Apache CarbonData 2.2.0 Release

Apache CarbonData community is pleased to announce the release of the Version 2.2.0 in The Apache Software Foundation (ASF).

CarbonData is a high-performance data solution that supports various data analytic scenarios, including BI analysis, ad-hoc SQL query, fast filter lookup on detail record, streaming analytics, and so on. CarbonData has been deployed in many enterprise production environments, in one of the largest scenarios, it supports queries on a single table with 3PB data (more than 5 trillion records) with response time less than 3 seconds!

We encourage you to use the release https://archive.apache.org/dist/carbondata/2.2.0/, and feedback through the CarbonData user mailing lists!

This release note provides information on the new features, improvements, and bug fixes of this release.

What’s New in CarbonData Version 2.2.0?

In CarbonData 2.2.0, 48 JIRA tickets related to improvements, and bugs have been resolved. Please find the summary of the important features that are developed with this release.

Support Add, Drop and rename column support for the complex column

CarbonData now supports the alter operations on the complex columns also, user can add complex column, drop and also rename columns and it is supported for the nested columns too.

Spark-3.1 support

CarbonData is now integrated with Spark-3.1 to leverage the improvements on spark3.

Secondary Index Support for Presto

Now query from presto can leverage the advantage of Secondary index using the index server and get faster query results.

CDC Performance improvement

CDC merge performance is improved with many optimizations and introduced new APIs for Upsert, Update, Delete and Insert operations during CDC.

Local sort Partition Load and Compaction improvement

Performance of load and compaction is improved for the partition table.

Geo Spatial Query enhancements

Support IN_POLYGON_LIST and IN_POLYLINE_LIST with SELECT QUERY on the polygon table.
Support IN_POLYGON filter as join condition for spatial JOIN queries which.

Improve table status and metadata writing

Writing table status file is improved to avoid any reliability issues in the long run and avoid any cache issues.