



Processing Spatial Data with Spatial Hadoop

SNo	Name	USN	Class/Section
1	Sanath Bhimsen	01FB15ECS260	E
2	Sandeep Mysore	01FB15ECS262	E
3	Roshan U	01FB15ECS246	E

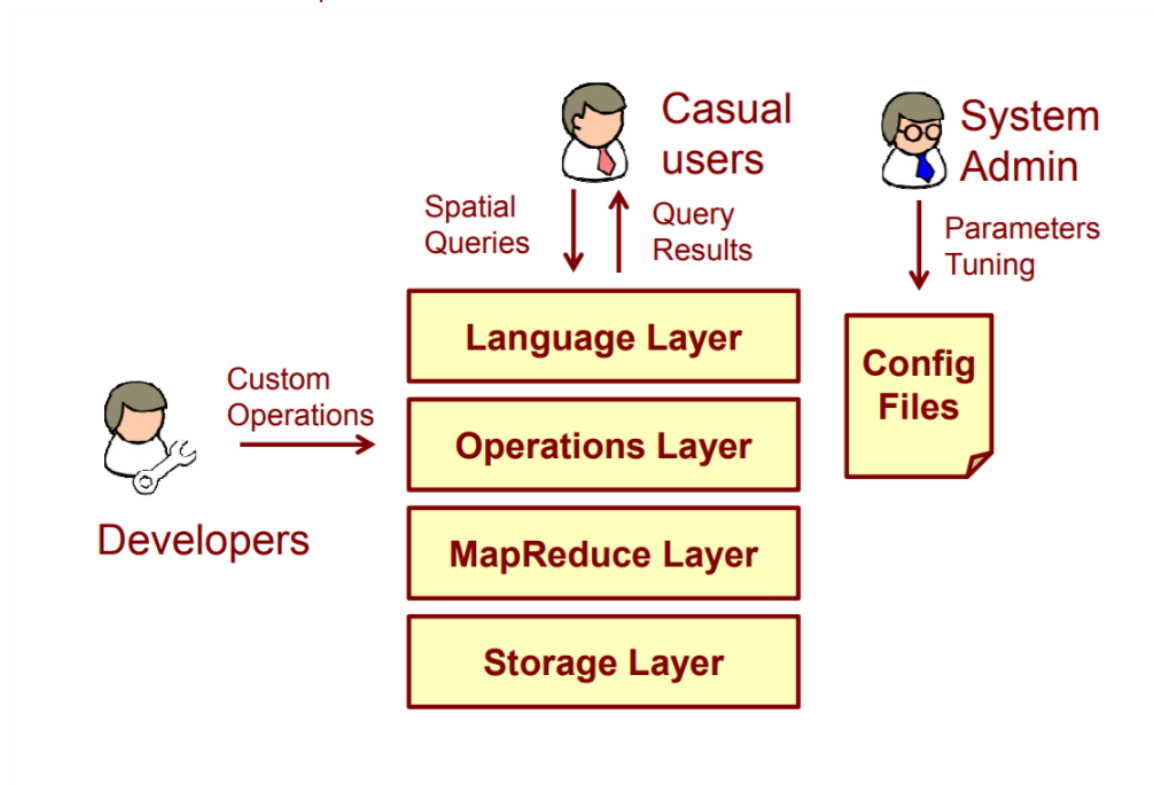
Introduction

- SpatialHadoop; a full-fledged MapReduce framework with native support for spatial data. SpatialHadoop is a comprehensive extension to Hadoop that injects spatial data awareness in each Hadoop layer, namely, the language, storage, MapReduce, and operations layers.

Related work

- <http://spatialhadoop.cs.umn.edu/>

ALGORITHM/DESIGN



Spatial Hadoop comprises of 4 layers .

In the language layer Spatial Hadoop adds a high level language for processing Spatial data types.

In the storage layer it adopts traditional Spatial index structures namely Grid, R-tree, R+tree to form a two level Spatial index (Global and Local indexing). These are used to make processing more efficient.

On top of the MapReduce layer Spatial Hadoop adds two new components called Spatial File Splitter and Spatial Record Reader to make data processing more scalable and efficient.

Spatial Hadoop is well equipped with Spatial operations like range query, Knn and Spatial join.

EXPERIMENTAL RESULTS

We generated rectangle datasets by using the spatial hadoop commands to generate rectangle files of various file sizes ranging from 100MB to 4GB.

The command used was:

```
shadoop generate rect1 mbr:0 , 0 , 1000000, 1000000 size:1.gb  
shape:rect
```

This command generates rectangle file of size 1GB

Now we index the above file by using the command :

```
shadoop index test test.grid mbr:0 , 0 , 1000000, 1000000 sinned:grid  
shape:rect
```

This command generated a grid index for the above file, similarly we can do the same for the R-tree and the R+tree index.

Now we perform the operations on the indexed file.

Range query is done by

```
shadoop rangequery test.grid rq_results rect:500, 500, 1000, 1000  
shape:rect
```

Knn query is done by

```
shadoop knn test.grid knn_results point:1000, 1000 k:1000 shape:rect
```

On the grid indexed file we perform the operations for all the file sizes we mentioned and we obtained the results and visualized it by a plot.

From the plots, we observe that the indexing time increases as the file size increases for all the indexing methods.

The time taken by grid indexing is in the order of magnitude of 5 for a file size of 4GB, whereas the R-tree and R+tree goes until the order of magnitude of even 7.

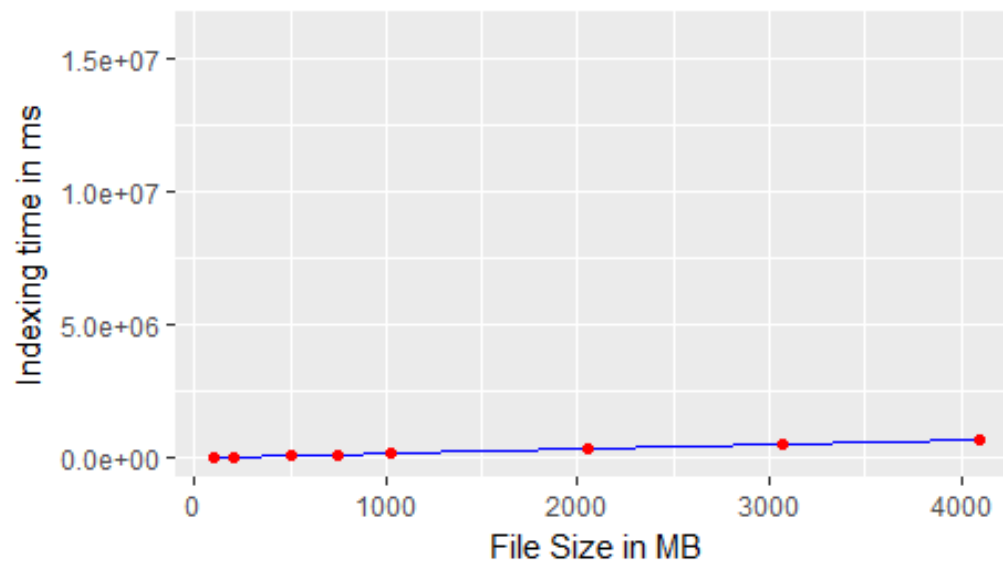
We also find that the grid indexing is more consistent and it takes lesser time when compared to the other two.

R-tree and R+ tree indexing methods is almost the same.

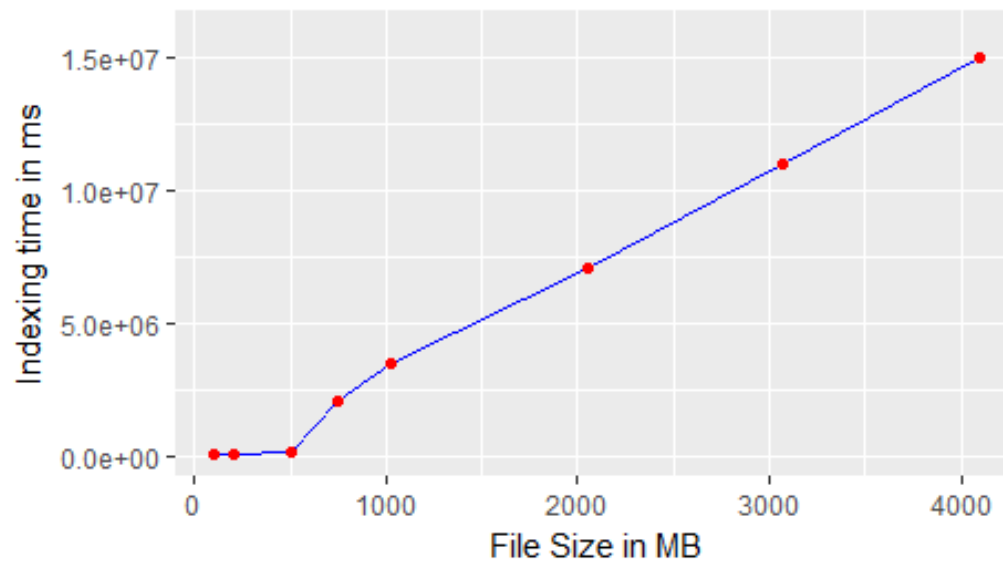
Plots on operations are displayed below

Based on these plots, we can infer that R-tree and R+ tree are more efficient when performing range queries (Blue line) as it uses global

GRID Indexing



R+tree Indexing



Rtree Indexing



indexing to prune out blocks that don't contribute to the answer (query area).

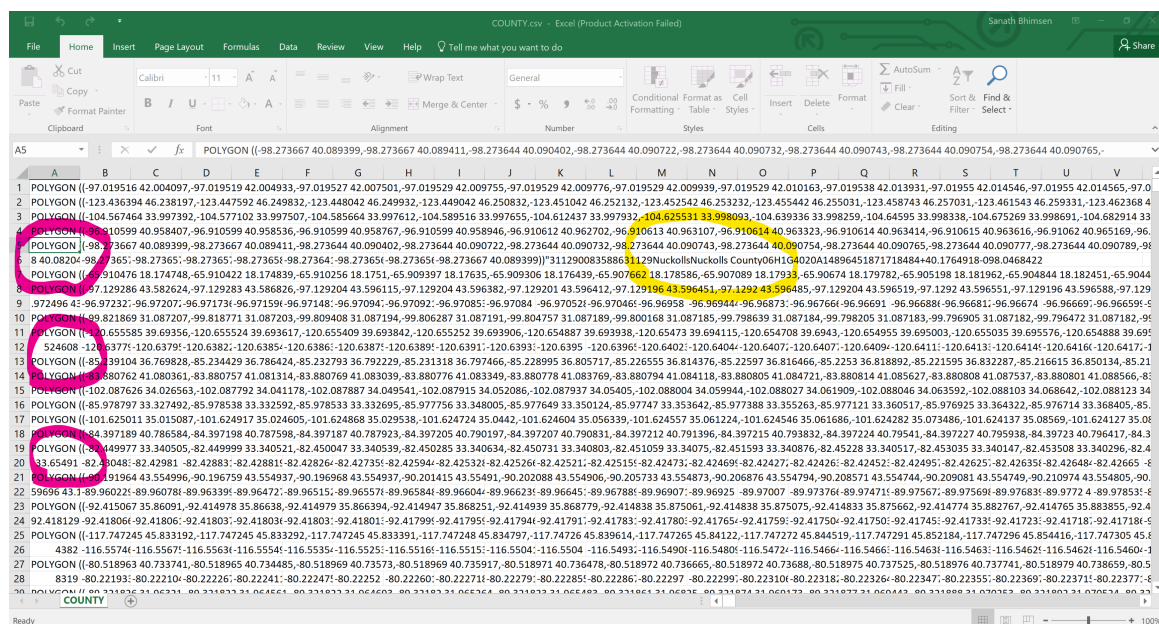
The Knn query (red line) is almost the same in all the 3 indexing methods

From the Spatial Hadoop website we downloaded Spatial datasets in order to perform the same operations on the files after indexing them.

The spatial datatypes/shapes that spatial Hadoop recognizes are Point , Rectangle , Polygon and Tiger(For Tiger Datasets).

We need to specify the shape while entering the command on the terminal.

The datasets are large and cannot be viewed in excel , but we were able to view one tiger dataset called COUNTY.csv.



The screenshot shows an Excel spreadsheet titled "COUNTY.csv - Excel (Product Activation Failed)". The spreadsheet contains a large table of coordinates for a polygon. The first row is a header row with columns labeled A through V. The data rows start with "POLYGON ((-98.273667 40.089399, -98.273667 40.089411, -98.273644 40.090402, -98.273644 40.090722, -98.273644 40.090732, -98.273644 40.090743, -98.273644 40.090754, -98.273644 40.090765, -98.273667 40.089399))". The coordinates are listed in a long, continuous string. A red circle highlights a specific row of data, which is the 10th row in the visible area. The coordinates in this row are: -98.273667 40.089399, -98.273667 40.089411, -98.273644 40.090402, -98.273644 40.090722, -98.273644 40.090732, -98.273644 40.090743, -98.273644 40.090754, -98.273644 40.090765, -98.273667 40.089399.

Operations on grid indexed files



Operations on r+tree indexed files



Operations on rtree indexed files



The Java Program that partitions the data looks for the shape (Polygon , linestring ,rectangle) in the first column of every row followed by spatial coordinates in the the rest of the row which denote a record.

For indexing the above file it looks for shape=Polygon in the first column of every row as per the WKT(Well Known Text) format.

But , from the dataset we can see some lines start with coordinates and some random text values instead of coordinates(Highlighted Yellow).

This is causing the spatial indexing to run improperly or halt suddenly.

FUTURE ENHANCEMENTS

- To process a large Spatial data set efficiently and get satisfactory output
- Our bigger goal would be to take satellite images in raster format and deploy them onto Spatial Hadoop, by converting it into vector format.

REFERENCES

- Eldawy, Ahmed, Yuan Li, Mohamed F. Mokbel, and Ravi Janardan. "CG_Hadoop: computational geometry in MapReduce." In *Proceedings of the 21st ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*, pp. 294-303. ACM, 2013.
- Eldawy, Ahmed, Louai Alarabi, and Mohamed F. Mokbel. "Spatial partitioning techniques in SpatialHadoop." *Proceedings of the VLDB Endowment* 8, no. 12 (2015): 1602-1605.
- <https://youtu.be/vM9OT7QUW4E>

EVALUATIONS (Leave this for the faculty)

Date	Evaluator	Comments	Score

CHECKLIST

SNo	Item	Status
	Source code documented	
2	Source code uploaded to CCBD server	
3	Recorded video of demo	
4	Instructions for building and running the code. Your code must be usable out of the box. Link to your gitlab account	
5	Dataset used for project uploaded. Please include a description of the dataset format. This includes input file format.	
6	Poster of your project	