

Partition Index and Data Locality Information

Background

Some of the challenges faced by distributed query engines are in the areas of join planning and scheduling tasks on physical nodes. In the area of join planning, information about the number of rows in a partition are very helpful in determining whether a broadcast join or a shuffle join should be used. Similarly, while the individual splits are scheduled, availability of information about data locality helps in scheduling splits on the nodes where data is located. This document proposes extensions to existing manifest files to include this information.

Proposed Enhancements

The proposed enhancements are in 2 specific areas:

- Partition index
- Data locality information

Partition Index

A list of partitions along with stats per partition helps in figuring out the number of rows being processed. This information is useful in join planning. We propose storing the below information to capture information about all partitions; and also per-partition statistics to help with planning.

Within each [snapshot](#), we propose adding the following information:

- partition-index (string): The name of the file that contains information about all partitions in the table; and includes additional information that can help with planning

The schema of partition-list is as follows:

- partition-spec-id (int): This is the partition-spec-id for the partition
- partition-values: This is a list of values for each partition column. Uses the same format as the partition field in the data-file struct
- file-count (int): Number of files in this partition
- row-count(long): Number of rows in this partition

Data locality

Data locality helps in scheduling work on execution nodes where the data is co-located. This is particularly useful in HDFS clusters. Both the manifest files and the data files are enhanced to include data locality (affinity) information.

Affinity is defined as follows:

- host-name (string): The name of the host
- host-ip: The ip address of the host. One of the two fields, host-name or host-ip, has to be present
- size (long): Number of bytes stored on this host. This list will be sorted in descending order by this field
- timestamp-ms (long): Timestamp when this information was computed

The [manifest-list file](#) present in a snapshot optionally includes an additional field to capture affinity information for a manifest file:

- host-list (list): This is a list containing affinity for a manifest file. This field is optional and is not expected to be present when manifest files are stored on the cloud (e.g S3/Azure)

The data-file struct in the [manifest file](#) is modified to include the following information. The split-offsets field contains the offset of a split in the file. The split-offsets field is enhanced to include the following information per split:

- split-offset (long): Offset of the split in the file (already existing)
- split-size (long): Size of the split. Can be inferred from the offset of the next split
- host-list (list): This is a list containing affinity for this split. This field is optional and is not expected to be present for tables stored on the cloud (e.g S3/Azure)