Doris 的分区表和分桶表

1、分区表

分区表是将数据按照某个字段（通常是时间字段或其他离散值字段）划分为多个逻辑部分（Partition）。每个分区是一个独立的存储单元，可以单独管理和查询。询时可以快速定位到特定分区，减少数据扫描量。

作用

- **数据管理**：分区可以方便地按时间范围或其他条件删除或加载数据。例如，按月分区的数据表可以轻松删除旧数据。
- **查询优化**：通过分区裁剪（Partition Pruning），Doris 可以只扫描与查询条件相关的分区，从而减少数据扫描量。
- **并行处理**：分区数据可以在多个节点上并行处理，提升查询性能。

使用场景

- 数据按时间分布（如日志数据、交易记录等）。
- 需要定期删除旧数据（如保留最近一年的数据）。
- 查询通常包含分区字段作为过滤条件。

2、分桶表

分桶表是将数据按照某个字段的哈希值划分到多个桶（Bucket）中。每个桶是一个物理存储单元，数据在桶内有序存储。分桶使得某些特定查询能够有效进行并行处理，提升查询效率。

作用

- **数据分布**：分桶可以确保数据均匀分布在集群的各个节点上，避免数据倾斜。
- **查询加速**：分桶字段通常是查询中的过滤条件或联结字段，Doris 可以利用分桶信息快速定位数据。
- **局部排序**：分桶内的数据是有序的，有助于提高范围查询和聚合查询的性能。

使用场景

- 数据需要均匀分布在多个节点上。
- 查询中经常使用某些字段作为过滤条件或联结条件。
- 数据量较大，需要通过分桶来提高查询并发能力。

3、分区表与分桶表的区别

维度	分区表（Partition Table）	分桶表（Bucket Table）
划分方式	按指定列的取值范围（如时间、地区）划分为多个分区	按哈希算法将数据均匀分布到多个桶（Bucket）
适用场景	适用于按时间、类别等维度进行查询	适用于高并发查询、大规模数据的 Join 计算
数据存储	每个分区独立存储，提高查询效率	数据在桶中均匀分布，优化并行查询
查询优化	通过**分区裁剪（Partition Pruning）** 跳过不相关分区，减少扫描数据	通过 **分桶均衡数据分布**，提高查询并行度
适合的数据规模	适合 TB 级别的大数据集，便于管理	适合高并发小数据查询，提高查询速度
Join 计算	需要数据重新分布（Shuffle），Join 可能较慢	相同分桶键的表可以加速 Join

4、选择合适的表类型

数据量非常大，并且需要频繁基于某个字段（如时间）进行查询，选择**分区表**。

数据包含大量高基数字段（如用户 ID、订单 ID 等），并且需要并行处理这些数据，选择**分桶表**。

在复杂查询场景下，可以结合分区和分桶的特性构建**复合表**：

- 使用 **分区** 按时间或其他离散值划分数据，优化范围查询和数据管理。
- 使用 **分桶** 按高基数字段分布数据，确保数据均匀分布并加速过滤和联结操作。

这种组合方式能够更好地满足复杂的查询需求，同时兼顾数据管理和查询性能。