НИЯУ МИФИ
Национальный исследовательский ядерный университет МИФИ
Кафедра 75 «Кафедра финансового мониторинга»

Лекция 1

Специальные технологии баз данных и информационных систем

Авторы:

Мелконян С.Е.
Айрапетян С.В.

Часть 1: OLTP и DSS системы

Из курсов «Базы данных» нам известно различие между системами оперативной обработки транзакций (Online Transaction Processing) и системами поддержки принятия решений (Decision Support System). Тем не менее, для лучшего понимания назначения нашей дисциплины вспомним эти и связанные с ними понятия.

OLTP-системы

Основное назначение OLTP-систем – это быстрая обработка небольших операций изменения данных (добавление, редактирование, удаление), поступающих большим потоком в режиме реального времени.

Основное требование к таким системам – это обеспечение выполнения операций в рамках заданных временных ограничений.

Область применения таких систем крайне широка. Это может быть информационная система магазина, в которой, прежде всего, необходимо быстро оформить и сохранить записи о совершённых покупках. Это может быть информационная система оперативной обработки банковских транзакций. Это может быть и часть полноценной ERP-системы (Enterprise Resource Planning, планирование ресурсов предприятия) крупной промышленной организации, отвечающая за ввод операционной информации.

OLTP-системы

Так как во главу угла в OLTP-системах ставится скорость обработки небольших транзакций, то в них база данных обычно проектируется в сильно нормализованном виде.

То есть, если говорить совсем упрощённо, большие таблицы разделяются на минимальные смысловые порции, образующие много маленьких (по количеству полей) таблиц.

Очень часто OLTP-системы содержат информацию актуальную только в текущий момент или в течении небольшого периода. Для долговременных исследований, требующих данные за длительные периоды времени OLTP-системы не подходят.

Обычно каждая OLTP-система отвечают за свою довольно узкую часть бизнес процесса кампании и содержит минимально требуемую детализацию данных.

Таким образом, использовать OLTP-системы в чистом виде для серьезных аналитических или предиктивных исследований затруднительно.

DSS-системы

В противоположность OLTP-системам, основной целью систем поддержки принятия решений является не быстрое преоброзование данных, а предоставление аналитических и предиктивных механизмов.

Обычно DSS-системы аккумулируют данные из нескольких источников и/или за длительный период времени.

В зависимости от назначения DSS-системы могут решать разные задачи. Это могут быть простые системы, которые при помощи заранее написанных запросов формируют различные статистические данные. Например, сводную информацию о продажах за месяц по всем точкам дистрибьюторской сети. Это могут быть системы, позволяющие формировать произвольные статистические запросы с различными элементами агрегаций данных. И это могут быть системы, позволяющие выявить на основе имеющихся данных новые. Например, выполнить прогнозирование расходов топлива в ближайшие месяцы на основе собранных за последние пять лет данных.

DSS-системы

Данные **в DSS-системы поступают редко** (по сравнению с OLTP-системами). Причём обычно это происходит автоматическим путём, а не при помощи оператора. Работают с DSS-системами аналитики. Как следствие, минимальное время отклика уже не так важно. Аналитик может ждать результата и несколько часов.

Так как работа идёт не одной записью, а с большим числом подробно разобранных данных, то для **DSS-систем характерно применение денормализации данных.**

Data Mining

Особым видов DSS-систем являются системы, помогающие аналитику обнаружить новые, заранее ему неизвестные знания в имеющемся в его распоряжении наборе данных.

Такой процесс называется **интеллектуальным анализом данных.**

В англоязычной литературе он более известен под термином Data Mining.

Основные этапы

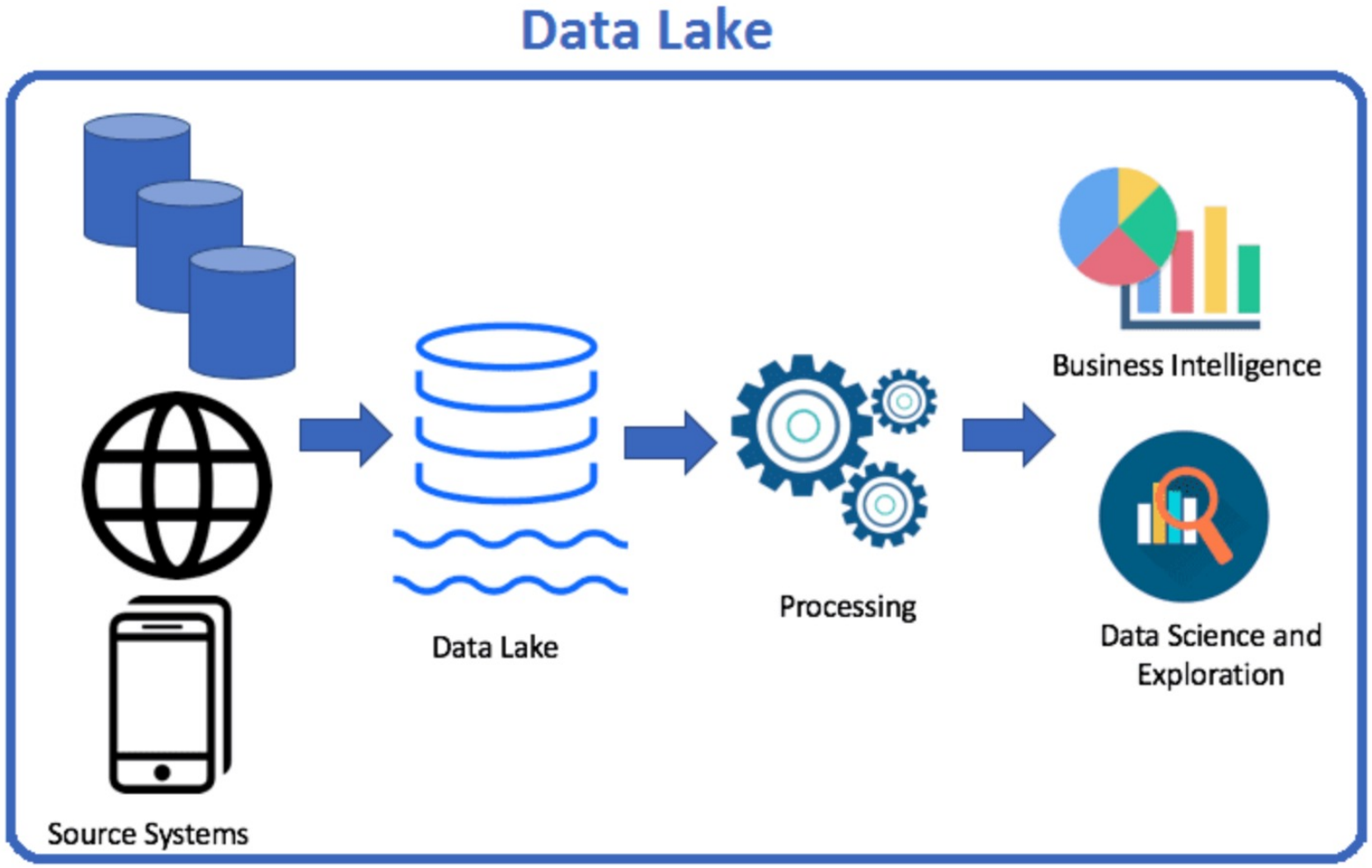
Любая DSS-система обычно решает следующие задачи:

1) сбор информации;
2) преобразование информации;
3) хранение информации;
4) представление информации в виде, пригодном для аналитического исследования и (но не обязательно) выявление новых знаний методами Data Mining.

Очевидно, что решать задачи Data Mining не возможно без решения первых трёх задач.

Озеро Данных – Data Lake

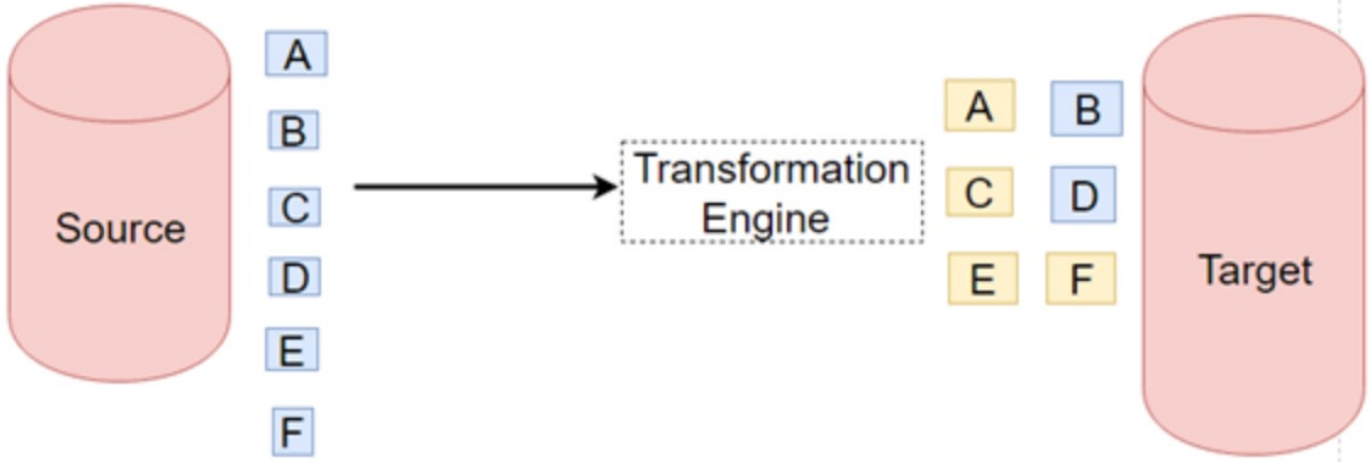
Data lake — это огромное хранилище, которое принимает любые файлы всех форматов. Источник данных тоже не имеет никакого значения. Озеро данных может принимать данные из CRM- или ERP-систем, продуктовых каталогов, банковских программ, датчиков или умных устройств — любых систем, которые использует бизнес.



Некоторые термины и понятия

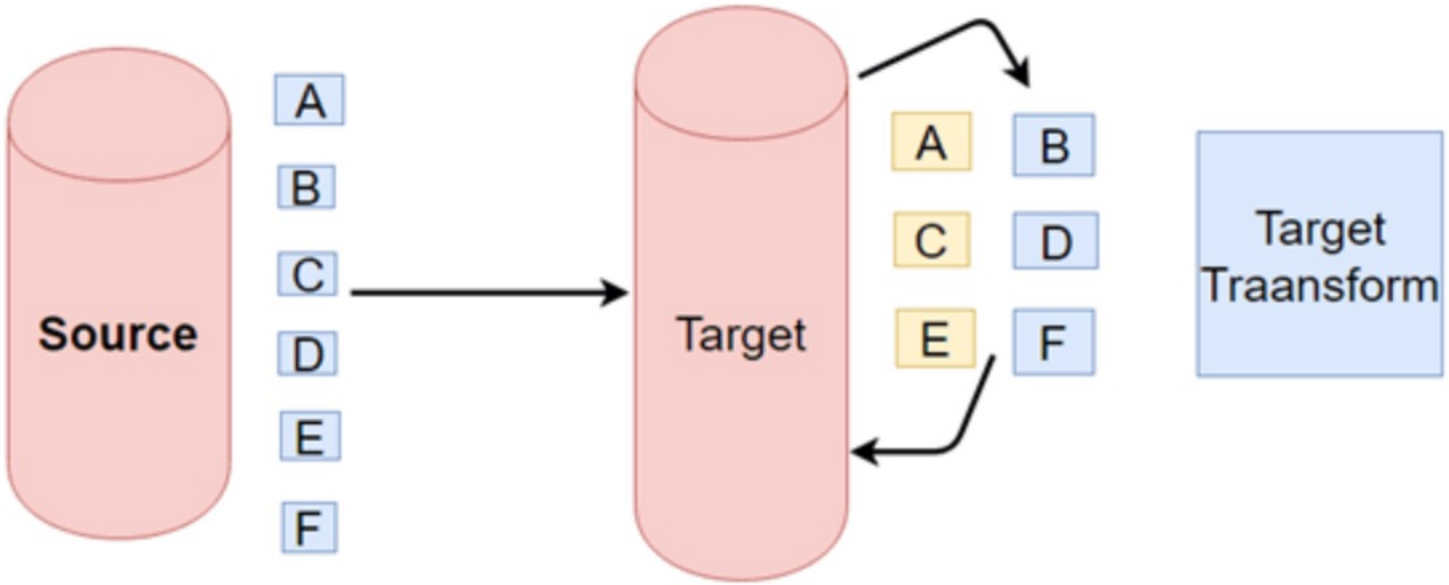
ETL – аббревиатура от Extract, Transform, Load. один из основных процессов в управлении хранилищами данных, который включает в себя:

- ❑ извлечение данных из внешних источников;
- ❑ их трансформация и очистка, чтобы они соответствовали
- ❑ потребностям бизнес-модели;
- ❑ и загрузка их в хранилище данных.

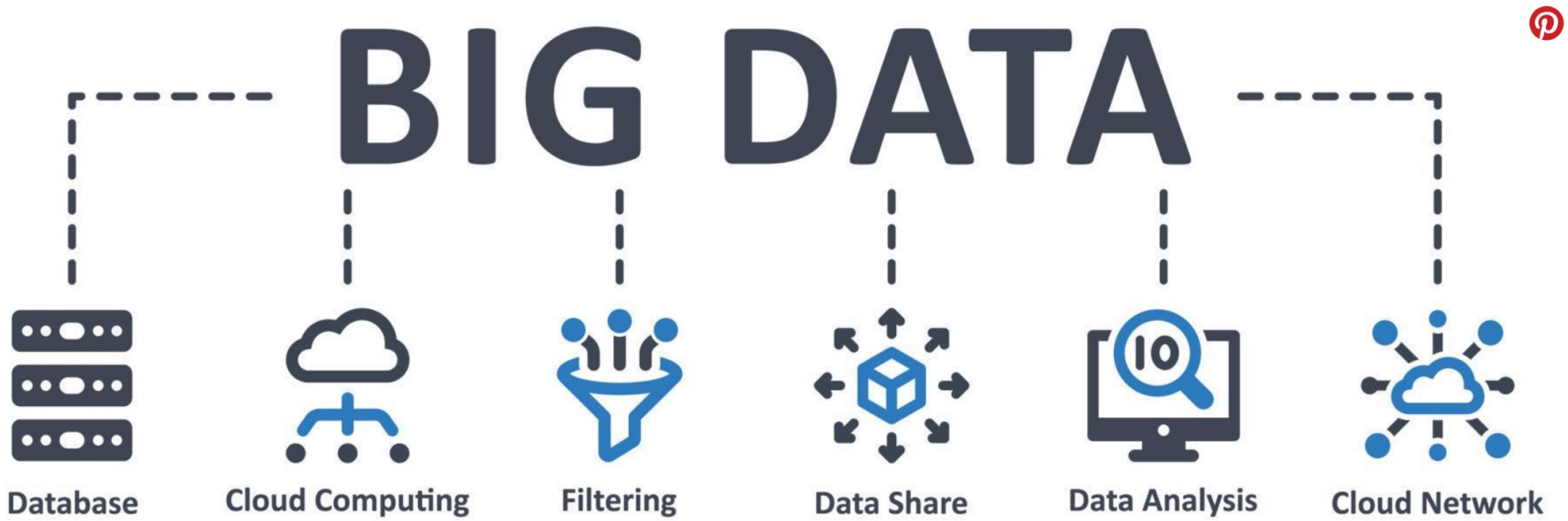


Некоторые термины и понятия

ELT – это другой метод рассмотрения инструментального подхода к перемещению данных. Вместо преобразования данных перед их записью ELT позволяет целевой системе выполнить преобразование. Данные сначала копируются в цель, а затем преобразуются на место.



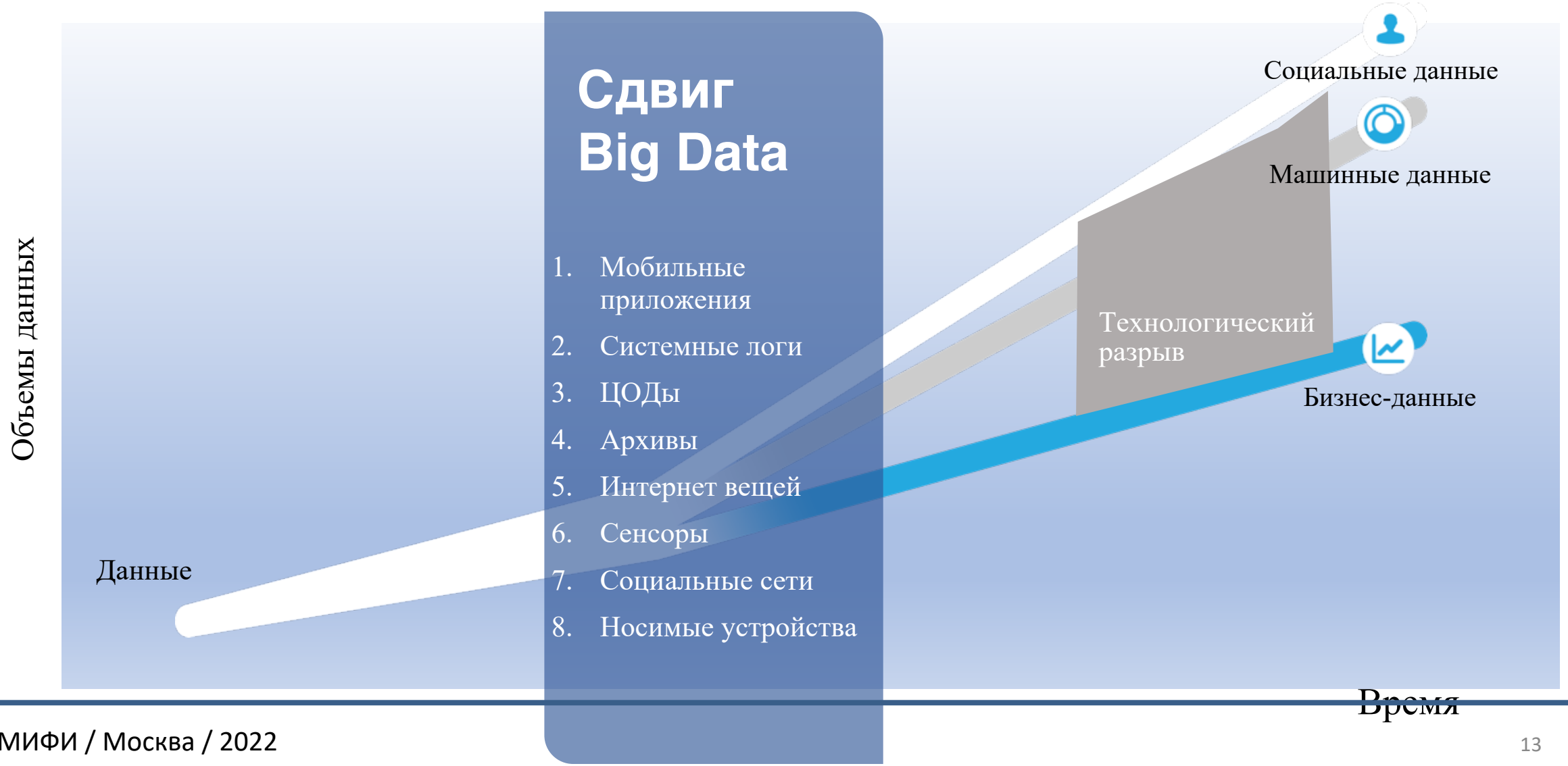
Обоснование актуальности применения распределенных технологий



Большие данные – это наборы данных такого объёма, что традиционные инструменты не способны осуществлять их захват, управление и обработку за приемлемое для практики время.

Большие данные объединяют техники и технологии, которые извлекают смысл из данных на экстремальном пределе практичности

Обоснование актуальности применения распределенных технологий



Парадигма больших данных

3 группы задач:

- ❑ хранение и управление объёмом данных в сотни терабайт или петабайт, которые обычные реляционные базы данных не позволяют эффективно использовать

- ❑ организация неструктурированной информации, состоящей из текстов, изображений, видео и других типов данных

- ❑ анализ Big Data, рассматривающий способы работы с неструктурированной информацией, генерацию аналитических отчётов и внедрение прогностических моделей.

Направления работы экспертов по большим данным:

- Аналитикой занимаются **Data Scientist** и **Data Analyst**, в их обязанности входит формирование гипотез, поиск закономерностей в наборах данных (dataset), визуализация информации, подготовка данных к моделированию, разработка алгоритмов Machine Learning (машинного обучения), интерпретация полученных данных, а также изучение предметной области или бизнес-процесса.

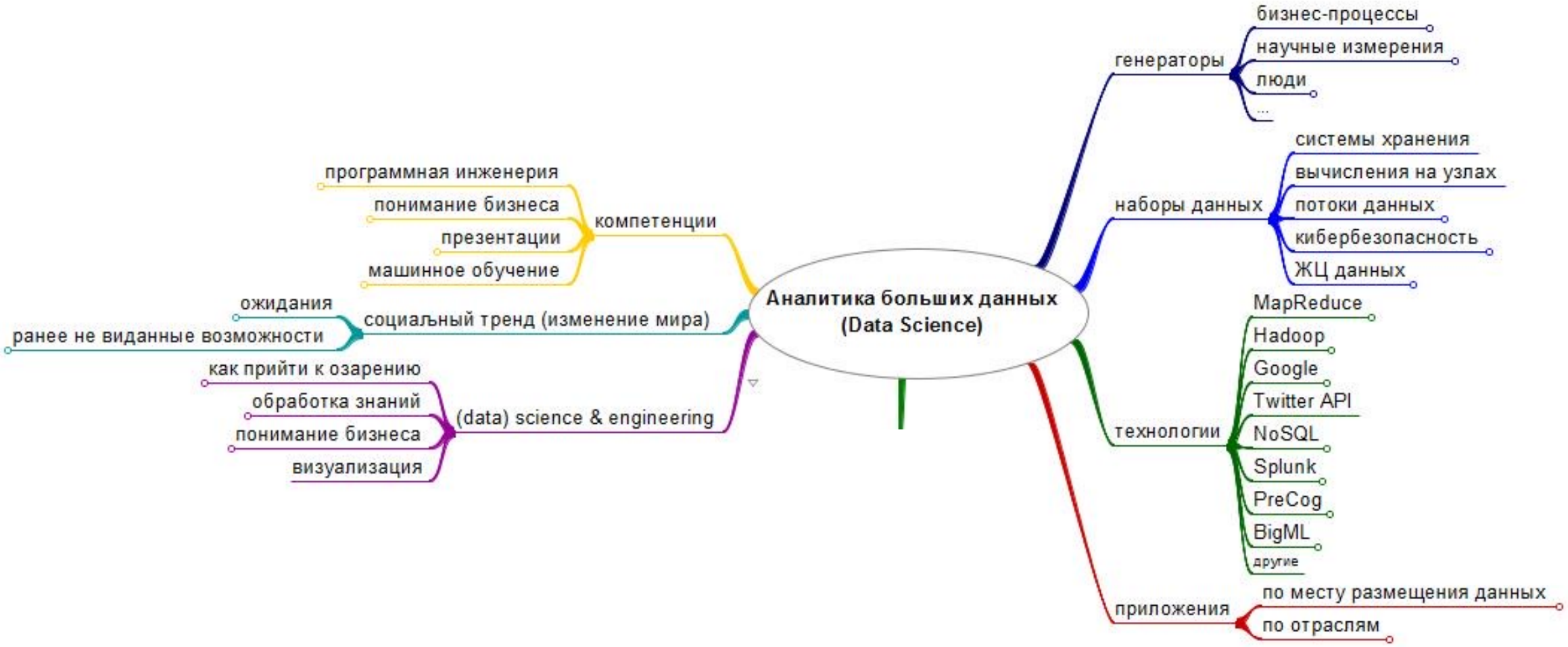
- Инженерия относится к профессиям **Data Engineer** и **администратор**. Такие специалисты занимаются поддержкой, созданием и настройкой программной и аппаратной инфраструктуры системы сбора, хранения и обработки информации, а также аналитикой массивов и информационных потоков, в том числе конфигурированием облачных (Cloud) и локальных кластеров.

Обоснование актуальности применения распределенных технологий

Для работы с большими данными, необходимо иметь базовые знания:

- Архитектуры компьютеров и серверов;
- Работы операционных систем и их взаимодействия с железом;
- СУБД (MySQL, Oracle, Postgres, Amazon Redshift, Microsoft Azure, Mongo, Hadoop, BigQuery или др.);
- По математическому анализу;
- По теории вероятностей и статистике.

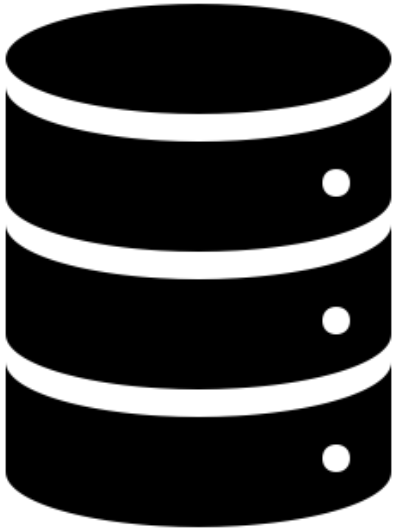
Обоснование актуальности применения распределенных технологий



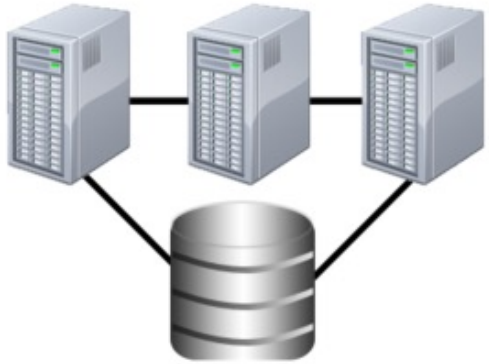
17

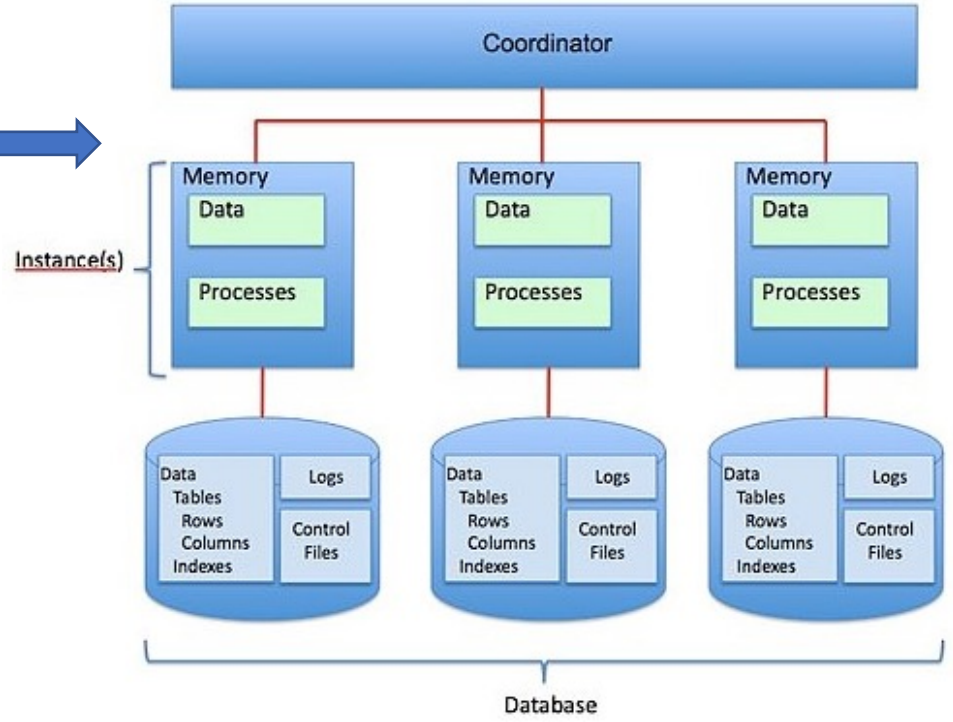
Обоснование актуальности применения распределенных технологий

Классический СУБД

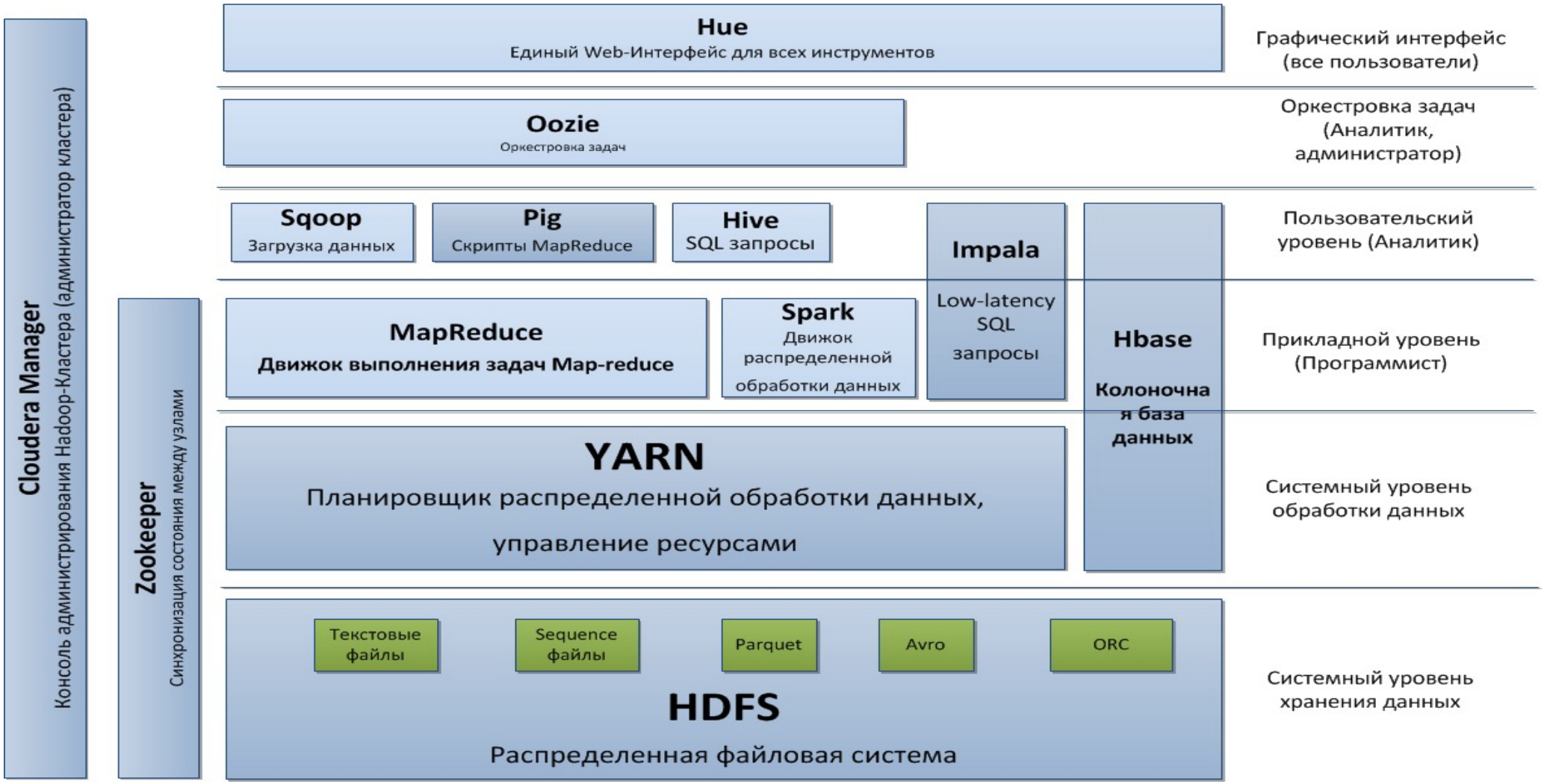


Горизонтально масштабируемые базы данных





Обоснование актуальности применения распределенных технологий



Кластер	несколько машин, соединенных друг с другом и взаимодействующих для выполнения общей работы.
Распределённые вычисления	способ решения трудоёмких вычислительных задач с использованием нескольких компьютеров, чаще всего объединённых в параллельную вычислительную систему.
Hadoop	это ПО с открытым исходным кодом, предназначенный для создания и запуска распределенных приложений, обрабатывающих большие объемы данных.

Отличительные особенности Hadoop

Доступность – Hadoop работает на крупных кластерах, собранных из стандартных компьютеров, или в вычислительном облаке, например на базе службы Elastic Compute Cloud (EC2), предлагаемой компанией Amazon.

Надежность – поскольку Hadoop должен работать на стандартном оборудовании, его архитектура разработана с учетом возможности частых отказов. Большинство отказов можно обработать так, что характеристики кластера будут ухудшаться постепенно.

Масштабируемость – Hadoop масштабируется линейно, то есть при увеличении объема данных достаточно добавить новые узлы в кластер.

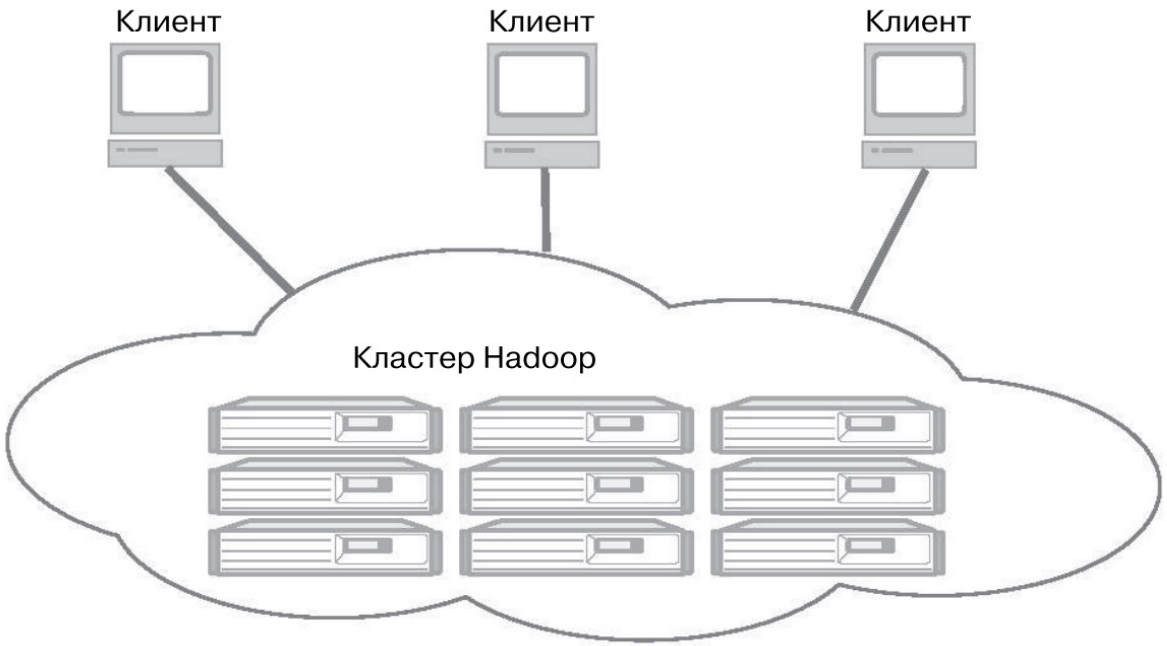
Простота – Hadoop позволяет пользователю быстро создавать эффективный параллельный код.

Введение в Hadoop

- **Кластер Hadoop** состоит из многих машин, которые хранят и параллельно обрабатывают большие наборы данных. Клиентские компьютеры посылают в это вычислительное облако задания и получают результаты.

- **Hadoop** представляет собой набор стандартных компьютеров, объединенных в сеть, физически расположенную в одном месте

- Как хранение, так и обработка данных происходят внутри этого **«облака»** машин.



Некоторые тезисы:

- СУБД на SQL ориентирован на работу со структурированными данными.
- Hadoop имеют дело как со структурированными данными так и с неструктурированными данными, например текстовыми. С этой точки зрения, Hadoop предлагает более общую парадигму, чем SQL.
- SQL и Hadoop могут дополнять друг друга, поскольку SQL – это язык запросов, который можно реализовать поверх Hadoop, выступающего в роли подсистемы исполнения.

Сравнение СУБД с Hadoop

Основные отличия:

- Масштабирование по горизонтали, а не по вертикали
- Можно хранить данные как Пары ключ/значение вместо реляционных таблиц
- Функциональное программирование (MapReduce) вместо декларативных запросов (SQL)
- Автономная пакетная обработка вместо оперативных транзакций

Большинство объектов физического мира неимоверно сложны по своей организации. Когда мы пытаемся описать какой-либо из таких объектов, мы на самом деле придумываем модель, соответствующую ему в нашем понимании. Если объекты можно поделить на некоторые группы, удовлетворяющие одинаковым моделям, то мы получаем ситуацию, когда внутри базы данных хранятся две группы сущностей:

- описания моделей объектов;
- записи, удовлетворяющие какой-либо из моделей и соответствующие различным представителям объектов.

Но бывают ситуации, когда объекты настолько различны, что их нельзя классифицировать. Тогда база данных представляет из себя набор из одних лишь моделей.

Когда мы говорим о моделях данных, мы должны понимать, что нужно уметь хранить не только сами объекты, но и взаимосвязи между ними. Существует несколько видов взаимосвязей между объектами:

- один к одному – 1:1;
- один ко многим – 1:M;
- многие ко многим – M:M.

Язык SQL

Язык SQL (Structured Query Language1) – это основной язык взаимодействия с информацией, размещённой в реляционной базе данных. В той или иной модификации язык SQL есть во всех основных СУБД, но для каждой из них характерен свой диалект языка.

Язык SQL позволяет создавать и манипулировать объектами реляционной БД. Например, SQL позволяет создавать таблицы, модифицировать их, наполнять строками, модифицировать строки, удалять строки и т.д.

Язык SQL

Название	Назначение	Примеры команд
DDL (Data Definition Language)	Набор действий языка SQL, отвечающих за определение данных. То есть команды создания/удаления/модификации объектов. Например, таблиц.	CREATE TABLE DROP TABLE ALTER TABLE RENAME TABLE
DML (Data Manipulation Language)	Набор действий языка SQL, отвечающих за манипуляцию с данными. То есть команды, отвечающие за наполнение объектов или выборку из них. Например, добавление, удаление, выборка или модификация строк таблиц.	SELECT INSERT UPDATE DELETE
DCL (Data Control Language)	Набор действий языка SQL, отвечающих за разграничение прав доступ. Например, разрешение на выборку строк из таблицы.	GRANT REVOKE

Полезные ресурсы

1. https://hadoop.apache.org
2. https://www.cloudera.com/products/open-source/apache-hadoop.html
3. https://spark.apache.org
4. https://www.youtube.com/watch?v=8tzCrau5JuE
5. https://www.youtube.com/watch?v=Rr-tBaFPEZY
6. https://docs.microsoft.com/en-us/previous-versions/sql/sql-server-2008-r2/ms187669(v=sql.105)?redirectedfrom=MSDN