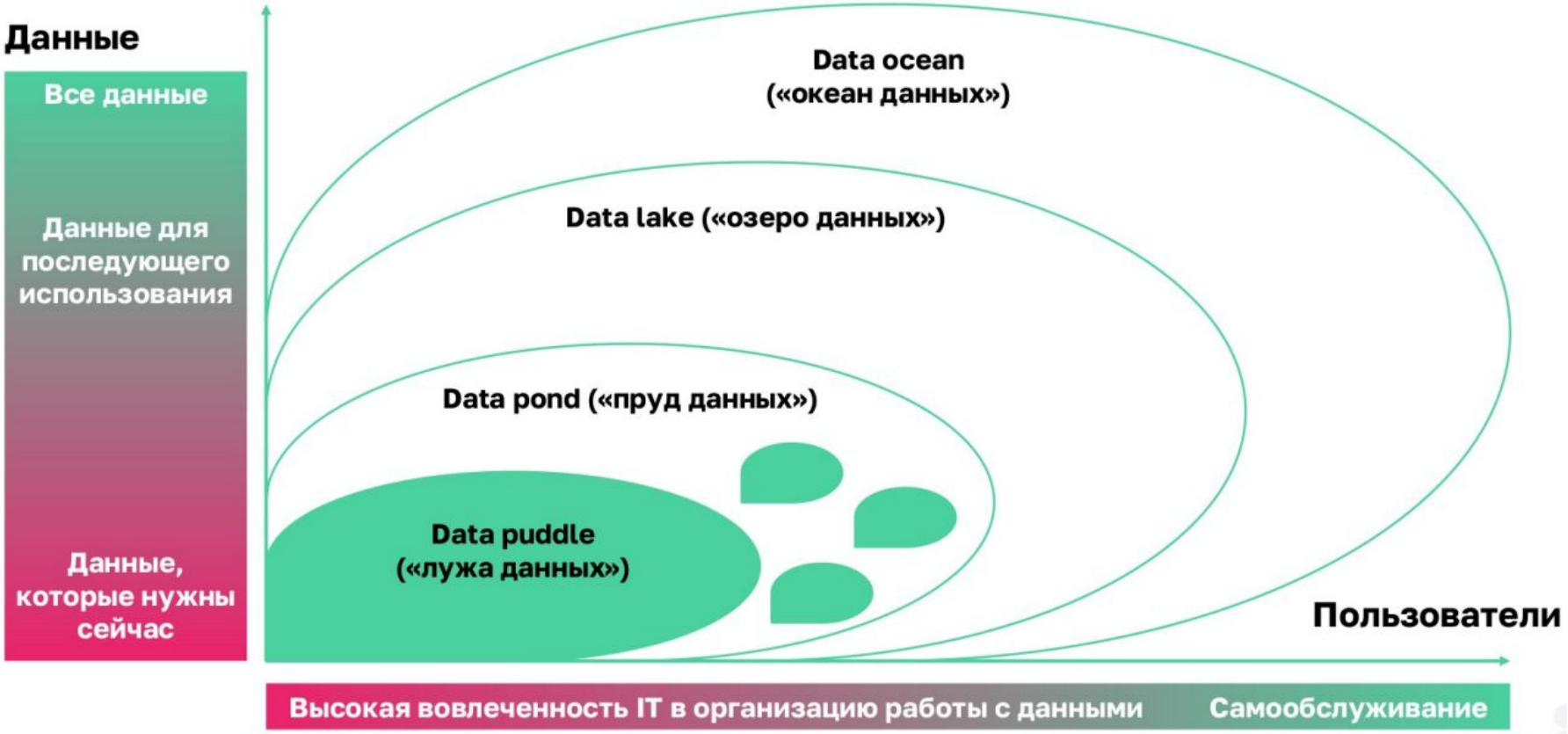
Data warehouse vs Data Lake



Зрелость данных



Зрелость данных



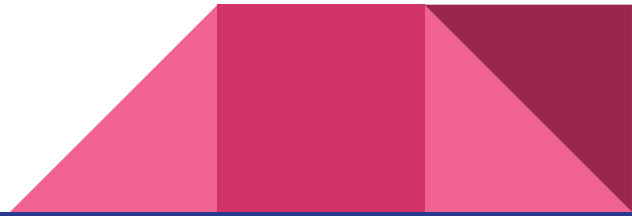
Предостережение

«Болото данных» — это испорченное и неуправляемое озеро данных, недоступное для пользователей или представляющее небольшую ценность.

«Кладбище больших данных»: сбрасываем всё в распределённую файловую систему и теряем представление о том, что там находится.

Успешное Озеро Данных

- Правильная платформа
- Правильные данные
- Правильные интерфейсы

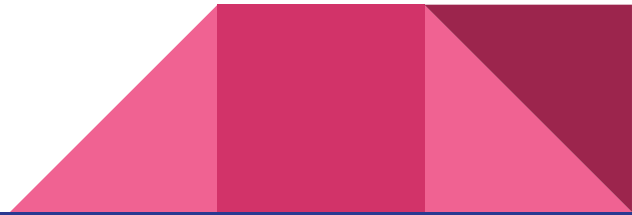


СМОТРИ В БУДУЩЕЕ. ИНВЕСТИРУЙ В ЗНАНИЯ.

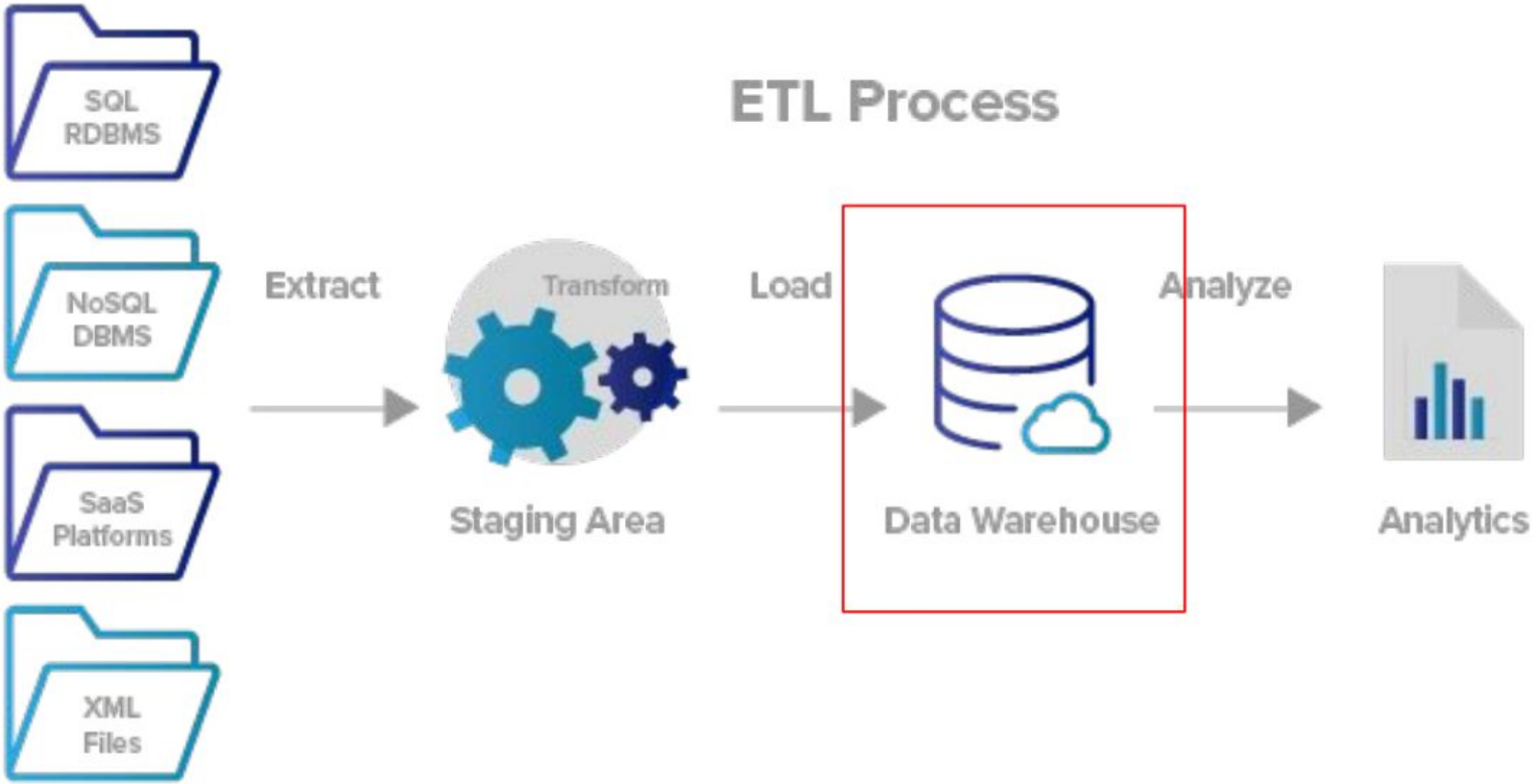
Цели и задачи Data warehouse

Аналитическая база данных / Data Warehouse

- Под аналитической базой данных традиционно понимают реляционное хранилище структурированных данных
- DWH предназначен для ручной (и не только) аналитики и принятия бизнес-решений BI - business intelligence
- DWH является инструментом стратегии “data-driven”
- Типичные пользователи DWH:
  - Продажи
  - Развитие бизнеса
  - Управление
  - Кадры



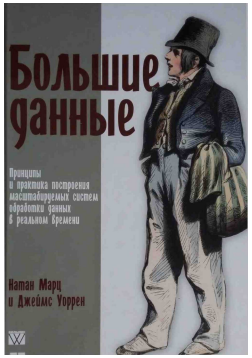
Аналитическая база данных / Data warehouse



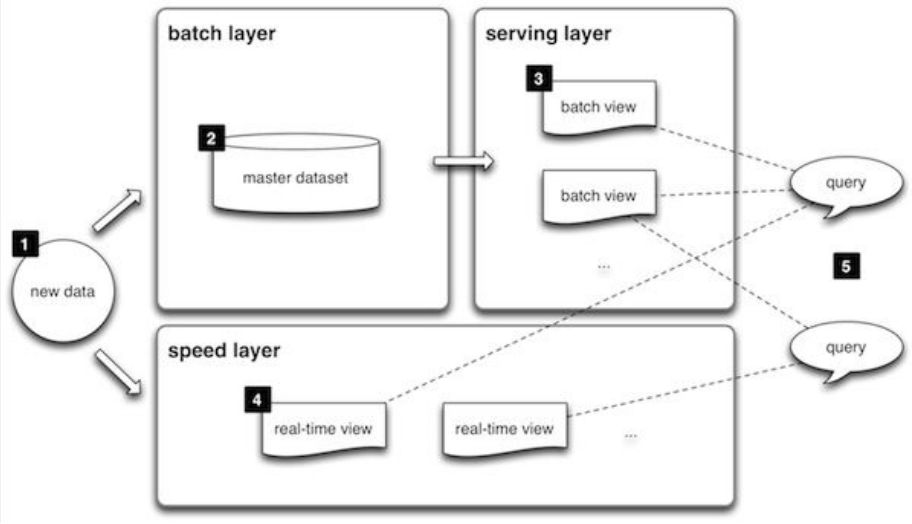
СМОТРИ В БУДУЩЕЕ. ИНВЕСТИРУЙ В ЗНАНИЯ.

Примеры архитектур

Lambda



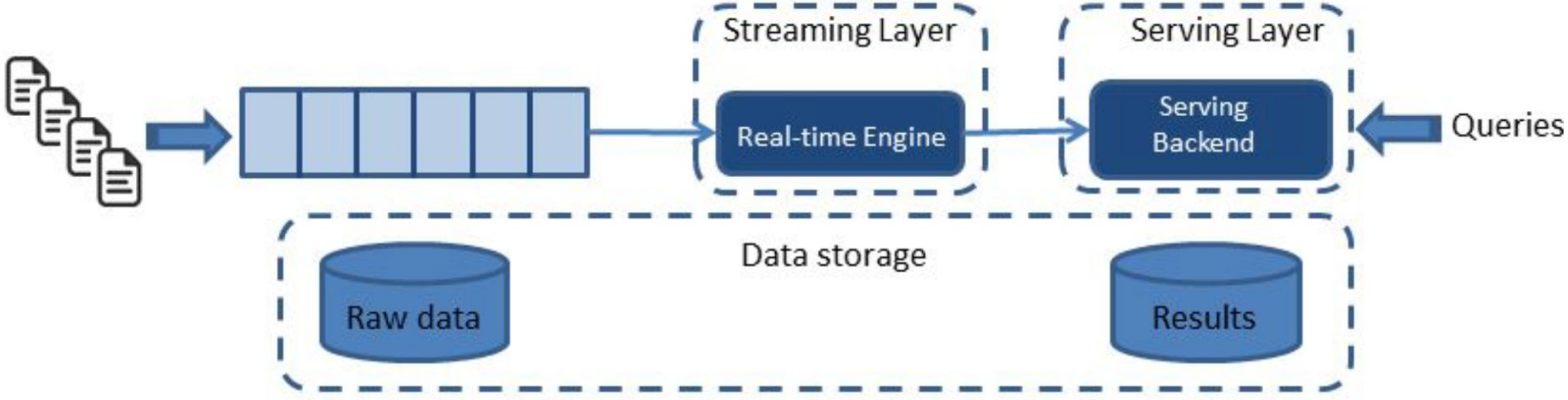
1) Все данные отправляются в batch и speed уровни
2) Пакетный (batch) уровень:
    1) Управление основным набором данных (неизменяемый, только добавляемый)
    2) Предварительное вычисление пакетных представлений
3) Обслуживающий уровень индексирует пакетные представления
4) Скоростной уровень обрабатывает только новые данные
5) Для ответа на входящие запросы объединяются результаты пакетных представлений и представлений реального времени



Архитектурный паттерн для обработки больших данных: Lambda

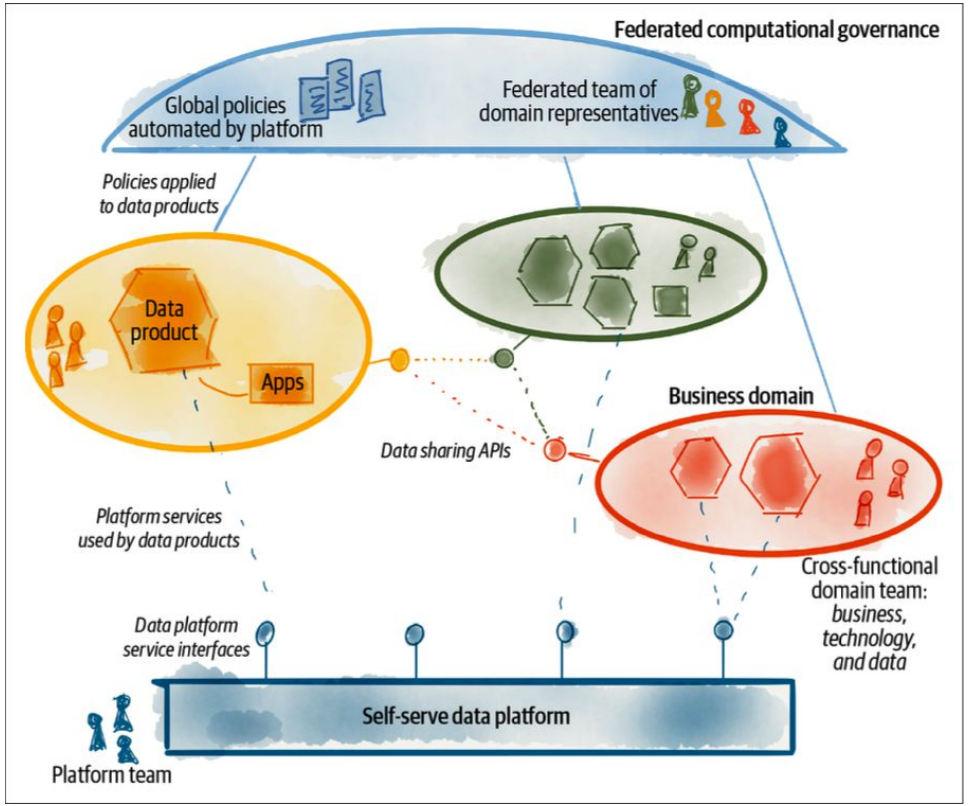
Kappa

- Убираем пакетную обработку
- Потоковая обработка всех данных



https://www.oreilly.com/radar/questioning-the-lambda-architecture/

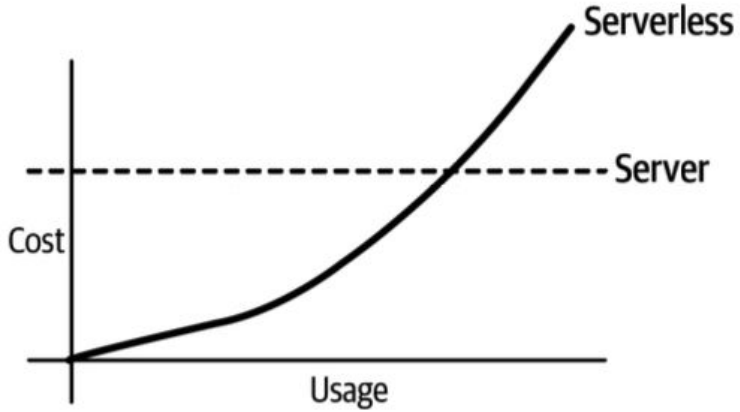
Data Mesh



https://martinfowler.com/articles/data-mesh-principles.html

Выбор технологий

Выбор технологий



- Размер и возможности команды
- Скорость вывода на рынок
- Совместимость
- Оптимизация затрат и повышение ценности бизнеса
- Сегодня против будущего: неизменные технологии против преходящих
- Варианты развёртывания
- Строить или покупать
- Монолитный или модульный
- Бессерверный или серверный

Варианты развёртывания

- **On-Premise** — своё оборудование, выделенное для сервисов
- **Private Cloud** — облачная инфраструктура на своём оборудовании
- **Public Cloud** — виртуальная инфраструктура принадлежит провайдеру и предоставляется в аренду

On-Premise

Плюсы:

- Полный контроль
- Данные "дома"

Минусы:

- Высокий CapEx
- Требуется персонал для сопровождения

https://en.wikipedia.org/wiki/Capital_expenditure

Private Cloud

Плюсы:

- Полный контроль
- Данные "дома"
- Большая гибкость

Минусы:

- Высокий CapEx
- Требуется опытный персонал для сопровождения

Public Cloud

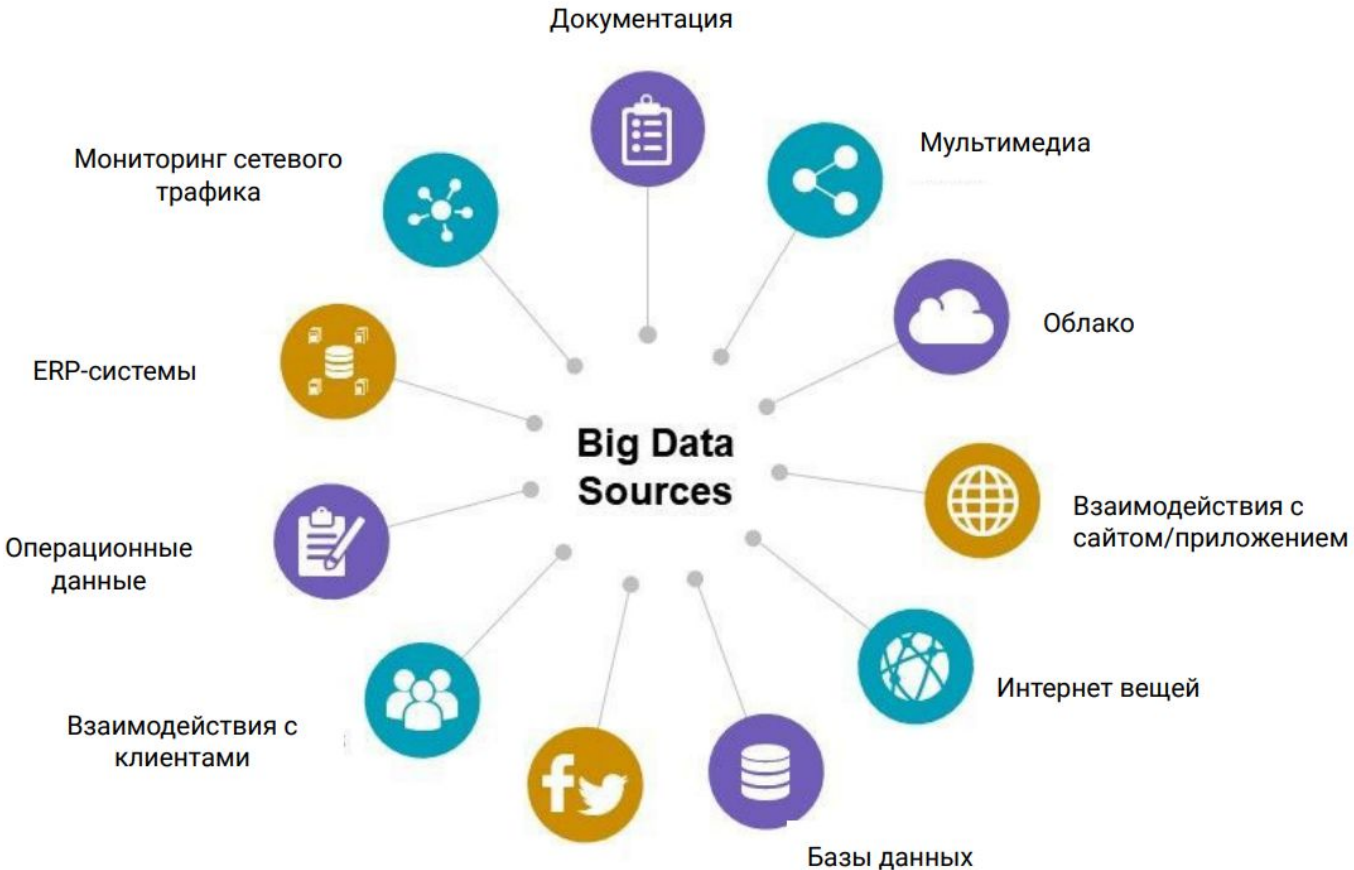
Плюсы:

- Низкий CapEx
- Большая гибкость
- Легко масштабировать

Минусы:

- Данные на чужом оборудовании
- Vendor Lock
- высоких OpEx

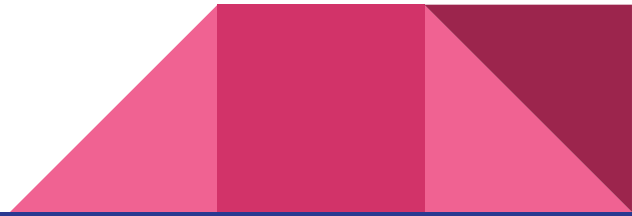
Источники данных



Data Storage

Хранилище данных - это различные технологии хранения данных, зависящие от:

1. Источников данных;
2. Типа данных;
3. Характера использования данных;
4. Объема данных;
5. Бюджета, стратегии компании;
6. ...



OLAP vs OLTP

Два типа систем обработки данных

Помимо типа данных, объема и источников, выбор хранилища также зависит от типа использования данных.

Различают два больших класса систем обработки данных:

1. **OLTP** - транзакционная система;
2. **OLAP** - аналитическая система;

Два типа использования данных

On**L**ine **T**ransaction **P**rocessing **(OLTP)** - *система обработки транзакций в реальном времени.*

Типичные операции:

- *Создание (CREATE)*
- *Чтение (SELECT)*
- *Модификация (UPDATE)*
- *Удаление (DELETE)*

Пример: *Система бронирования авиабилетов*

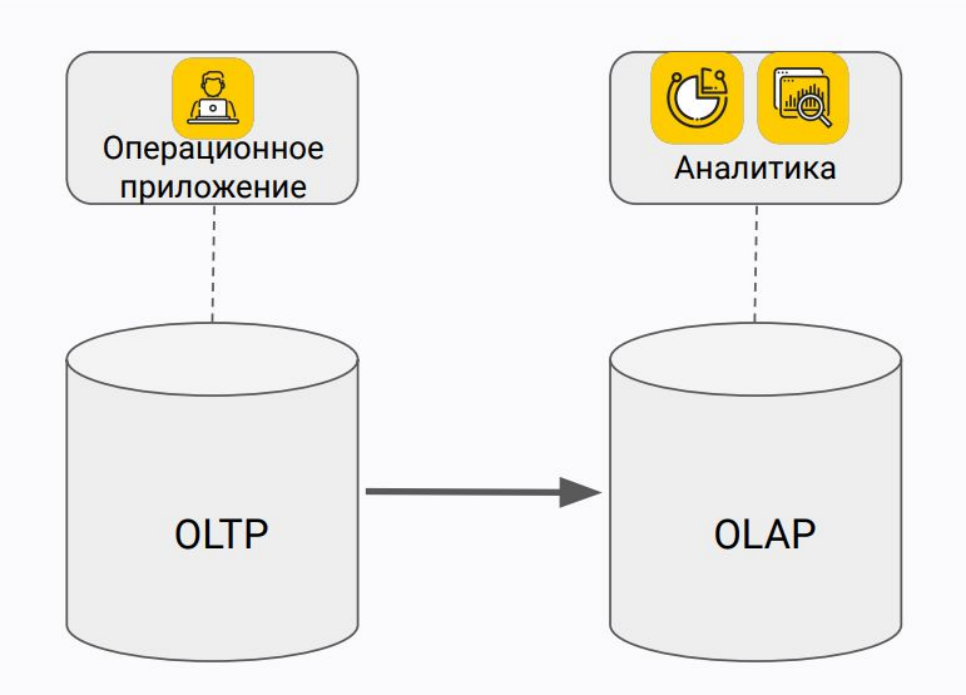
On**L**ine **A**nalytical **P**rocessing **(OLAP)** - *интерактивная аналитическая обработка данных. Данные суммируются, агрегируются.*

Типичные операции:

- *Чтение*
- *Агрегация (GROUP BY, OVER PARTITION BY)*

Пример: *Система отслеживания объемов продаж авиабилетов*

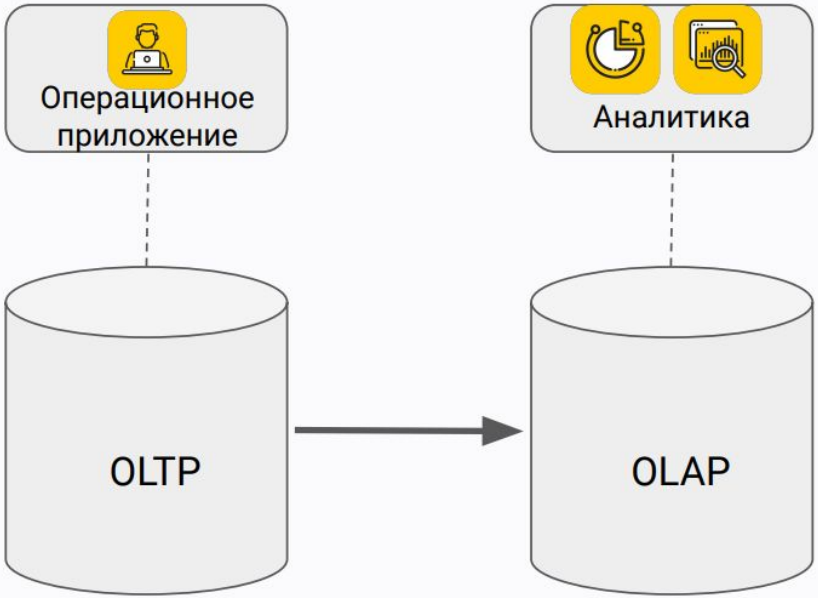
OLAP vs OLTP



Зачем разделять OLTP и OLAP?

OLAP vs OLTP

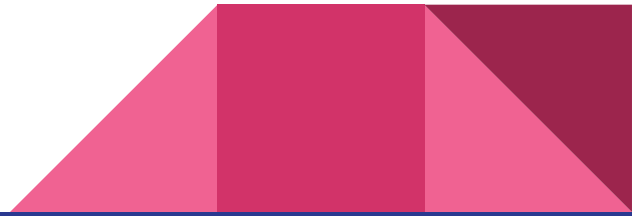
- Большой объем операций
- Высокая скорость обработки
- Много таблиц
- Данные нормализованы
- *“Кто* купил *X*?”



- Большой объем данных
- Низкая скорость обработки запросов
- Мало таблиц
- Данные денормализованы
- “Сколько человек купило *X*?”

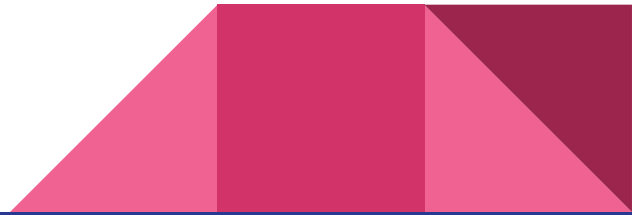
Кейс

Есть таблица с данными о состоянии удаленного оборудования, которые приходят раз в 15 минут от каждого терминала. Страница, которая показывает последнюю пришедшую информацию по всем терминалам. Как вы реализуете такой функционал?
OLTP или OLAP нагрузка?



Кейс

Есть сервис, в который поступают данные по заказам с разных магазинов (онлайн, оффлайн). Спустя время менеджеры обрабатывают заказы. В результате часть из них уходит в отправку как есть, часть меняет свой состав (после уточнения деталей или наличия фактических остатков), часть просто отменяются. Есть потребность анализировать данные о заказах (топ продаваемых товаров, динамика новых клиентов, динамика заказов, средний чек, динамика повторных заказов). Как вы реализуете такой функционал?
OLTP или OLAP нагрузка?



СПАСИБО ЗА ВНИМАНИЕ!

#аис
#учисьваис