# Spatial queries

## Overview

Hadoop-GIS is a scalable and high performance spatial data warehousing system for running large-scale spatial queries on Hadoop. Hadoop-GIS relies on RESQUE for spatial query processing. RESQUE is a internally developed tile based spatial query engine which is written in C++ and deployed as shared library.

**Hive$^{SP}$**: we integrate Hadoop-GIS with Hive, to support both structured queries and spatial queries with a unified query language (HQL) and interface (Hive Shell).

# Query Language

At the language layer, Hadoop-GIS extends HQL to support spatial data types and query constructs.

JOIN operator – we kept JOIN keyword for backward compatibility. However, whenever there is a spatial operator in the join predicate, the query is considered as spatial join query and a spatial join query processing pipeline is applied to process this query.

e.g SELECT *  FROM a JOIN b on ST_INTERSECTS (a.spatialcolumn ,b.spatialcolumn)  = TRUE ;

**Data Type**

We will add a spatial data type in Hive: **GEOMETRY**. Geometry is an extension of String type with special serialization/deserialization, and operation support. For example, users can create a table with spatial column as shown in following example:

CREATE TABLE IF NOT EXISTS spatial_table ( tile_id  STRING, d_id STRING, rec_id STRING, outline **GEOMETRY**) ROW FORMAT DELIMITED FIELDS TERMINATED BY '\t' ;

**Query Pipelines**

We have created efficient query pipelines between Hive and RESQUE, to support various spatial queries. Again, with the philosophy of “minimum change to Hive”, current query pipelines implemented as set of “custom MapReduce codes” which interacts with Hive via **TRANSFORM** mechanism.

Specifically, a spatial SQL query will be intercepted at the **query translation** phase to translate the query into a Hive executable query operators. Basically, the spatial query processing part will be translated into set of custom Map and Reduce scripts which will interact with Hive via STDIN and STDOUT.

Before Hive submits the spatial query operator to the RESQUE for processing, it will use appropriate serialization method to transform data into a format that RESQUE can recognize. Then after REQUE processing, RESQUE will desterilize the data into a format that can be recognized by Hive.

# Spatial Predicates

At this moment, Hadoop-GIS support the following spatial predicates which implemented as Hive UDF. More predicates being developed and will be integrate in future.

- *st_intersects*
- *st_touches*
- *st_crosses*
- *st_contains*
- *st_adjacent*
- *st_disjoint*
- *st_equals*
- *st_dwithin*
- *st_within*
- *st_overlaps*

We can use the spatial query just like using standard HQL in Hive shell. For example, if we want to *spatially join* two tables (say *ta* and *tb*), we can issue following HQL sentence in Hive Shell:

- *SELECT ta.rec_id, tb.rec_id FROM ta JOIN tb ON (st_intersects(ta.outline, tb.outline) = TRUE);*

We will get the following output for the above *st_intersects* query:

- *……*
- *Hadoop job information for Stage-1: number of mappers: 2; number of reducers: 1*
- *2013-09-08 01:23:18,838 Stage-1 map = 0%,  reduce = 0%*
- *2013-09-08 01:23:24,889 Stage-1 map = 100%,  reduce = 0%*
- *2013-09-08 01:23:33,954 Stage-1 map = 100%,  reduce = 100%*
- *Ended Job = job_201309080121_0001*
- *MapReduce Jobs Launched:*
- *Job 0: Map: 2  Reduce: 1   HDFS Read: 187984 HDFS Write: 40 SUCCESS*

```
Total MapReduce CPU Time Spent: 0 msec
OK
1       1
1       2
1       3
1       4
15      87
34      78
54      74
61      54
Time taken: 28.207 seconds
```

# Changes in Hive

We have tried to make minimum change to Hive to not to compromise the compatibility.

Changes are mostly at the language, and query optimization layer.

**Lanague layer**: Hive.g is changed to add data types and other spatial language support.

**Parsing/Analyzing:** Mostly the *SemanticAnalyzer* is changed (by adding functions) to generate an executable query plan.

**Optimization**: The generated query plan is optimized with a function which can produce optimal query plan according to the spatial predicate and table information.

The RESQUE library will be deployed as shared library, and a path to this library will be provided to hive to invoke functions in the library via RANSFORM mechanism.