**Ex1: Online Extraction vs Offline Extraction. Example?**

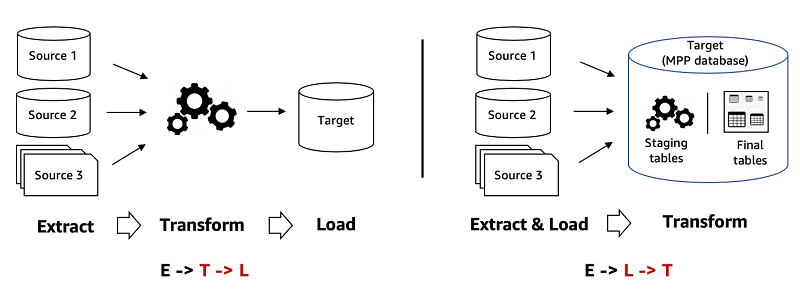
* **Online Extraction:**

In data processing or analytics, online extraction refers to the process of extracting information from data streams in real-time or near real-time as the data becomes available. For example, in online advertising, user clickstream data is continuously analyzed to extract patterns and insights that inform real-time ad targeting decisions.

* **Offline Extraction:**

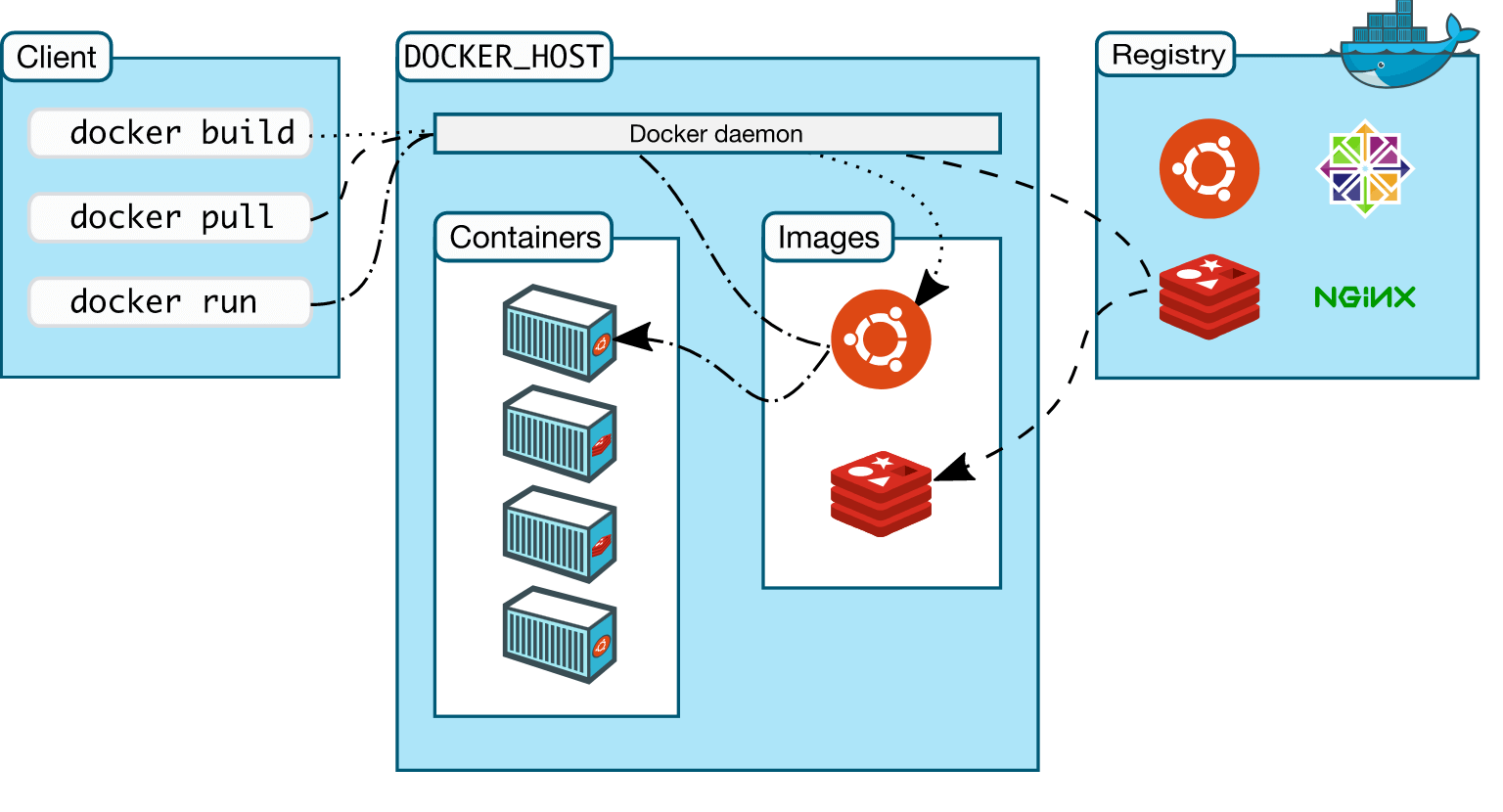
In contrast, offline extraction involves extracting information from static or stored data sets that are not continuously updated. For example, in data warehousing, offline extraction refers to the process of extracting data from various sources, transforming it, and loading it into a data warehouse for subsequent analysis. This extraction process typically occurs at scheduled intervals, such as nightly or weekly batch jobs.

**Ex2: Motivation of ELT model. What's the difference between ETL and ELT?**



|  |  |  |
| --- | --- | --- |
| **Danh mục** | **ETL** | **ELT** |
| Quy trình | Lấy dữ liệu thô và chuyển đổi dữ liệu thô thành một định dạng đã xác định trước, sau đó tải vào kho dữ liệu mục tiêu. | Lấy dữ liệu thô, tải dữ liệu thô vào kho dữ liệu mục tiêu, sau đó chuyển đổi dữ liệu thô ngay trước khi phân tích. |
| Vị trí chuyển đổi và tải | Quá trình chuyển đổi diễn ra trong một máy chủ xử lý thứ cấp. | Quá trình chuyển đổi diễn ra trong kho dữ liệu mục tiêu. |
| Khả năng tương thích với dữ liệu | Phù hợp nhất với dữ liệu có cấu trúc. | Có thể xử lý dữ liệu có cấu trúc, phi cấu trúc và bán cấu trúc. |
| Tốc độ | ETL chậm hơn ELT. | ELT nhanh hơn ETL vì nó có thể sử dụng các tài nguyên nội bộ của kho dữ liệu. |
| Chi phí | Có thể cần nhiều thời gian và chi phí để thiết lập, tùy thuộc vào các công cụ ETL được sử dụng. | Tiết kiệm chi phí hơn, tùy thuộc vào cơ sở hạ tầng ELT được sử dụng. |
| Bảo mật | Có thể cần phải xây dựng các ứng dụng tùy chỉnh để đáp ứng các yêu cầu bảo vệ dữ liệu. | Bạn có thể sử dụng các tính năng tích hợp sẵn của cơ sở dữ liệu mục tiêu để quản lý việc bảo vệ dữ liệu |

**Ex3: Investigate the overall architecture of Docker; The differences between Docker Container and Docker Image; How to build a docker image; How to run a container from a self-built docker image; How to run a container from docker hub image.**

1. Overall Architecture of Docker:

Docker makes use of a client-server architecture. The Docker client talks with the docker daemon which helps in building, running, and distributing the docker containers. The Docker client runs with the daemon on the same system or we can connect the Docker client with the Docker daemon remotely. With the help of REST API over a  UNIX socket or a network, the docker client and daemon interact with each other.

2. Differences between Docker Container and Docker Image:

- Docker Image: A Docker image is a read-only template used to create containers. It contains everything needed to run an application, including the code, runtime, libraries, and dependencies.

- Docker Container: A Docker container is a runnable instance of a Docker image. It encapsulates an application and its dependencies, providing an isolated environment for execution.

3. Building a Docker Image:

- Create a Dockerfile: This is a text file that contains instructions for building the image.

- Define the base image: Specify the base image from which your image will inherit.

- Copy application files: Copy your application code and any necessary files into the image.

- Define commands: Use Dockerfile instructions like “RUN”, “COPY”, and “CMD” to configure the image.

- Build the image: Use the “docker build” command to build the image from the Dockerfile.

4. Running a Container from a Self-built Docker Image:

- After building the Docker image, you can run a container from it using the “docker run” command followed by the image name.

- For example: docker run <image\_name>

5. Running a Container from Docker Hub Image:

- Docker Hub is a public registry of Docker images.

- To run a container from an image on Docker Hub, use the `docker run` command followed by the image name.

- If the image is not available locally, Docker will automatically pull it from Docker Hub before running the container.

- For example: docker run <username>/<image\_name>

**Ex4: Try Multi-stage build to check if the image size is reduced.**

**Ex5: Study SQL, DDL, DML, DQL.**

SQL (Structured Query Language) is the language used to interact with relational databases. It's divided into three main categories:

1. DDL (Data Definition Language):

- DDL commands are used to define, modify, and delete database objects such as tables, indexes, and views.

- Common DDL commands include:

- `CREATE`: Used to create new database objects like tables, indexes, views, etc.

- `ALTER`: Modifies existing database objects, such as adding or deleting columns in a table.

- `DROP`: Deletes existing database objects, such as tables or views.

- `TRUNCATE`: Deletes all data from a table but keeps the table structure intact.

- `RENAME`: Renames an existing database object.

2. DML (Data Manipulation Language):

- DML commands are used to manipulate data within the database.

- Common DML commands include:

- `SELECT`: Retrieves data from one or more tables.

- `INSERT`: Adds new rows of data into a table.

- `UPDATE`: Modifies existing data within a table.

- `DELETE`: Removes rows of data from a table.

3. DQL (Data Query Language):

- DQL is a subset of SQL used specifically for querying data from the database.

- The primary DQL command is `SELECT`, which is used to retrieve data from one or more tables based on specified criteria.

- `SELECT` statements can include various clauses such as `FROM`, `WHERE`, `GROUP BY`, `HAVING`, `ORDER BY`, etc., to filter, sort, and manipulate the retrieved data.