Hive performance tuning

To maximize performance of your Apache Hive query workloads, you need to optimize cluster configurations, queries, and underlying Hive table design. This includes the following:

- Configure CDH clusters for the maximum allowed heap memory size, loadbalance concurrent connections across your CDH Hive components, and allocate adequate memory to support HiveServer2 and Hive metastore operations.
- Review your Hive query workloads to make sure queries are not overly complex, that they do not access large numbers of Hive table partitions, or that they force the system to materialize all columns of accessed Hive tables when only a subset is necessary.
- Review the underlying Hive table design, which is crucial to maximizing the throughput of Hive query workloads. Do not create thousands of table partitions that might cause queries containing JOINs to overtax HiveServer2 and the Hive metastore. Limit column width, and keep the number of columns under 1,000.

HiveServer2 and the Hive metastore require sufficient memory to run correctly. The default heap size of 256 MB for each component is inadequate for production workloads. Consider the following guidelines for sizing the heap for each component, based on your cluster size.

Number of Concurrent Connections	HiveServer2 Heap Size Recommended Range	Hive Metastore Heap Size Recommended Range
Up to 40 concurrent connections Cloudera recommends splitting HiveServer2 into multiple instances and load-balancing once you start allocating over 16 GB to HiveServer2. This reduces the impact of Java garbage collection on active processing by the service.	12 - 16 GB	12 - 16 GB
Up to 20 concurrent connections	6 - 12 GB	10 - 12 GB
Up to 10 concurrent connections	4 - 6 GB	4 - 10 GB
One connection	4 GB	4 GB