18. DATA LAKE VS. DATA WAREHOUSE

ADS ISE II

–PARSHWA HERWADE (22510064)

Data lake vs. data warehouse

- Introduction & Context
- Definition and Purpose
- Architectural Overview
- Key Differences & Decision Factors
- Pros, Cons, and Use Cases
- Emerging Trends & Hybrid Approaches
- Summary & Conclusion
- References & Further Reading

Literature Survey

- Data Quality & Structure:

  Warehouses: Enforce clean, structured data via rigorous ETL.

  Lakes: Ingest raw data with flexible schemas, requiring robust metadata management.

- Query Performance:

  Warehouses: Optimized with pre-aggregated data and indexing for rapid, predictable queries.

  Lakes: Rely on distributed processing (e.g., Apache Spark) to handle complex queries over diverse data.

- Scalability & Flexibility:

  Warehouses: Effective in regulated environments but can be resource - intensive to scale.

  Lakes: Offer scalable, cost-effective storage for large, varied data sets.

- Industry Adoption:

  Warehouses are well-established in traditional BI environments.

  Lakes and emerging hybrid models (Lakehouses) are validated by recent research in big data analytics.

Introduction & Context

- **Background:**

I. Rapid data growth from IoT, social media, and enterprise applications.

2. Increasing need to manage structured and unstructured data.

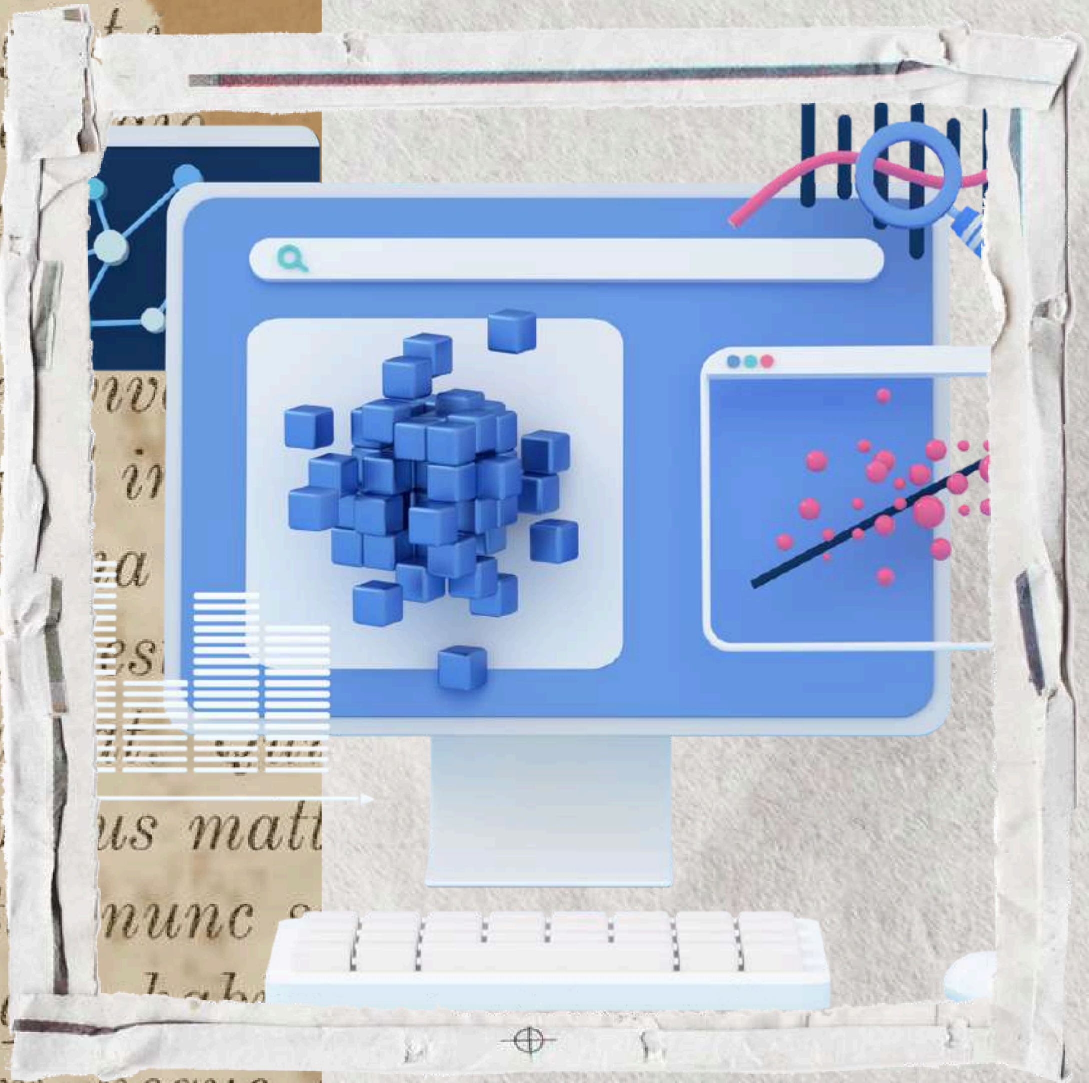
- **Why Compare?**

3. Organizations must choose the right data storage/processing architecture based on business needs.

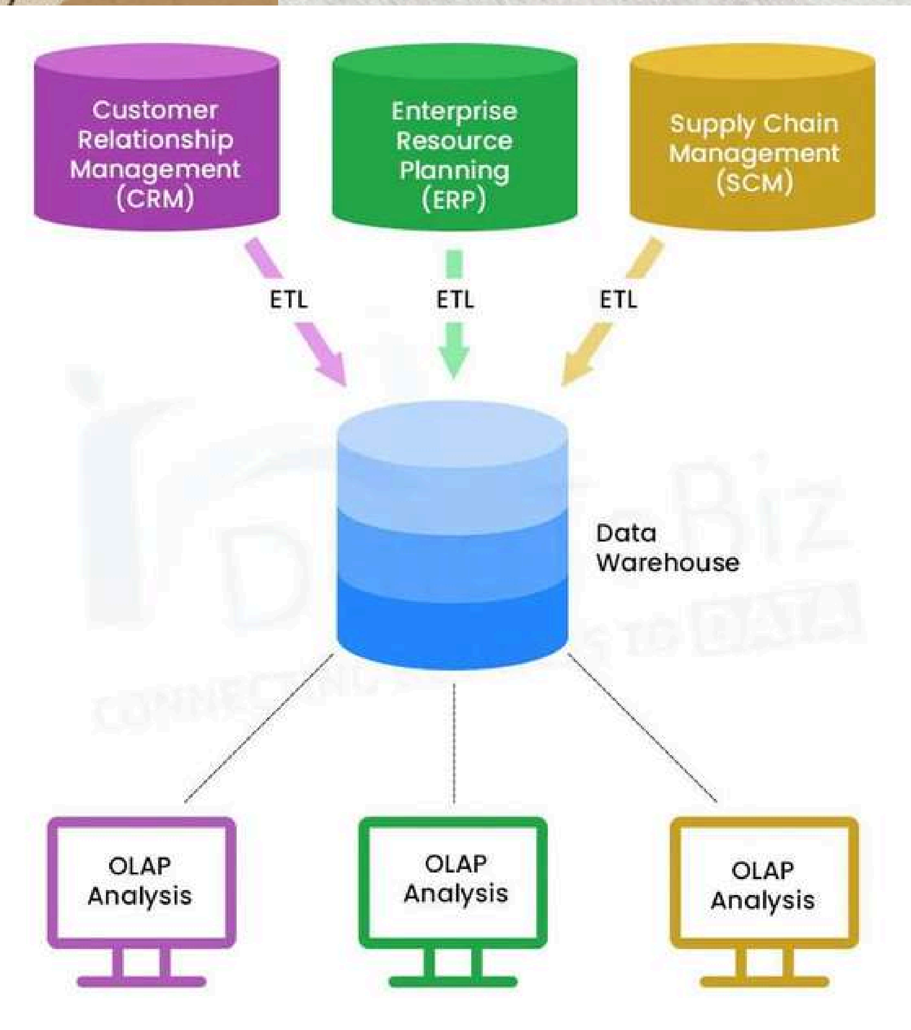
4. This comparison helps decision-makers balance cost, scalability, and performance.

- **Visual/Statistic:**

5. Consider an infographic showing data growth trends globally.



What is a Data Warehouse?


Customer Relationship Management (CRM)
Enterprise Resource Planning (ERP)
Supply Chain Management (SCM)
ETL
ETL
ETL
Data Warehouse
OLAP Analysis
OLAP Analysis
OLAP Analysis

- Definition:

I. A centralized repository designed to store structured data for reporting and analysis.

- Purpose and Usage:

2. Facilitates business intelligence, historical analysis, and decision-support.

- Core Features:

3. Schema-on-Write: Data is cleansed, transformed, and structured before storage.

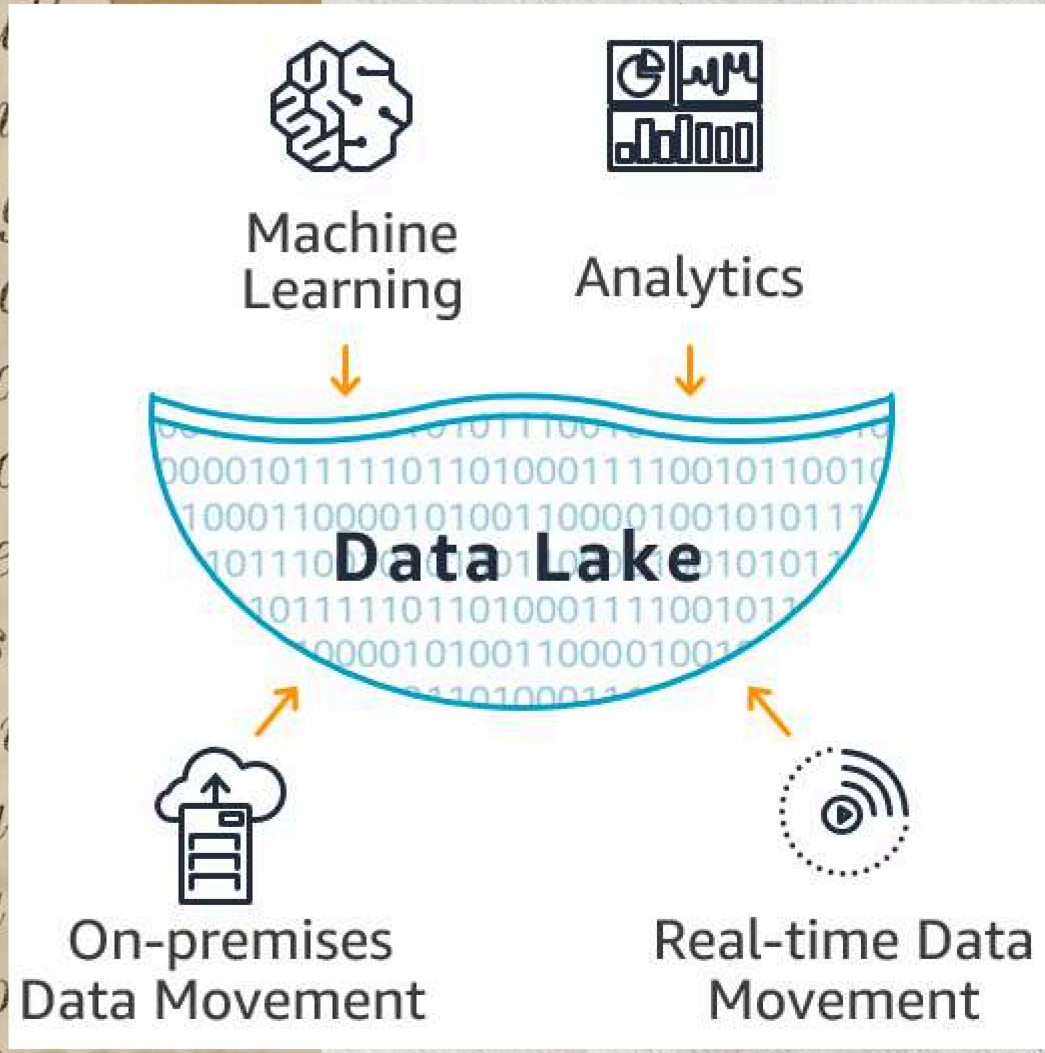
4. Optimized for OLAP: Supports fast querying and reporting.

5. Data Integration: Combines data from disparate sources through ETL/ELT processes.

- Visual Aid:

6. Diagram of ETL flow from source systems to Data Warehouse beside.

What is a Data Lake?



I. **Definition:**

   a. A storage repository that holds raw, unstructured, semi-structured, and structured data in its native format.

2. **Purpose and Usage:**

   a. Enables flexible data storage; supports data exploration, analytics, and machine learning.

3. **Core Features:**

   a. Schema-on-Read: Structure is applied when data is accessed.

   b. High Scalability: Uses distributed storage systems (e.g., Hadoop, cloud object storage).

   c. Data Diversity: Supports video, images, logs, and text as well as traditional structured data.

4. **Visual Aid:**

   a. Diagram with a layered approach: ingestion → raw data storage → processing framework (e.g., Apache Spark).

Architectural Comparison & Data Flow

1. Data Warehouse Architecture:
    - ETL Process: Extract, Transform, Load data with a focus on data quality and consistency.
    - Storage: Relational databases on dedicated hardware.
    - Query Optimization: Uses indexing and aggregations to support complex queries.
2. Data Lake Architecture:
    - Data Ingestion: Stores data in raw form with minimal upfront processing.
    - Storage: Utilizes distributed file systems or cloud object storage.
    - Processing Frameworks: Analysis through scalable systems (Apache Spark, Presto, etc.).
    - Governance Concerns: Emphasis on metadata management to avoid “data swamp” issues.
3. Visual Comparison:
    - ETL vs. raw data ingestion, schema-on-write vs. schema-on-read, cost models, and performance aspects.

Key Differences & Decision Factors

I. <u>Organization & Schema Approach:</u>

- Data Warehouse: Predefined, rigid schema for consistent data.
- Data Lake: Flexible, varied schema applied at read time.

2. <u>Performance and Efficiency:</u>

- Warehouse: Faster queries on structured data, reliable performance.
- Lake: Potential processing delays; requires robust indexing/metadata for efficient querying.

3. <u>Cost & Scalability:</u>

- Warehouse: Higher cost and maintenance overhead due to specialized hardware and ETL.
- Lake: Lower storage cost with scalable, commodity hardware; hidden costs in processing.

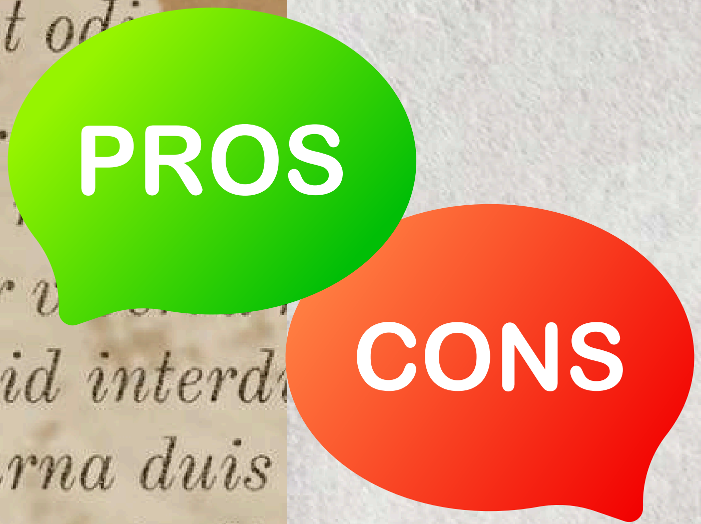
4. <u>Security & Governance:</u>

- Warehouse: Mature data governance and security frameworks.
- Lake: Demands additional data cataloging and governance tools.

5. <u>Decision Factors:</u>

- Evaluate your use case: reporting vs. exploratory analytics, structured vs. diverse data sources.

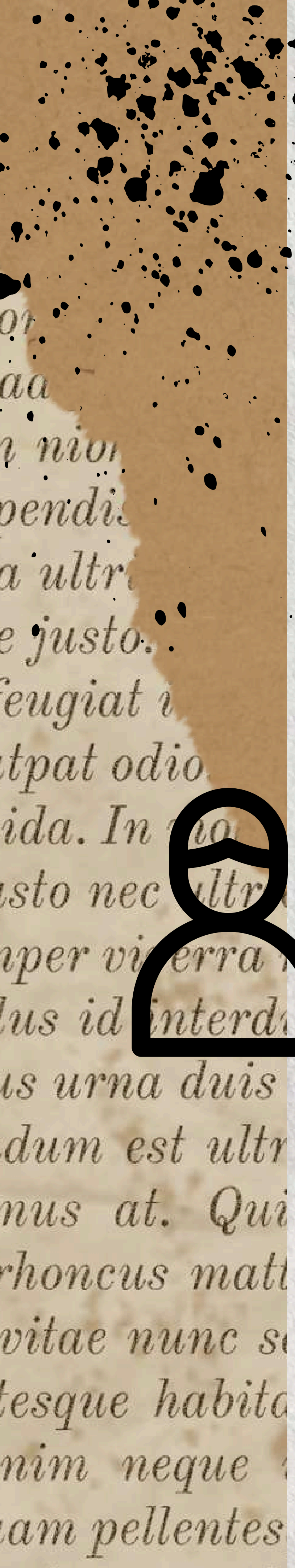
Pros and Cons



- I. Data Warehouse Advantages:
  - a. High data quality and consistency, optimized performance for complex queries.
  - b. Strong data governance, which is ideal for regulatory compliance.

2. Data Warehouse Limitations:
   - a. Inflexible structure that may struggle with rapidly changing data types.
   - b. Higher cost and slower integration of new/unstructured data sources.
3. Data Lake Advantages:
   - a. Flexibility to store all types of data with lower upfront processing costs.
   - b. Scalability ideal for big data analytics and machine learning applications.
4. Data Lake Limitations:
   - a. Can degrade into a “data swamp” without proper metadata management.
   - b. Requires more complex processing and security measures for efficient querying.
5. Visual Comparison:
   - a. A bullet list or two-column table to directly compare benefits and drawbacks.

Use Cases & Industry Examples



I. **Data Warehouse Use Cases:**

   a. Enterprise reporting, sales analysis, financial forecasting, and historical trend analysis.

   b. Examples: Financial institutions, retail chains using centralized data for decision support.

2. **Data Lake Use Cases:**

   a. Data science projects, real-time analytics, storing sensor, social media, and IoT data.

   b. Examples: E-commerce companies analyzing user behavior or healthcare institutions storing medical imaging alongside clinical data.

3. **Visual Examples:**

   a. Include icons/logos of industries, or sample case study snapshots.

4. **Real-World Insights:**

   a. Mention how hybrid approaches have been successfully used in industries like finance and healthcare.

Emerging Trends & Hybrid Approaches

I. Trends in Data Management:

- Hybrid Solutions: Adoption of Data Lakehouse architectures that bring together low-cost storage and robust query performance.
- Cloud Integration: Increasing move towards integrated cloud platforms (AWS, Azure, Google Cloud) that support both data lakes and warehouses.
- Advancements in Tools: Introduction of technologies like Delta Lake and Apache Iceberg to improve metadata management and query performance on data lakes.

2. Future Outlook:

- Evolution of real-time analytics with machine learning integration.
- Increasing emphasis on building flexible yet governed data environments.

References

I. **Books & Publications:**

- “The Data Warehouse Toolkit” by Ralph Kimball.
- “Data Lake Architecture: Designing the Data Lake and Avoiding the Garbage Dump” (various authors).

2. **Research Papers:**

- Papers on data lake governance and hybrid data architectures from IEEE, ACM, or similar reputable sources.

3. **Online Resources:**

- Articles from cloud providers (AWS, Azure, Google Cloud) comparing data lakes and data warehouses.
- Technical blogs and white papers such as “Understanding the Data Lakehouse”.

Thank you