M.Tech. Thesis Presentation

Spatial Data: Crawling, Metadata Discovery, Publishing & Query Orchestration

Submitted by
Deepak Punjabi
15IT60R17

Under Guidance of
Prof. Soumya K. Ghosh

Introduction

- Definitions & Terminology

Spatial Data

Spatial data

is data containing Information about

the locations and shapes

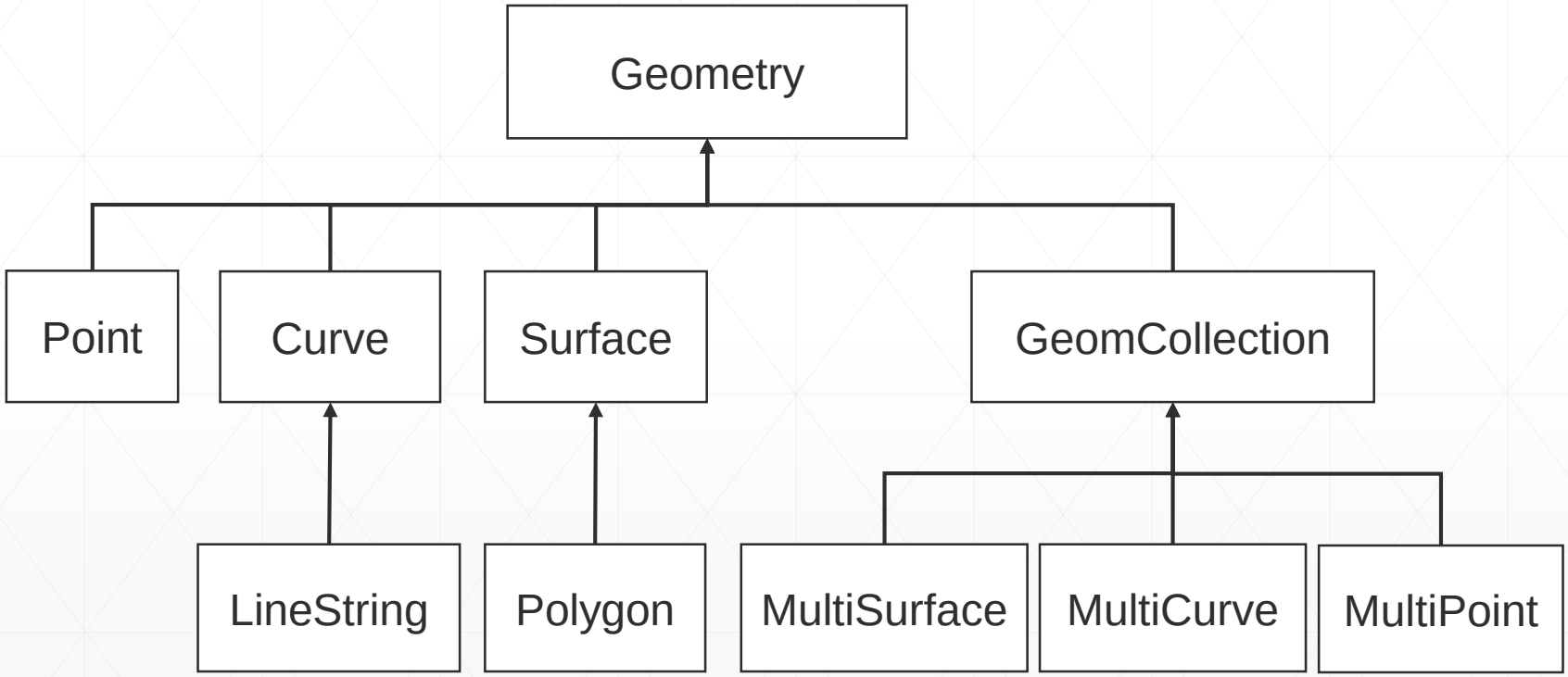
of geographic features

and the relationships between them,

usually stored as coordinates and topology.

Source: http://support.esri.com/

Spatial Data Types



Spatial Web Services

Web Map Service (WMS)

Web Feature Service (WFS) Web Coverage Service (WCS)

Problem Statement

- Defining the problem
- Motivation
- Defining the solution objectives
- Solution Model

Aim

Create a foundation platform

to crawl, store, maintain, and publish

metadata information about spatial data

and it's providers and to utilize this information

to perform efficient query orchestration.

Motivation

Use Cases:

- Remote Sensing
- Area affected by flood/disease
- Spatial-Temporal analysis
- Spatial data mining
- Telecom & Network Services
- Urban Planning and Hot spot analysis
- Navigation
- And many more…

Challenges

- Availability: not all spatial data is publicly available
- Current Solutions: Available search engines do not handle spatial data efficiently
- Heterogeneity: Spatial data has complex data types and operations

Objectives

1. Build a topical crawler to crawl the web and store geo-spatial metadata.
2. Build an Open Geospatial Consortium (OGC) compliant catalog service to publish and search accumulated metadata.
3. Build Query orchestration service to perform real-time query with heterogeneous data sources and cost matrices associated with them.

Solution Model:
3 stage approach

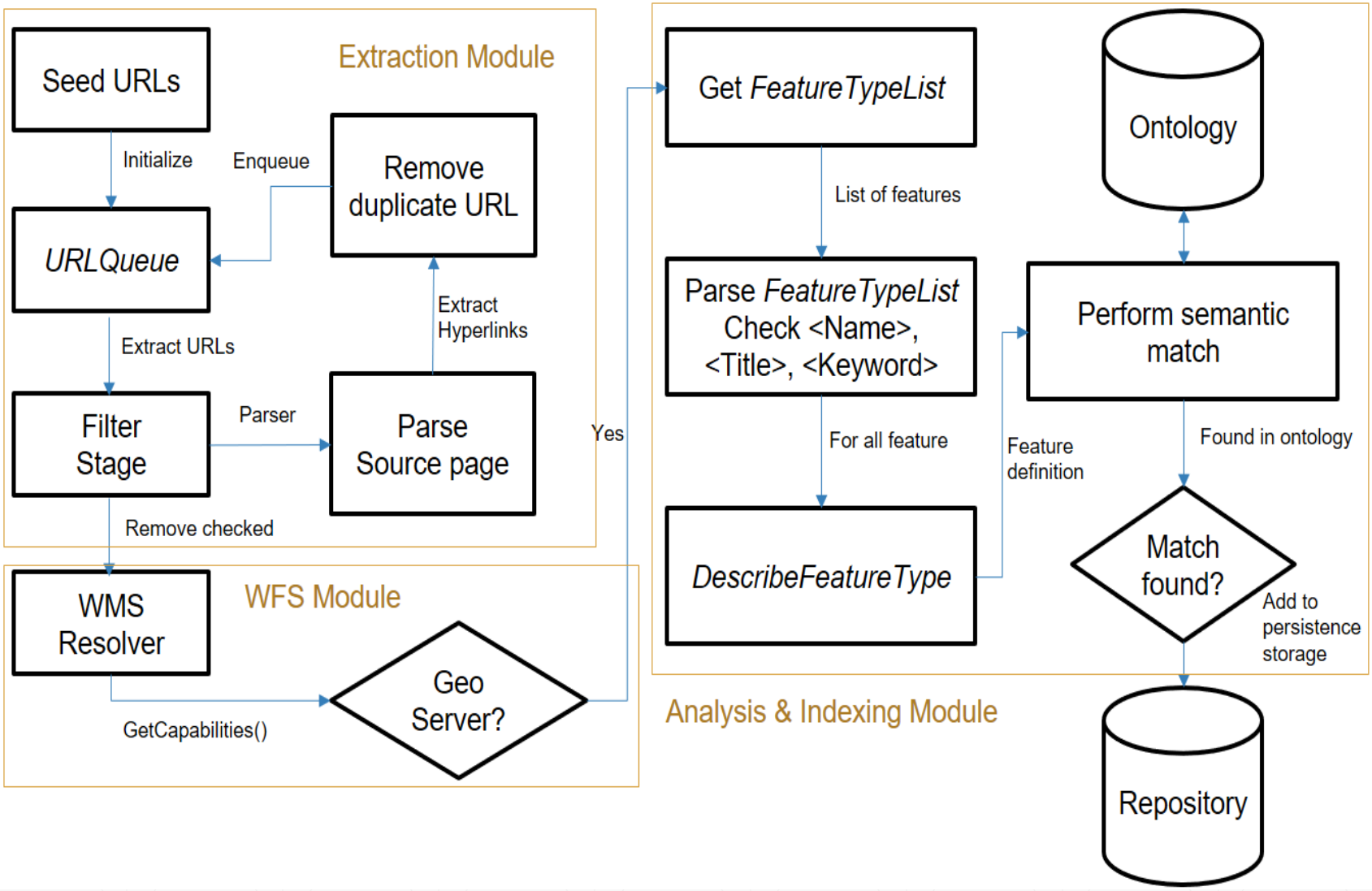
Spatial Web Crawler

- Objectives
- Architecture

Spatial web crawler: Objectives

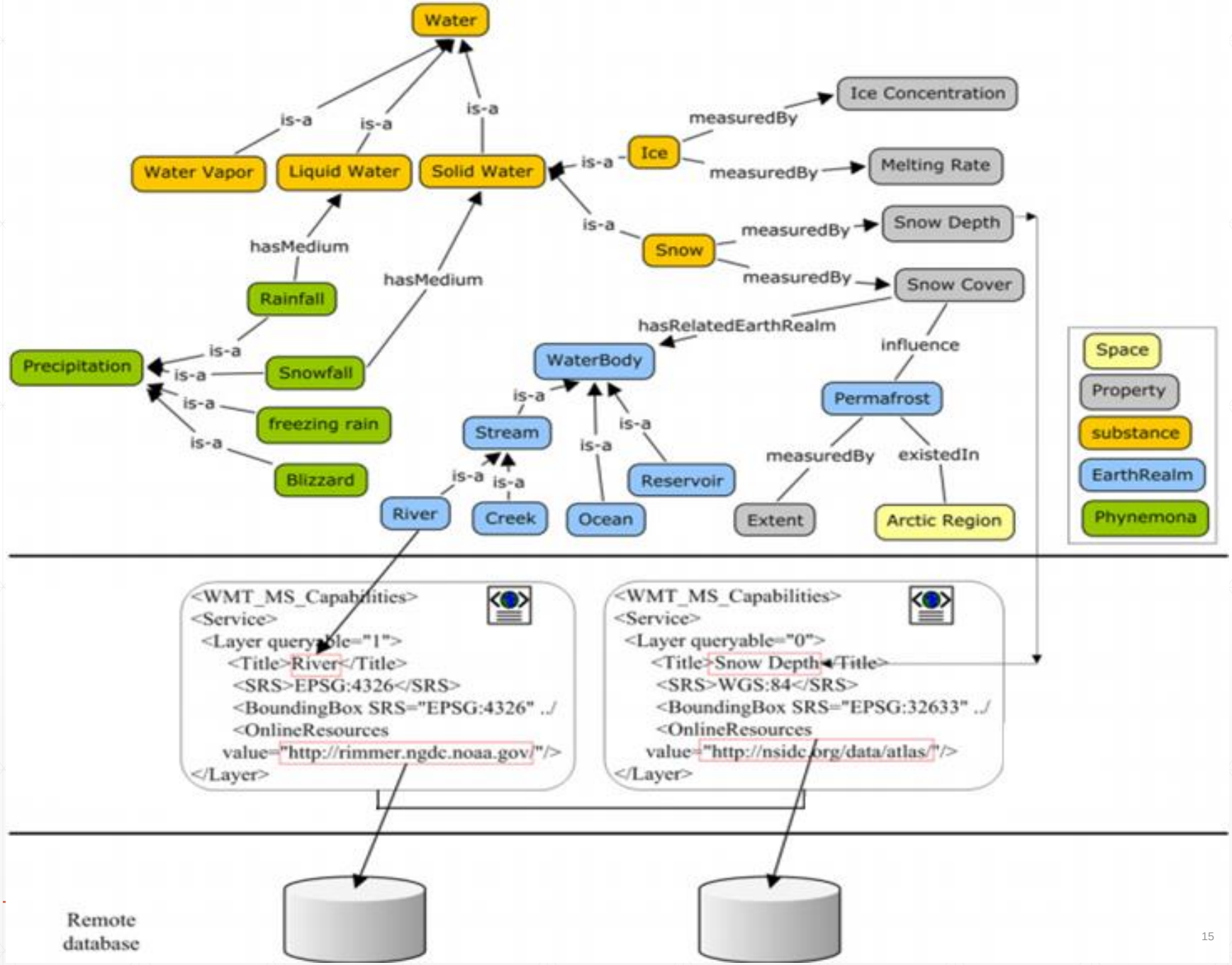
1. Build a spatial web crawler which crawlers through geo-servers which offers Web Feature Service(WFS) based OGC compliant services.
2. Build a domain specific vocabulary(ontology) for this features which can be helpful to compare found features with wanted features.
3. Perform semantic matching of found features from crawled web-pages with given ontology for filtering the correct features and storing them in the permanent repository.

Spatial web crawler: Architecture



WFS Capabilities


```
− <FeatureType>
    <Name>kgp:bnk_block_hq</Name>
    <Title>bnk_block_hq</Title>
    <Abstract/>
  − <ows:Keywords>
      <ows:Keyword>features</ows:Keyword>
      <ows:Keyword>bnk_block_hq</ows:Keyword>
    </ows:Keywords>
    <DefaultSRS>urn:x-ogc:def:crs:EPSG:4326</DefaultSRS>
  − <ows:WGS84BoundingBox>
      <ows:LowerCorner>86.78687987110824 22.7698</ows:LowerCorner>
      <ows:UpperCorner>87.6219 23.5656</ows:UpperCorner>
    </ows:WGS84BoundingBox>
  </FeatureType>
− <FeatureType>
    <Name>kgp:bnk_district_boundary</Name>
    <Title>bnk_district_boundary</Title>
    <Abstract/>
  − <ows:Keywords>
      <ows:Keyword>features</ows:Keyword>
      <ows:Keyword>bnk_district_boundary</ows:Keyword>
    </ows:Keywords>
    <DefaultSRS>urn:x-ogc:def:crs:EPSG:4326</DefaultSRS>
  − <ows:WGS84BoundingBox>
      <ows:LowerCorner>86.6121826171875 22.626935958862305</ows:LowerCorner>
      <ows:UpperCorner>87.7676773071289 23.635831832885742</ows:UpperCorner>
    </ows:WGS84BoundingBox>
  </FeatureType>
```



source: Li, W., *et al*. Computers & Geosciences,2011.

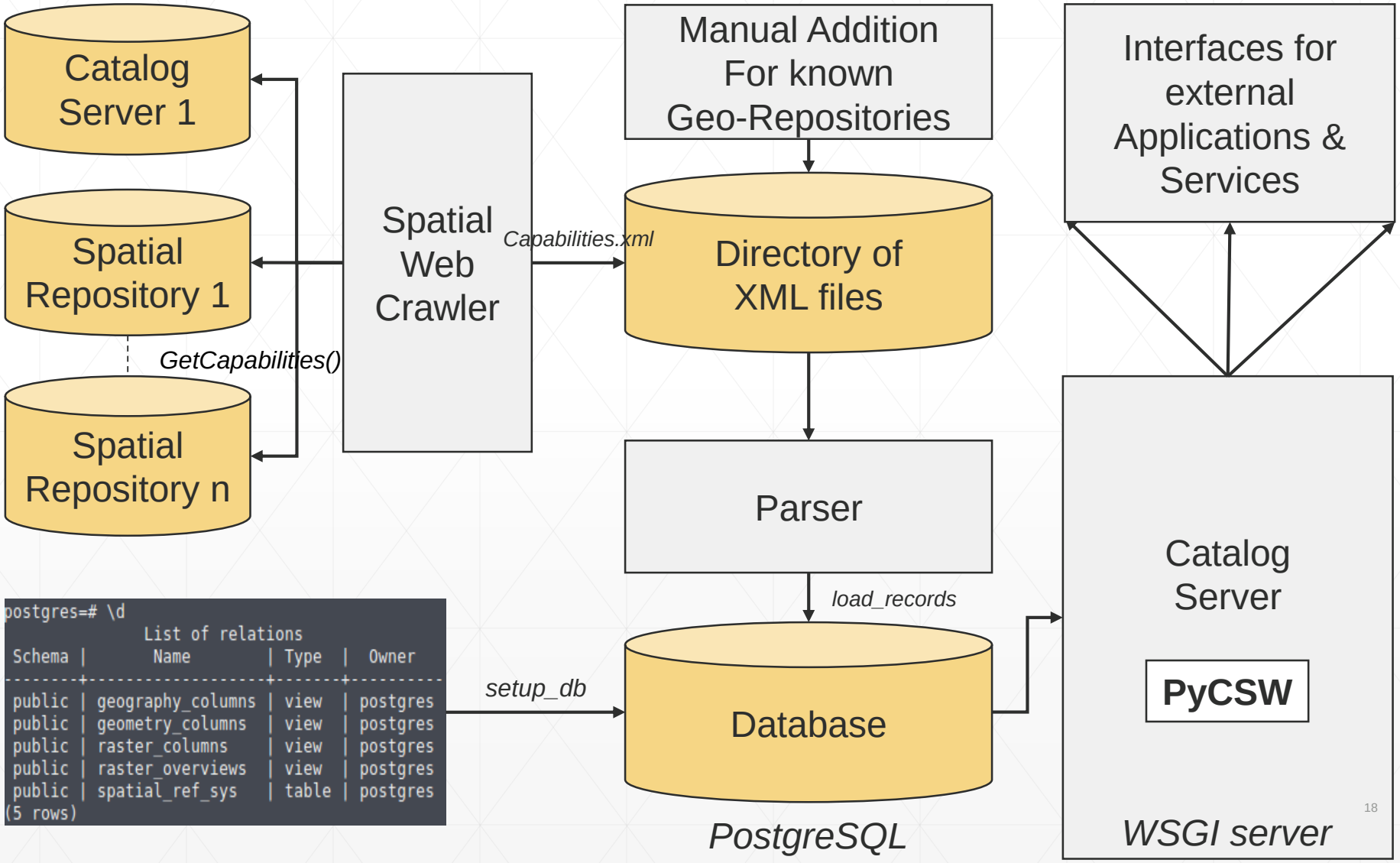
Spatial Catalog Service

- Objectives
- Architecture

Spatial Catalog Service: Objectives

1. Parse the crawled metadata and store it into a permanent database in a structured manner.
2. Parse the crawled metadata and store it into a permanent database in a structured manner.

Architecture: Spatial Catalog Service



Design Choices

PyCSW:

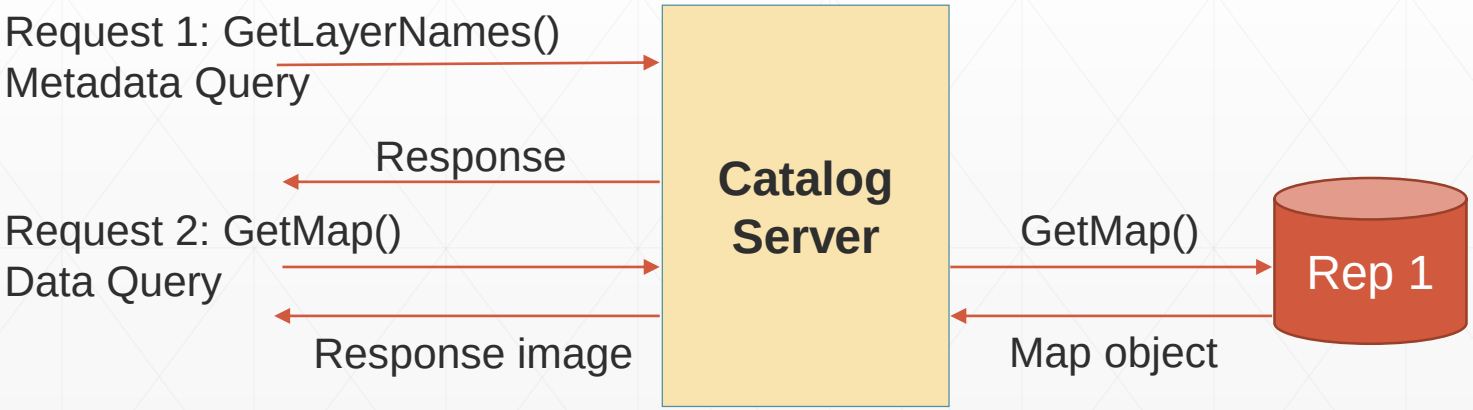
- Stores metadata information
- Less resource intensive
- Provides APIs for spatial web services
- Stores metadata information as XML files.
- Two step delivery for data queries

GeoServer:

- Can store data as well as metadata information
- More resource intensive
- Provides graphical admin panel and APIs for spatial web services
- Supports more file formats like XML, GML for metadata & shape for data
- Can directly response with data
- Provides extensions

PyCSW Query Processor

- Database holds all metadata information about the data available from repositories.
- Type of queries
  - Metadata Query
  - Request for the data object



GeoServer

Username | Remember me | Login

About & Status
About GeoServer
Data
Layer Preview
Demos

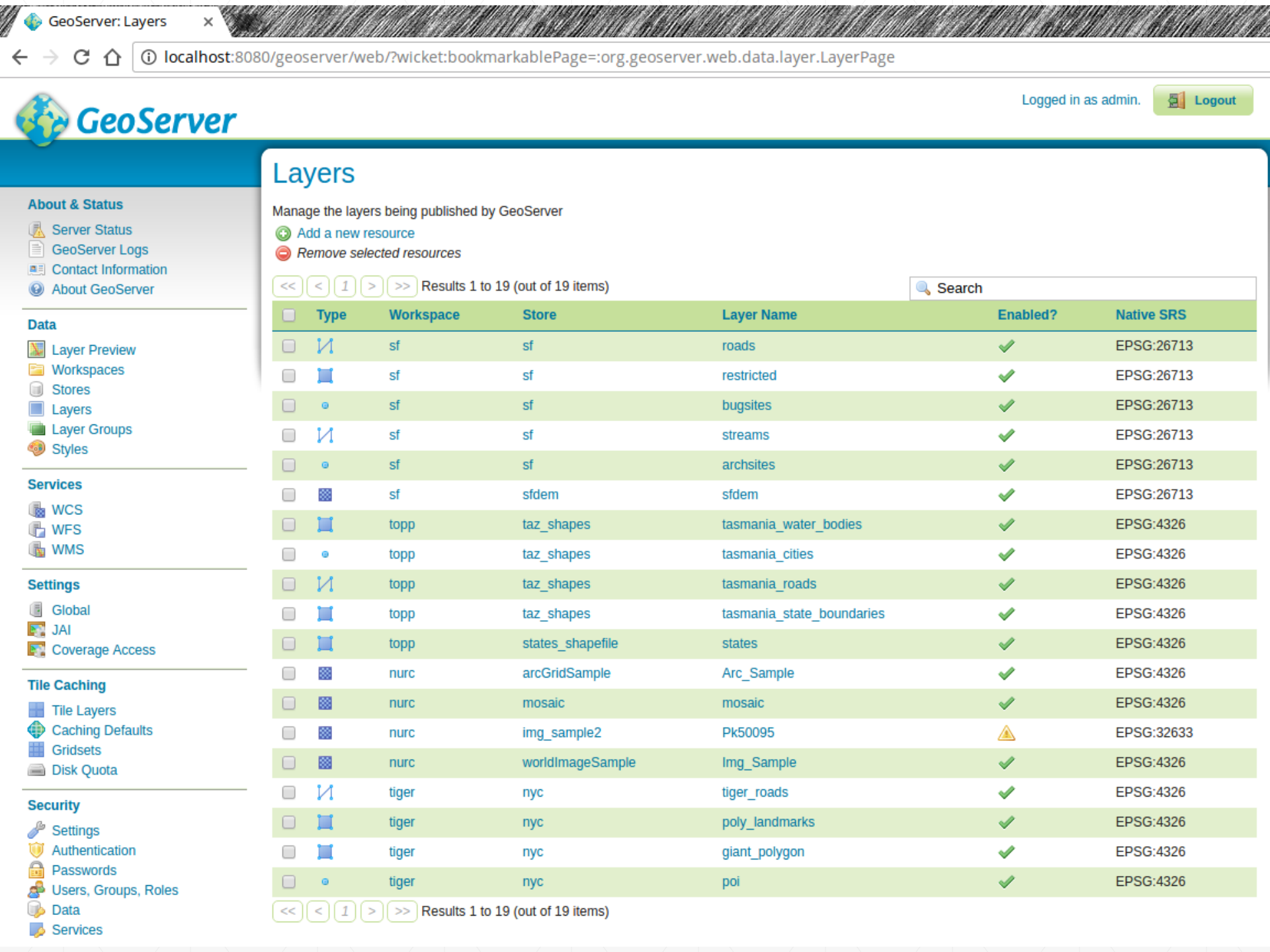
Layer Preview

List of all layers configured in GeoServer and provides previews in various formats for each.

<< < 1 > >> Results 1 to 21 (out of 21 items)

Search

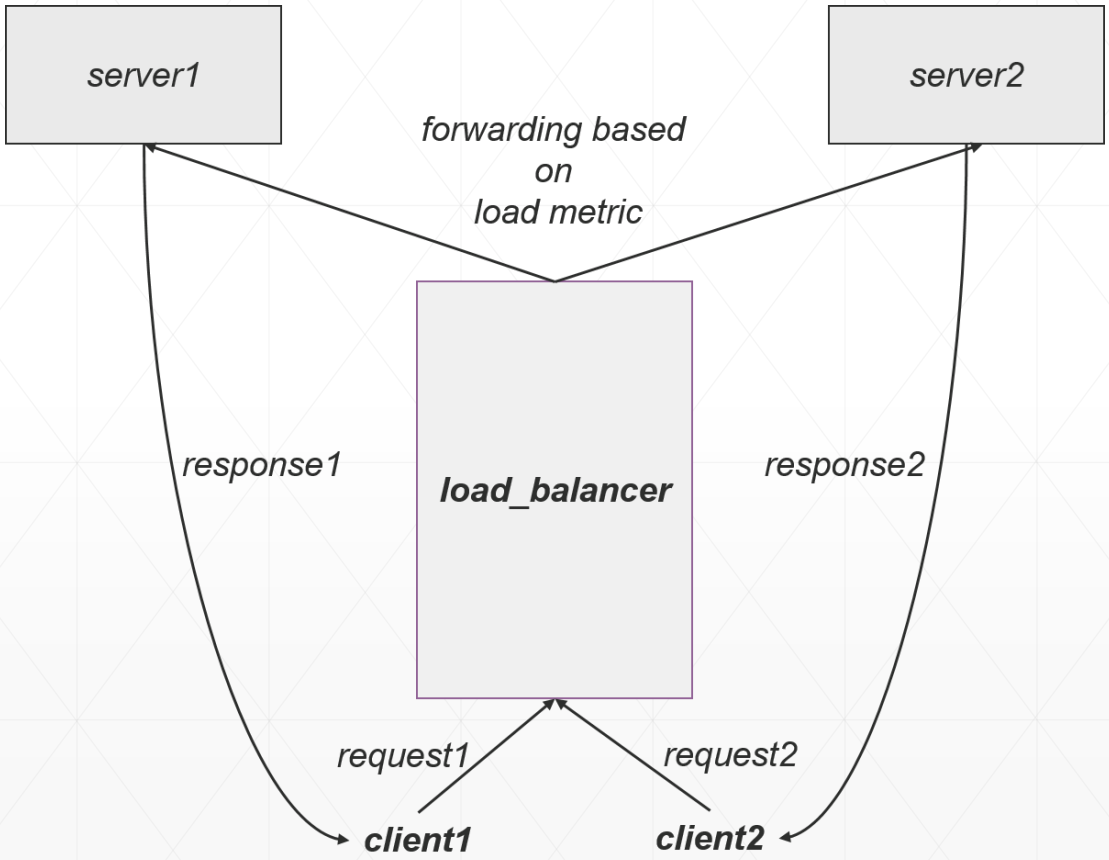
Type	Name	Title	Common Formats	All Formats
	sf:roads	Spearfish roads	OpenLayers KML GML	Select one
	sf:restricted	Spearfish restricted areas	OpenLayers KML GML	Select one
	sf:bugsites	Spearfish bug locations	OpenLayers KML GML	Select one
	sf:streams	Spearfish streams	OpenLayers KML GML	Select one
	sf:archsites	Spearfish archeological sites	OpenLayers KML GML	Select one
	sf:sfdem	sfdem is a Tagged Image File Format with Geographic information	OpenLayers KML	Select one
	topp:tasmania_water_bodies	Tasmania water bodies	OpenLayers KML GML	Select one
	topp:tasmania_cities	Tasmania cities	OpenLayers KML GML	Select one
	topp:tasmania_roads	Tasmania roads	OpenLayers KML GML	Select one
	topp:tasmania_state_boundaries	Tasmania state boundaries	OpenLayers KML GML	Select one
	topp:states	USA Population	OpenLayers KML GML	Select one
	nurc:Arc_Sample	A sample ArcGrid file	OpenLayers KML	Select one
	nurc:mosaic	mosaic	OpenLayers KML	Select one
	nurc:Img_Sample	North America sample imagery	OpenLayers KML	Select one
	tiger:tiger_roads	Manhattan (NY) roads	OpenLayers KML GML	Select one


GeoServer: Layers
localhost:8080/geoserver/web/?wicket:bookmarkablePage=:org.geoserver.web.data.layer.LayerPage
GeoServer
Logged in as admin.
Logout
About & Status
Server Status
GeoServer Logs
Contact Information
About GeoServer
Data
Layer Preview
Workspaces
Stores
Layers
Layer Groups
Styles
Services
WCS
WFS
WMS
Settings
Global
JAI
Coverage Access
Tile Caching
Tile Layers
Caching Defaults
Gridsets
Disk Quota
Security
Settings
Authentication
Passwords
Users, Groups, Roles
Data
Services
Layers
Manage the layers being published by GeoServer
Add a new resource
Remove selected resources
<< < 1 > >> Results 1 to 19 (out of 19 items)
Search
Type Workspace Store Layer Name Enabled? Native SRS
sf sf roads EPSG:26713
sf sf restricted EPSG:26713
sf sf bugsites EPSG:26713
sf sf streams EPSG:26713
sf sf archsites EPSG:26713
sf sfdem sfdem EPSG:26713
topp taz_shapes tasmania_water_bodies EPSG:4326
topp taz_shapes tasmania_cities EPSG:4326
topp taz_shapes tasmania_roads EPSG:4326
topp taz_shapes tasmania_state_boundaries EPSG:4326
topp states_shapefile states EPSG:4326
nurc arcGridSample Arc_Sample EPSG:4326
nurc mosaic mosaic EPSG:4326
nurc img_sample2 Pk50095 EPSG:32633
nurc worldImageSample Img_Sample EPSG:4326
tiger nyc tiger_roads EPSG:4326
tiger nyc poly_landmarks EPSG:4326
tiger nyc giant_polygon EPSG:4326
tiger nyc poi EPSG:4326
<< < 1 > >> Results 1 to 19 (out of 19 items)

Extending to Cloud Characteristics

Load Balancer:

- Higher Availability
- Horizontal Scaling
- Middleware or Gateway
- Access Control



```
niku@slab ~/Desktop/thsis/py
niku@slab ~/Desktop/thsis/py $ python server.py
load balancer running on: 10.14.1.163:7845
senidng reply from GeoServer: 10.14.1.201:8080
OGC:WMS
senidng reply from GeoServer: 10.14.1.163:8080
OGC:WMS
senidng reply from GeoServer: 10.14.1.201:8080
OGC:WMS
senidng reply from GeoServer: 10.14.1.163:8080
OGC:WMS
senidng reply from GeoServer: 10.14.1.201:8080
OGC:WMS
senidng reply from GeoServer: 10.14.1.163:8080
OGC:WMS
```

```
niku@slab ~/Desktop/thsis/py
niku@slab ~/Desktop/thsis/py $ python client.py
Connection Successful
size  1163796
<class 'bytes'>
<owslib.wms.WebMapService object at 0x7f60a53609b0>
tiger-ny
tasmania
spearfish
nurc:Arc_Sample
nurc:Img_Sample
sf:archsites
sf:bugsites
tiger:giant_polygon
nurc:mosaic
tiger:poi
tiger:poly_landmarks
sf:restricted
sf:roads
sf:sfdem
topp:states
sf:streams
topp:tasmania_cities
topp:tasmania_roads
topp:tasmania_state_boundaries
topp:tasmania_water_bodies
tiger:tiger_roads
niku@slab ~/Desktop/thsis/py $ python client.py
Connection Successful
size  1163796
<class 'bytes'>
<owslib.wms.WebMapService object at 0x7f0a538f69b0>
tasmania
spearfish
tiger-ny
nurc:Arc_Sample
nurc:Img_Sample
sf:archsites
sf:bugsites
tiger:giant_polygon
nurc:mosaic
tiger:poi
tiger:poly_landmarks
sf:restricted
sf:roads
sf:sfdem
topp:states
sf:streams
topp:tasmania_cities
```

Results

List of available Layers:

- kgp:POPULATION
- kgp:bnk block boundary
- kgp:bnk block hq
- kgp:bnk district boundary
- kgp:bnk drainage
- kgp:bnk grampanchayat boundary
- kgp:bnk mouza boundary
- kgp:bnk road

List of available Operations

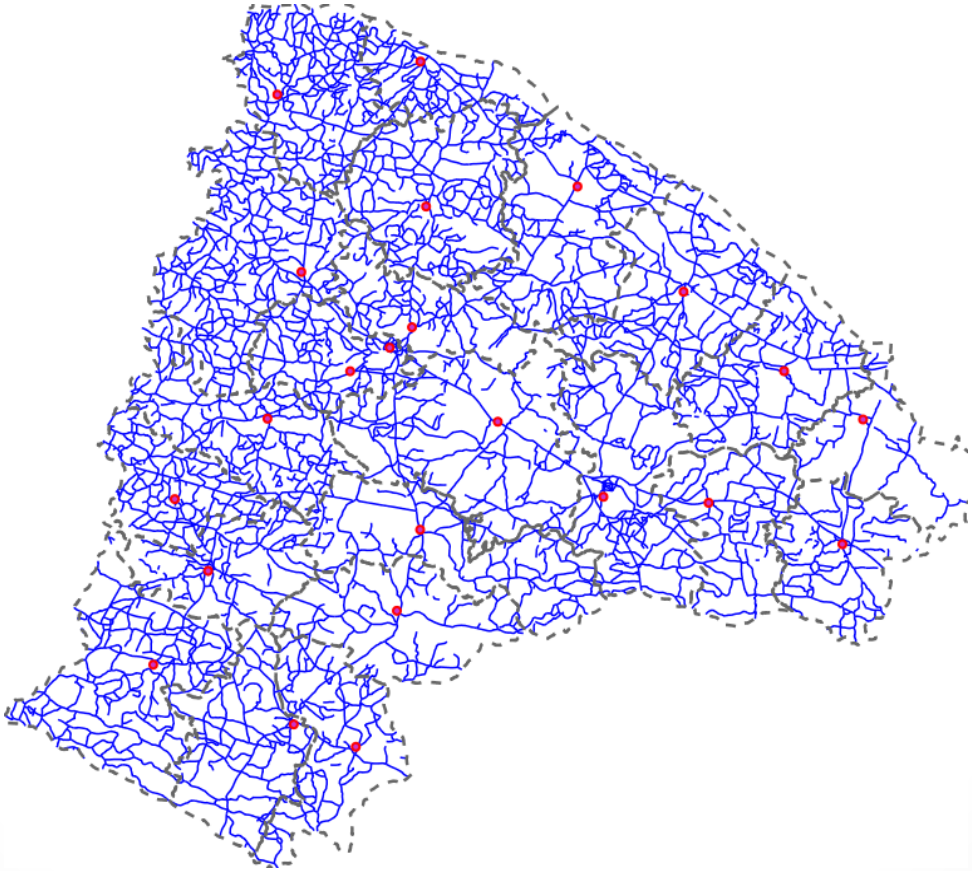
- GetCapabilities
- GetMap
- GetFeatureInfo
- DescribeLayer
- GetLegendGraphic
- GetStyles

GetMap

GetMap returns a map image of the layer(s) in available formats.

Options:

- Layers=kgp:bnk_road
- Width=768
- Height=679
- Format=image/png



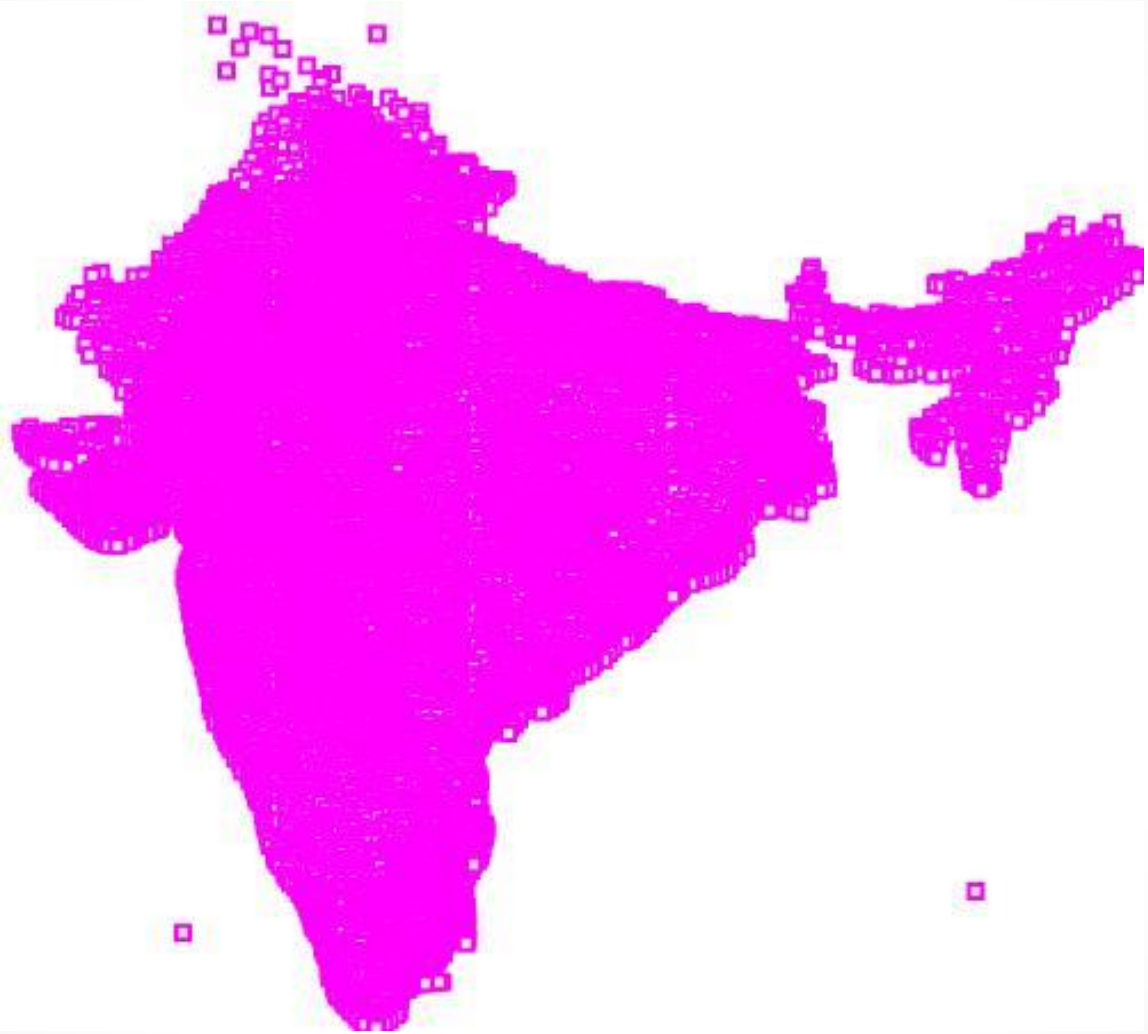
DrawMap

DrawMap Overlays different map images on top of each other.

Useful to find affected area.

Options:

- Layers = { kgp:bnk_road, kgp:bnk_block_hq, kgp:bnk_block_boundary }
- Width = 768
- Height = 679
- Format = image/png



* This image shows population density in India, without any information on boundaries.

Information about specific layer *

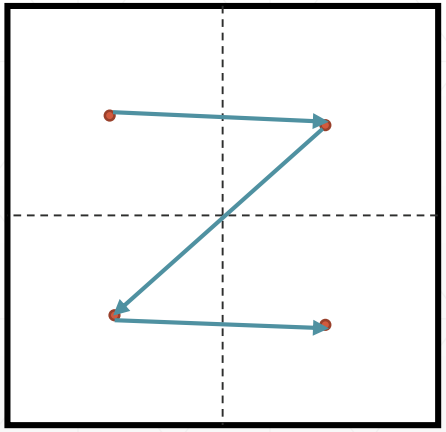
- Title | POPULATION
- Name | kgp:POPULATION
- Is Queryable | 1
- Is Opaque | 0
- Bounding Box |
- minx | 68.52669525146484
- miny | 8.086045265197754
- maxx | 97.3387680053711
- maxy | 35.8697509765625

Spatial Query Orchestration

- Applications of spatial data
- Problems with currently available solution

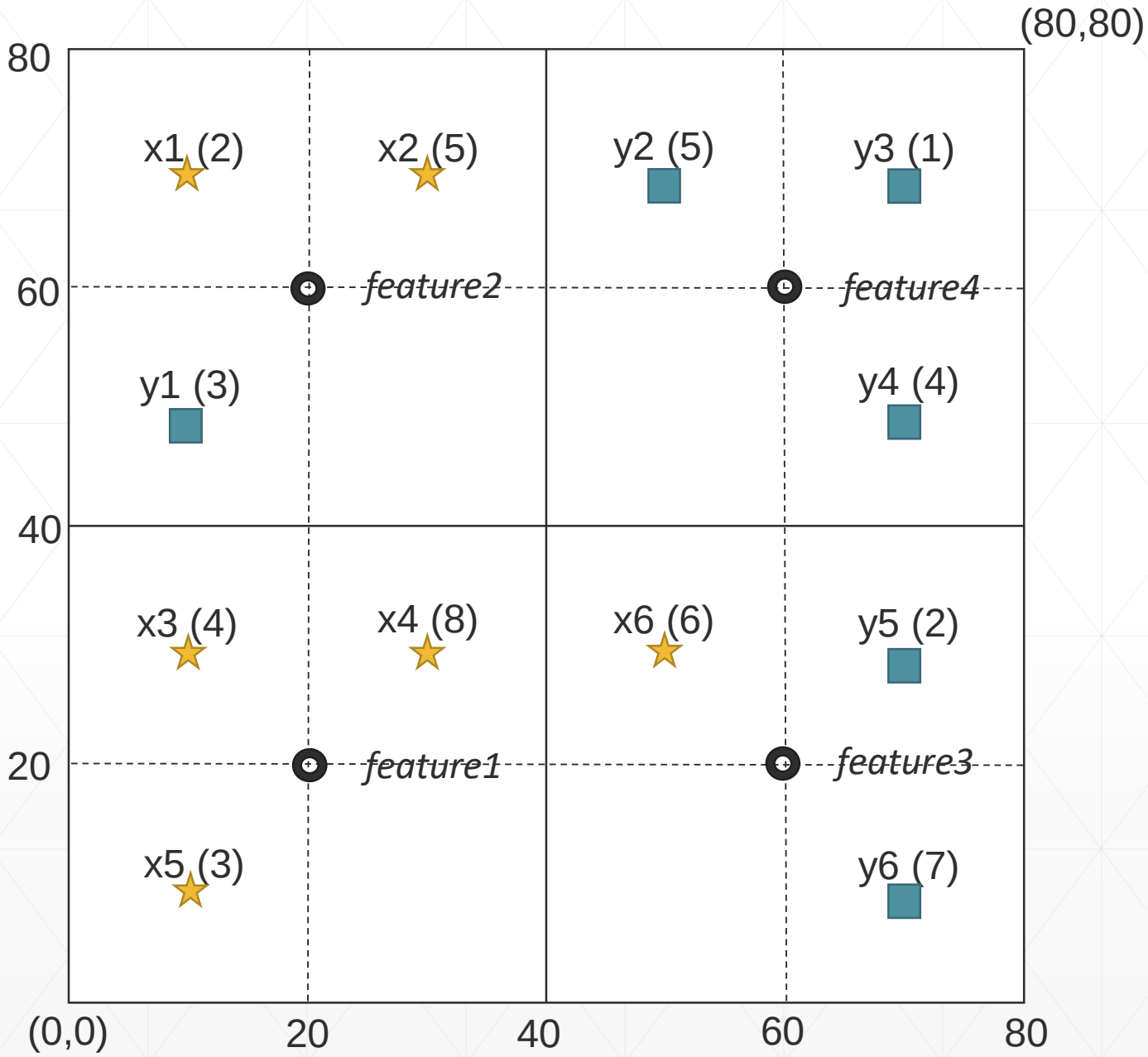
Quadtree based Indexing

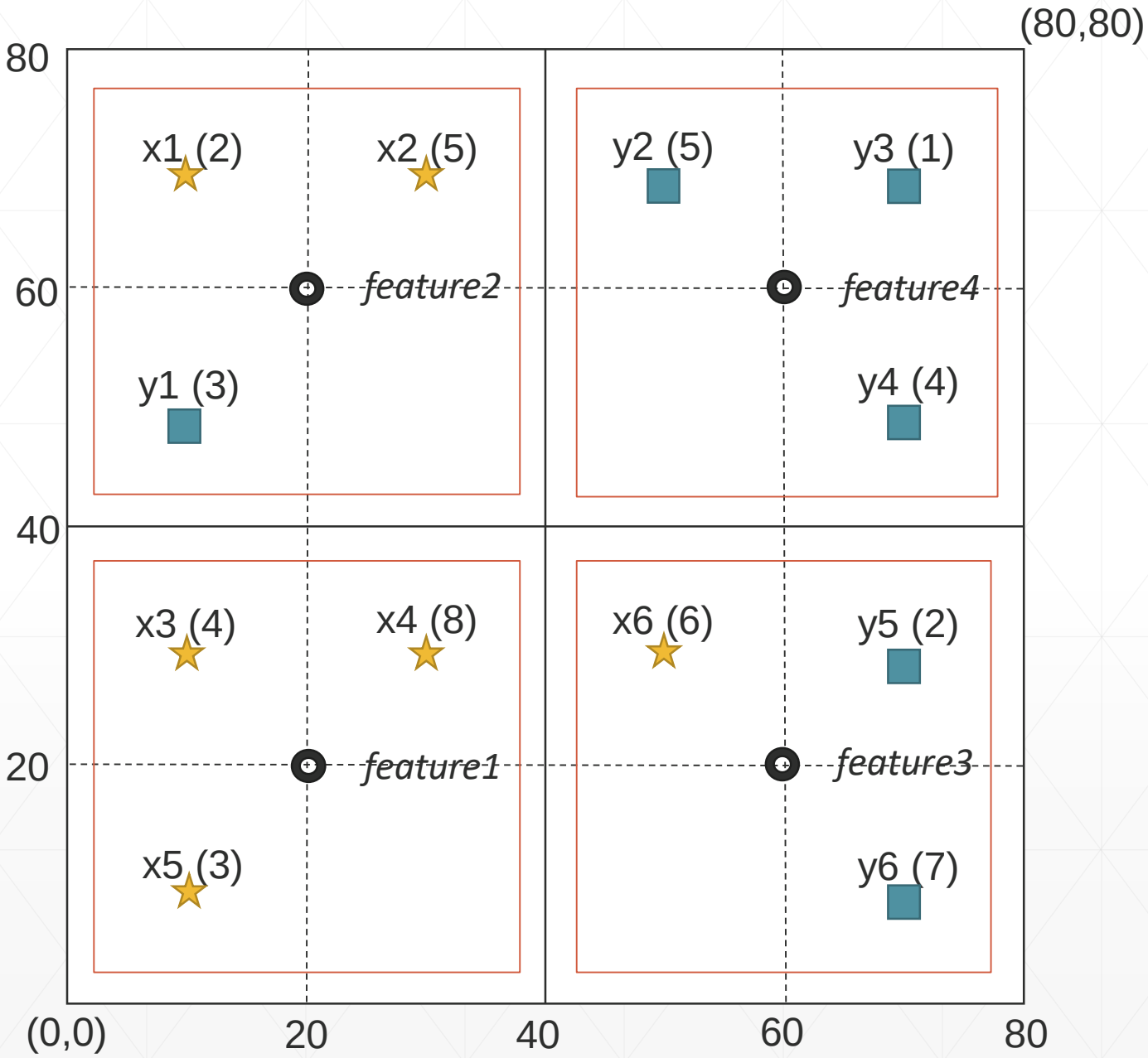
- Quadtree is a special kind of data structure used for spatial division.
- Each node has 4 child nodes.
- Most commonly used indexing technique in Quadtree is Z-order.
- Naive technique for indexing

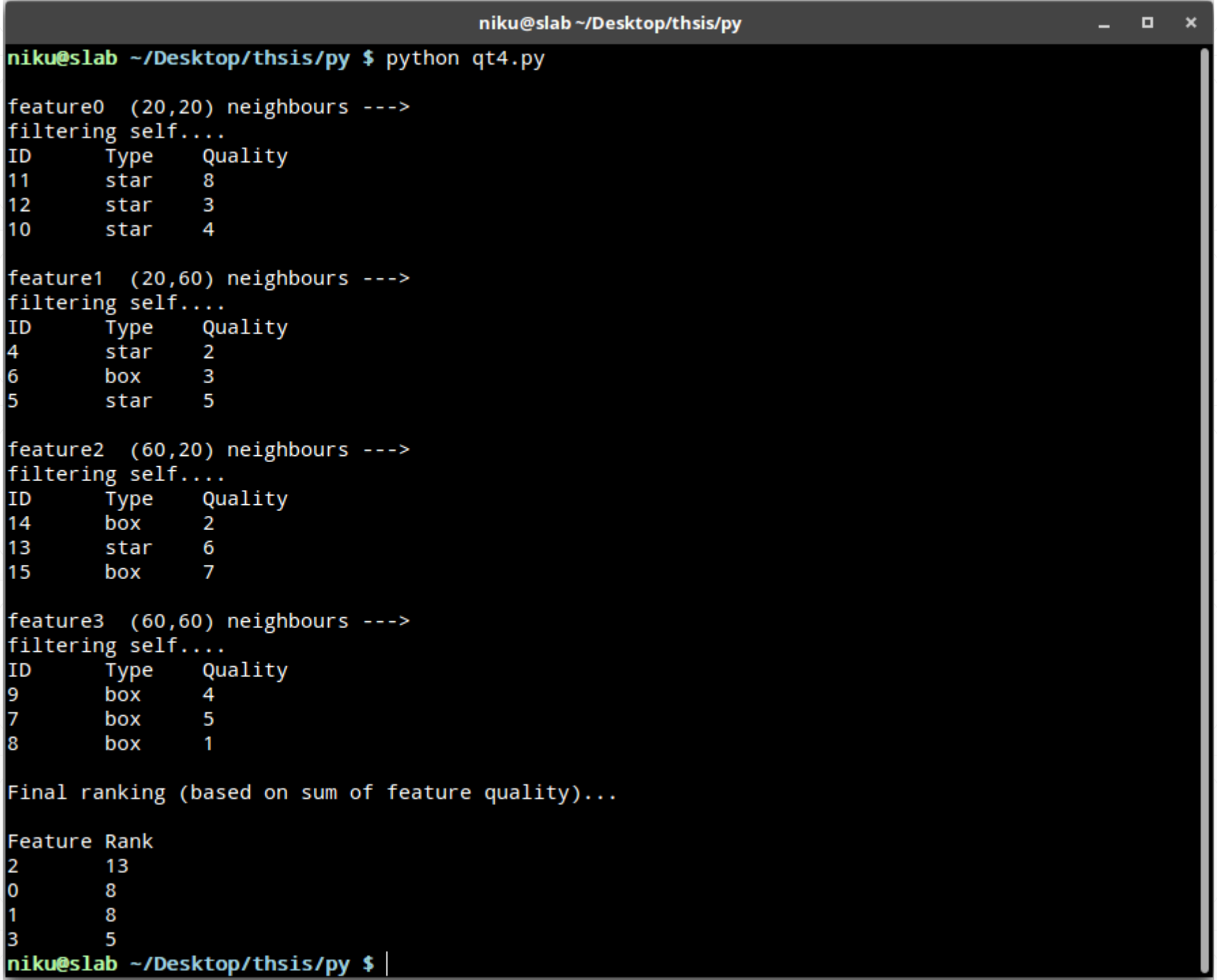


Ranking by Quality Preferences

- Quality preferences can be used for ranking when extra information about the spatial neighborhood is available.
- In this type of ranking, the total score of a feature depends on quality of its spatial neighborhood.
- E.g. Purchasing a house
- Many type of mathematical qualities can be applied to find quality of neighborhood like sum, average for each or all feature type.


(80,80)
80
60
40
20
(0,0)
20
40
60
80
x1 (2)
x2 (5)
y2 (5)
y3 (1)
feature2
feature4
y1 (3)
y4 (4)
x3 (4)
x4 (8)
x6 (6)
y5 (2)
feature1
feature3
x5 (3)
y6 (7)


(80,80)
80
60
40
20
(0,0)
20
40
60
80
x1 (2)
x2 (5)
y2 (5)
y3 (1)
feature2
feature4
y1 (3)
y4 (4)
x3 (4)
x4 (8)
x6 (6)
y5 (2)
feature1
feature3
x5 (3)
y6 (7)


niku@slab ~/Desktop/thsis/py
niku@slab ~/Desktop/thsis/py $ python qt4.py
feature0 (20,20) neighbours --->
filtering self....
ID Type Quality
11 star 8
12 star 3
10 star 4
feature1 (20,60) neighbours --->
filtering self....
ID Type Quality
4 star 2
6 box 3
5 star 5
feature2 (60,20) neighbours --->
filtering self....
ID Type Quality
14 box 2
13 star 6
15 box 7
feature3 (60,60) neighbours --->
filtering self....
ID Type Quality
9 box 4
7 box 5
8 box 1
Final ranking (based on sum of feature quality)...
Feature Rank
2 13
0 8
1 8
3 5
niku@slab ~/Desktop/thsis/py $

Conclusion

- Geo-service portal acts as a underlying framework or foundation for various kind of higher level use cases.
- Building an OGC compliant web service catalog can also be beneficiary as already available software and services can use the registry for various kinds of services with little to no modification of their original code-base.

Future Scope

- Build a cloud based implementation for the spatial web crawler, catalog service and query processing.
- Build interfaces and implementation for more complex queries.
- Provide parallel query processing for same data occurring in multiple repositories.

References

- Sonal Patil, Shrutilipi Bhattacharjee, and Soumya K. Ghosh. **A spatial web crawler for discovering geo-servers and semantic referencing with spatial features.** *International Conference on Distributed Computing and Internet Technology*. Springer International Publishing, 2014.
- Li, Wenwen, Chaowei Yang, and Chongjun Yang. **An active crawler for discovering geospatial web services and their distribution pattern. A case study of OGC Web Map Service.** *International Journal of Geographical Information Science* 24.8 (2010): 1127-1147.
- Ahlers, Dirk, and Susanne Boll. **Location-based Web search.** *The Geospatial Web*. Springer London, 2009. 55-66.
- Li, W., et al. **Semantic-based web service discovery and chaining for building an Arctic spatial data infrastructure.** *Computers & Geosciences* 37.11 (2011): 1752-1762
- ***https://github.com/karimbahgat/Pyqtree***
- Paul, Manoj, and S. K. Ghosh. **An approach for service oriented discovery and retrieval of spatial data.** In *Proceedings of the 2006 international workshop on Service-oriented software engineering*, pp. 88-94. ACM, 2006.

References (continue)

- Jiang, Jun, Chong-jun Yang, and Ying-chao Ren. **A Spatial Information Crawler for OpenGIS WFS**. *Sixth International Conference on Advanced Optical Materials and Devices*. International Society for Optics and Photonics, 2008.
- ***http://geopython.github.io/pycsw-workshop/***
- ***https://geopython.github.io/OWSLib/***
- Lopez-Pellicer, Francisco J., et al. **Discovering geographic web services in search engines.** *Online Information Review* 35.6 (2011): 909-927.
- ***https://github.com/geoserver/geoserver***
- Yiu, Man Lung, Hua Lu, Nikos Mamoulis, and Michail Vaitis. **Ranking spatial data by quality preferences**. *IEEE Transactions on Knowledge and Data Engineering* 23, no. 3 (2011): 433-446.
- Hjaltason, Gisli, and Hanan Samet. **Ranking in spatial databases**. In *Advances in Spatial Databases*, pp. 83-95. Springer Berlin/Heidelberg, 1995.

Thank You