巻末資料 6

Web-GIS

WebGIS System Development II-1a

Joel Bandibas
Geological Survey of Japan, AIST

WebGIS System Development II-1a

Brief review about WebGIS and WebGIS Components

AIST Geological Survey of Japan
AIST=Innovation

WebGIS System Development II-1a

I. Brief review about WebGIS and WebGIS Components

II. Brief review about PostgreSQL/PostGIS Database Creation

III. Brief review about PostgreSQL/PostGIS Table Creation

IV. Database Population

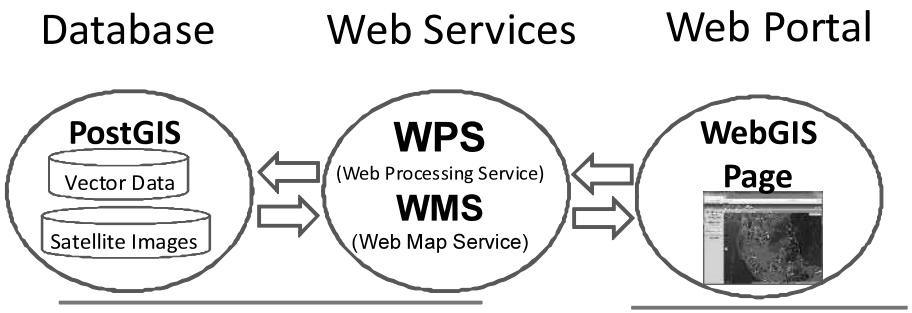
V. Brief review about Web Map Service (WMS)

AIST Geological Survey of Japan
AIST=Innovation

WEBGIS

Information System that allows users to view, edit and manipulate spatial data over the WEB. Most WebGIS are powered by Web Processing Service (WPS) and Web Map Service (WMS).

Major Components of a WebGIS System



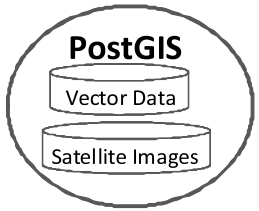
PostGIS
Apache
Linux
Mapserver
PHP (Customization)
C++ (Customization)

J-iView
AJAX
Javascript

POSTGRESQL (Postgres)

- is an object-relational database management system (ORDBMS) available for many platforms including Linux, FreeBSD, Solaris, Microsoft Windows and Mac OS X.
- PostgreSQL is developed by the PostgreSQL Global Development Group
- It implements the majority of the SQL:2008 standard (Wikipedia).

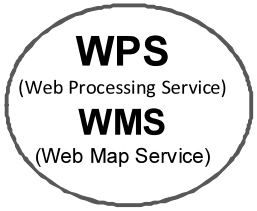
Database



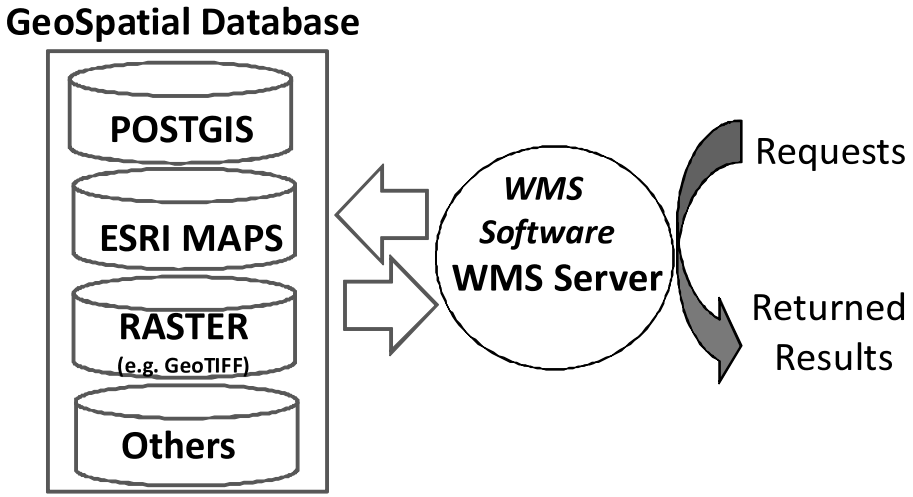
POSTGRESQL -> PostGIS

PostGIS a spatially enabled PostGreSQL. PostGIS database is a PostGreSQL with Geogmetry and Spatial Reference System columns.

Web Services



Web Map Service (WMS)



Web Services

Web Coverage Service (WCS)
Web Coverage Processing Service (WCPS)
Web Feature Service (WFS)
Web Map Service (WMS)
Web Map Tile Service (WMTS)
Web Processing Service (WPS)

Web Map Service (WMS)

Requests

Required

- **GetCapabilities** - returns parameters about the WMS and the available layers
- **GetMap** - returns a map image

Optional

- GetFeatureInfo
- DescribeLayer
- GetLegendGraphic

Web Map Service (WMS)

Request Examples

GetCapabilities

http://ows.geogrid.org/GSJ_CCOP_Combined_Bedrock_and_Superficial_Geology_and_Age/wms?SERVICE=WMS&VERSION=1.1.1&REQUEST=GetCapabilities&

Web Map Service (WMS)

Request Examples

GetMap

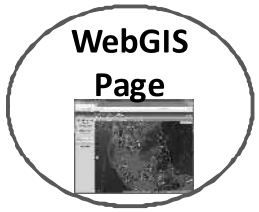
http://ows.geogrid.org/GSJ_CCOP_Combined_Bedrock_and_Superficial_Geology_and_Age/wms?REQUEST=GetMap&SERVICE=WMS&VERSION=1.1.1&LAYERS=EASIA_CCOP_2M_Combined_BLT_SLT_BA&FORMAT=image/png&SRS=EPSG:4326&BBOX=92.190331,-16.633300,155.954768,47.612849&WIDTH=256&HEIGHT=256

Result


```
WMT_MS_Capabilities version="1.1.1">
<Service>
  <Name>OGC:WMS</Name>
  <Title>GSJ CCOP Combined Bedrock and Superficial Geology and Age</Title>
  <Abstract>This is the geological map produced and compiled by the Coordinating
Committee for Geoscience Programmes in East and Southeast Asia (CCOP). The attribute
each polygon includes the bedrock and superficial geology information and age.</Abstra
      <KeywordList>
        <Keyword>Bedrock</Keyword>
        <Keyword>Surficial</Keyword>
        <Keyword>Geology</Keyword>
        <Keyword>Age</Keyword>
        <Keyword>OneGeology</Keyword>
        <Keyword>East Asia</Keyword>
        <Keyword>continent@Asia</Keyword>
        <Keyword>subcontinent@Eastern Asia</Keyword>
        <Keyword>geographicarea@Eastern Asia</Keyword>
        <Keyword>dataprovider@Coordinating Committee for Geoscience Programmes in
and Southeast Asia</Keyword>
```


Result

Web Portal



Web Map Service (WMS)

WMS Clients

1. Open Source

OpenLayers
OneGeology

2. Commercial

Google Maps
Google Earth
Bing Maps

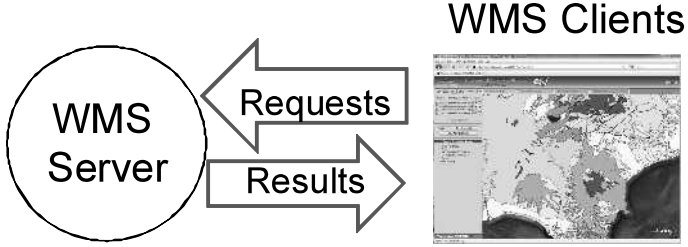
3. GSJ-AIST Dedicated WMS Client

J-iView

Web Map Service (WMS)

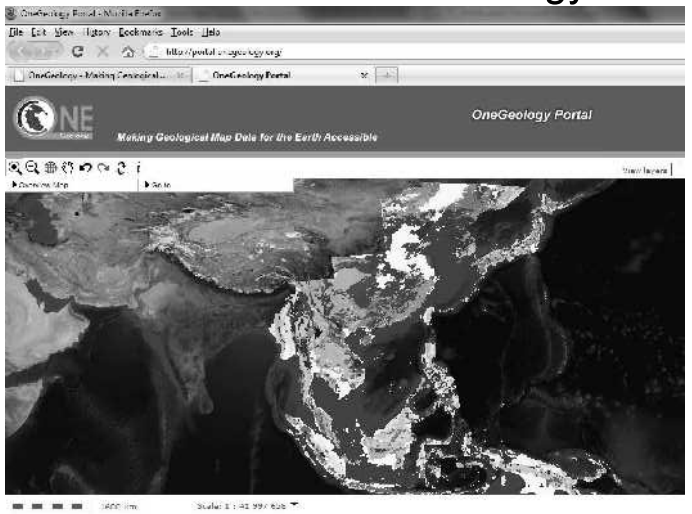
WMS Clients

WMS Clients are software that provide Graphic User Interface for implementing WMS requests (mostly GetMap request) and viewing results.

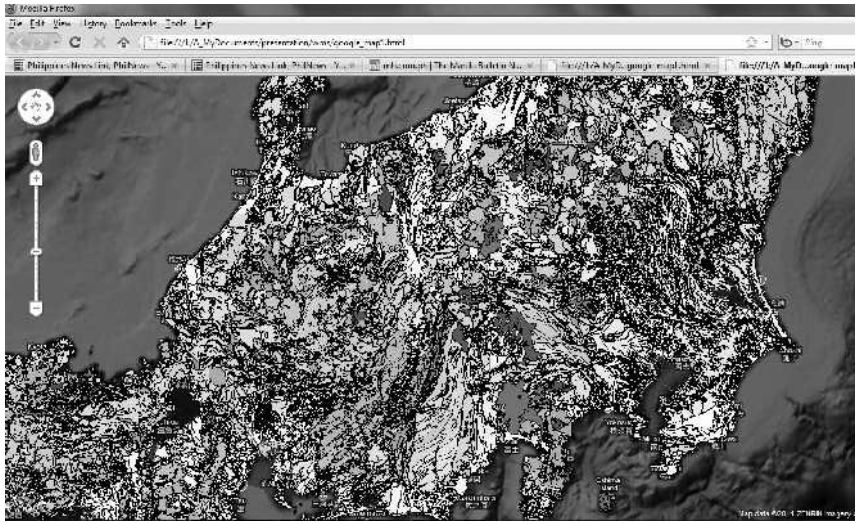


Web Map Service (WMS)

WMS Clients: OneGeology



Web Map Service (WMS)
WMS Clients: Google Maps (Google Maps API)

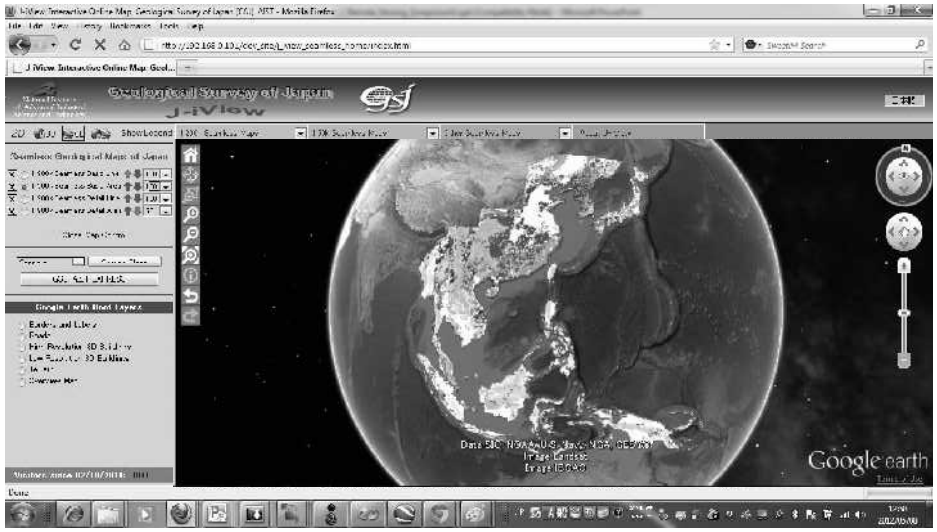


AIST Geological Survey of Japan

AIST=Innovation

Web Map Service (WMS)
WMS Clients: OpenLayers

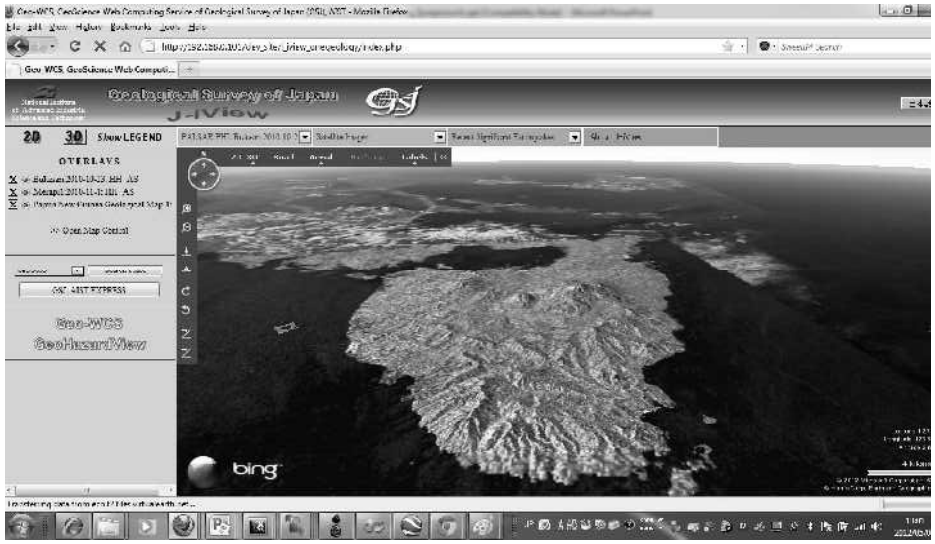
WMS Clients: Google Earth API



AIST Geological Survey of Japan

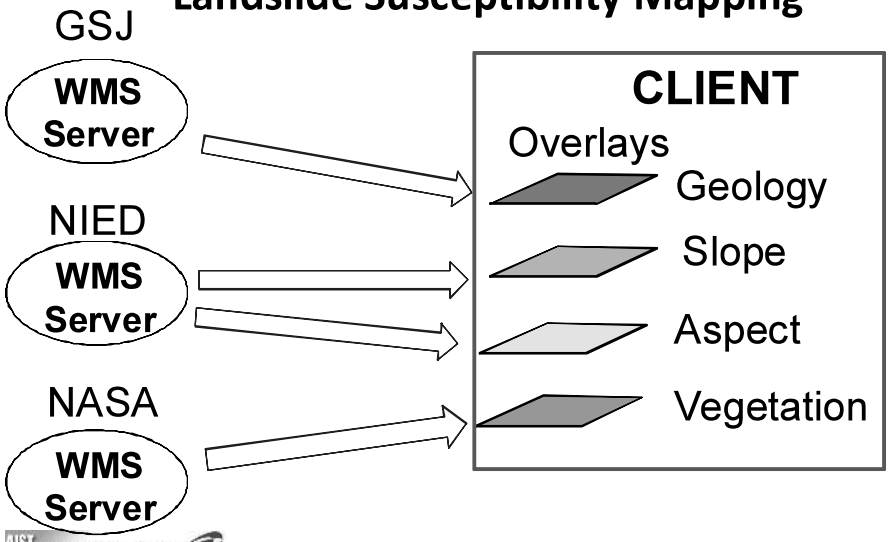
AIST=Innovation

Web Map Service (WMS)
WMS Clients: J-iView



Web Map Service (WMS)

Landslide Susceptibility Mapping



Creating Database

Command:

createdb Database_Name

Spatially enable the database

$psql –f postgis.sql –d Database_Name
$psql –f patial_ref_sys.sql –d Database_Name

WebGIS System Development II-1a

Brief review about PostgreSQL/PostGIS Database Creation

Creating Database

$su – postgres
$/usr/local/pgsql/bin/createdb PHL_DB
$cd /usr/local/pgsql/bin

Spatially enable the database

$psql –f /usr/local/pgsql/share/contrib/postgis-1.5/postgis.sql –d PHL_DB
$psql –f /usr/local/pgsql/share/contrib/postgis-1.5/spatial_ref_sys.sql –d PHL_DB

Postgres Password: joelbandibas

Creating Database

$su – postgres
$/usr/local/pgsql/bin/createdb PHL_DB
$cd /usr/local/pgsql/bin

Spatially enable the database

```
$psql –f /usr/local/pgsql/share/contrib/postgis-1.5/postgis.sql –d PHL_DB
$psql –f /usr/local/pgsql/share/contrib/postgis-1.5/spatial_ref_sys.sql –d PHL_DB
```

Postgres Password: joelbandibas

Table Creation

Shapefile to PostGIS Table

shp2pgsql –W *LATIN1* -s epsg_code *shape_fname table_name dbname* > *output_file.sql*

psql –d *dbname* –f *output_file.sql*

WebGIS System Development II-1a

Brief review about PostgreSQL/PostGIS
Table Creation

Table Creation

Shapefile to PostGIS Table

```
$shp2pgsql –W LATIN1 –s 4326
/home/co2_trainee/training_phl_group
/sample_data/east_asia_geology.shp east_asia_geology_table
PHL_DB > phl.sql
```

```
$psql –f phl.sql –d PHL_DB
```

Linux Console

Inputting maps to the database

shp2pgsql –W *LATIN1* -s epsg_code *shape_fname table_name dbname* > *output_file.sql*

psql –d *dbname* –f *output_file.sql*

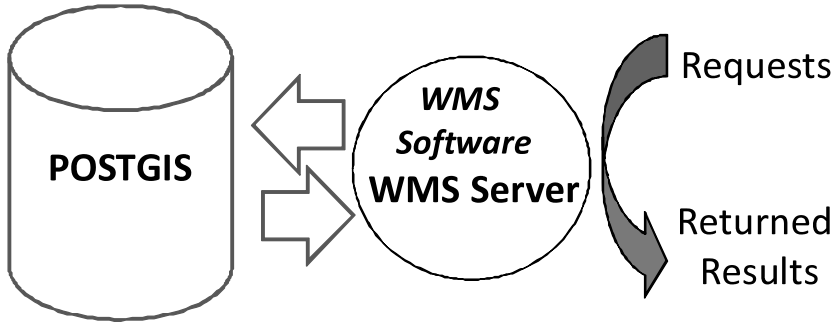
e.g.

$shp2pgsql –W LATIN1 –s 4326 /home/co2_trainee/training_phl_group /sample_data/east_asia_geology.shp east_asia_geology_table PHL_DB > phl.sql

$psql –f phl.sql –d PHL_DB

Creating Web Map Service (WMS)

GeoSpatial Database



WebGIS System Development II-1a

Brief review about Web Map Service (WMS) Formulation

Creating WMS

Two Documents Needed for WMS

1. **MapFile**
2. **WMS Script**

Creating WMS Service

MapFiles

```
MAP
 NAME OGC_WMS
 STATUS ON
 SIZE 256 256
 EXTENT 92.190331 -16.633300 155.954768  47.612849
 UNITS dd
 IMAGECOLOR  255 255 255
 IMAGETYPE gif
 PROJECTION
  "init=epsg:4326"
 END
```

East Asia Extent: WGS84 (epsg:4326)

Min X: 92.19
Min Y: -16.63
Max X: 155.954
Max Y: 47.612

Creating WMS Service

Edit MapFiles

```
CLASSITEM “geo_des”

class
name "Q_S: Sedimentary Rocks, Quaternary"
expression " Q_S: Sedimentary Rocks, Quaternary "
color 255 255 203
End

class
name "NQ_S: Sedimentary Rocks, Neogene to Quaternary"
expression " NQ_S: Sedimentary Rocks, Neogene to Quaternary "
color 255 255 153
End

class
name "N3Q_S: Sedimentary Rocks, Late Neogene to Quaternary"
expression "N3Q_S: Sedimentary Rocks, Late Neogene to Quaternary "
color 255 255 101
end
```

For POLYGON

Creating WMS Service

Edit MapFiles

```
LAYER
  NAME east_asia_geology_table
  METADATA
    WMS_TITLE “Geological Map of East Asia"
    WMS_ABSTRACT "Autogenerated map file"
    WMS_SRS “EPSG:32648 EPSG:4326 ESRI:54004 EPSG:4612 EPSG:4301 EPSG:900913"
  END

connectiontype postgis
connection "user=postgres password=joelbandibas dbname=PHL_DB host=localhost port=5432"
data "the_geom from east_asia_geology_table using unique gid"

  TYPE POLYGON
  STATUS ON
  DUMP TRUE
  PROJECTION
    "init=epsg:4326"
  END
```

Creating WMS Service

Edit MapFiles

```
CLASSITEM "mineral"

  class
   name “Gold"
   expression "Au"
     style
       symbol "Circle"
       color  236 61 156
       outlinecolor 0 0 0
       size 15
     end
  end

  class
   name “Cu"
   expression “Copper"
     style
       symbol "Circle"
       color  72 104 122
       outlinecolor 0 0 0
       size 15
     end
  end
```

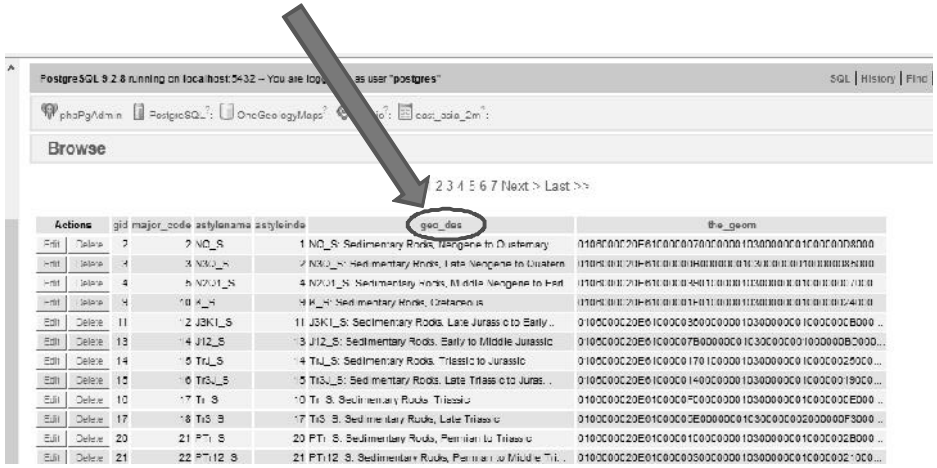
For POINT

Creating WMS Service

Edit MapFiles

Use **phpPgAdmin** to determine CLASSITEM and Values

CLASSITEM "geo_des"



Creating WMS Service

Determine the WMS URL

Format:

http://HOSTNAME/cgi-bin/training_wms/wms_script

Example:

http://**203.247.181.228** /cgi-bin/training_wms/phl_wms

Creating WMS Service

WMS Script

WMS Script Format

#!/bin/sh
export MS_MAPFILE=MapFile Path
exec /usr/local/bin/mapserv

Example

#!/bin/sh
export MS_MAPFILE=/var/www/training_mapfiles/phl.map
exec /usr/local/bin/mapserv

Creating WMS Service

Testing Your WMS

GetCapabilities

GetCapabilities Universal Resource Identifier (URI)

URI Format:

WMS URL?SERVICE=WMS&VERSION=1.1.1&REQUEST=GetCapabilities&

URI Example:

http://onegeology-asia.org/ows/GSJ_MGB_Combined_Bedrock_and_Superficial_Geology_and_Age/wms
?SERVICE=WMS&VERSION=1.1.1&REQUEST=GetCapabilities&

Testing Your WMS

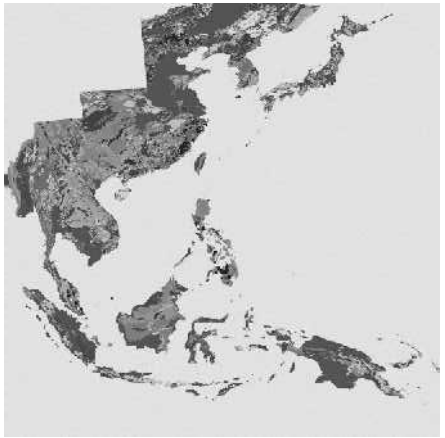
GetCapabilities Result

WMT_MS_Capabilities version="1.1.1">
<Service>
<Name>OGC:WMS</Name>
<Title>GSJ MGB Combined Bedrock and Superficial Geology and Age</Title>
<Abstract>This is the geological map produced and compiled by the Mines and Geosciences Bureau (MGB) of the Philippines. The attribute of each polygon includes the bedrock and superficial geology information and age.</Abstract>
<KeywordList>
<Keyword>Bedrock</Keyword>
<Keyword>Surficial</Keyword>
<Keyword>Geology</Keyword>
<Keyword>Age</Keyword>
<Keyword>OneGeology</Keyword>
<Keyword>Philippines</Keyword>
<Keyword>continent@Asia</Keyword>
<Keyword>subcontinent@South-eastern Asia</Keyword>
<Keyword>geographicarea@Philippines</Keyword>
<Keyword>dataprovider@Mines and Geoscience Bureau of the Philippines</Keyword>

Creating WMS Service

Testing Your WMS

GetMap Result after pointing a brower to the GetMap URI



AIST Geological Survey of Japan

AIST=Innovation

Creating WMS Service

Testing Your WMS

GetMap

GetMap Universal Resource Identifier (URI)

URI Format

WMS URL?REQUEST=GetMap&SERVICE=WMS&VERSION=1.1.1&LAYERS=Layer Name&FORMAT=image/png&SRS=EPSG:4326&BBOX=102,10,108,15&WIDTH=256&HEIGHT=256

URI Format

http://ccop-geoinfo.org/ows/group/co2_phl/east_asia_geology_table_wms?REQUEST=GetMap&SERVICE=WMS&VERSION=1.1.1&LAYERS=east_asia_geology_table&FORMAT=image/png&SRS=EPSG:4326&BBOX=92.19,-16.63,155.95, 47.612849&WIDTH=512&HEIGHT=512

Execise 1

Implement GetCapabilities and GetMap for the WMS at

http://onegeology-asia.org/ows/GSJ_Combined_Bedrock_and_Superficial_Geology_and_Age/wms

Exercises

Display the following WMSs using the client at
http://ccop-geoinfo.org/my_client.html

1. http://jcbwebgis.com/cgi-bin/wms/group/onegeology_new/east_asia_2m_wms?
2. http://jcbwebgis.com/cgi-bin/wms/group/onegeology_new/thailand_1m_wms?
3. http://jcbwebgis.com/cgi-bin/wms/group/onegeology_new/malaysia_pen_1m_wms?
4. http://jcbwebgis.com/cgi-bin/wms/group/onegeology_new/myanmar_1m_wms?
5. http://jcbwebgis.com/cgi-bin/wms/group/onegeology_new/cambodia_geology_wms?
6. http://jcbwebgis.com/cgi-bin/wms/group/onegeology_new/japan_1m_wms?
7. http://jcbwebgis.com/cgi-bin/wms/group/onegeology_new/vietnam_new_1m_wms?
8. http://jcbwebgis.com/cgi-bin/wms/group/onegeology_new/indonesia_1m_wms?
9. http://jcbwebgis.com/cgi-bin/wms/group/onegeology_new/philippine_geological_map_1m_wms?
10. http://jcbwebgis.com/cgi-bin/wms/group/groundwater/vietnam_groundwater_wms?
11. http://jcbwebgis.com/cgi-bin/minerals_wms/asean_gold_wms?
12. http://jcbwebgis.com/cgi-bin/minerals_wms/loindo_2012_wms?

Execise 3

http://150.29.14.124/phpPgAdmin/index.php

id: postgres

WebGIS System Development II-1b

Joel Bandibas
Geological Survey of Japan, AIST

WebGIS System Development II

PostGIS and Simple Features

AIST Geological Survey of Japan
AIST=Innovation

WebGIS System Development II

I. PostGIS and Simple Features

II. PostGIS and Spatial Information Processing Functions

III. Spatial Queries and Sequential Query Language (SQL)

IV. SQL Formulation and Implementation(SQL)

AIST Geological Survey of Japan
AIST=Innovation

Simple Features

PostGIS supports "**Simple Features**" objects defined by OGC. It also supports the "**Simple Features SQL**". In PostGIS, spatial objects are expressed in two standard ways: The **Well-Known Text** (WKT) and **Well-Known Binary** (WKB).

AIST Geological Survey of Japan
AIST=Innovation

Simple Features

Well-Known Text (WKT)

- Elements of the simple feature objects are pairs of x and y values separated by space.

- Adjacent x and y pairs are separated by a comma.

Simple Features (Spatial Objects) expressed in Well-Known Text (WKT)

- POINT(0 0)
- LINESTRING(0 0,1 1,1 2)
- POLYGON((0 0,4 0,4 4,0 4,0 0),(1 1, 2 1, 2 2, 1 2,1 1))
- MULTIPOINT(0 0,1 2)
- MULTILINESTRING((0 0,1 1,1 2),(2 3,3 2,5 4))
- MULTIPOLYGON(((0 0,4 0,4 4,0 4,0 0),(1 1,2 1,2 2,1 2,1 1)), ((-1 -1,-1 -2,-2 -2,-2 -1,-1 -1)))
- GEOMETRYCOLLECTION(POINT(2 3),LINESTRING(2 3,3 4))

Simple Features
Well-Known Text (WKT)

Examples:

Point coordinates: **125.2** deg. long **40.56** deg. lat

POINT (125.2 40.56).

Polygon coordinates: **98.3 1.53,103.18 1.8,104.97 -3.28,98.6 -3.6,98.3 1.53**

POLYGON((98.3 1.53,103.18 1.8,104.97 -3.28,98.6 -3.6,98.3 1.53));

In the PostGIS Database, Geometry (spatial objects) are stored in Well-Known Binary

Actions	gid	major_code	assfilename	assstyleinde	geo_desc	the_geom
Edit Delete	1	1	Q_S	0	Q_S: Sedimentary Rocks, Quaternary	0106000020E6...
Edit Delete	2	2	NQ_S	1	NQ_S: Sedimentary Rocks, Neogene to Quaternary	0106000020E6...
Edit Delete	3	3	N2Q_S	2	N2Q_S: Sedimentary Rocks, Late Neogene to Quaternary	0106000020E6...
Edit Delete	4	5	N2Q1_S	4	N2Q1_S: Sedimentary Rocks, Middle Neogene to Quaternary	0106000020E6...
Edit Delete	5	6	N_S	5	N_S: Sedimentary rocks Neogene	0106000020E6...
Edit Delete	6	7	Pg_S	6	Pg_S: Sedimentary Rocks, Paleogene	0106000020E6...
Edit Delete	7	8	T_S	7	T_S: Sedimentary rocks Tertiary	0106000020E6...
Edit Delete	8	9	KPg_S	8	KPg_S: Sedimentary Rocks, Cretaceous to Paleogene	0106000020E6...
Edit Delete	9	10	K_S	9	K_S: Sedimentary Rocks Cretaceous	0106000020E6...
Edit Delete	10	11	JK_S	10	JK_S: Sedimentary Rocks, Jurassic to Cretaceous	0106000020E6...
Edit Delete	11	12	JSK1_S	11	JSK1_S: Sedimentary Rocks, Late Jurassic to Early	0106000020E6...
Edit Delete	12	13	J_S	12	J_S: Sedimentary Rocks, Jurassic	0106000020E6...
Edit Delete	13	14	J12_S	13	J12_S: Sedimentary Rocks, Early to Middle Jurassic	0106000020E6...
Edit Delete	14	15	TrJ_S	14	TrJ_S: Sedimentary Rocks, Triassic to Jurassic	0106000020E6...
Edit Delete	15	16	TrJ1_S	15	TrJ1_S: Sedimentary Rocks, Late Triassic to Jurassic	0106000020E6...
Edit Delete	16	17	Tr_S	16	Tr_S: Sedimentary Rocks, Triassic	0106000020E6...
Edit Delete	17	18	Tr3_S	17	Tr3_S: Sedimentary Rocks, Late Triassic	0106000020E6...
Edit Delete	18	19	Tr12_S	18	Tr12_S: Sedimentary Rocks, Early to Middle Triass...	0106000020E6...
Edit Delete	19	20	Mz_S	19	Mz_S: Sedimentary Rocks, Mesozoic	0106000020E6...
Edit Delete	20	21	PTr_S	20	PTr_S: Sedimentary Rocks, Permian to Triassic	0106000020E6...
Edit Delete	21	22	P1Tr2_S	21	P1Tr2_S: Sedimentary rocks Permian to Middle Tr...	0106000020E6...
Edit Delete	22	23	C1Tr_S	22	C1Tr_S: Sedimentary Rocks, Carboniferous to Triass...	0106000020E6...
Edit Delete	23	24	PzMz_S	23	PzMz_S: Sedimentary Rocks, Paleozoic to Mesozoic	0106000020E6...
Edit Delete	24	25	P_S	24	P_S: Sedimentary Rocks, Permian	0106000020E6...
Edit Delete	25	26	P2_S	25	P2_S: Sedimentary Rocks, Late Permian	0106000020E6...
Edit Delete	26	27	CP_S	26	CP_S: Sedimentary Rocks, Carboniferous to Permian	0106000020E6...
Edit Delete	27	28	C3P_S	27	C3P_S: Sedimentary Rocks, Late Carboniferous to P	0106000020E6...
Edit Delete	28	29	C_S	28	C_S: Sedimentary Rocks, Carboniferous	0106000020E6...
Edit Delete	29	30	DC_S	29	DC_S: Sedimentary Rocks, Devonian to Carboniferous	0106000020E6...
Edit Delete	30	31	D_S	30	D_S: Sedimentary rocks, Devonian	0106000020E6...

WebGIS System Development II

PostGIS and Spatial Information Geometry Processing Functions

AIST Geological Survey of Japan

AIST=Innovation

Sample SQLs using the Functions

Login to phpPgAdmin

http://hostname/phpPgAdmin/index.php

Username: **postgres**
Password: **joelbandibas**

Important geometry processing functions that convert objects between WKT and WKB

bytea WKB = ST_AsBinary(geometry)

text WKT = ST_AsText(geometry)

geometry = ST_GeomFromWKB(bytea WKB, SRID)

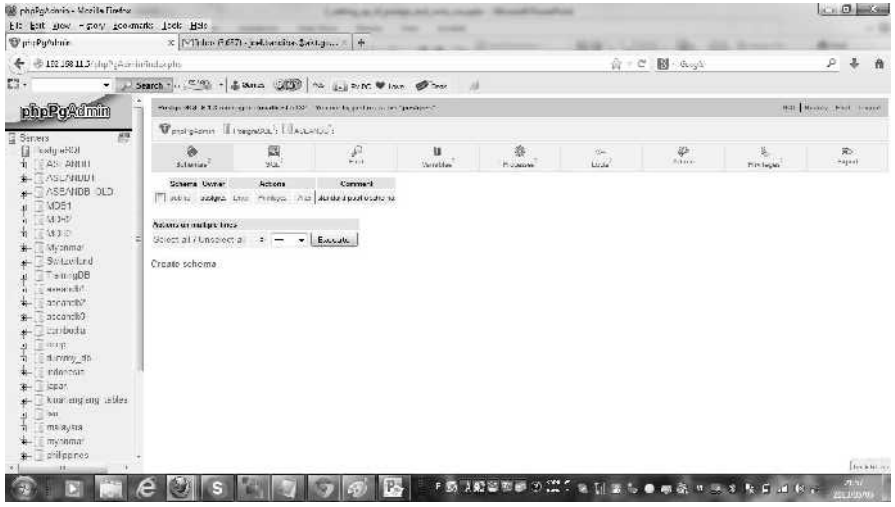
geometry = ST_GeometryFromText(text WKT, SRID)

AIST Geological Survey of Japan

AIST=Innovation

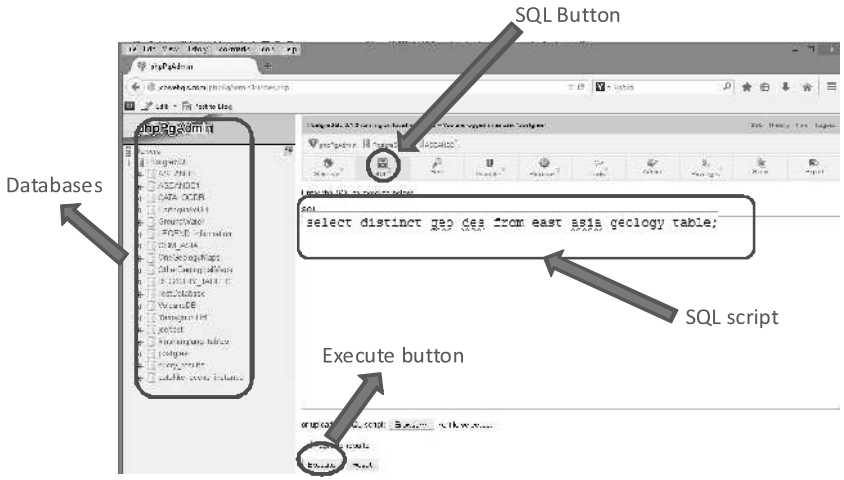
Sample SQLs using the Functions

Login to phpPgAdmin



Sample SQLs using the Functions

Login to phpPgAdmin



Sample SQLs using the Geometry Processing Functions

byte WKB = ST_AsBinary(geometry)

Actual SQL

```
select ST_AsBinary(the_geom) from myanmar_minerals_table
where GID=1;
```

Sample SQLs using the Geometry Processing Functions

byte WKB = ST_AsBinary(geometry)

Actual SQL

```
select ST_AsBinary(the_geom) from myanmar_minerals_table
where GID=1;
```

Sample SQLs using the Geometry Processing Functions

text WKT = ST_AsText(geometry)

Actual SQL

```
select ST_AsText(the_geom) from myanmar_minerals_table
where GID=1;
```

Sample SQLs using the Geometry Processing Functions

geometry = ST_GeomFromWKB(bytea WKB, SRID)

Actual SQL

```
select ST_GeomFromWKB(the_geom,4326)
from myanmar_minerals_table where GID=1;
```

Sample SQLs using the Geometry Processing Functions

An example of inserting spatial object into a database using one of the aforementioned functions:

```
INSERT INTO geotable (the_geom)
VALUES (ST_GeomFromText('POINT(-126.4 45.32)',
4326));
```


Sample SQLs using the Geometry Processing Functions

geometry = ST_GeometryFromText(text WKT, SRID)

e.g.
Point Location: *Longitude=-126.4; Latitude=45.32*
WKT='POINT(-126.4 45.32)'

Actual SQL

```
select ST_GeometryFromText('POINT(-126.4  45.32)',4326);
```

WebGIS System Development II

Spatial Queries and Sequential Query Language (SQL)

Spatial Queries and Sequential Query Language (SQL)

SQL Format

select distinct geological _description_column
from table_name;

Determine all mineral types found in Myanmar

select distinct geo_des from east_asia_geology_table;

Spatial Queries and Sequential Query Language (SQL)

Extracting records containing gold from Myanmar Database

SQL Format

select * from table_name where column_name='Gold' ;

Extracting records (sites) containing gold using Myanmar data.

select * from myanmar_minerals_table where mineral_co='Gold' ;

Spatial Queries and Sequential Query Language (SQL)

Extracting polygon with geology
"K_Mi: Intermediate Grade Metamorphic Rocks, Cretaceous" from east_asia_geology_table

The SQL

select * from east_asia_geology_table where geo_des='***K_Mi: Intermediate Grade Metamorphic Rocks, Cretaceous*** ';

Spatial Queries and Sequential Query Language (SQL)

Spatial Query

Querying the database involving geometry. Question like "Which polygon is nearest or covers Bangkok" could be answered using spatial query.

The SQL to answer that above question:

(*Central Location Bangkok):* *Longitude: 100.47; Latitude: 13.75*

WKT of Location: *POINT(100.47 13.75)*
EPSG Code for WGS84: *4326* **)**

Operator:

float **ST_Distance**(geometry ***g1***, geometry ***g2***);

The SQL:

```
select * from east_asia_geology_table
ORDER by ST_Distance(the_geom, ST_Geomfromtext('POINT(100.47 13.75) ',4326 )) ASC
LIMIT 1;
```

Querying the Database Using
phpPgAdmin and Sequential Query Language (SQL)

Spatial Query

Querying the database involving geometry. Question like "Which known mineral site is nearest to Nay Pyi Taw" could be answered using spatial query.

The SQL to answer that above question:

(*Central Location of Nay Pyi Taw (City Hall):* *Longitude: 96.129; Latitude: 19.744*

WKT of Location: *POINT(96.129 19.744)*
EPSG Code for WGS84: *4326* **)**

Operator:

float **ST_Distance**(geometry ***g1***, geometry ***g2***);

The SQL:

```
select * from myanmar_minerals_table
ORDER by  ST_Distance(the_geom, ST_Geomfromtext('POINT(96.129 19.744) ',4326 ))  ASC
LIMIT 1;
```

Spatial Queries and Sequential Query Language (SQL)

Spatial Query

Question like "Which site containing gold is nearest to Nay Pyi Taw, Myanmar" could be answered using spatial query.

Operator:

float **ST_Distance**(geometry ***g1***, geometry ***g2***);

The SQL:

```
select * from myanmar_minerals_table  where  mineral_co='Gold'  ORDER by
ST_Distance(the_geom, ST_Geomfromtext('POINT(96.129 19.744)',4326 ))  ASC
LIMIT 1;
```

Querying the Database Using
phpPgAdmin and Sequential Query Language (SQL)

Spatial Query

Displaying the distance between Nay Pyi Taw and all sites with Gold.

Distance in Radiance

The SQL:

```
select mineral_co,mine_depos, ST_Distance
(
        the_geom,
         ST_Geomfromtext('POINT(96.129 19.744) ',4326 )
)
from myanmar_minerals_table
where mineral_co='Gold';
```

Distance in Meters (Myanmar is UTM 46N; EPSG Code: 23946

The SQL:

```
select mineral_co,mine_depos, ST_Distance
(
                ST_Transform(the_geom,23946) ,
                ST_Transform(ST_Geomfromtext('POINT(96.129  19.744) ',4326 ),23946)
)
from myanmar_minerals_table
where mineral_co='Gold';
```

Spatial Queries and Sequential Query Language (SQL)

Spatial Query: Distance

The distance between Bangkok and the nearest polygon with "K_Mi: Intermediate Grade Metamorphic Rocks, Cretaceous" could be determined answered using spatial query.

Distance in Radiance

The SQL:

```
select  geo_des , ST_Distance
(
        the_geom,
         ST_Geomfromtext('POINT(100.47 13.75) ',4326 )
)
from  east_asia_geology_table
where geo_des='K_Mi: Intermediate Grade Metamorphic Rocks, Cretaceous'
ORDER by  ST_Distance(the_geom, ST_Geomfromtext('POINT(100.47 13.75) ', 4326 ))  ASC
LIMIT 1;
```

Spatial Queries and Sequential Query Language (SQL)

Spatial Query

Determining the Distance

Distance in Meters (Thailand is UTM 47N; EPSG Code: 24047)

The SQL:

```
Select geo_des, ST_Distance
(
        ST_Transform(the_geom, 24047) ,
        ST_Transform(ST_Geomfromtext('POINT(100.47 13.75) ',4326 ), 24047)
)
from east_asia_geology_table
where geo_des='K_Mi: Intermediate Grade Metamorphic Rocks, Cretaceous'
ORDER by ST_Distance(the_geom, ST_Geomfromtext('POINT(100.47 13.75) ', 4326 )) ASC LIMIT 1;
```

Creating and Populating Table Using Spreadsheet Data and SQL

gid	mineral_co	mine_depos	state_devi	township	reserve_to	longitude	latitude
0	Barite	Konesut	Kayah	Loikaw	0.0045	97.18	19.87
1	Barite	Kyun Taing	Mandalay	Pyin Oo Lwin	0.003831	96.45	22.20
2	Barite	Chanpelun	Mandalay	Kyaukse	400	96.31	21.66
3	Barite	Ngaok Twin	Mandalay	Kyaukse	0.0014	96.30	21.68
4	Barite	Sithar	Mandalay	Pyin Oo Lwin	0.002503	96.42	21.89
5	Barite	Setaw Lay	Mandalay	Patheingyi	0.063	96.25	21.91
6	Barite	Nyaung Ye	Mandalay	Kyaukse	0.002	96.34	21.68
7	Barite	Taungpatlan	Mandalay	Kyaukse	0.010585	96.31	21.69
8	Barite	Dahatpin	Mandalay	Kyaukse	0.0018	96.30	21.72
9	Barite	Dattaw	Mandalay	Pyin Oo Lwin	0.004	96.40	21.98
10	Barite	Naungpain	Mandalay	Pyin Oo Lwin	0.002	96.45	22.22
11	Barite	Padaukpin	Mandalay	Kyaukse	0.000975	96.34	21.68

The SQL Format:

CREATE TABLE Table_Name (colum_name datatype, column_name datatype, column_name datatype ...);

The SQL:

```
CREATE TABLE myanmar_mineral_new (gid integer, mineral_co VARCHAR, mine_depos VARCHAR,
state_devi VARCHAR, township VARCHAR, reserve_to FLOAT, longitude FLOAT, latitude FLOAT);
```

Creating and Populating Table Using Spreadsheet Data and SQL

Sample Tabular Data (Myanmar Minerals)
(An Excel file saved as text, tab delimited)

gid	mineral_co	mine_depos	state_devi	township	reserve_to	longitude	latitude
0	Barite	Konesut	Kayah	Loikaw	0.0045	97.18	19.87
1	Barite	Kyun Taing	Mandalay	Pyin Oo Lwin	0.003831	96.45	22.20
2	Barite	Chanpelun	Mandalay	Kyaukse	400	96.31	21.66
3	Barite	Ngaok Twin	Mandalay	Kyaukse	0.0014	96.30	21.68
4	Barite	Sithar	Mandalay	Pyin Oo Lwin	0.002503	96.42	21.89
5	Barite	Setaw Lay	Mandalay	Patheingyi	0.063	96.25	21.91
6	Barite	Nyaung Ye	Mandalay	Kyaukse	0.002	96.34	21.68
7	Barite	Taungpatlan	Mandalay	Kyaukse	0.010585	96.31	21.69
8	Barite	Dahatpin	Mandalay	Kyaukse	0.0018	96.30	21.72
9	Barite	Dattaw	Mandalay	Pyin Oo Lwin	0.004	96.40	21.98
10	Barite	Naungpain	Mandalay	Pyin Oo Lwin	0.002	96.45	22.22
11	Barite	Padaukpin	Mandalay	Kyaukse	0.000975	96.34	21.68

Creating and Populating Table Using Spreadsheet Data and SQL

Populating the Table with Data from Text File

gid	mineral_co	mine_depos	state_devi	township	reserve_to	longitude	latitude
0	Barite	Konesut	Kayah	Loikaw	0.0045	97.18	19.87
1	Barite	Kyun Taing	Mandalay	Pyin Oo Lwin	0.003831	96.45	22.20
2	Barite	Chanpelun	Mandalay	Kyaukse	400	96.31	21.66
3	Barite	Ngaok Twin	Mandalay	Kyaukse	0.0014	96.30	21.68
4	Barite	Sithar	Mandalay	Pyin Oo Lwin	0.002503	96.42	21.89
5	Barite	Setaw Lay	Mandalay	Patheingyi	0.063	96.25	21.91
6	Barite	Nyaung Ye	Mandalay	Kyaukse	0.002	96.34	21.68
7	Barite	Taungpatlan	Mandalay	Kyaukse	0.010585	96.31	21.69
8	Barite	Dahatpin	Mandalay	Kyaukse	0.0018	96.30	21.72
9	Barite	Dattaw	Mandalay	Pyin Oo Lwin	0.004	96.40	21.98
10	Barite	Naungpain	Mandalay	Pyin Oo Lwin	0.002	96.45	22.22
11	Barite	Padaukpin	Mandalay	Kyaukse	0.000975	96.34	21.68

The SQL Format:

COPY table_name FROM 'text_file_filename';

The SQL:

```
COPY myanmar_mineral_new FROM '/home/postgres/sample_mineral_table.txt';
```

Creating and Populating Table Using Spreadsheet Data and SQL

Inserting Geometry Column

The SQL Format:

```
SELECT AddGeometryColumn('table_name', 'the_geom',SRID,'geometry_type',2);
```

The SQL:

```
SELECT AddGeometryColumn('myanmar_mineral_new','the_geom',4326,'POINT',2);
```

Creating and Populating Table Using Spreadsheet Data and SQL

Complete SQL Sequence

The SQL:

```
CREATE TABLE myanmar_mineral_new (gid integer, mineral_co VARCHAR, mine_depos
VARCHAR, state_devi VARCHAR, township VARCHAR, reserve_to FLOAT, longitude FLOAT,
latitude FLOAT);

COPY myanmar_mineral_new FROM '/home/co2_trainee/myanmar_mineral_data.txt';

SELECT AddGeometryColumn('myanmar_mineral_new','the_geom',4326,'POINT',2);

UPDATE myanmar_mineral_new SET the_geom=GeomFromText('POINT('||longitude||'
'||latitude||')',4326);
```

Creating and Populating Table Using Spreadsheet Data and SQL

Populating Geometry Column

The SQL Format:

```
UPDATE Table_Name SET the_geom= GeomFromText ('POINT('||long_column||' '||lat_column||')',SRID);
```

The SQL:

```
UPDATE myanmar_mineral_new SET the_geom=GeomFromText('POINT('||longitude||' '||latitude||')',4326);
```

EXERCISES

1. Formulate an SQL to extract the records containing the mineral Antimony from Myanmar mineral data in your database.
2. Formulate an SQL to show the distances in meters between Yangon (longitude: 96.144; Latitude: 16.793) mineral sites containing Gypsum.
3. Input the mineral data of the Philippines (philippine_minerals.exe) in Excel file into your PostGIS database.