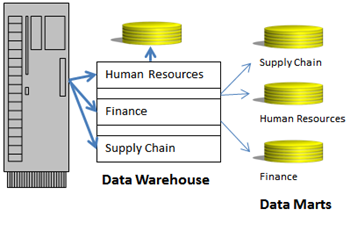
**TASK 2 DATA ENGINEERING BASICS – 2 15-03-23**

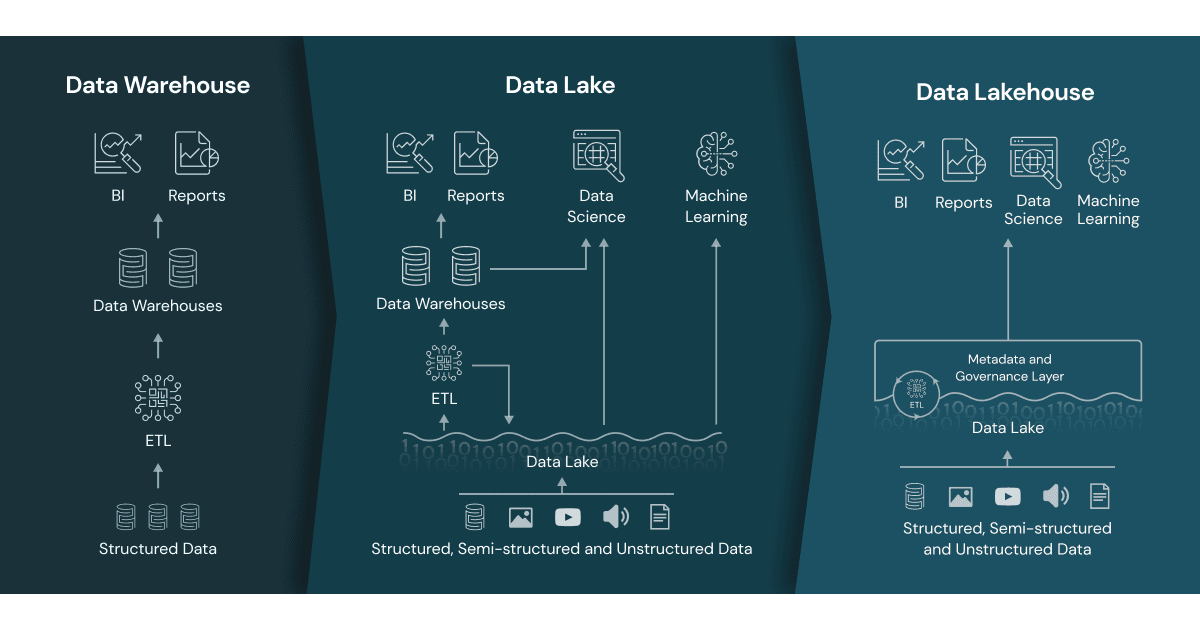
**DATA MARTS**

A data mart is a subset of a data warehouse focused on a particular line of business, department, or subject area. Data marts make specific data available to a defined group of users, which allows those users to quickly access critical insights without wasting time searching through an entire data warehouse. For example, many companies may have a data mart that aligns with a specific department in the business, such as finance, sales, or marketing.



**DATA LAKE HOUSE**

A data lake house is a new, open data management architecture that combines the flexibility, cost-efficiency, and scale of data lakes with the data management and ACID transactions of data warehouses, enabling business intelligence (BI) and machine learning (ML) on all data.



**DATA LAKE VS DATA LAKEHOUSE**

While adoption for both data lakes and data warehouses will only increase with the growth of new data sources, the limitations of both data repositories are leading to a convergence in these technologies. A data lake house couples the cost benefits of a data lake with the data structure and data management capabilities of a data warehouse.

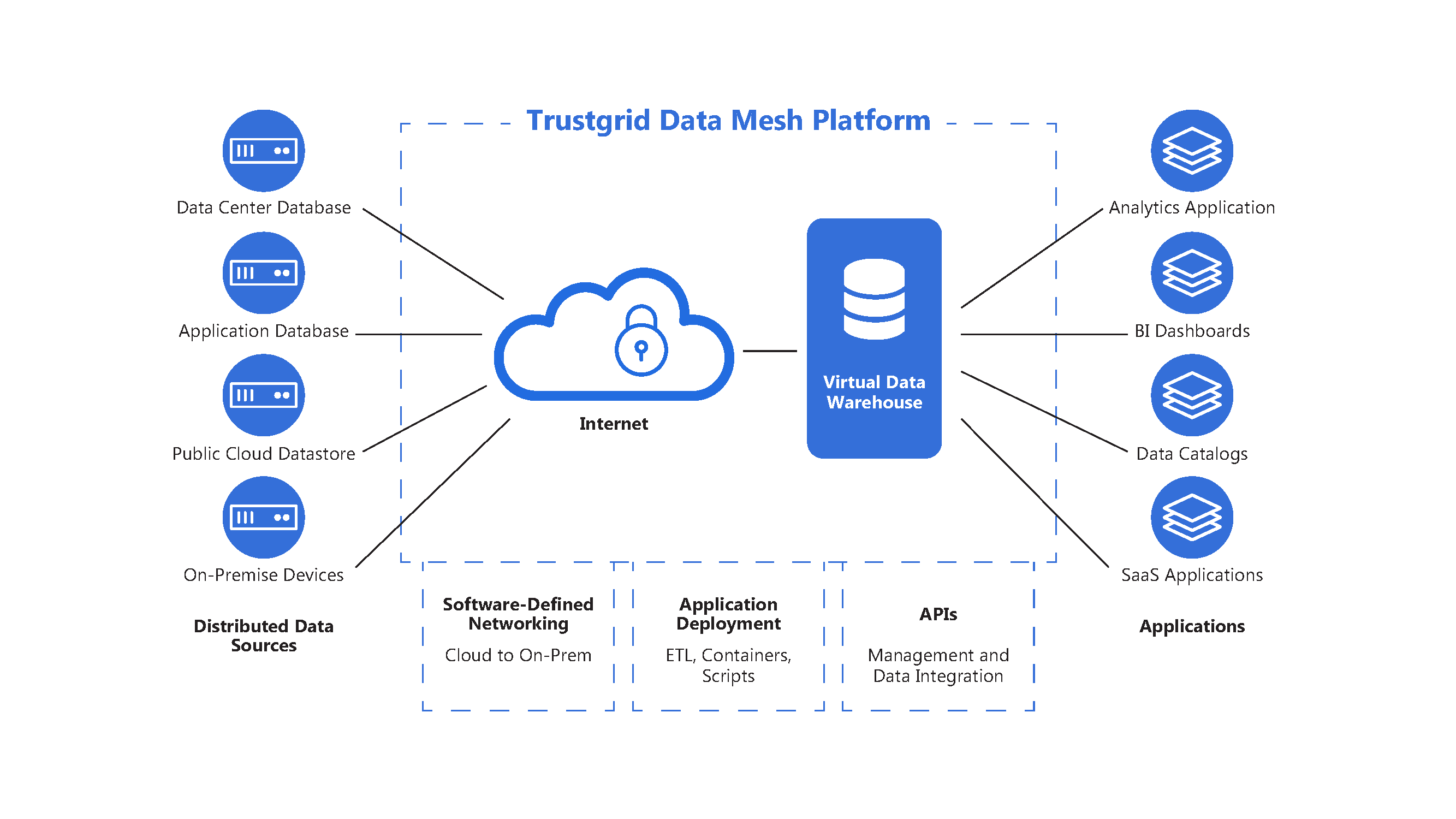
**DATA MESH**

A data mesh is a decentralized data architecture that organizes data by a specific business domain—for example, marketing, sales, customer service, and more—providing more ownership to the producers of a given dataset. The producers’ understanding of the domain data positions them to set data governance policies focused on documentation, quality, and access. This, in turn, enables self-service use across an organization. While this federated approach eliminates many operational bottlenecks associated with centralized, monolithic systems, it doesn't necessarily mean that you can't use traditional storage systems, like data lakes or data warehouses. It just means that their use has shifted from a single, centralized data platform to multiple decentralized data repositories.

It's worth noting that data mesh promotes the adoption of cloud native and cloud platform technologies to scale and achieve the goals of data management. This concept is commonly compared to micro services to help audiences understand its use within this landscape. As this distributed architecture is particularly helpful in scaling data needs across an organization, it can be inferred that a data mesh may not be for all types of businesses; that is, smaller businesses may not reap the benefits of a data mesh as their enterprise data may not be as complex as a larger organization.

**HOW DOES A DATA MESH WORK**

A data mesh involves a cultural shift in the way that companies think about their data. Instead of data acting as a by-product of a process, it becomes the product, where data producers act as data product owners. Historically, a centralized infrastructure team would maintain data ownership across domains, but the product thinking focus under a data mesh model shifts this ownership to the producers as they are the subject matter experts. Their understanding of the primary data consumers and how they leverage the domain’s operational and analytical data allows them to design APIs with their best interests in mind. While this domain-driven design also makes data producers responsible for documenting semantic definitions, cataloguing metadata and setting policies for permissions and usage, there is still a centralized data governance team to enforce these standards and procedures around the data. Additionally, while domain teams become responsible for their ETL data pipelines under a data mesh architecture, it doesn't eliminate the need for a centralized data engineering team. However, their responsibility becomes more focused on determining the best data infrastructure solutions for the data products being stored.



**DATA WAREHOUSE VS DATA LAKES**

While data lakes and data warehouses both store data, each repository has its own requirements for storage, which makes it an ideal choice for different scenarios. For instance, data warehouses require a defined schema to fit specific data analytics requirements for data outputs, such as dashboards, data visualizations, and other business intelligence tasks. These requirements are usually specified by business users and other relevant stakeholders, who will utilize the reporting output on a regular basis. The underlying structure of a data warehouse is typically organized as a relational system (i.e. in a structured data format), sourcing data from transactional databases. Data lakes, on the other hand, incorporate data from both relational and non-relational systems, allowing data scientists to incorporate structured and unstructured data into more data science projects.

Each system also has its own set of advantages and disadvantages. For example, data warehouses tend to be more performant, but it comes at a higher cost. Data lakes may be slower in returning query results, but they have lower storage costs. Additionally, the storage capacity of data lakes makes it ideal for enterprise data.

**OLAP VS OLTP**

| **Sr. No.** | **Category** | **OLAP (Online analytical processing)** | **OLTP (Online transaction processing)** |
| --- | --- | --- | --- |
| **1.** | **Definition** | It is well-known as an online database query management system. | It is well-known as an online database modifying system. |
| **2.** | **Data source** | Consists of historical data from various Databases. | Consists of only of operational current data. |
| **3.** | **Method used** | It makes use of a data warehouse. | It makes use of a standard database management system (DBMS). |
| **4.** | **Application** | It is subject-oriented. Used for Data Mining, Analytics, Decisions making, etc. | It is application-oriented. Used for business tasks. |
| **5.** | **Normalized** | In an OLAP database, tables are not normalized. | In an OLTP database, tables are normalized (3NF). |
| **6.** | **Usage of data** | The data is used in planning, problem-solving, and decision-making. | The data is used to perform day-to-day fundamental operations. |
| **7.** | **Task** | It provides a multi-dimensional view of different business tasks. | It reveals a snapshot of present business tasks. |
| **8.** | **Purpose** | It serves the purpose to extract information for analysis and decision-making. | It serves the purpose to Insert, Update, and Delete information from the database. |
| **9.** | **Volume of data** | A large amount of data is stored typically in TB, PB | The size of the data is relatively small as the historical data is archived. For ex MB, GB |
| **10.** | **Queries** | Relatively slow as the amount of data involved is large. Queries may take hours. | Very Fast as the queries operate on 5% of the data. |
| **11.** | **Update** | The OLAP database is not often updated. As a result, data integrity is unaffected. | The data integrity constraint must be maintained in an OLTP database. |
| **12.** | **Backup and Recovery** | It only need backup from time to time as compared to OLTP. | Backup and recovery process is maintained rigorously |
| **13.** | **Processing time** | The processing of complex queries can take a lengthy time. | It is comparatively fast in processing because of simple and straightforward queries. |
| **14.** | **Types of users** | This data is generally managed by CEO, MD, GM. | This data is managed by clerks, managers. |
| **15.** | **Operations** | Only read and rarely write operation. | Both read and write operations. |
| **16.** | **Updates** | With lengthy, scheduled batch operations, data is refreshed on a regular basis. | The user initiates data updates, which are brief and quick. |
| **17.** | **Nature of audience** | Process that is focused on the customer. | Process that is focused on the market. |
| **18.** | **Database Design** | Design with a focus on the subject. | Design that is focused on the application. |
| **19.** | **Productivity** | Improves the efficiency of business analysts. | Enhances the user’s productivity. |