CCIS كلية علوم الحاسب والمعلومات
COLLEGE OF COMPUTER & INFORMATION SCIENCES

جامعة الأمير سلطان
PRINCE SULTAN UNIVERSITY
PSU.1998

Data Management Technologies

IS465: Data Management and Governance

Outline

- Introduction
- Data Warehousing and Data Mart
- Data Lakes and Data Reservoirs
- Cloud Computing and Data Management
- Big Data and NoSQL Databases

Introduction

Data Management Technologies: An Overview

- Data management technologies refer to the tools and systems used to collect, store, organize, and manage data across various industries and applications.
- Effective data management is crucial for making informed decisions, optimizing processes, and ensuring data security and privacy.

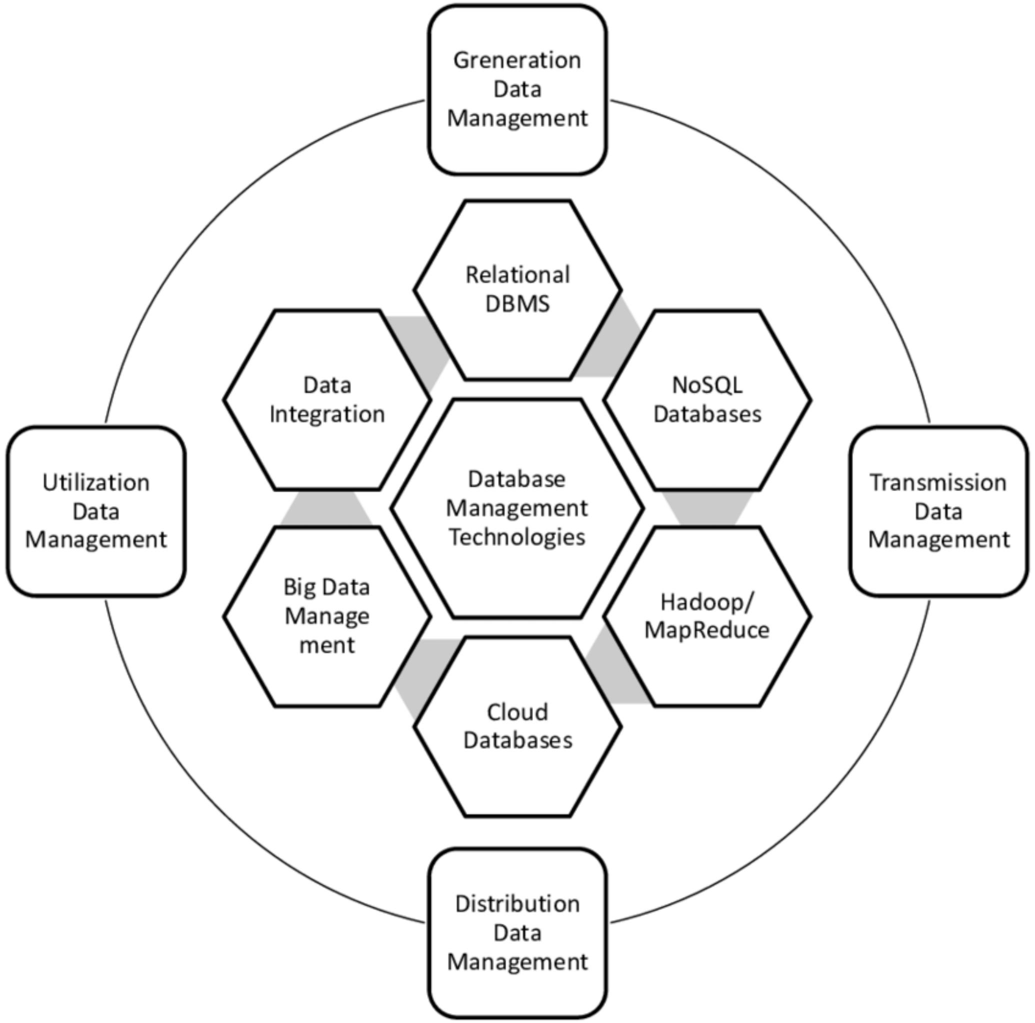
Types of Data Management Technologies

- Data storage technologies
  - Systems used to store and retrieve data, such as relational databases, NoSQL databases, and data warehouses. Examples include MySQL, MongoDB, and Amazon Redshift.
- Data processing technologies
  - Tools used to process, transform, and analyze data, such as data integration, data cleansing, and data mining. Examples include Apache NiFi, Talend, and RapidMiner.

Types of Data Management Technologies

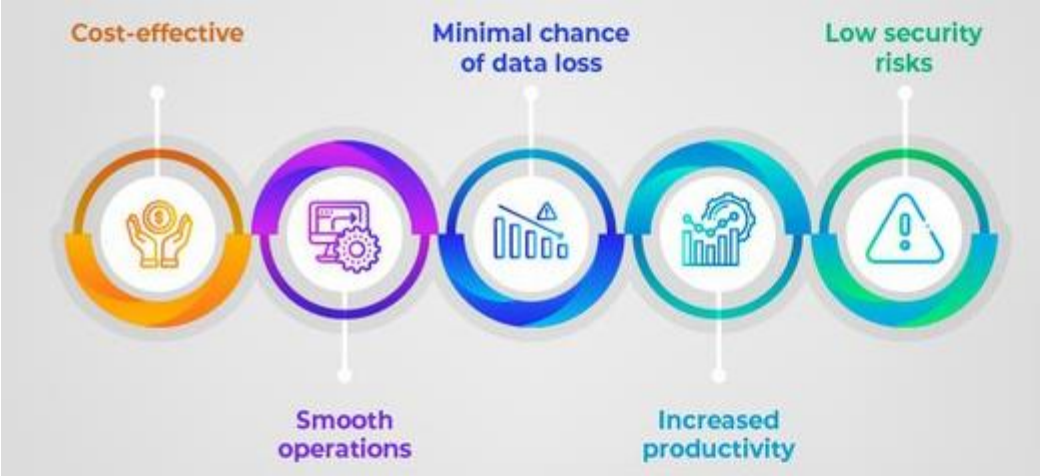
- Data governance technologies
  - Systems used to manage data quality, security, and privacy, such as data catalogs, data dictionaries, and data access control. Examples include Apache Atlas, AWS Lake Formation, and DataClarity.
- Data visualization technologies
  - Tools used to create interactive and intuitive visualizations of data, such as business intelligence, data analytics, and data dashboards. Examples include Tableau, Power BI, and QlikView.

Modern database management technologies



Importance of Data Management Technologies

- Data management technologies are critical for organizations to collect, store, process, and analyze large volumes of data.
- Data-driven decision making has become a norm in today's business landscape.
- Data management technologies help organizations to make informed decisions, improve operational efficiency, and stay competitive.



Improving Business Outcomes

- Data management technologies help organizations to:
  - Improve data quality and accuracy
  - Enhance data security and privacy
  - Increase operational efficiency and productivity
  - Improve customer experience and satisfaction
  - Make informed decisions based on data insights
  - Stay competitive in the marketplace

Improving Business Outcomes

- Data management technologies also enable organizations to:
  - Identify new business opportunities
  - Optimize business processes
  - Improve financial performance
  - Enhance strategic decision making

Types of Data Management Technologies

- Data storage technologies:
    - Relational databases (e.g., MySQL, Oracle)
    - NoSQL databases (e.g., MongoDB, Cassandra)
    - Data warehouses (e.g., Amazon Redshift, Google BigQuery)
    - Data lakes (e.g., Apache Hadoop, AWS S3)

- Data processing technologies:
    - Data integration (e.g., Talend, Informatica)
    - Data cleansing (e.g., Trifacta, DataClarity)
    - Data transformation (e.g., Apache Beam, AWS Lambda)
    - Data streaming (e.g., Apache Kafka, AWS Kinesis)

Types of Data Management Technologies

- Data governance technologies:
  - Data catalogs (e.g., Apache Atlas, AWS Lake Formation)
  - Data dictionaries (e.g., Apache Hive, AWS Glue)
  - Data access control (e.g., Apache Ranger, AWS IAM)

- Data analytics technologies:
  - Business intelligence (e.g., Tableau, Power BI)
  - Data visualization (e.g., D3.js, Matplotlib)
  - Predictive analytics (e.g., R, Python)
  - Machine learning (e.g., TensorFlow, PyTorch)

Key Features and Use Cases

Technology	Key Features	Use Cases
Relational databases	SQL querying, data consistency, transaction management, data security	- Traditional data storage and management - Online transactions and e-commerce - Enterprise data management
NoSQL databases	Scalability, flexible schema, high availability, big data analytics	- Big data storage and processing - Real-time web analytics - IoT data management
Cloud storage	Scalability, cost-effectiveness, data accessibility, collaboration	- File sharing and collaboration - Backup and archiving - Cloud-based data storage
Data warehousing	Data integration, data cleansing, data transformation, data mining	- Data analysis and reporting - Business intelligence - Data-driven decision making
Data lakes	Data storage, data processing, data analytics, data visualization	- Big data storage and processing - Data science and machine learning - Data-driven decision making

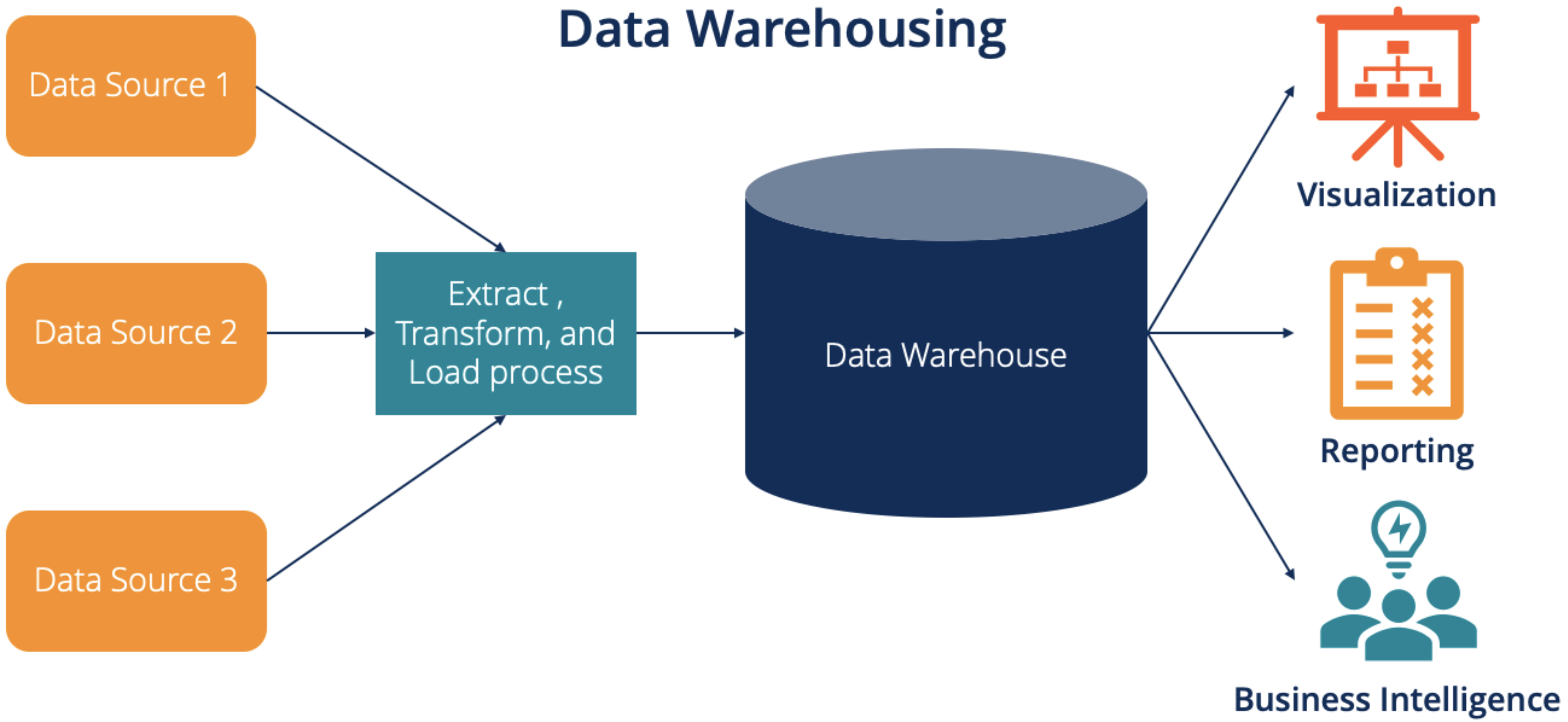
Key Features and Use Cases

Technology	Key Features	Use Cases
Data governance	Data quality, data security, data compliance, data accessibility	- Data management and oversight - Data risk management - Data privacy and protection
Master data management	Data integration, data cleansing, data transformation, data security	- Data management and oversight - Data quality and consistency - Data-driven decision making
Data integration	Data integration, data transformation, data mapping, data validation	- Data migration and integration - Data synchronization and replication - Data integration for analytics
Data quality	Data validation, data cleansing, data normalization, data enrichment	- Data quality and consistency - Data accuracy and completeness - Data-driven decision making
Data security	Data encryption, data access controls, data authentication, data backup	- Data protection and privacy - Data risk management - Compliance and regulatory requirements
Data backup	Data backup and recovery, data archiving, data retention, data restore	- Data protection and recovery - Data backup and archiving - Compliance and regulatory requirements
Data analytics	Data visualization, data mining, data predictive analytics, data prescriptive analytics	- Data-driven decision making - Business intelligence - Predictive and prescriptive analytics

Data Warehousing and Data Mart

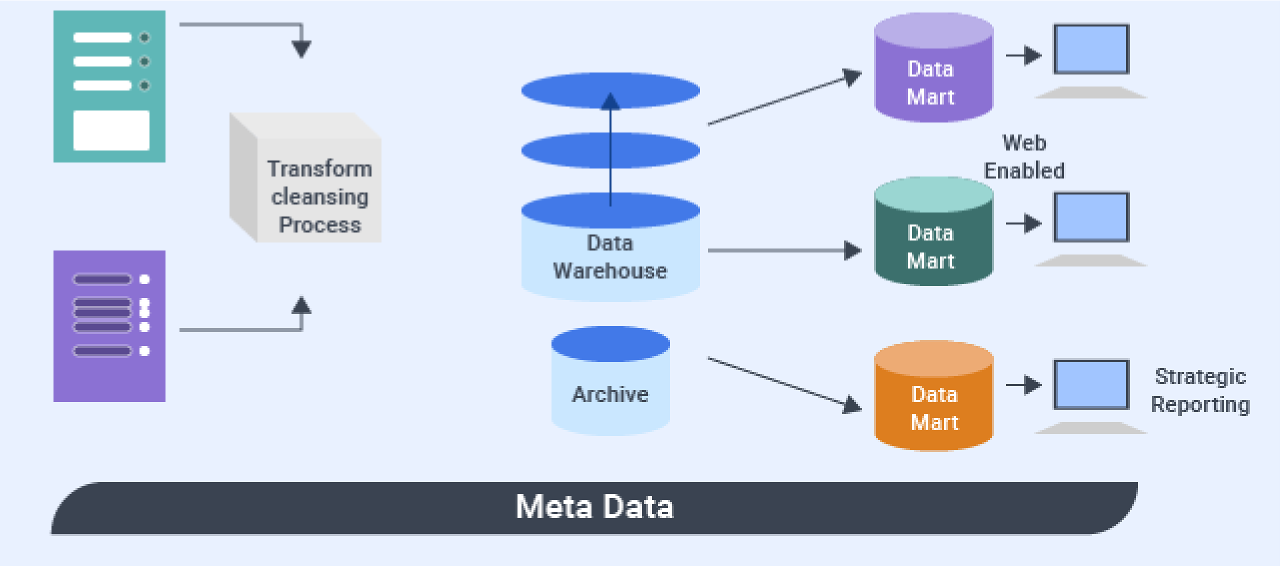
Understanding the Basics

- Data warehousing is a process of collecting and storing data from various sources in a centralized repository



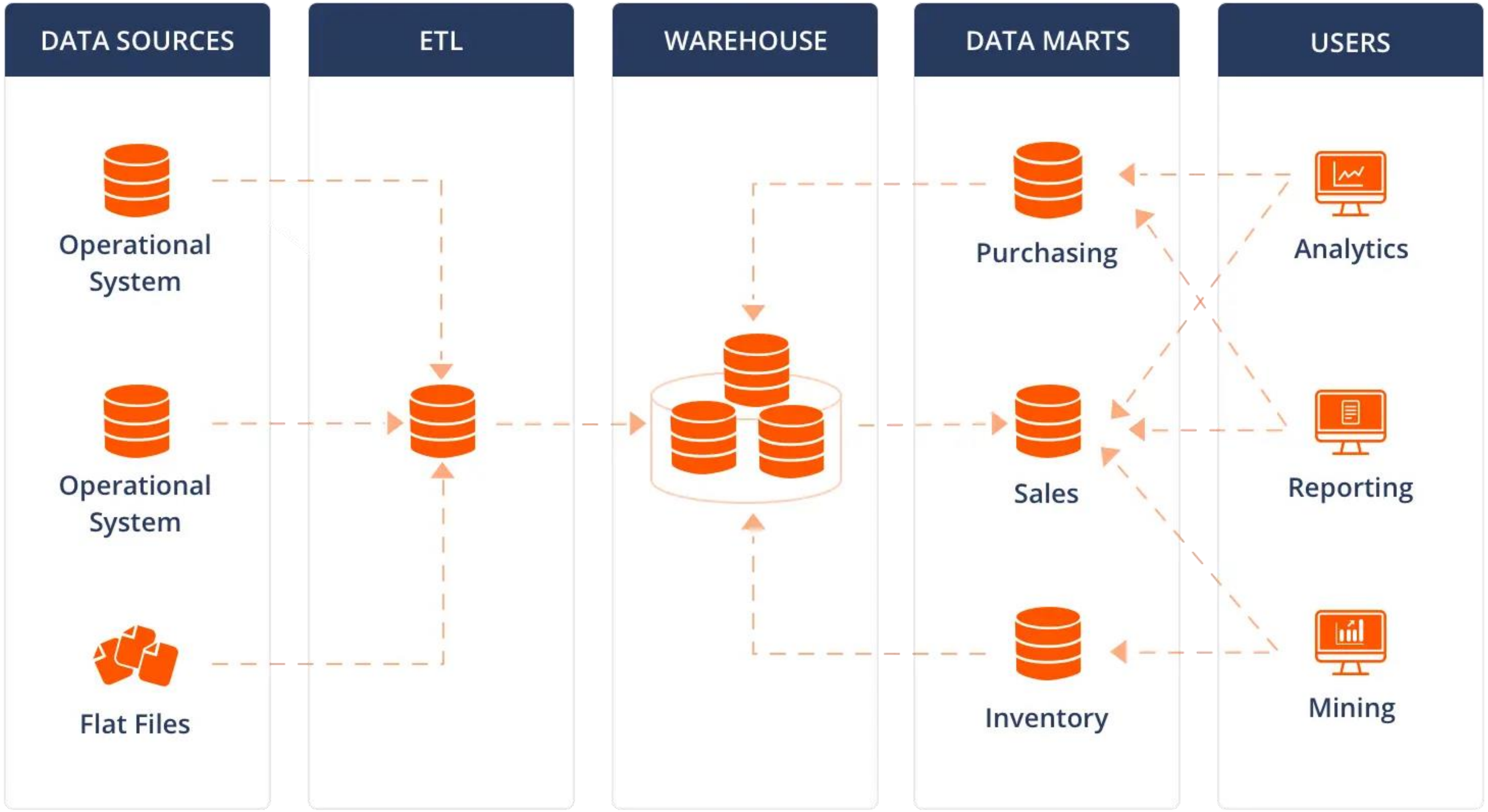
Understanding the Basics

- Data mart is a subset of a data warehouse that serves a specific business function or department

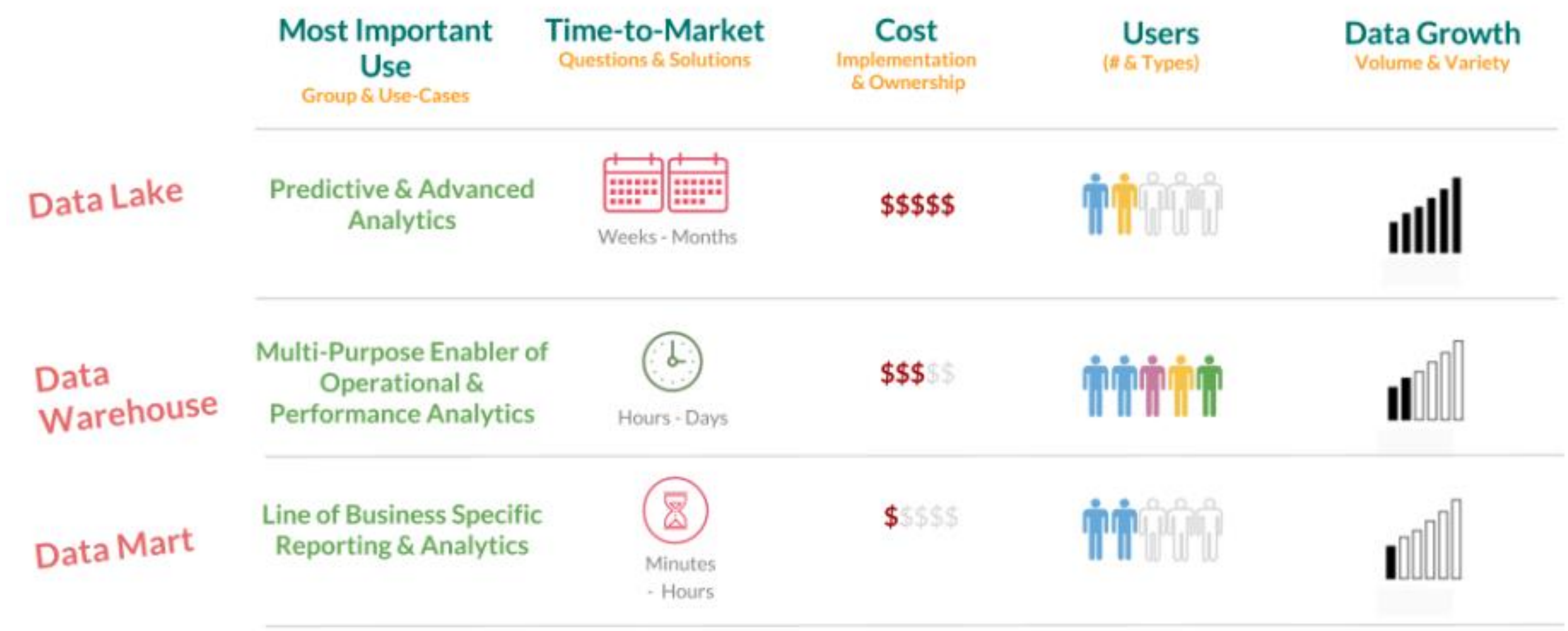


Understanding the Basics

- Both data warehousing and data mart are used for data analysis and reporting



Lake, Warehouse, Mart



	Most Important Use Group & Use-Cases	Time-to-Market Questions & Solutions	Cost Implementation & Ownership	Users (# & Types)	Data Growth Volume & Variety
Data Lake	Predictive & Advanced Analytics	Weeks - Months	$$$$$	👤👤○○○	▮▮▮▮▮
Data Warehouse	Multi-Purpose Enabler of Operational & Performance Analytics	Hours - Days	$$$$$	👤👤👤👤👤	▮▮▯▯▯
Data Mart	Line of Business Specific Reporting & Analytics	Minutes - Hours	$$$$$	👤👤○○○	▮▯▯▯▯

A Centralized Repository for Data

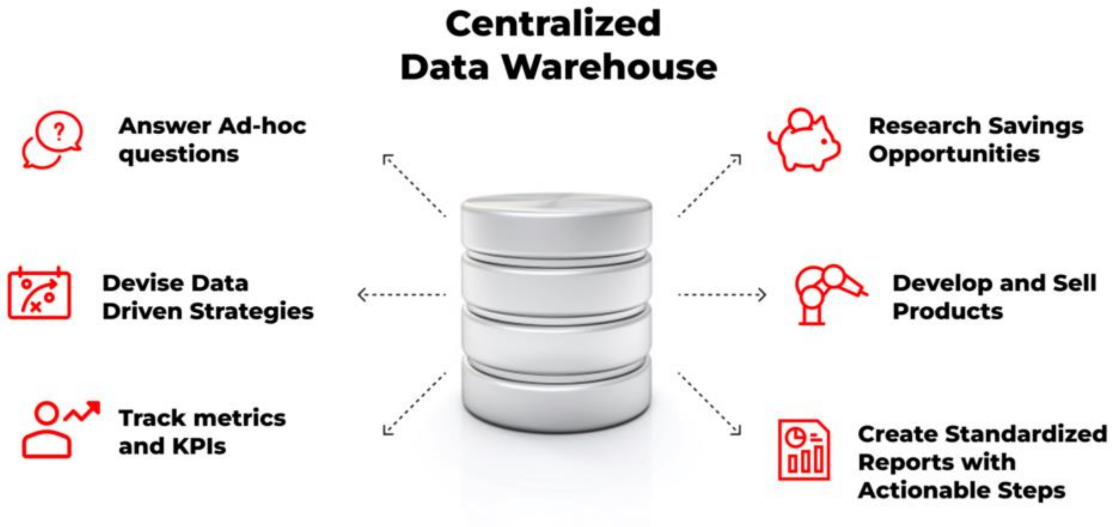
- Data warehousing is a process of collecting and storing data from various sources in a centralized repository
- Data warehouse is a large, centralized repository that stores data from various sources, such as transactional databases, log files, and external data sources
- Data warehouse is designed to handle large volumes of data and support complex queries and analytical tasks
- Data warehouse is used for data analysis, reporting, and business intelligence

A Subset of a Data Warehouse

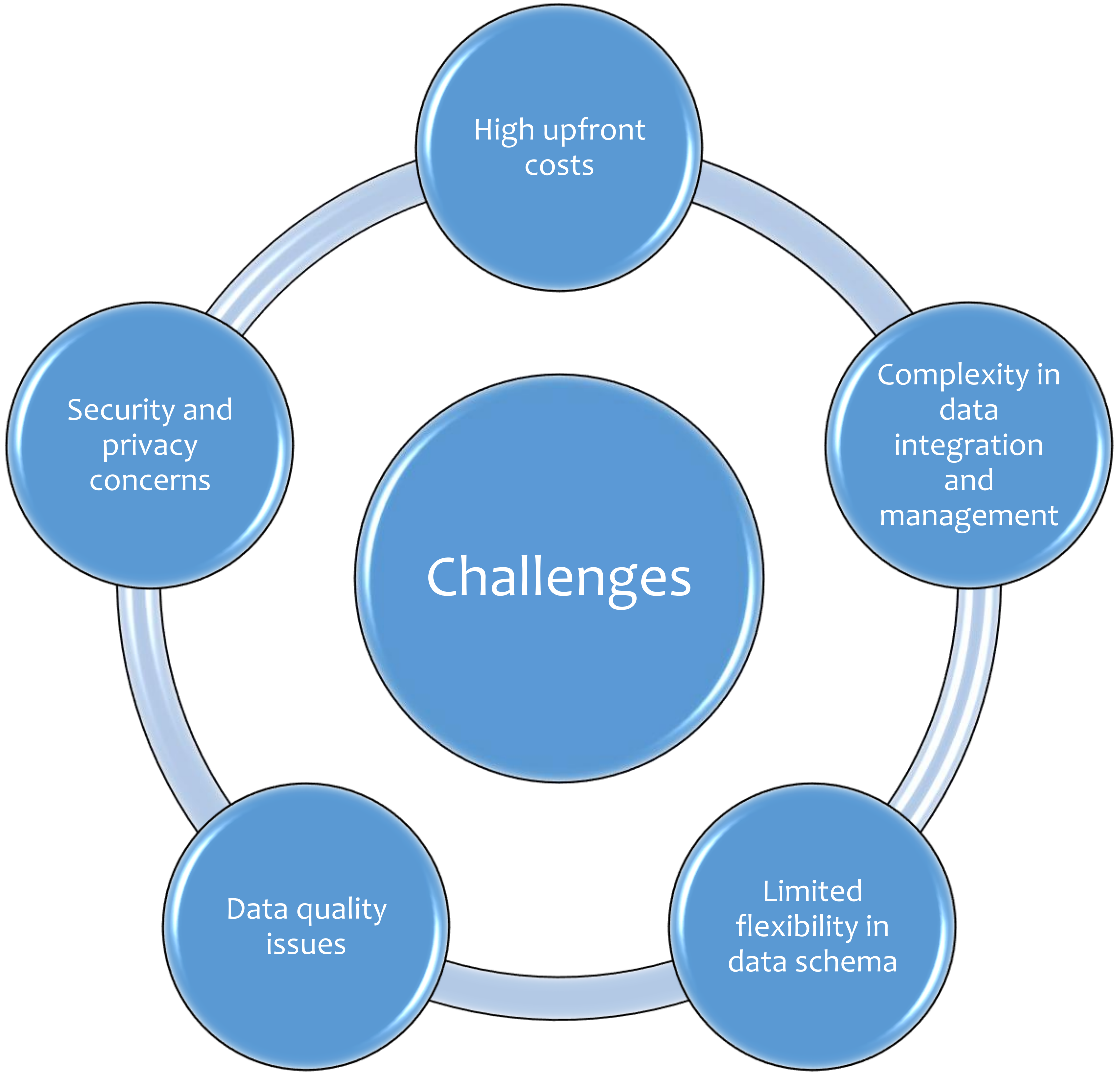
- Data mart is a subset of a data warehouse that serves a specific business function or department
- Data mart is designed to meet the needs of a specific business area or team, such as sales, marketing, or finance
- Data mart is a smaller, more focused repository that stores data relevant to a specific business function or department
- Data mart is used for data analysis and reporting within a specific business area or department

Benefits of Centralized Data Storage

- Improved data quality and consistency
- Enhanced data analysis and reporting capabilities
- Better data-driven decision making
- Improved data security and compliance
- Scalability and flexibility


Centralized Data Warehouse
Answer Ad-hoc questions
Devise Data Driven Strategies
Track metrics and KPIs
Research Savings Opportunities
Develop and Sell Products
Create Standardized Reports with Actionable Steps

Challenges of Centralized Data Storage



Challenges

High upfront costs

Complexity in data integration and management

Limited flexibility in data schema

Data quality issues

Security and privacy concerns

Benefits of Focused Data Storage

- Improved data analysis and reporting for specific business areas
- Better data management and organization
- Increased data accessibility and usability
- Reduced data complexity and cost
- Improved data freshness and timeliness

Challenges of Focused Data Storage

- Limited data scope and coverage
- Data duplication and inconsistency
- Lack of data integration and standardization
- Data security and privacy concerns
- Limited scalability and flexibility

Industry Examples

- Healthcare: patient data, medical records, and clinical trials
- Finance: transactional data, financial performance, and risk management
- Retail: customer data, sales data, and inventory management
- Manufacturing: production data, supply chain data, and quality control

Departmental Examples

- Sales: customer data, sales data, and sales performance
- Marketing: customer data, marketing campaigns, and lead generation
- Finance: financial data, budgeting, and forecasting
- Human Resources: employee data, performance management, and training

Data Warehousing Tools

Summary of Key Points

- Data warehousing and data mart are important concepts in data management
- Both have their advantages and disadvantages
- Use cases for data warehousing and data mart vary by industry and department
- Choosing the right tool depends on the organization's needs and other factors such as suitability, cost and familiarity.

Data Lakes and Data Reservoirs

Data Lakes and Data Reservoirs

- Data lakes and data reservoirs are data storage and management solutions
- They are designed to handle large amounts of data from various sources
- They are used for data processing, analysis, and reporting

Data Lakes

- A data lake is a centralized repository that stores raw, unprocessed data
- Data lakes are designed to handle large amounts of data and can scale horizontally
- Data lakes are schema-on-read, meaning the schema is defined when the data is queried
- Data lakes are ideal for data warehousing, big data analytics, and data science

Data Reservoirs

- A data reservoir is a repository that stores processed data
- Data reservoirs are designed to handle smaller amounts of data and are optimized for query performance
- Data reservoirs are schema-on-write, meaning the schema is defined when the data is ingested
- Data reservoirs are ideal for operational reporting, data visualization, and real-time analytics

When to Use Each

- Data lakes are suitable for use cases such as:
  - Data warehousing and big data analytics
  - Data science and machine learning
  - Data integration and data transformation
  - Data archiving and data retention
- Data reservoirs are suitable for use cases such as:
  - Operational reporting and data visualization
  - Real-time analytics and dashboarding
  - Data integration and data transformation
  - Data quality and data governance

Data Lakes Pros and Cons

- Pros:
  - Scalability
  - Flexibility
  - Data Integration
  - Data Agility
  - Cost-effective
- Cons:
  - Complexity
  - Data Quality
  - Data Security
  - Data Governance
  - Cost

Advantages and Disadvantages of Data Reservoirs

- Advantages:
  - Data reservoirs can handle large amounts of data and scale horizontally as needed
  - Data reservoirs can store data in various formats and schema, making it easier to handle different data sources
  - Data reservoirs can integrate data from multiple sources, creating a single source of truth
  - Data reservoirs can be more cost-effective than traditional data storage solutions, especially for large amounts of data
- Disadvantages:
  - Data reservoirs can be complex to set up and manage, requiring specialized skills and resources
  - Data reservoirs can store data of varying quality, which can impact the accuracy of insights and analytics
  - Data reservoirs can be challenging to govern, making it difficult to ensure data accuracy, completeness, and compliance
  - Data reservoirs can be vulnerable to security breaches, especially if not properly secured

Real-World Examples of Data Lake Implementations

- There are various industries that leverage data lake implementations to gain insights from their data.
  - **Coca-Cola Andina**
    - They built a data lake on AWS to increase their analytics team productivity by 80%
  - **Adobe Systems**
    - They implemented a data lake infrastructure to manage their analytics data. "Adobe Experience Cloud," is built on Amazon Web Services (AWS).

Use Cases

- Data Lakes
  - Data warehousing and big data analytics
  - Data science and machine learning
  - Data integration and data transformation
  - Data archiving and data retention
  - Real-time analytics and stream processing

Use Cases

- Data Reservoirs
  - Operational reporting and data visualization
  - Real-time analytics and dashboarding
  - Data integration and data transformation
  - Data quality and data governance
  - Customer 360 and personalization

Comparing Data Lakes and Data Reservoirs

	Data Lakes	Data Reservoirs
Purpose	store raw, unprocessed data in its native format, usually for data warehousing and big data analytics	store processed, transformed, and cleaned data, usually for data integration and data sharing
Data Structure	use a flat, schema-on-read structure, which means the data is stored in a flat file system or database, and the schema is defined when the data is queried	use a hierarchical, schema-on-write structure, which means the data is stored in a structured format, such as a relational database, and the schema is defined when the data is ingested
Data Processing	batch processing and are often used for big data analytics, machine learning, and data science workloads	real-time or near-real-time data processing and are often used for operational data stores, data integration, and data sharing

Data Lake Tools

APACHE
hadoop
™

AWS
Lake
Formation

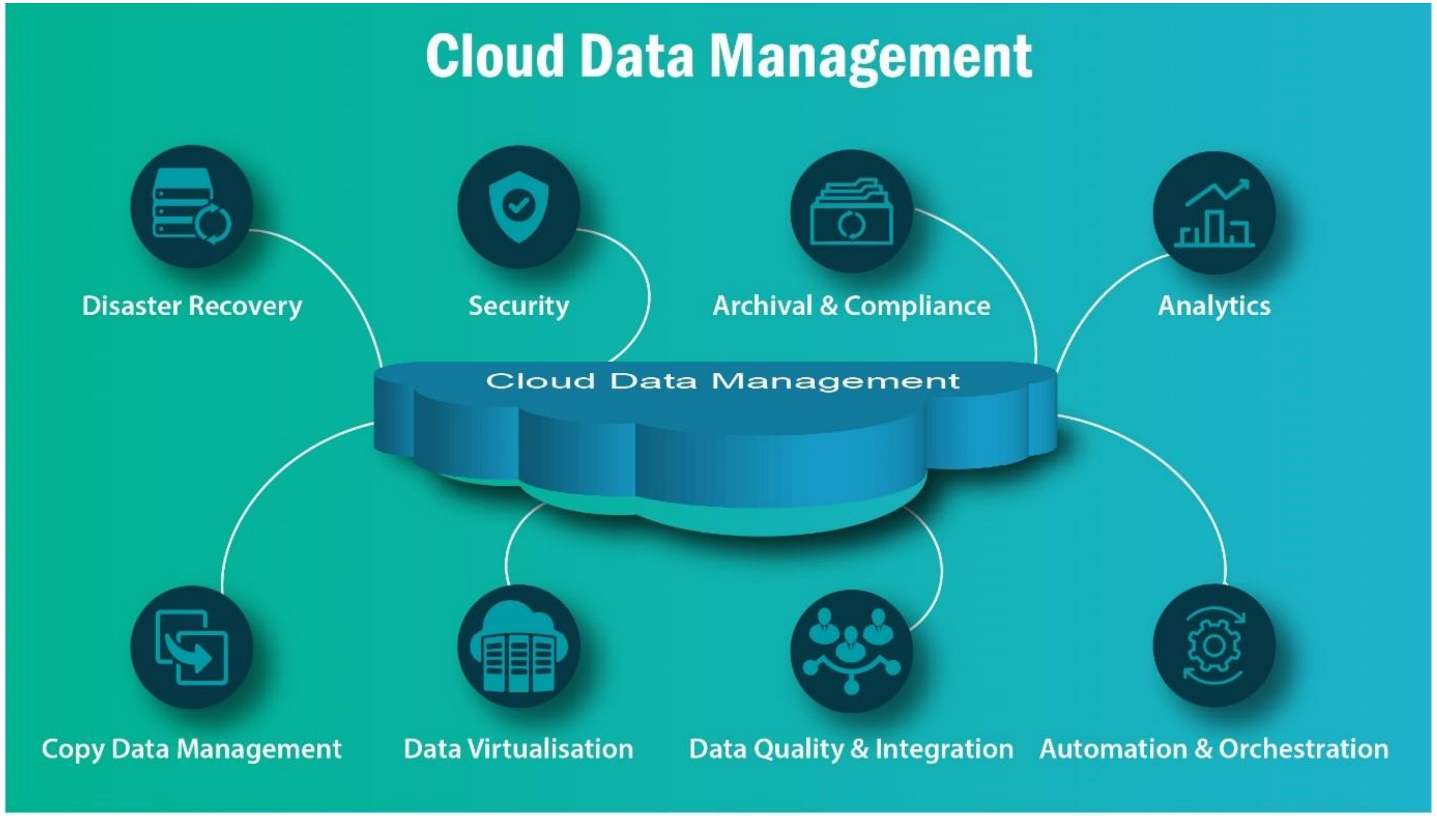
APACHE
Spark
™

Azure Databricks

Cloud Computing and Data Management

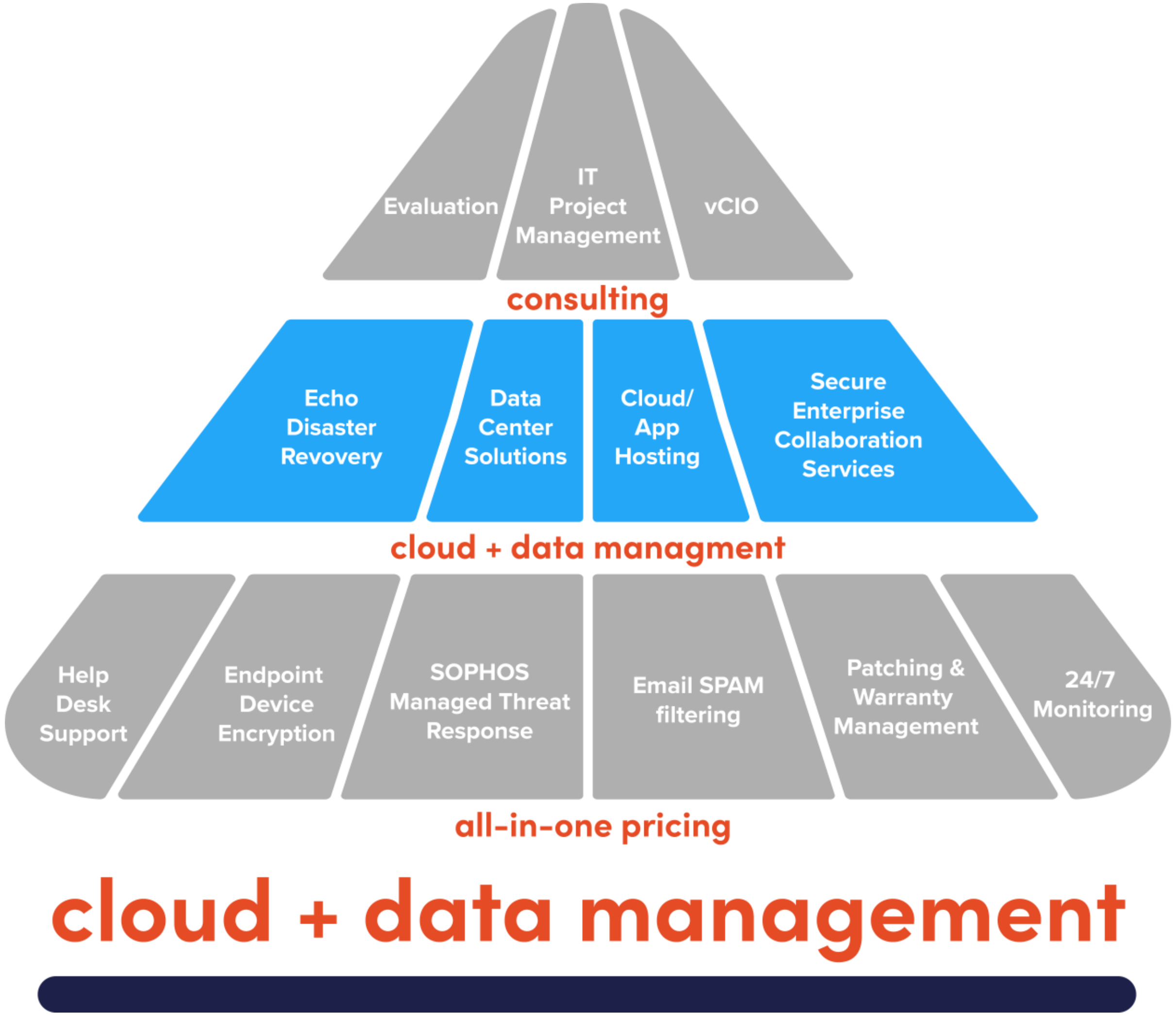
An Overview

- Cloud computing: a model for delivering computing services over the internet



Importance of data management in cloud computing

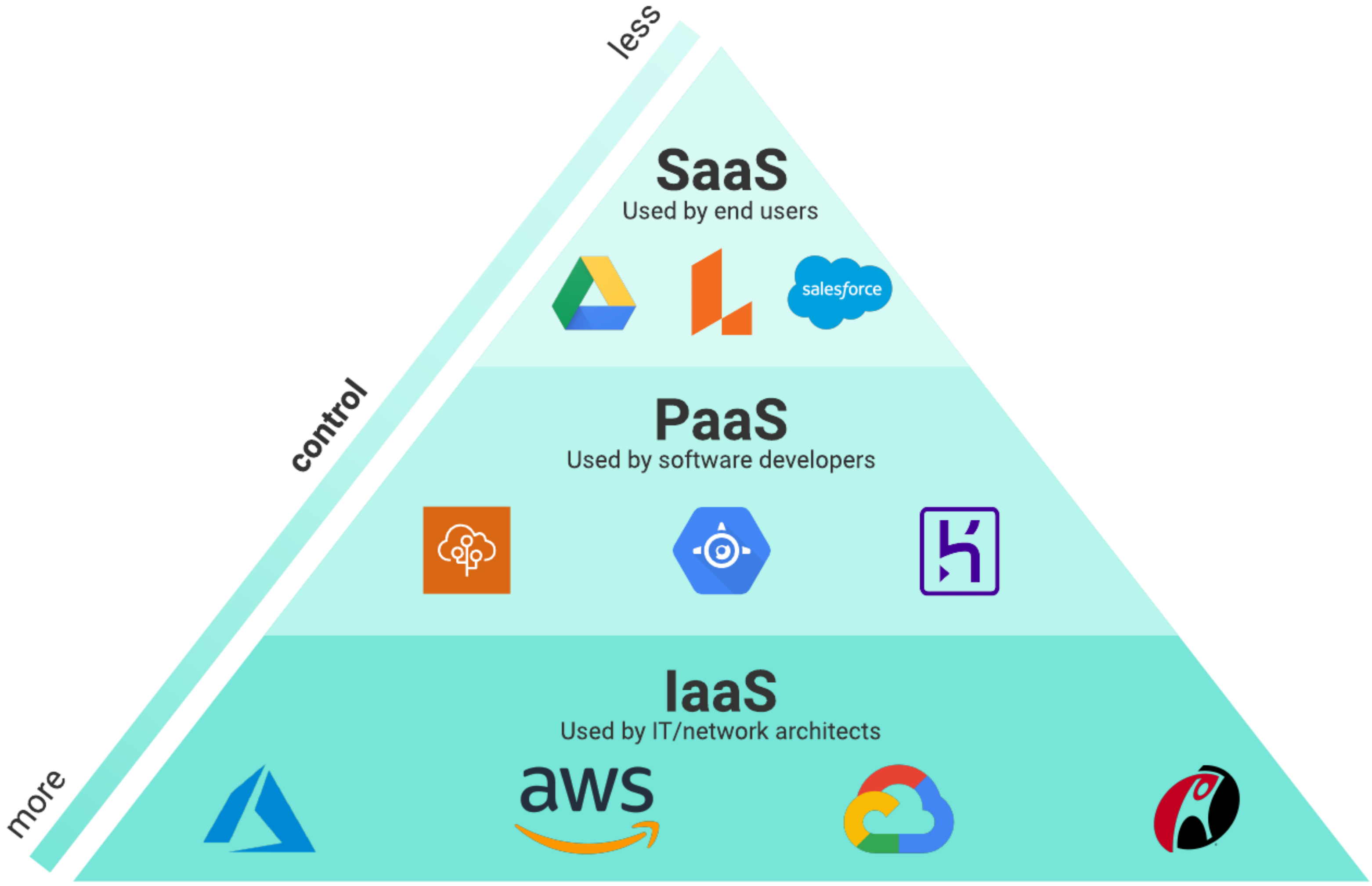
- Data Security
- Data Compliance
- Data Availability
- Data Integrity
- Data Governance
- Cost Optimization



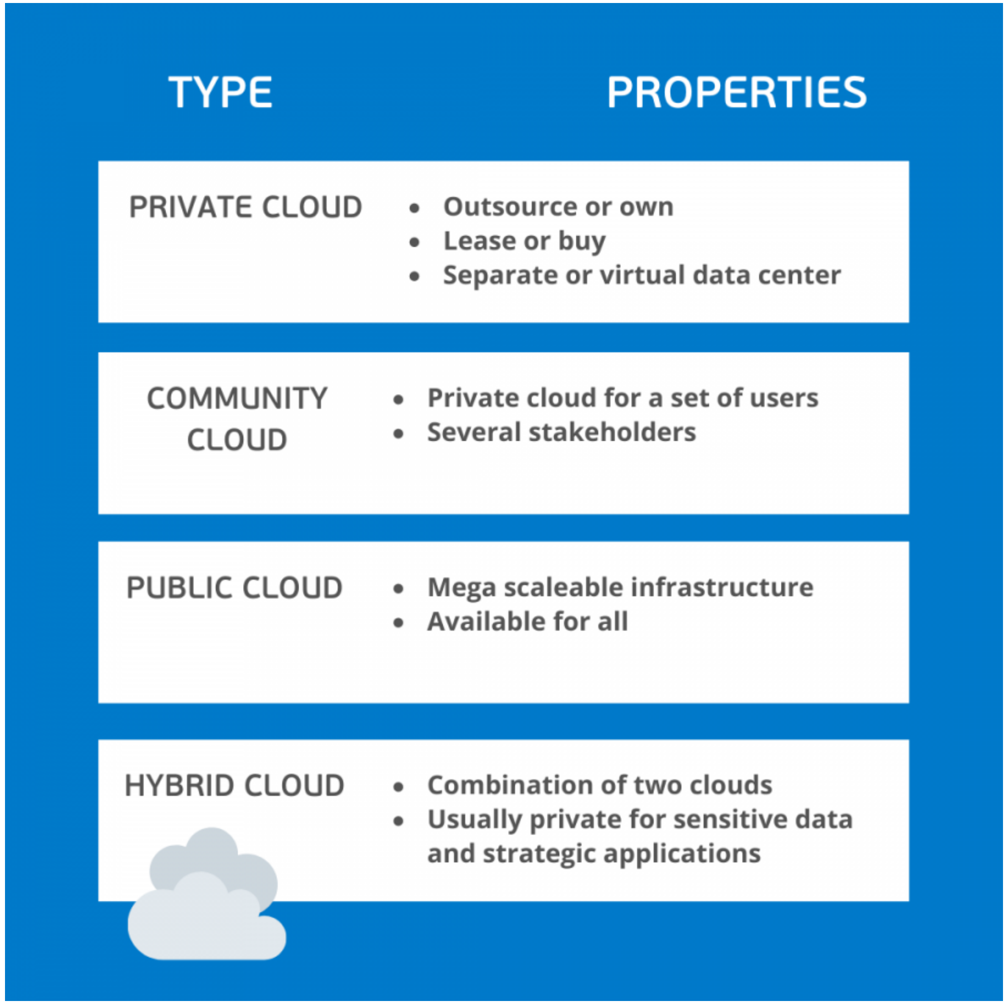
A Closer Look

- Cloud computing is a model for delivering computing services over the internet
- Resources such as servers, storage, databases, software, and applications are provided as a service
- Users can access and use these resources on-demand and pay only for what they use

Cloud Computing Service Models



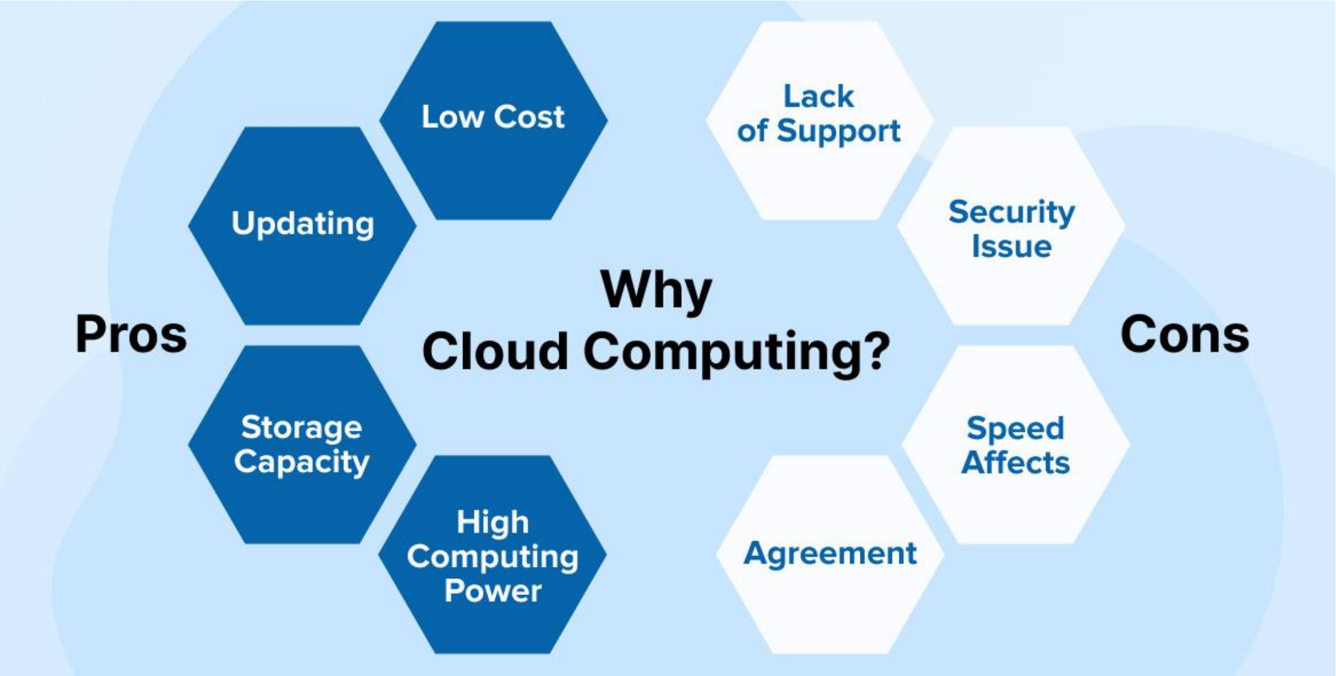
Cloud Computing Deployment Models



TYPE	PROPERTIES
PRIVATE CLOUD	• Outsource or own • Lease or buy • Separate or virtual data center
COMMUNITY CLOUD	• Private cloud for a set of users • Several stakeholders
PUBLIC CLOUD	• Mega scaleable infrastructure • Available for all
HYBRID CLOUD	• Combination of two clouds • Usually private for sensitive data and strategic applications

Cloud Computing for Data Management

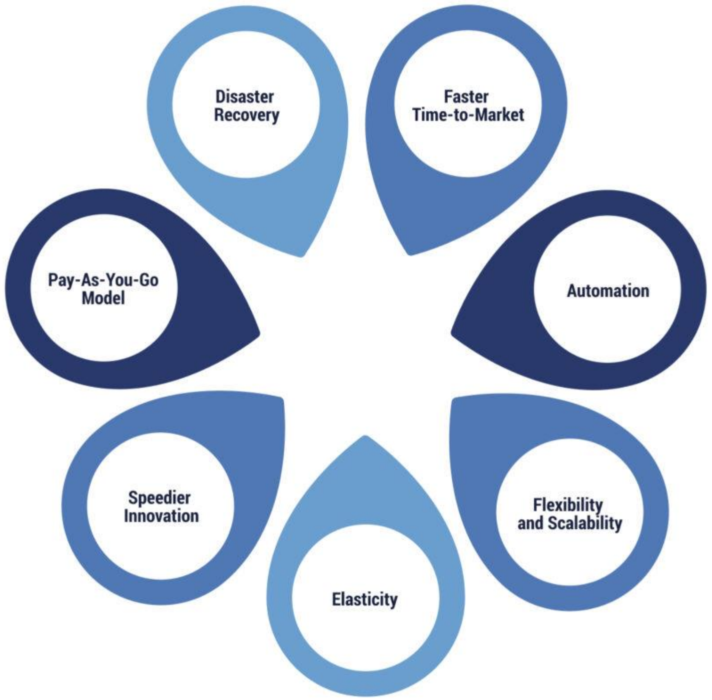
- Pros
  - Scalability, Flexibility, Cost-Effectiveness
- Cons
  - Security, Compliance, Data Sovereignty



Example of Cloud Computing Platforms

Azure
amazon web services
IBM Cloud
Google Cloud Platform
Alibaba Cloud
WIRE19

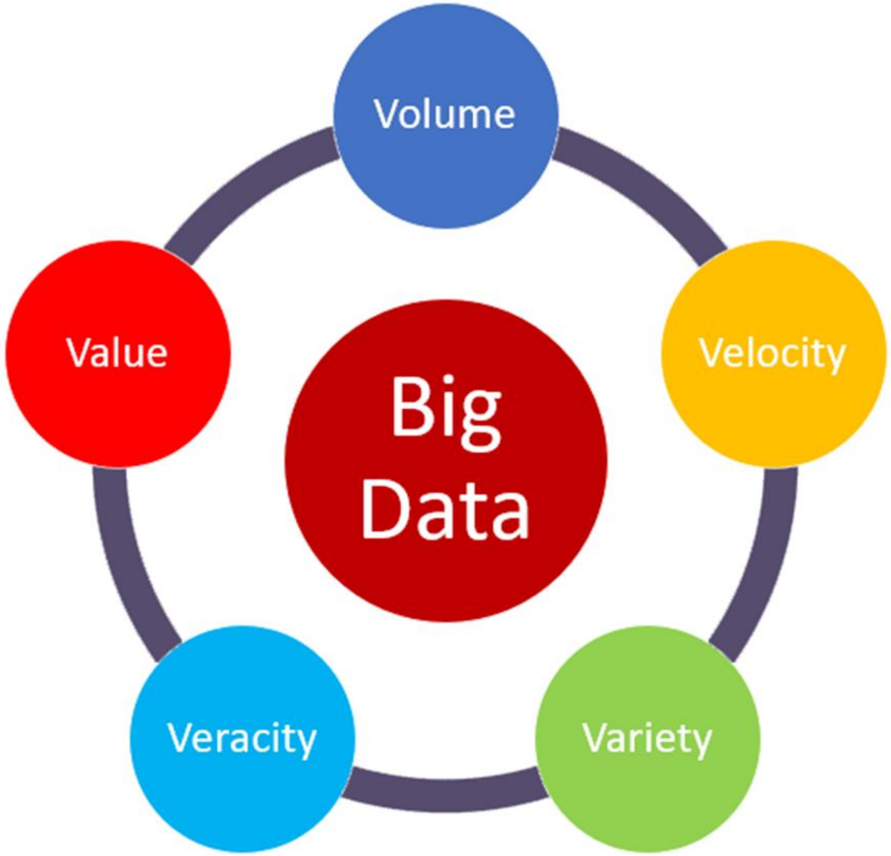
Agility, Innovation, Collaboration



Big Data and NoSQL Databases

An Overview

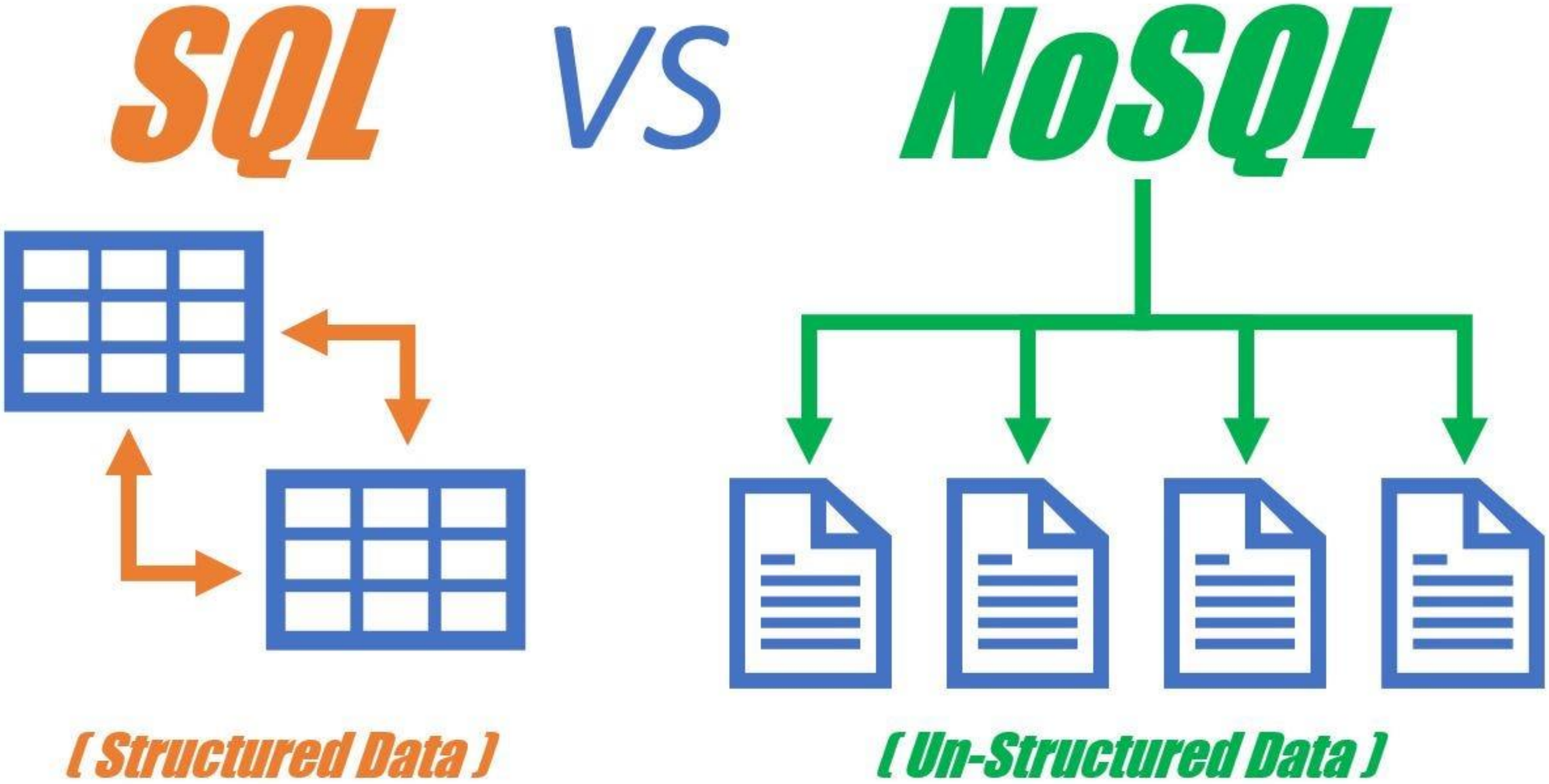
- Big data
  - large volumes of data that cannot be processed by traditional data processing tools



A Closer Look

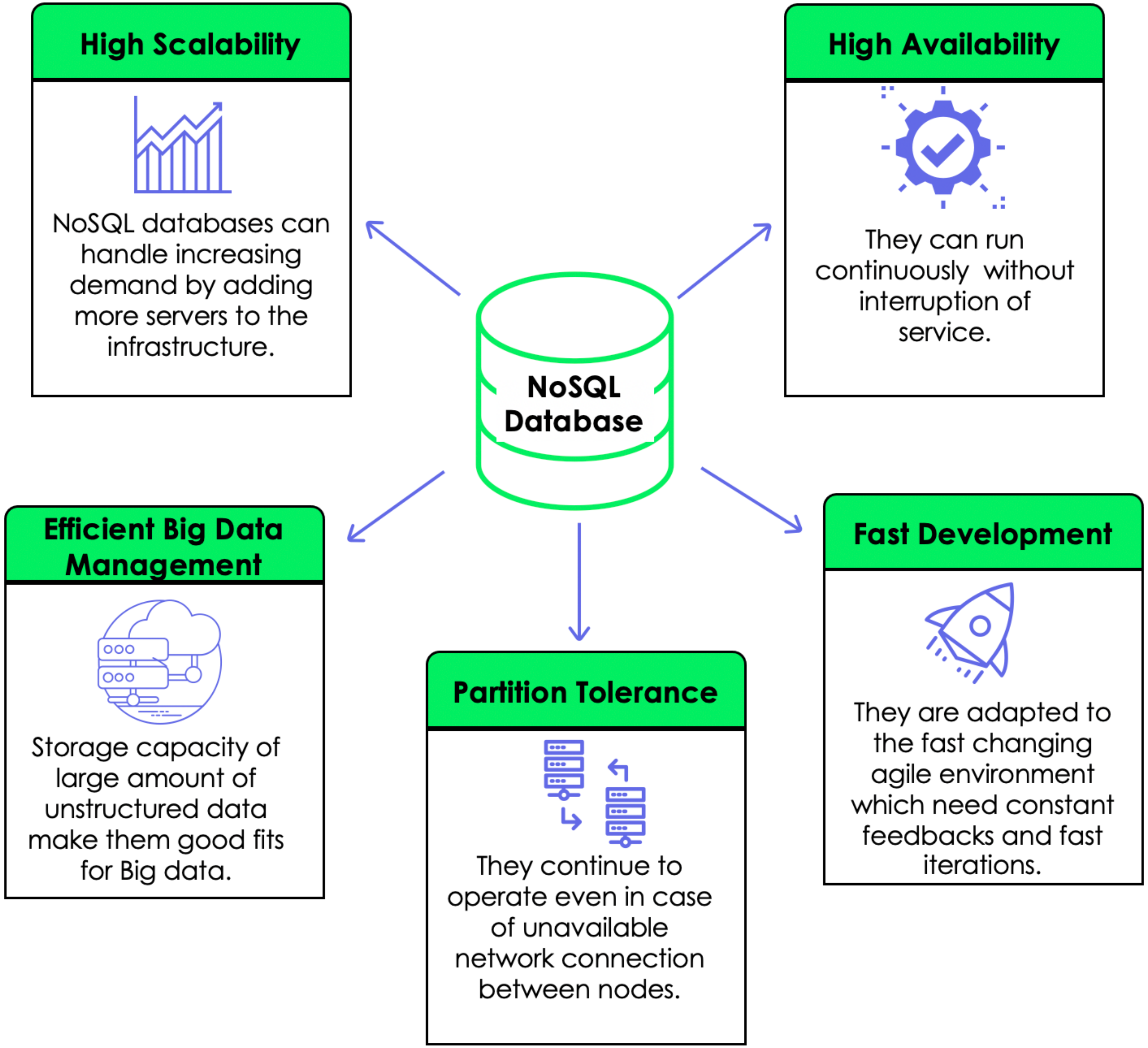
- Volume: large amounts of data generated from various sources
- Variety: diverse data types, including structured, semi-structured, and unstructured data
- Velocity: high speed of data generation and processing
- Veracity: data quality and accuracy
- Value: data must provide business value

SQL vs NoSQL



A Different Approach to Data Storage

- NoSQL databases are designed to handle large amounts of unstructured and semi-structured data
- They are horizontally scalable, allowing for easy data distribution and processing
- They are schema-less, allowing for flexibility in data storage and retrieval



NoSQL Database

High Scalability

NoSQL databases can handle increasing demand by adding more servers to the infrastructure.

High Availability

They can run continuously without interruption of service.

Efficient Big Data Management

Storage capacity of large amount of unstructured data make them good fits for Big data.

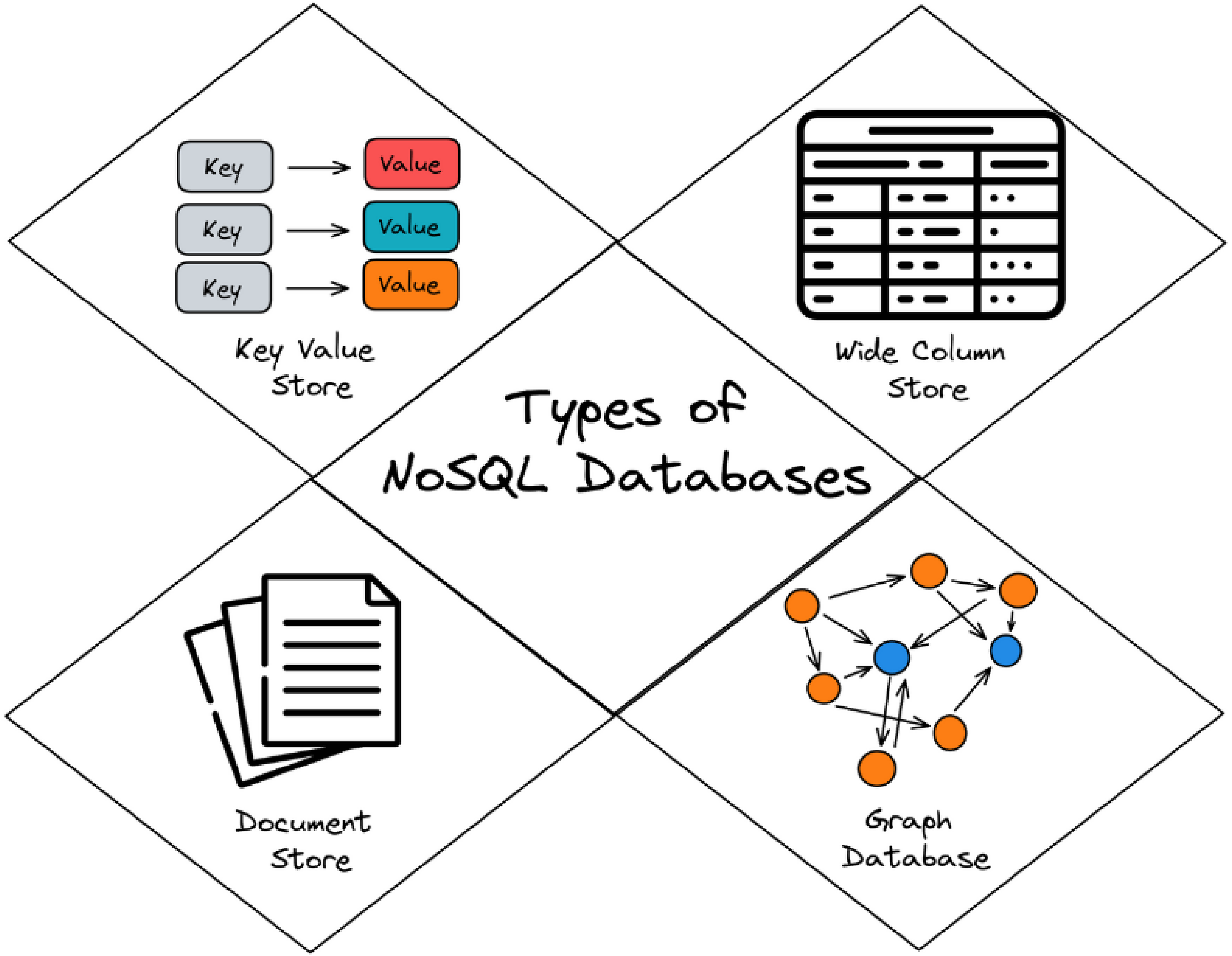
Partition Tolerance

They continue to operate even in case of unavailable network connection between nodes.

Fast Development

They are adapted to the fast changing agile environment which need constant feedbacks and fast iterations.

Types of NoSQL Databases



Types of NoSQL Databases

- Key-value databases
  - store data as a collection of key-value pairs
- Document databases
  - store data in the form of documents, such as JSON or XML
- Column-family databases
  - store data in a column-family format, similar to a relational database
- Graph databases
  - store data in the form of nodes and edges, useful for social networks and recommendation systems

Use Cases for Big Data and NoSQL Databases

- IoT data processing and analysis
- Social media data analysis and sentiment analysis
- Fraud detection and prevention
- Recommendation systems and personalization
- Predictive maintenance and machine learning

NoSQL Databases

- E-commerce Applications
- Healthcare
- Social Media Platforms
- Transportation
- Real-World Examples:
  - Netflix: store customer profiles, viewing histories, and content recommendations.
  - Uber: ride-sharing platform, managing driver and rider profiles, trip histories, and real-time location data.
  - Airbnb: handle its extensive listings and user interactions data1.

Example of Big Data and NoSQL Database Tools

APACHE
hadoop
™

APACHE
HBASE

Comparison of NoSQL databases

Feature	Hadoop	MongoDB	Cassandra	Neo4j	HBase
Data Model	Binary, key-value pairs	Document-oriented	Column-oriented	Graph-based	Column-family based
Data Processing	MapReduce	MongoDB query language	CQL (Cassandra Query Language)	Cypher (Graph query language)	HBase query language
Data Indexing	Secondary indexes	Indexing	Secondary indexes	Indexing	Indexing