Dask is a parallel computing framework that extends Python's standard data science libraries like Pandas, NumPy, and scikit-learn to handle larger-than-memory datasets and distributed computation. Here is how it works in our project:

**Parallel Reading of Large Dataset**

* **Code:** dd.read\_csv(dataset\_path)
* Dask reads the CSV file in parallel by splitting it into chunks (partitions). Each partition is processed independently, enabling efficient use of memory and computation resources. Unlike Pandas, which loads the entire dataset into memory, Dask processes only chunks that fit into memory, making it suitable for handling large datasets.

**Lazy Evaluation**

* Operations like groupby, sum, count, and mean in Dask are **lazy**.
* When you call ddf.groupby("payment\_type")["total\_amount"].sum(), Dask builds a computation graph but doesn’t execute it immediately. The computation is triggered only when you call .compute(). This approach:
  + Reduces memory usage since computations are performed on demand.
  + Allows Dask to optimize the execution by combining tasks and avoiding redundant computations.

**Distributed Data Processing**

* Your dataset is divided into partitions across cores. Each operation, such as groupby, is executed on these smaller partitions independently. For example:
  + **Summing total revenue per payment type:** Each partition calculates the sum for its subset of data, and the results are combined in a final aggregation step.

**Scalable Aggregations**

* Aggregation functions like .sum() or .mean() operate on intermediate results from each partition.
  + For instance, mean is computed using partial means and counts from each partition, avoiding the need to load all data at once.
  + Take this dataset for instance [10,20,30,40,50,60]

Divide this dataset into 3

Partition 1: Sum = 30, Count = 2

Partition 2: Sum = 70, Count = 2

Partition 3: Sum = 110, Count = 2

A = total of Sum, B = total of count

A/B => 210/6 => 35

* This process ensures scalability for large datasets.

**5. Efficiency**

* Dask's multi-threading or distributed execution model allows it to parallelize operations across multiple CPU cores or even machines, leading to faster computation than single-threaded libraries like Pandas for large datasets.

**Benefits of Using Dask in our Project**

1. **Handles Large Datasets:** Works seamlessly even if the dataset is larger than available RAM.
2. **Faster Execution:** Parallel processing accelerates operations compared to Pandas.
3. **Flexible Deployment:** Can scale from a single machine to a distributed cluster if needed.
4. **Efficient Resource Use:** Processes only chunks of data at a time, reducing memory overhead.