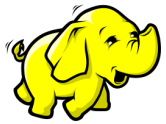
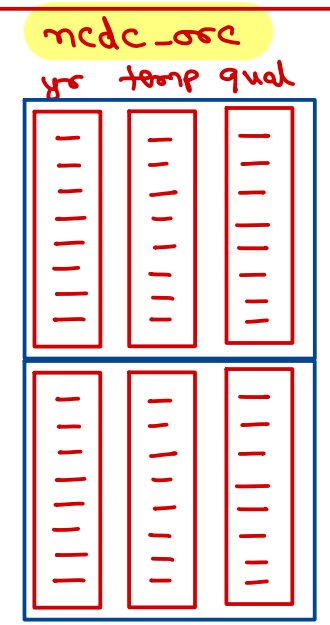


Apache Hive

Sunbeam Infotech

File formats

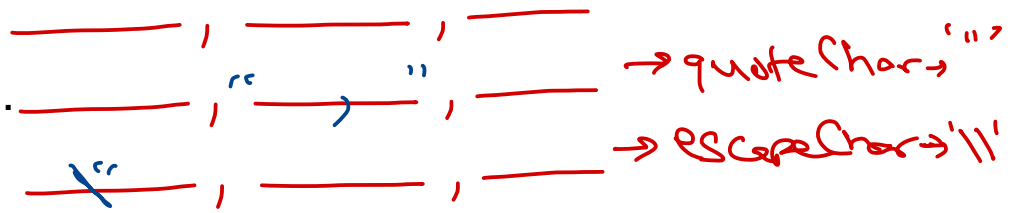
- Hive data is stored in HDFS in supported file formats.
- The file formats decide data processing in generated MR job.
- Popular file formats
  - TEXTFILE
    - If not mentioned or "STORED AS TEXTFILE", the hive table data is stored in text format.
    - Internally it use InputTextFormat/OutputTextFormat for processing it. By default one record is one line.
  - RC
  - ORC - Optimized Row Columnar
    - Text processing is not efficient.
    - ORC file format is designed for optimized execution of Hive queries.
    - ORC (Optimized Row Columnar) stores data in columnar way in binary format.
  - Parquet
  - Json
  - AVRO



Hive SerDe

- Serde is Serializer & Deserializer.
  - Internally encapusulate Hadoop InputFormat (& RecordReader) and OutputFormat (& RecordWriter).
  - Types: Built-in Serdes (e.g. OpenCSVSerde), Third party Serdes, Custom Serdes

- OpenCSVSerde
  - Loads CSV file into hive table
    - Comma separated file
    - If data contains comma, cell is enclosed in double quote.
    - If data contains double quotes, it is escaped by "\".



- RegexSerde
  - Only Deserializer i.e. only used to read records.
  - Mainly used for data cleansing/extraction.

Hive Views

- Hive views are same as RDBMS views.
- Hive 2.x views are not materialized.
  - Only Hive view create query is stored in Hive metastore.

- Hive 3.x support materialized views.
  - `CREATE MATERIALIZED VIEW mv_booksummary AS SELECT subject, SUM(price) total, AVG(price) avgprice FROM books_orc GROUP BY subject;`
  - `ALTER MATERIALIZED VIEW mv_booksummary REBUILD;`
  - `SHOW MATERIALIZED VIEWS;`
  - `DROP MATERIALIZED VIEW mv_booksummary;`
- Applications
  - Simplify few queries.
  - Security.
  - Improve performance.

Hive Joins

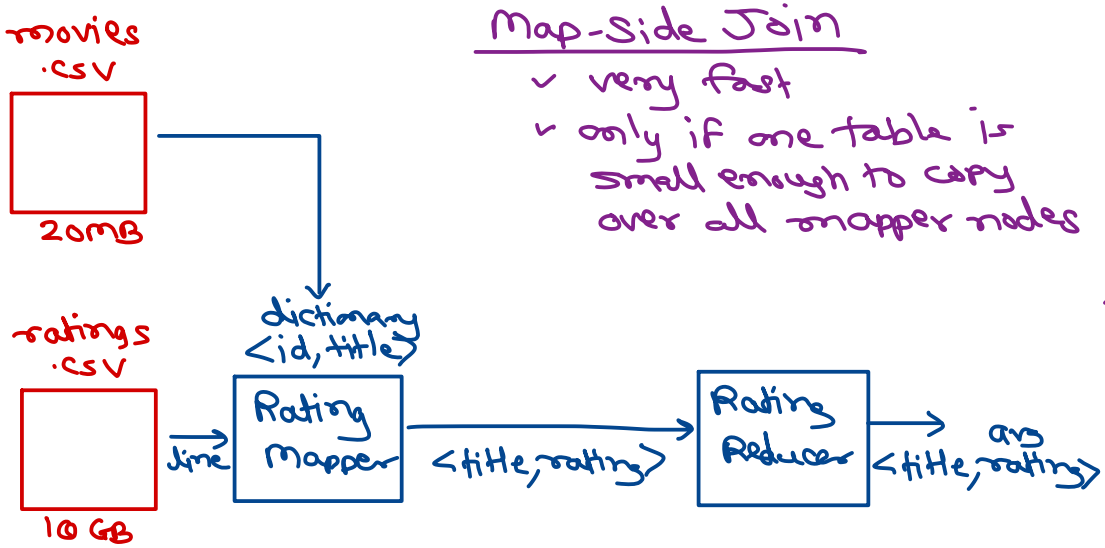
- Hadoop supports two types of joins i.e. Map-side and Reduce-side join.

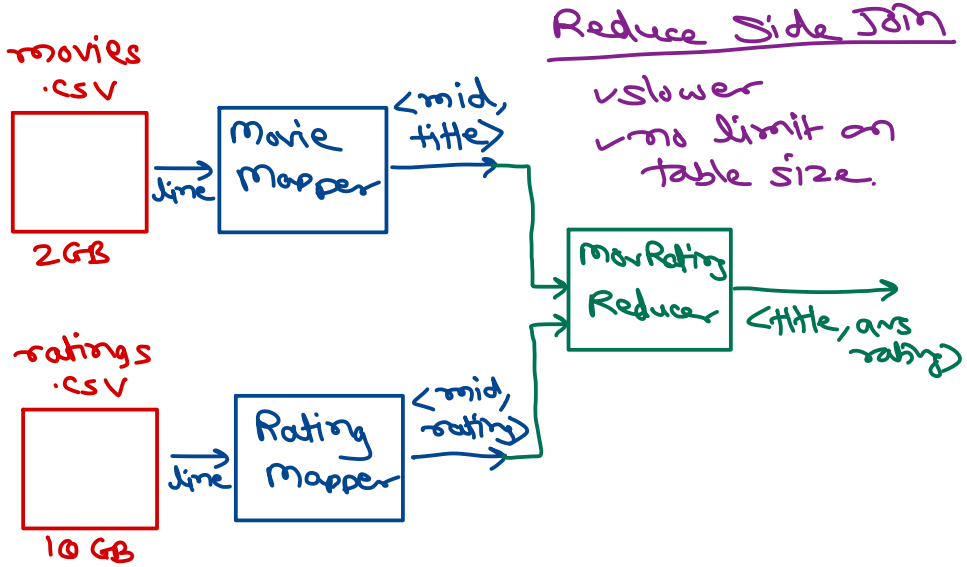
old hive- SELECT /*+MAPJOIN*/ m.title, ...

- In map-side join, smaller table is copied on all processing nodes and connected with other table in mapper only. Reducer does only aggregation.

- In reduce-side join, both tables are processed by individual mappers and produce common key. Using this reducer connect both tables and also does aggregation job.

- SET hive.auto.convert.join=false; (hive 2.x, true → hive 3.x)
- SET hive.auto.convert.join.noconditionaltask=false;





Movie Recommendation

movies

id	title	…
…	…	…

ratings

uid	mid	rating
17	70	3.0
35	21	1.0
49	19	2.0
49	21	1.0
49	70	4.0
87	19	1.0
87	19	1.0
87	21	2.0
98	19	2.0

$$r = \frac{n(\Sigma xy) - (\Sigma x)(\Sigma y)}{\sqrt{[\,n\Sigma x^2 - (\Sigma x)^2\,]\,[\,n\Sigma y^2 - (\Sigma y)^2\,]}}$$

user movies

m1	m2	r1	r2
21	70	1.0	4.0
19	70	2.0	4.0
19	21	2.0	1.0
19	21	1.0	2.0

corr movies

m1	m2	r1	r2
21	70	1	0.0
19	70	1	0.0
19	21	2	-1.0

Hive DML

- Hive was designed for OLAP.
- HDFS is write once read multiple times (no edit).
- Hive doesn't allow UPDATE or DELETE operations (before 0.14).
- Newer versions enable operations. Should be avoided for efficiency.
- Need to set transaction manager, concurrency and compactor properties into hive-site.xml.
- UPDATE/DELETE is allowed only if table meets following conditions.
  - ORC file format, Bucketing, Transactional attribute
- Each DML operations save modifications into delta files. Hive process them when data is queries.
- The delta files are merged over the period by compactor threads.

Hive connectivity

- HiveServer2 must be running to connect Hive from external programs.
- The thrift service in HiveServer2 accept client query and execute it on server.
- Hive can be connected from Python with multiple packages.
- pyhive package is most popular option for it.
  - sudo apt-get install python3.7-dev libsasl2-dev
  - python3.7 -m pip install thrift sasl thrift_sasl pyhive
- Hive can be connected from Java using JDBC.
  - JDBC Jar is available in $HIVE_HOME/jdbc and it should be added in project classpath.

Thank you!

Nilesh Ghule <nilesh@sunbeaminfo.com>