A Demonstration of SpatialHadoop

An Efficient MapReduce Framework for Spatial Data

Mohamed F. Mokbel Ahmed Eldawy Department of Computer Science and Engineering University of Minnesota

Presented By :

Henni Karam

Plan

- Introduction
- SpatialHadoop
- SpatialHadoop's architecture
- Language Layer
- Storage Layer
- MapReduce Layer
- Operations Layer
- Demonstration
- Conclusion

Introduction

Nowadays,there is a recent explosion of spatial datasets generated by different sources.

Hadoop is ill-equipped for supporting spatial data because it focuses mainly on specific data types and operations.

SpatialHadoop as the first full-fledged MapReduce framework with native support for spatial data

Introduction

HADOOP :

```
Objects = LOAD 'points' AS (id:int, x:int, y:int);
FILTER Objects BY x < x2 AND x > x1 AND y < y2 AND y > y1;
```

SPATIALHADOOP :

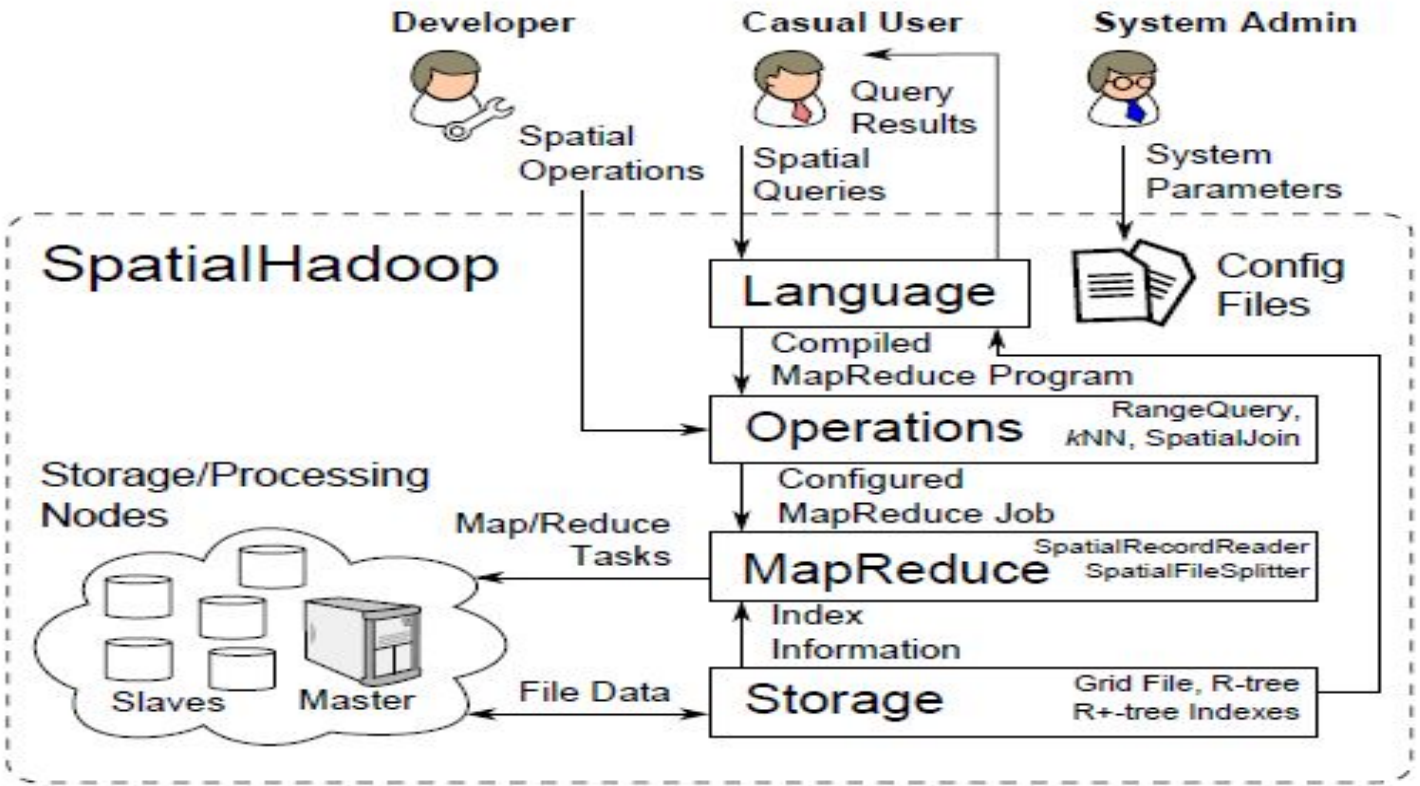
```
Objects = LOAD 'points' AS (id:int, Location:POINT);
FILTER Objects BY Overlaps (Location, Rectangle(x1, y1, x2, y2));
```

SpatialHadoop :

SpatialHadoop is a comprehensive extension to Hadoop that pushes spatial constructs and the awareness of spatial data inside Hadoop code base.

SpatialHadoop pushes its spatial constructs in all layers of Hadoop, namely, **language**, **storage**, **MapReduce** and **operations** layers.

SpatialHadoop's Architecture:



Language layer:

Spatialhadoop provides a built-in support for spatial data types, spatial primitive functions, and spatial operations.

Datatypes : POINT,RECTANGLE,POLYGON

Spatial Primitive Functions : Distance,Overlaps,MBR

Spatial operations : range query, k-nearest neighbor, and spatial join

Language layer:

SpatialHadoop extends Pig Latin by adding new spatial constructs while preserving the original functionality.

SpatialHadoop language overrides the keywords FILTER and JOIN to perform range query and spatial join.

```
houses = LOAD ’houses’ AS (id:int, loc:point);
nearest_houses = KNN houses WITH_K=100
USING Distance(loc, query_loc);
```

Storage layer:

SpatialHadoop adds new spatial indexes that are well adapted for the MapReduce environment.

This new technology is built in order to overcome the following challenges :

- traditional indexes are designed for the procedural programming paradigm.
- traditional indexes are designed for local file systems.

Storage layer:

SpatialHadoop implemented a new technology ; it organises it index's into two level :

- Global indexing
- Local Indexing

The global index is stored in the master node while each local index is stored in a one file block of (64MB) in a slave node.

MapReduce Layer:

SpatialHadoop introduces two new components in the MapReduce layer :

- SpatialFileSplitter
- SpatialRecordReader.

MapReduce Layer:

SpatialFileSplitter :

The SpatialFileSplitter takes as input one or two spatially indexed files in addition to a user provided filter function.Then, it uses the global index to prune file blocks that do not contribute to the query answer

MapReduce Layer:

SpatialRecordReader:

The SpatialRecordReader utilizes the local index by allowing records in one block to be accessed through the local index instead of iterating over all records one-by-one.

Operations Layer:

Range query, **kNN**, and **spatial join** as three case studies of how to exploit the new storage and MapReduce layers in SpatialHadoop

Operations Layer:

Range query :

- the SpatialFileSplitter uses the global index to select only the partitions that overlap the query rang
- Each of the selected partitions goes through a SpatialRecordReader
- It executes a traditional range query on that index to find matching records
- The reference point duplicate avoidance technique is employed on the matching records to ensure that each answer record is reported exactly once.

Operations Layer:

k-nearest-neighbor: The operation is carried out in two iterations :

- the SpatialFileSplitter uses the global index to select the partition that contains the query point.
- The local index in that partition is extracted and used to find the kNN in that partition
- a test circle is drawn with the query point as the center and the distance to the kth neighbor as radius
- If it overlaps with other partitions, a second iteration is carried out to process those overlapping partitions.

Operations Layer:

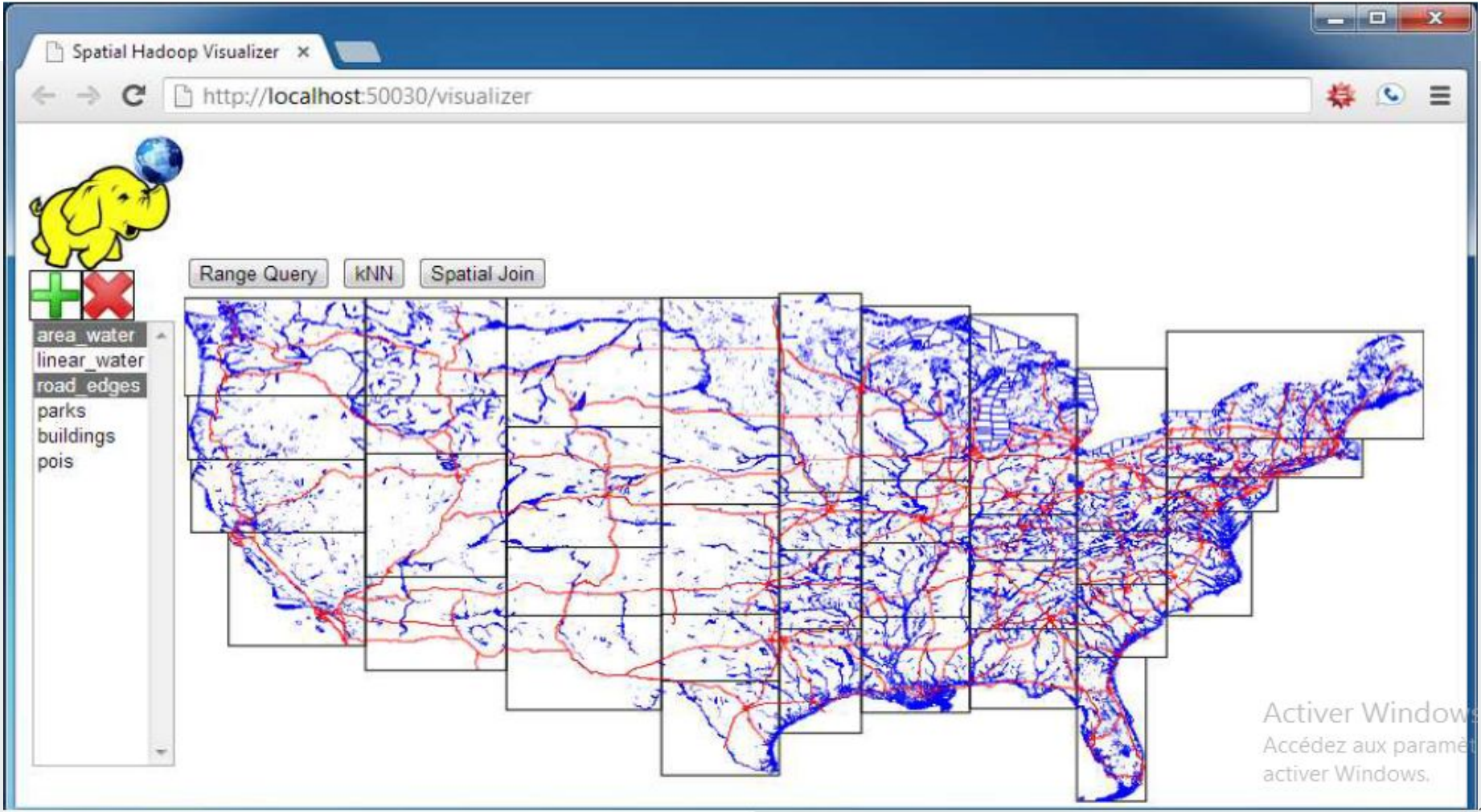
Spatial join :

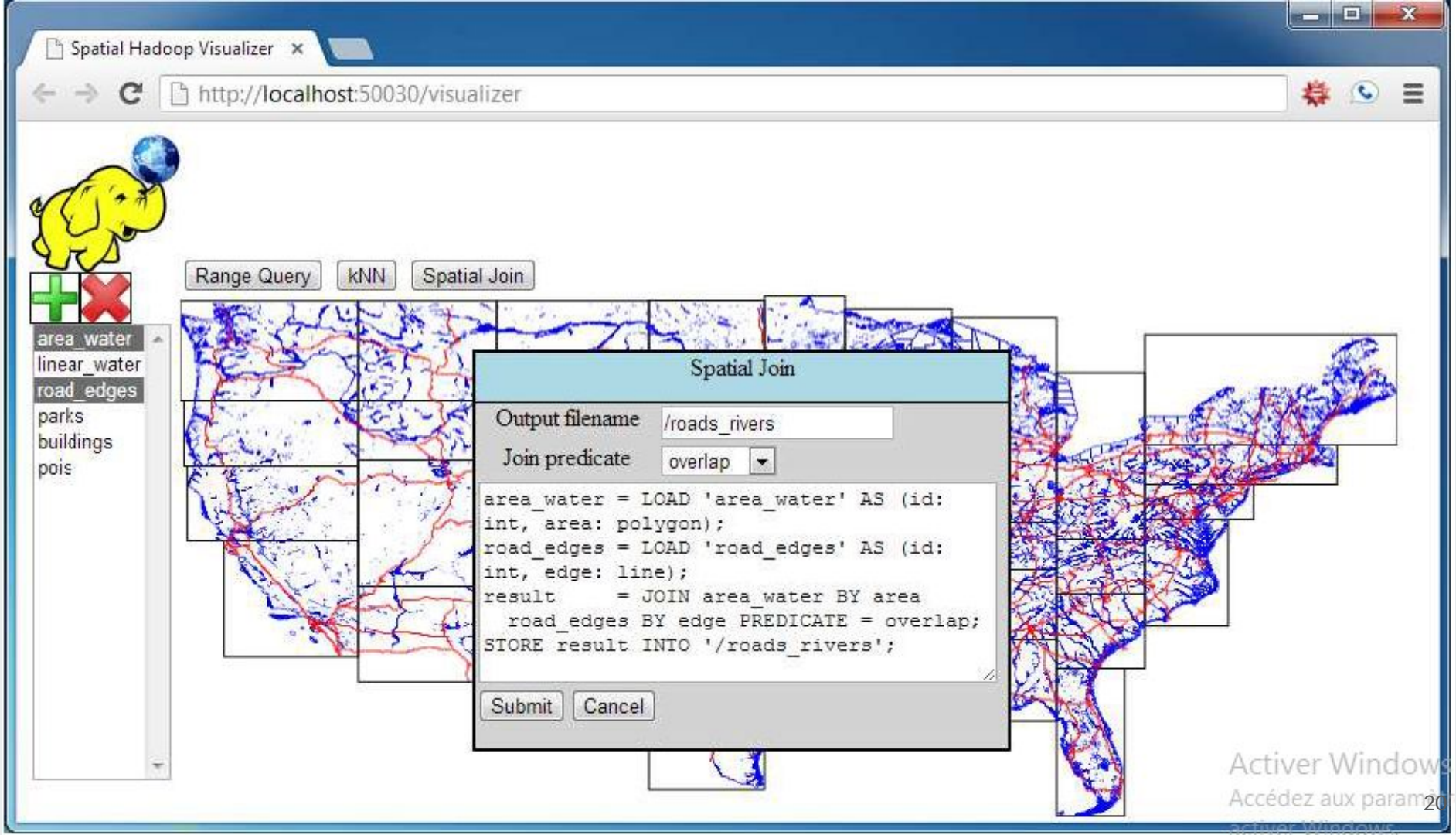
- The SpatialFileSplitter uses the two global indexes in both files to find all pairs of overlapping partitions.
- Each pair of overlapping partitions is processed by a SpatialRecordReader, which uses the local indexes in both files to find overlapping records.

Demonstration :

A prototype system was deployed in order to demonstrate how SpatialHadoop works .

The cluster is loaded with two real datasets obtained from Tiger files and OpenStreetMap


Spatial Hadoop Visualizer
http://localhost:50030/visualizer
Range Query
kNN
Spatial Join
area_water
linear_water
road_edges
parks
buildings
pois
Activer Windows
Accédez aux paramèt
activer Windows.


Spatial Hadoop Visualizer
http://localhost:50030/visualizer
Range Query
kNN
Spatial Join
area_water
linear_water
road_edges
parks
buildings
pois
Spatial Join
Output filename
/roads_rivers
Join predicate
overlap
area_water = LOAD 'area_water' AS (id: int, area: polygon);
road_edges = LOAD 'road_edges' AS (id: int, edge: line);
result = JOIN area_water BY area road_edges BY edge PREDICATE = overlap;
STORE result INTO '/roads_rivers';
Submit
Cancel
Activer Windows
Accédez aux paramè
activer Windows.

Conclusion:

SpatialHadoop is an efficient and a great addition to GeoSpatial data analysis .

It is dedicated to be implemented in GIS that contains a lot of data to process , with it’s geospatial oriented functions .SpatialHadoop has proven itself as the first goto big-data analysis framework

References :

A Demonstration of SpatialHadoop: An Efficient MapReduce Framework for Spatial Data.

Mohamed F. Mokbel Ahmed Eldawy Department of Computer Science and Engineering

University of Minnesota