Yashoda Shikshan Prasarak Mandal's
Yashoda Technical Campus
Approved by AICTE Delhi/ Govt. of Maharashtra
NH-4, Wadhe, Satara 415011
Email : principalpoly_ytc@yes.edu.in Call: 02162-271238/39 Mob. 9172220775
Faculty of Polytechnic
DEPARTMENT OF INFORMATION TECHNOLOGY

DWM NOTES

Chapter 1: Unit - I Basics of Data Warehousing				
Teaching Hours: 12	**Marks Distribution**			
	Remember =04 M	**Understanding=10 M**	**Applying =04 M**	**Total =18 M**

Topics and subtopics:

Unit - I Basics of Data Warehousing

1.1 Introduction to Data Warehouse

1.2 Need of Data Warehousing

1.3 Differences between Operational Database Systems and Data Warehouses

1.4 A Multi-Tiered Architecture of Data Warehouse

1.5 Data Warehouse Models: Enterprise Warehouse, Data Mart, And virtual Warehouse

1.6 Extraction, Transformation and Loading (ETL)

1.7 Metadata Repository

1.8 Concept of data pond, data lake, data ocean

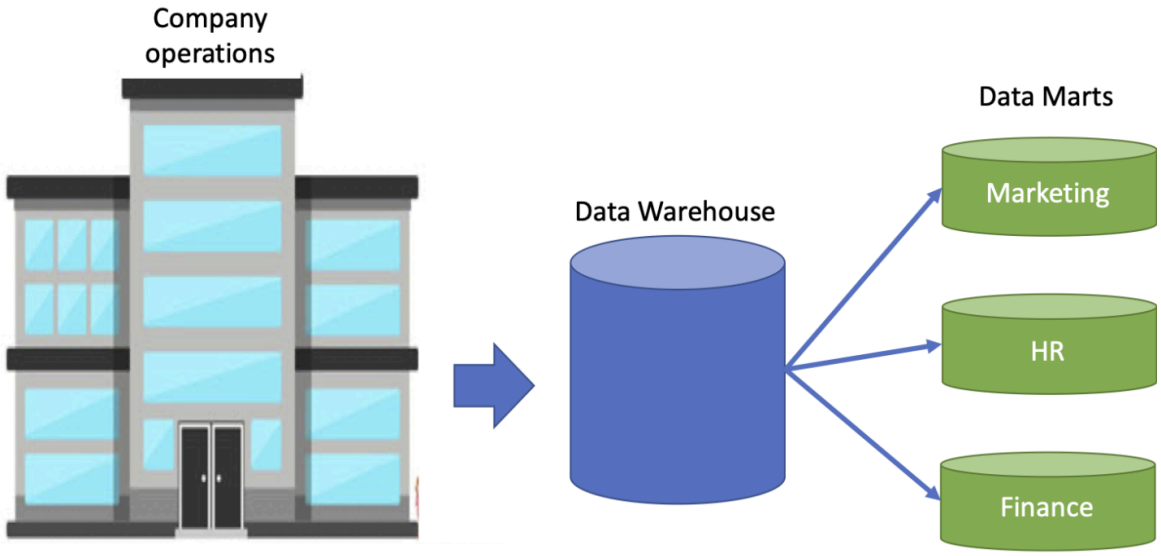
1.1 Introduction to Data Warehouse

A **Data Warehouse (DW)** is a large, central storage system where data from different sources (databases, applications, files, etc.) is collected, cleaned, and stored for analysis and decision-making.

It is mainly used by organizations to **analyze historical data**, generate reports, and support strategic decisions.

A **data warehouse** is a **big storage place** where a company keeps all its important data from different sources in **one place**.
It is mainly used for **reporting, analysis, and making business decisions**, not for daily transactions.

It is different from normal databases (used for day-to-day work) because a data warehouse is designed to **study data**, **find patterns**, and **understand trends over time**.



Key Characteristics of a Data Warehouse

1. Subject-Oriented

- Data is arranged based on **important topics** of the business like **sales**, **customers**, **products**, etc.
- This makes it easy to study each area separately.

2. Integrated

- Data comes from many different places (databases, files, online systems).
- In the data warehouse, all this data is **combined and cleaned**, so everything is in the **same format**.

3. Time-Variant

- A data warehouse stores **old as well as current data**.
- This helps in understanding how things have **changed over time**, like sales over the last 5 years.

4. Non-Volatile

- Once data is stored in the warehouse, it is **not changed or deleted** frequently.
- New data is added from time to time, but older data remains as it is for analysis.

1.2 Need of Data Warehousing

A **data warehouse** is needed because modern businesses generate a *lot of data* from many different sources.
To make good decisions, companies must **combine this data**, **analyze it**, and **understand trends**. A data warehouse helps do all this in an efficient way.

1. Data comes from many different sources

- Companies use many systems—sales, HR, marketing, finance, website, etc.
- All this data is stored **separately**.
- A data warehouse brings everything **together in one place**.

2. Helps in better decision-making

- Managers need reports like:
  ✔ Best-selling products

✔ Monthly sales growth
✔ Customer behavior

- A data warehouse makes it easy to get these answers quickly.

3. Stores historical data

- Day-to-day systems may keep only recent data.
- A data warehouse stores **old + new data**.
- This helps in studying **changes over years** and finding **trends**.

4. Faster analysis

- Operational databases are slow when running big reports.
- Data warehouses are designed for **complex queries**, so analysis becomes faster.

5. Improves data quality & consistency

- Data from different sources may use **different formats** or names.
- The warehouse cleans and converts it into **one standard format**, making reports accurate.

6. Supports Business Intelligence (BI)

- BI tools like Power BI, Tableau, and dashboards work best with a data warehouse because the data is:
    - Clean
    - Organized
    - Ready for analysis

7. Reduces load on operational systems

- Daily systems (OLTP) should focus only on transactions like billing or booking.
- Heavy reports can slow them down.
- A data warehouse handles all reporting separately.

8. Helps find patterns and trends

- Helps answer questions like:
  ✔ Which customer group buys the most?
  ✔ What time of year do sales increase?
  ✔ Which products are not performing well?

1.3 Differences between Operational Database Systems and Data Warehouses

Feature	Operational Database Systems (OLTP)	Data Warehouses (OLAP)
Purpose	Used for **day-to-day operations** like billing, payments, booking, updating customer records	Used for **analysis, reporting, and strategic decision-making**
Data Focus	Stores **current, real-time data**	Stores **historical + consolidated data** from many years
Processing Type	**OLTP** (Online Transaction Processing)	**OLAP** (Online Analytical Processing)
Operations	Handles **simple and frequent CRUD operations** (Create, Read, Update, Delete)	Mostly **read-only** with **complex analytical queries**
Data Structure	**Normalized tables** to avoid duplication and ensure accuracy	**Denormalized structures** (Star, Snowflake) to make queries faster

Data Updates	Updated constantly; data changes very frequently	**Non-volatile**: Once loaded, data is rarely changed; new data added in batches
Optimization	Optimized for **fast transactions** and high throughput	Optimized for **fast querying** across huge data volumes
Data Source	Acts as the **original source** of data	Combines data from **multiple operational systems**
Query Type	Short, simple queries (e.g., "Add new order", "Update profile")	Long, complex queries (e.g., "Find sales trend of last 5 years")
User Type	Used by clerks, staff, operators	Used by managers, analysts, decision-makers
Data Volume	Usually smaller because it stores only current data	Very large because it stores years of historical data
Purpose Summary	Helps **run** the business	Helps **analyze and improve** the business

1. Operational Database (OLTP)

- Used for **routine operations** like adding new customers, updating orders, or processing payments.
- Data changes often.
- Examples: ATM system, Shopping app, Ticket booking system.

2. Data Warehouse (OLAP)

- Used for **analyzing data**, creating reports, and helping managers make decisions.
- Stores large amounts of past data.
- Examples: Monthly sales report, yearly profit analysis, customer trend reports.

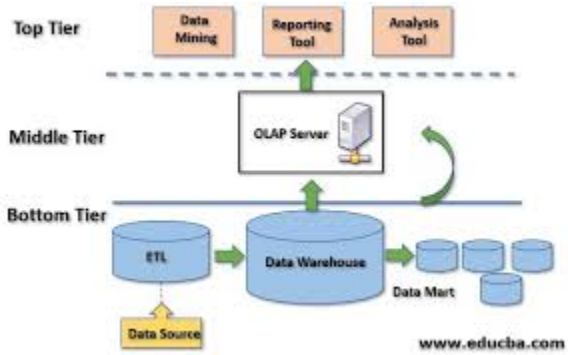
1.4 A Multi-Tiered Architecture of Data Warehouse

A multi-tiered data warehouse architecture, typically a three-tiered model, separates the system into a data source tier, a middle tier, and a top tier for better organization, performance, and scalability.

The bottom tier includes data sources and the ETL process, the middle tier is the application layer where data is cleaned and stored in the data warehouse or data marts, and the top tier provides end-user tools for reporting and analysis.

A **multi-tiered architecture** means the data warehouse is built in **different layers**, and each layer has a specific job.

This makes the system **organized, efficient, and easy to manage**.



1. Bottom Tier: Data Sources and Staging

Data Sources:

This tier consists of the various operational and external data sources where raw data originates.

- This is where all the **raw data** comes from.

- Sources include:
  - ✔ Operational databases
  - ✔ Excel files
  - ✔ Web data
  - ✔ External systems

ETL Process

Extract → Take data from different sources

Transform → Clean and convert data into a proper format

Load → Put the cleaned data into the warehouse

- ETL Process: Data is extracted from these sources, transformed to meet business requirements, and loaded into the data warehouse.

Staging Area: A temporary storage area where data is cleaned, transformed, and prepared for the data warehouse before being loaded.

- A **temporary storage place**
- Here the data is cleaned and prepared **before** going into the data warehouse.

2. Middle Tier: Data Warehouse Server

Data Warehouse

- This is the **main storage** where all processed data is kept in an organized way.
- Data Warehouse: The central repository where processed and integrated data is stored in a structured format.

Data Marts

- Smaller sections of the data warehouse
- Created for **specific departments** like sales, HR, finance, etc.

- Data Marts: Specialized subsets of the data warehouse that serve specific business functions, such as sales or finance.

OLAP Server

- Helps users do **fast and complex analysis**

- Supports multi-dimensional views like product-wise, region-wise, year-wise analysis.

- OLAP Server: An Online Analytical Processing server that enables multi-dimensional analysis and allows users to perform complex queries.

3. Top Tier: Front-End Tools

Data Access Tools

- These are the tools that help users **view and interact** with data.
- Data Access Tools: This tier provides the user interface for accessing and analyzing the data.

Reporting and Query Tools

- Dashboards

- Graphs
- Charts
- Reports

These tools help managers and analysts get insights and make business decisions.

- Reporting and Query Tools: Users interact with the data through tools like dashboards, graphs, and reports to gain insights and make decisions.

Here are **simple real-life examples** for each tier of the **Multi-Tier Architecture of a Data Warehouse**, so students can understand it easily:

Real-Life Examples of Each Tier

---

1. Bottom Tier – Data Sources & Staging

Real-Life Example

You go to a **shopping mall**.
Every time you buy something:

- The shop's **billing system** records your purchase
- The mall's **loyalty app** stores your points

- The mall's website tracks your **online orders**
- Social media ads capture your **clicks & preferences**

All these different sources generate data.

Example of ETL & Staging

- The mall collects all this data every night
- Data is cleaned (duplicates removed, wrong entries fixed)
- Data is converted to one format
- Staging area acts like a "temporary kitchen counter" where ingredients (data) are prepared before cooking (loading into warehouse)

---

2. Middle Tier – Data Warehouse Server

Real-Life Example

The mall stores all the cleaned data in one big system called a **data warehouse**.

This includes:

- All customer purchases over years

- All online shopping data
- All loyalty card data

Data Mart Example

From the full warehouse:

- The **sales team** gets a "sales data mart"
- The **marketing team** gets a "customer behavior data mart"
- The **finance team** gets a "revenue data mart"

Each department sees only what they need.

OLAP Example

A mall manager wants to check:

- "How did Diwali sales change in the last 5 years?"
- "Which product category sold most in Pune?"
- "Which age group buys more electronics?"

The OLAP server quickly answers these multi-dimensional questions.

3. Top Tier – Front-End Tools

Real-Life Example

The mall uses tools like **Power BI**, **Tableau**, or **Excel dashboards** to see the data.

These show:

- Monthly sales trends
- Most popular brands
- Customer buying patterns
- Store performance

Managers look at colorful graphs and charts to decide:

- Which products to restock
- When to run discounts
- Which stores are underperforming
- What marketing campaigns should run next

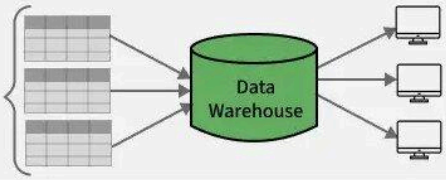
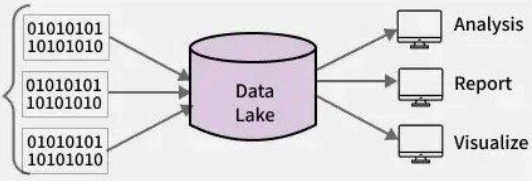
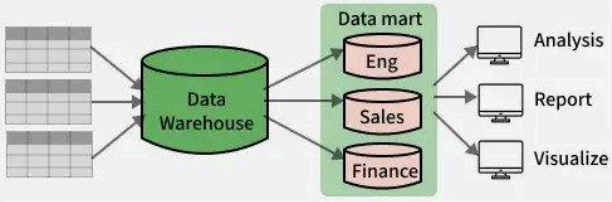
In Simple Words

- **Bottom Tier** → Collecting and cleaning data
  Like gathering ingredients in the kitchen.

- **Middle Tier** → Storing and organizing data
  Like storing cooked food in containers.

- **Top Tier** → Viewing and using data
  Like serving and eating the food.

1.5 Data Warehouse Models: Enterprise Warehouse, Data Mart, And virtual Warehouse

Name	Explanation	Illustration
Data Warehouse	A large, structured repository of integrated data from various sources, used for complex querying and historical analysis	
Data Mart	A more focused, department-specific subset of a data warehouse providing quick data retrieval and analysis	
Data Lake	A vast pool of raw, unstructured data stored in its native format until it's needed for use	

The three data warehouse models are:

Enterprise Warehouse, which is a centralized repository for an entire organization's data

Data Mart, a smaller, focused version of a data warehouse for a specific department or business unit

Virtual Warehouse, which acts as a logical view over operational data sources, allowing them to be queried as if they were a single, centralized data warehouse

1. Enterprise Data Warehouse (EDW)

An **Enterprise Data Warehouse** is a centralized warehouse that provides a comprehensive, integrated view of all enterprise data across the organization.

Characteristics

- Contains data from **all functional areas** (sales, finance, HR, operations, etc.)
- Highly **integrated**, **consistent**, and **standardized**
- Stores **historical** and **current** data
- Supports **strategic decision-making** across the entire enterprise

Advantages

- Single source of truth for the whole organization
- High data consistency and quality
- Excellent for cross-functional analytics

Disadvantages

- Expensive and time-consuming to build
- High complexity
- Requires significant management and maintenance

Scope: Covers the whole organization.

Purpose: Gives a complete, unified view of all company data for big-picture analysis and strategic decisions.

Data: Contains both detailed and summarized data from many internal systems and external sources.

Size: Very large—can grow from gigabytes to petabytes.

2. Data Mart

Definition

A **Data Mart** is a smaller, more focused version of a data warehouse, containing data for a **specific business line** or **department**, such as marketing, finance, or sales.

Types of Data Marts

1. **Dependent Data Mart** – sourced from an enterprise warehouse

2. **Independent Data Mart** – built directly from operational systems without an EDW

Characteristics

- Department-specific
- Smaller and easier to build
- May contain summarized or detailed data

Advantages

- Faster and cheaper to implement
- Tailored to user needs
- Less complex

Disadvantages

- Risk of data inconsistency if multiple marts are not integrated
- Limited enterprise-wide analytics

Scope: Focuses on one department (e.g., sales, finance, marketing).

Purpose: Supports the specific reporting and analysis needs of that department.

Data: A small part of the enterprise data, taken from a warehouse or operational systems.

Size: Much smaller than an EDW—usually up to a terabyte.

---

3. Virtual Warehouse

Definition

A **Virtual Warehouse** is a set of views or metadata that provides a unified interface to distributed data sources. It does **not store data physically** but gives real-time access to underlying operational databases or warehouse stores.

Characteristics

- Logical or *virtual* view of data
- No physical storage required (or minimal)
- Uses middleware, federation tools, or virtualization technologies

Advantages

- Low cost and fast deployment
- Real-time or near real-time data access
- Flexible and easy to update

Disadvantages

- Performance may be slower than a physical warehouse
- Suitable mainly for light analytical queries
- Depends heavily on underlying data source availability

Scope: Logical only—does not store data itself.

Purpose: Gives an easy, combined view of data that actually lives in many different databases.

Data: Stays in the original operational systems; the virtual warehouse just uses queries and views to access it.

Implementation: Needs very little storage but relies on the speed of the underlying databases.

Summary Table

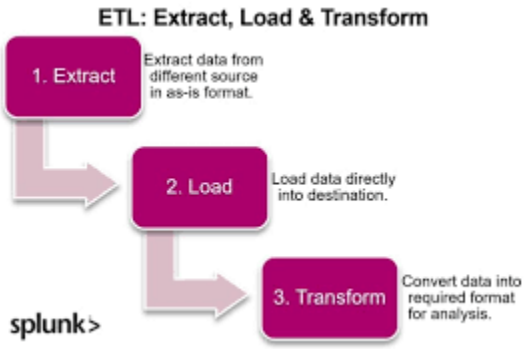
Feature	Enterprise Data Warehouse	Data Mart	Virtual Warehouse
Scope	Entire organization	Single department	Multiple distributed sources

Storage	Centralized physical storage	Smaller physical storage	No physical storage (mostly logical)
Cost	High	Low to medium	Low
Complexity	High	Low	Medium
Best for	Enterprise-wide BI	Departmental analytics	Quick access to distributed data

---

1.6 Extraction, Transformation and Loading (ETL)

ETL is the core process used in building and maintaining a data warehouse. It involves three main stages that move data from source systems into the warehouse.



1. Extraction

- This is the **first step**, where data is collected from various source systems.
- Sources may include **databases, applications, files, sensors, or external data providers**.
- The main goal is to **pull relevant data** without affecting the performance of operational systems.
- Extraction can be:
    - **Full extraction** – takes all data
    - **Incremental extraction** – takes only new or changed data

What it does: Collects data from many sources (databases, files, APIs, etc.).

Goal: Get all needed data without slowing down the source systems.

Key activities:

- Read raw data
- Validate records
- Check data types
- Remove unnecessary data

2. Transformation

- In this step, the extracted data is **cleaned, formatted, and processed** before loading into the warehouse.
- Common transformation tasks include:
    - **Data cleaning:** removing errors, duplicates, inconsistencies
    - **Data integration:** merging data from different sources
    - **Data standardization:** converting formats (e.g., date formats)
    - **Data enrichment:** adding calculated fields
    - **Data validation:** checking accuracy and completeness
- The goal is to make data **consistent, accurate, and usable** for analysis.

What it does: Cleans and converts the extracted data so it fits business rules and is consistent.

Goal: Improve data quality and prepare it for analysis.

Key activities:

- Clean errors and handle missing values
- Remove duplicate records

- Format data (units, dates, etc.)
- Combine data from different sources
- Create new calculated fields
- Encrypt sensitive information

---

3. Loading

- This is the final step, where transformed data is **moved into the data warehouse**.
- Loading can be:
    - **Batch loading:** large volumes of data loaded at scheduled times
    - **Real-time or streaming loading:** small, continuous updates
- The goal is to ensure the warehouse is **updated and ready** for query and analysis.

What it does: Moves the transformed data into the target system (data warehouse, database, etc.).

Goal: Store the final data for reporting and analytics.

Key activities:

- Write data to the destination
- Perform full load or incremental updates
- Often done during off-peak hours

Summary Table

ETL Stage	Purpose	Key Activities
Extraction	Gather data from different sources	Select, collect, and retrieve data
Transformation	Clean and prepare data	Clean, merge, convert, validate
Loading	Store data in warehouse	Batch load, real-time load

Importance and Benefits

ETL is crucial for business intelligence (BI), analytics, and machine learning initiatives because it transforms messy, scattered raw data into a reliable and usable resource for informed decision-making.

Key benefits include:

- **Unified View:** Combines data from many sources into one place.
- **Better Data Quality:** Cleansing and validation make data accurate and reliable.
- **Historical Insight:** Keeps old and new data together for long-term analysis.
- **Automation:** Modern ETL tools save time by handling repetitive tasks automatically

---

1.7 Metadata Repository

A **Metadata Repository** is a central place that stores **metadata**, which means "data about data."

In a data warehouse, it helps users understand where data comes from, how it is processed, and how it should be used.

What Metadata Includes

A metadata repository typically stores:

1. Technical Metadata

- Table names, column names, data types
- Source of the data (databases, files, systems)

- ETL rules (how data was extracted, transformed, loaded)
- Data lineage (where data originated and how it changed)

2. Business Metadata

- Definitions of business terms (e.g., "customer," "sales")
- Rules and calculations (e.g., how profit is calculated)
- Context for how data should be used

3. Operational Metadata

- ETL job schedules and logs
- Load times and data refresh status
- Error reports and performance statistics

Purpose of a Metadata Repository

- Helps users understand and trust the data
- Makes data easier to search, manage, and use
- Improves communication between business users and technical teams
- Supports troubleshooting by showing data lineage and processing steps

- Allows better control and governance of data assets

Why It's Important

- Ensures **data consistency** across the organization
- Provides **transparency** about how data is created and transformed
- Supports **better decision-making**
- Helps maintain **high data quality**
- Simplifies **maintenance** of the data warehouse

Components of a Metadata Repository

- **Metadata Database** (stores metadata tables)
- **Metadata Manager Tool** (interface to view/update metadata)
- **Metadata API** (allows other tools to access metadata)
- **Search & Query Engine** (to find metadata quickly)

Real-Life Example:

Imagine a college data warehouse storing student records. The metadata repository will tell you:

- Meaning of fields like *Attendance%, GPA, SemesterCode*
- How GPA is calculated
- Which source database provided the marks
- How marks were transformed before loading
- Last updated date of the student data

This helps teachers, administrators, and systems work smoothly.

1.8 Concept of data pond, data lake, data ocean

Modern organizations collect data from many different sources. Depending on the **size**, **complexity**, and **purpose** of data, it may be organized into:

- **Data Pond**
- **Data Lake**
- **Data Ocean**

These terms represent **increasing levels of data volume and variety**.

1. Data Pond

✔ Definition

A **Data Pond** is a **small, focused collection of raw data** created for a specific team, department, or project.

✔ Features

- Smaller in size
- Contains data from limited sources
- Usually created for a single purpose
- Easier to manage and maintain
- Often used for departmental analytics

✔ Example

The **IT Department** of a college keeps only student attendance and lab records for analysis.
This small, specialized dataset is a **data pond**.

---

2. Data Lake

✔ Definition

A **Data Lake** is a **large centralized storage** that holds **raw, semi-structured, and structured data** from many sources.

✔ **Features**

- Very large storage
- Stores data in its original/raw format
- Supports batch and real-time data
- Useful for Data Science, ML, and BI
- Can handle huge variety of data:
    - Logs
    - Images
    - Sensor data
    - Databases
    - Text files

✔ **Example**

A university stores:

- Attendance from ERP

- Exam results
- Library logs
- RFID entry records
- CCTV metadata
- IoT lab sensor data

All this mixed data stored together becomes a **data lake**.

---

3. Data Ocean

✔ Definition

A **Data Ocean** is a **massive ecosystem** of multiple data lakes and data repositories merged together, usually at organizational or enterprise level.

It supports **very high volume, velocity, and variety** of data.

✔ Features

- Extremely large scale (enterprise or global)
- Consists of multiple data lakes connected together

- Highly distributed architecture
- Used by multinational companies
- Can integrate cloud + on-premise + partner organization data

✔ **Example**

A global company like Amazon or Google collects data from:

- E-commerce
- Logistics
- Cloud services
- IoT devices
- Smart home devices
- Payment systems
- Advertising systems

All these large interconnected systems form a **data ocean**.

Comparison Table

Feature	Data Pond	Data Lake	Data Ocean
Size	Small	Large	Very Large / Global
Scope	Single department/project	Entire organization	Multi-organization / Global
Data Types	Limited	Structured + semi + unstructured	All types across ecosystems
Purpose	Focused analytics	BI, ML, Data Science	Enterprise-wide integrated analytics
Complexity	Low	Medium	High

Analogy to Understand Easily

- **Data Pond** → Small water body behind a house

- **Data Lake** → Large lake storing water from many rivers

- **Data Ocean** → Massive body connecting multiple lakes and rivers on a global scale

---