GEO_DISTANCE Feature Preview

Astra Documentation

The GEO_DISTANCE function is a new CQL operator capable of searching an indexed vector column of latitude and longitude pairs in StorageAttachedIndexes for locations within a queried radius.

Given the following table:

```
CREATE TABLE test.table1 (id text PRIMARY KEY, coordinates
vector<float, 2>);
```

We can create an SAI index using the `euclidean` similarity function:

```
CREATE CUSTOM INDEX ON test.table1(coordinates) USING
'org.apache.cassandra.index.sai.StorageAttachedIndex' WITH OPTIONS =
{ 'similarity_function': 'euclidean' };
```

Insert two sample rows:

```
INSERT INTO test.table1 (id, coordinates) VALUES ('New York City',
[40.7128, -74.0060]);
INSERT INTO test.table1 (id, coordinates) VALUES ('Washington DC',
[38.8951, -77.0364]);
```

And you can query like this to get the rows with lat/long pairs within 300 km of Philadelphia (39.9526, -75.1652):

```
SELECT * FROM test.table1 WHERE GEO_DISTANCE(coordinates, [39.9526,
-75.1652]) < 300000;
```

Notice that the distance, or radius, is in meters; to search using miles, multiply the parameter by conversion 1609.34 meters/mile. The above query returns both cities since DC is about 200 KM from Philadelphia and NYC is about 130 KM from Philadelphia. If we restrict the search radius to just 150 KM, we can exclude DC and get just NYC in the result set:

```
SELECT * FROM test.table1 WHERE GEO_DISTANCE(coordinates, [39.9526,
-75.1652]) < 150000;
```