



*GeoServer*

# WS05 - Basic to Advance GeoServer

• • •

Instructor : Sattawat Arab

# I am a instructor

•••

FOSS4G THAILAND 2019

“ การแสดงผลและให้บริการข้อมูลสภาพอากาศรายวันผ่านเครือข่ายด้วย GeoServer ”

ศตวรรษ อาร์บ  
GIS Analyst and Backend Engineer,  
i-bitz co.,ltd.

05 NOV 2019 WORKSHOP

FOSS4G THAILAND 2019

ศตวรรษ อาร์บ

GIS Analyst and Backend Engineer, i-bitz co.,ltd.



“เรื่องบางอย่างรู้คุณเดียวมันไม่สนุก ต้องบอกต่อให้คนอื่นรู้ด้วย และให้คนเหล่านั้นนำความรู้ที่ได้รับไปบอกต่อ”

จากประสบการณ์ด้านการวิเคราะห์ข้อมูลแบบ Automatic การจัดทำฐานข้อมูล ออกแบบ วิเคราะห์ และบริหารจัดการข้อมูลเชิงพื้นที่ (Spatial data) โดยให้บริการแพนที่ผ่านทางชุด โปรแกรมที่ชื่อว่า “GeoServer” ร่วมกับการออกแบบการแสดงผลแพนที่ (Map Style) ในรูปแบบต่าง ๆ เพื่อผลิตแพนที่ออนไลน์ที่สามารถตอบโจทย์เฉพาะเรื่องผ่านทาง Web Application ได้อย่างถูกต้องและเหมาะสม โดยเฉพาะข้อมูลที่มีเวลาเข้ามาเกี่ยวข้องต้องแสดงผลข้อมูลให้เหมาะสมกับสถานะการณ์นั้น ๆ เพื่อสื่อความหมายให้ประชาชนเข้าใจง่ายที่สุด

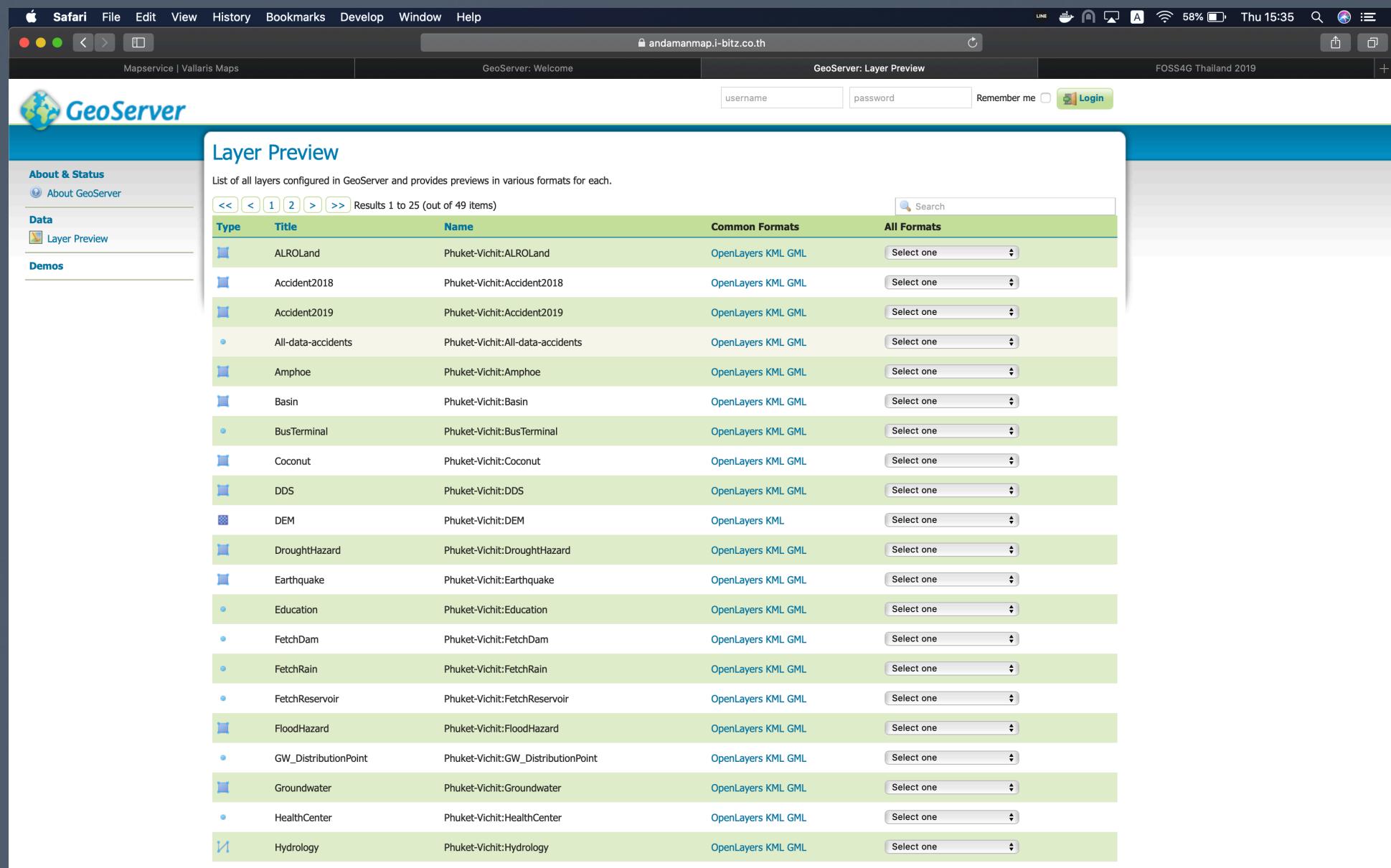
นอกเหนือจากการดูแลงานด้าน Geoinformatics ซึ่งเป็นภาระกิจหลักแล้ว ยังมีงานด้านการถ่ายทอดองค์ความรู้ที่มีแก่ผู้สนใจทั่วไป หน่วยงานราชการและเอกชน ในรูปแบบของ GIS Training&Workshop และกิจกรรม One day sharing UNI project แก่นิสิตนักศึกษาระดับมหาวิทยาลัยต่างๆอีกด้วย ภายใต้ GISBuildup Training Center

จากความรู้ข้างต้นผมจึงอยากที่จะถ่ายทอดองค์ความรู้ที่มีให้แก่ผู้ที่สนใจขึ้นมาโดยเฉพาะสำหรับงานประชุมครั้งนี้

ติดต่อ : Email : [sattawat.a@i-bitz.co.th](mailto:sattawat.a@i-bitz.co.th)

# CONTENT

...



The screenshot shows the GeoServer Layer Preview interface. The main area displays a table titled "Layer Preview" with columns: Type, Title, Name, Common Formats, and All Formats. The table lists various layers such as ALROLand, Accident2018, Accident2019, All-data-accidents, Amphoe, Basin, BusTerminal, Coconut, DDS, DEM, DroughtHazard, Earthquake, Education, FetchDam, FetchRain, FetchReservoir, FloodHazard, GW\_DistributionPoint, Groundwater, HealthCenter, and Hydrology. Each row includes dropdown menus for selecting formats like OpenLayers KML GML.

Type	Title	Name	Common Formats	All Formats
	ALROLand	Phuket-Vichit:ALROLand	OpenLayers KML GML	Select one
	Accident2018	Phuket-Vichit:Accident2018	OpenLayers KML GML	Select one
	Accident2019	Phuket-Vichit:Accident2019	OpenLayers KML GML	Select one
	All-data-accidents	Phuket-Vichit:All-data-accidents	OpenLayers KML GML	Select one
	Amphoe	Phuket-Vichit:Amphoe	OpenLayers KML GML	Select one
	Basin	Phuket-Vichit:Basin	OpenLayers KML GML	Select one
	BusTerminal	Phuket-Vichit:BusTerminal	OpenLayers KML GML	Select one
	Coconut	Phuket-Vichit:Coconut	OpenLayers KML GML	Select one
	DDS	Phuket-Vichit:DDS	OpenLayers KML GML	Select one
	DEM	Phuket-Vichit:DEM	OpenLayers KML	Select one
	DroughtHazard	Phuket-Vichit:DroughtHazard	OpenLayers KML GML	Select one
	Earthquake	Phuket-Vichit:Earthquake	OpenLayers KML GML	Select one
	Education	Phuket-Vichit:Education	OpenLayers KML GML	Select one
	FetchDam	Phuket-Vichit:FetchDam	OpenLayers KML GML	Select one
	FetchRain	Phuket-Vichit:FetchRain	OpenLayers KML GML	Select one
	FetchReservoir	Phuket-Vichit:FetchReservoir	OpenLayers KML GML	Select one
	FloodHazard	Phuket-Vichit:FloodHazard	OpenLayers KML GML	Select one
	GW_DistributionPoint	Phuket-Vichit:GW_DistributionPoint	OpenLayers KML GML	Select one
	Groundwater	Phuket-Vichit:Groundwater	OpenLayers KML GML	Select one
	HealthCenter	Phuket-Vichit:HealthCenter	OpenLayers KML GML	Select one
	Hydrology	Phuket-Vichit:Hydrology	OpenLayers KML GML	Select one

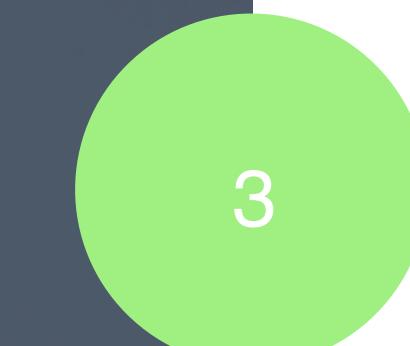
เอกสารที่ : cccc



i-bitz company limited  
THE GEOMATICS COMPANY



2



4

- Store data and management daily weather data in PostgreSQL

- Store data and display the map in GeoServer according to the conditions of time (Vector and Raster )

- Design weather map using Styled Layer Descriptor (SLD)

- Using the function SQL view, CQL and Viewparams of GeoServer for display the map

- Providing time-based weather information using the WFS of GeoServer

- Call WMS layer Daily weather map display in OpenLayers

# Software

...



OpenLayers  
5%



GeoServer  
50%



PostgreSQL  
30%

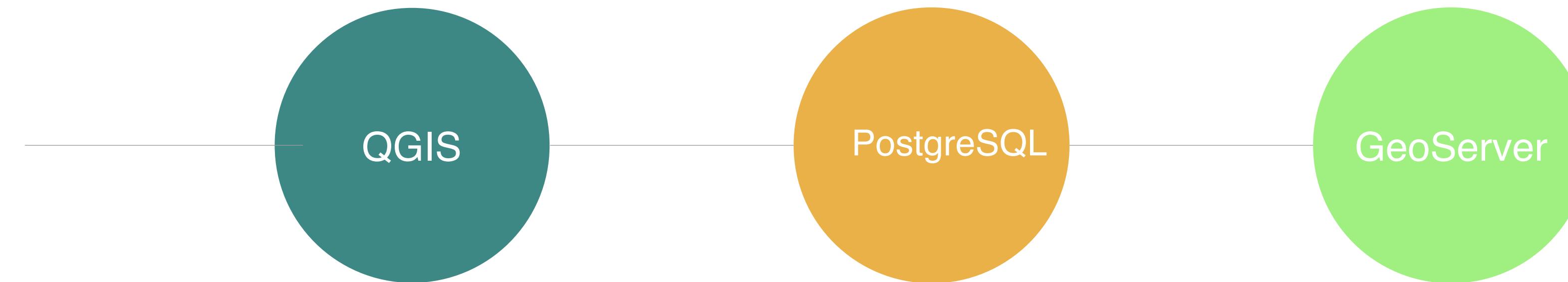


QGIS  
15%

# Concept



เมื่อมีการอัพเดทข้อมูลปริมาณน้ำฝนใหม่จากไฟล์ csv. เข้าไปในระบบ(ฐานข้อมูล PostgreSQL) แผนที่ใน GeoServer จะเปลี่ยนเองแบบอัตโนมัติ ตามเงื่อนไขที่เรากำหนดไว้ ยกตัวอย่างเช่น การแสดงแผนที่ปริมาณน้ำฝนรายสถานีตามเงื่อนไขของเวลา แผนที่น้ำวันนี้, แผนที่น้ำฝนเมื่อวาน, แผนที่น้ำฝน 3 วันที่แล้ว ทั้งแบบ Vector และ Raster



## Weather data

Management data csv ,  
Shapefile for import data  
to PostgreSQL

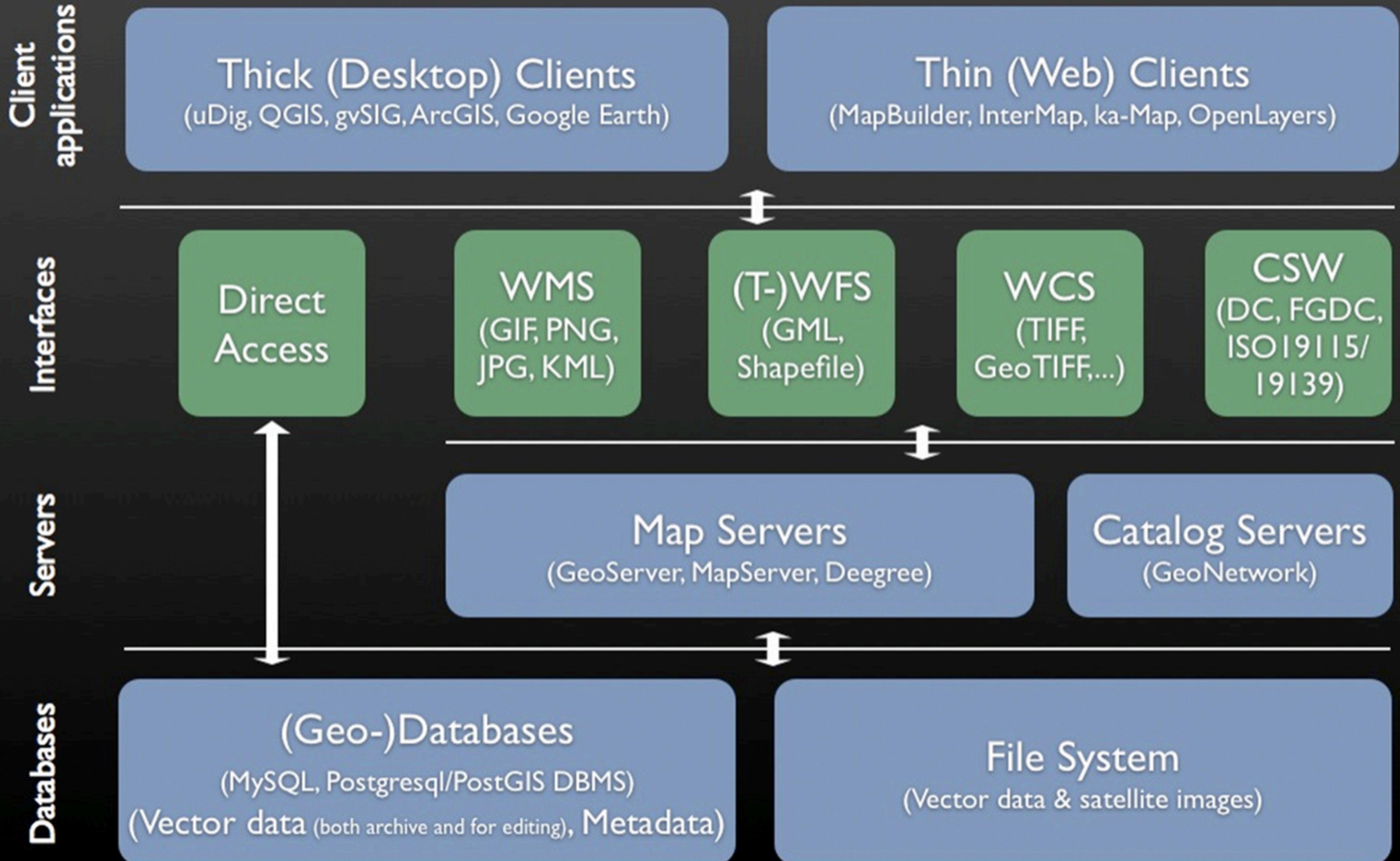
## SQL

Using SQL for Prepare  
Data data and create view  
data support layer in  
GeoServer

## Data service

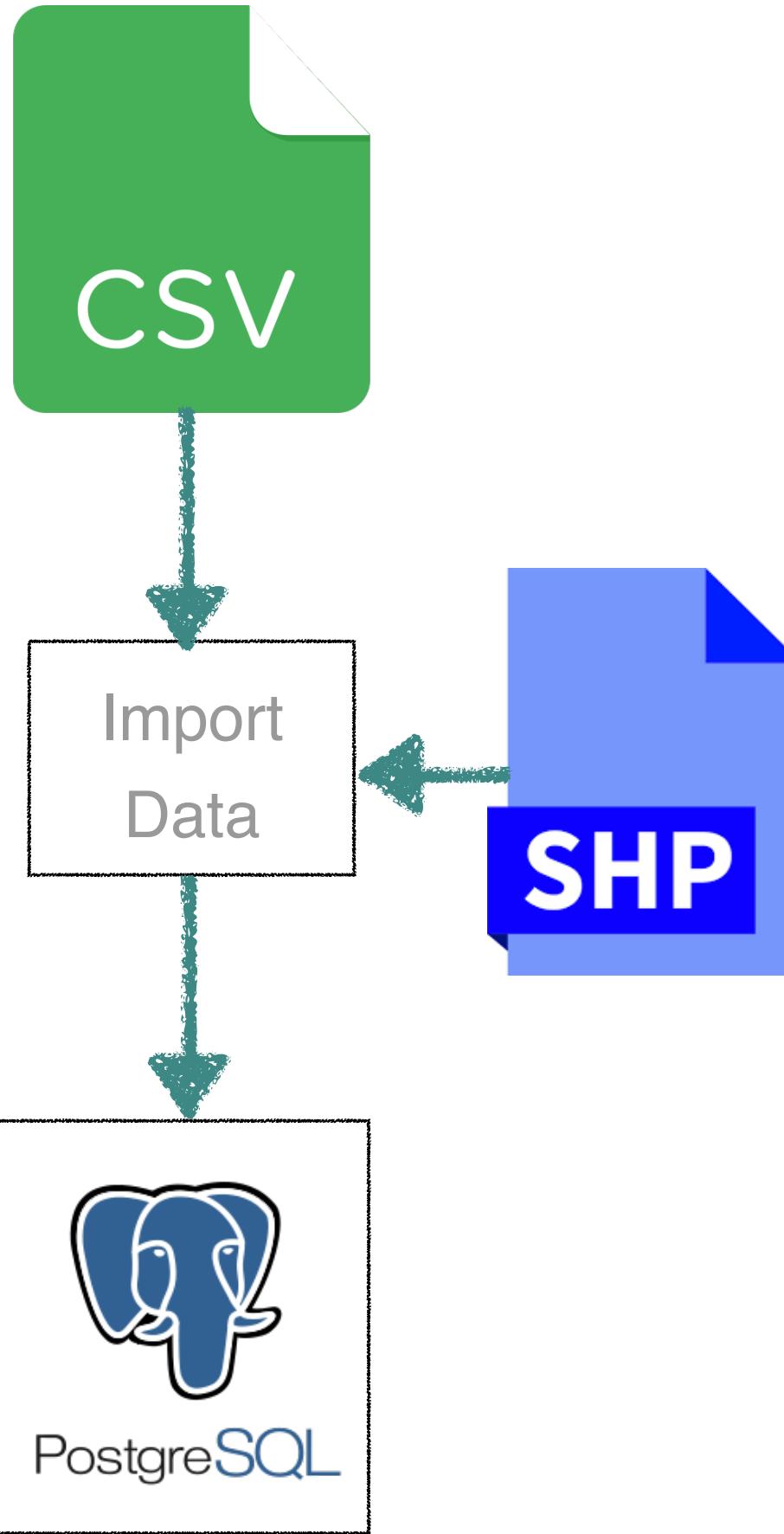
Using the tool of  
GeoServer for create  
services WMS,  
WMTS,WFS and WCS

# Spatial Data infrastructure & Services



i-bitz company limited  
THE GEOMATICS COMPANY

# Database management



**DB Manager**

Database Schema Table

Providers

- GeoPackage
- Oracle Spatial
- PostGIS
  - 10.0.0.120
  - Postgres
  - foss4g-2019
    - public
      - TMDStation
      - day-30-10-2019
      - geography\_columns
      - geometry\_columns
      - raster\_columns
      - raster\_overviews
      - spatial\_ref\_sys
    - topology
    - i-bitz
    - test
  - SpatiaLite
  - Virtual Layers

Import Layer/File... Export to File...

**TMDStation**

**General info**

Relation type: Table  
Owner: postgres  
Pages: 0  
Rows (estimation): 0  
Rows (counted): 122  
Privileges: select, insert, update, delete

**PostGIS**

Column: geom  
Geometry: MULTIPOLY  
Dimension: 2  
Spatial ref: WGS 84 (4326)  
Extent: (unknown) ([find out](#))

**Fields**

#	Name	Type	Length	Null	Default	Comment
1	id	int4	4	N	<code>nextval("TMDStation_id_seq")::regclass</code>	
2	geom	geometry (MultiPoint,4326)		Y		
3	wmonumber	varchar (80)		Y		
4	name	varchar (80)		Y		
5	province	varchar (80)		Y		

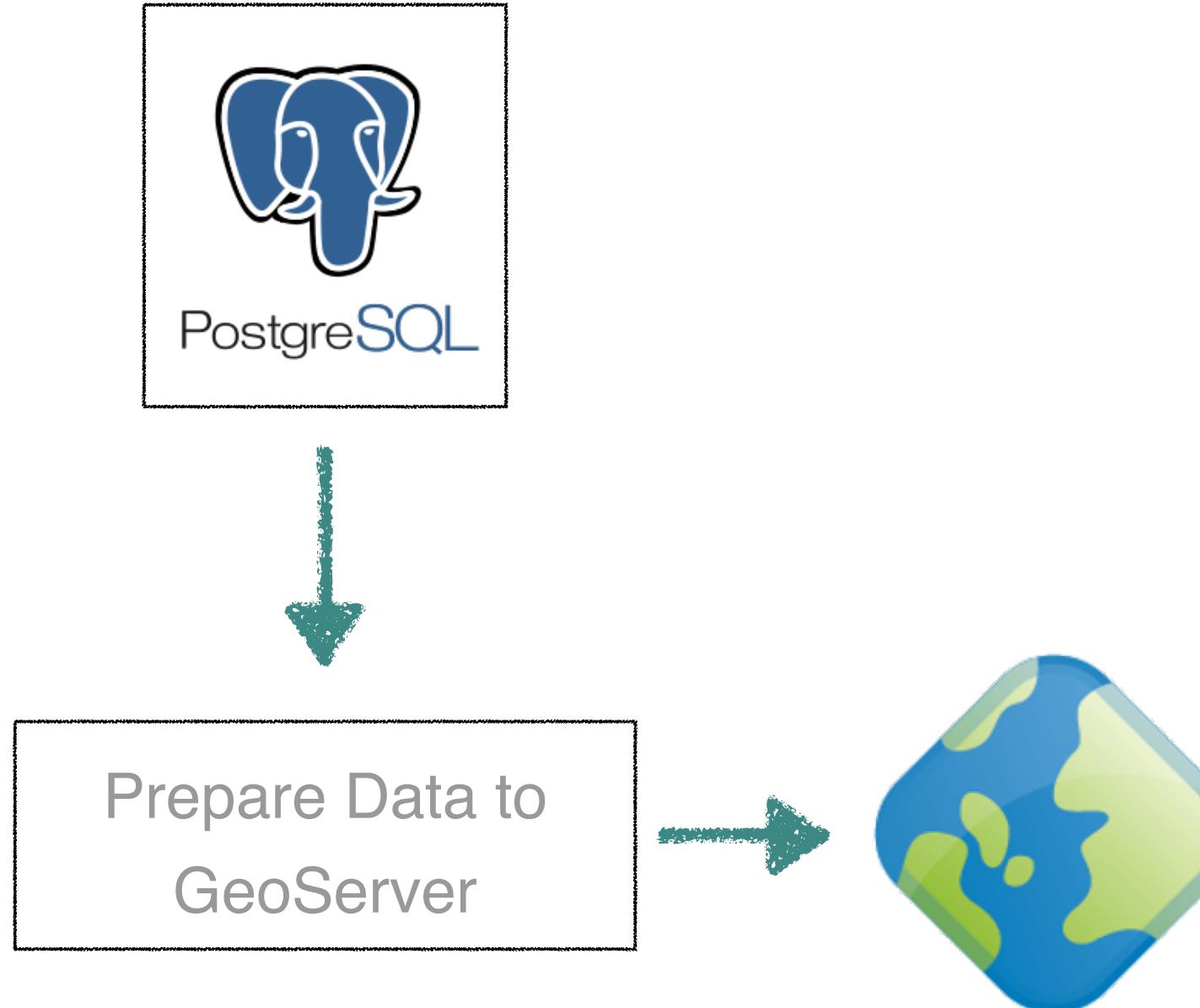
**Constraints**

Name	Type	Column(s)
TMDStation_pkey	Primary key	id

Import data to PostSQL using DB Manager in QGIS

# Database management

...



pgAdmin 4

Browser

- PostgreSQL 10
  - Databases (7)
    - foss4g
    - foss4g-2019
      - Casts
      - Catalogs
      - Event Triggers
      - Extensions
      - Foreign Data Wrappers
      - Languages
      - Schemas (2)
        - public
          - Collations
          - Domains
          - FTS Configurations
          - FTS Dictionaries
          - FTS Parsers
          - FTS Templates
          - Foreign Tables
          - Functions
          - Materialized Views
          - Sequences
          - Tables (3)
            - TMD-today
            - TMDStation
              - Columns
              - Constraints
              - Indexes
              - Rules
              - Triggers
            - spatial\_ref\_sys
          - Trigger Functions
          - Types
          - Views (6)
            - TMD-data
            - TMD-join
              - Columns
              - Constraints
              - Indexes
              - Rules
              - Triggers
            - geography\_columns
            - geometry\_columns
            - raster\_columns
            - raster\_overviews
    - Postgis
    - postgres

PostgreSQL 10 - foss4g-2019 - public.TMD-join

```

1 SELECT * FROM public."TMD-join"
2
  
```

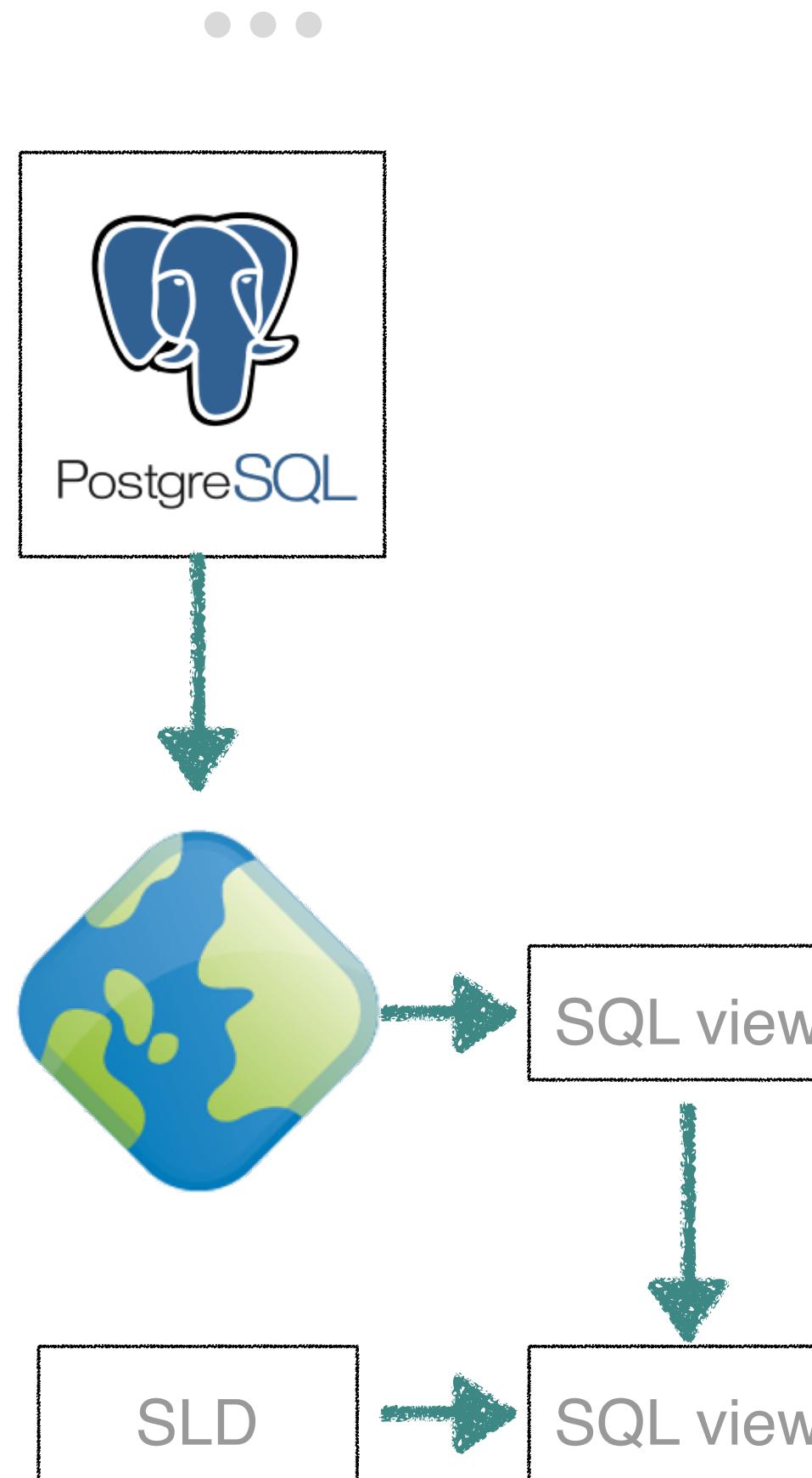
Data Output Explain Messages Query History

	WmoNumber character varying	rain character varying	temp character varying	to_date date	name character varying (80)	geom geometry
1	48325	9	21.8	2019-10-...	แม่สระบึง	01040002...
2	48303	0	22.4	2019-10-...	เชียงราย	01040002...
3	48304	1.4	22	2019-10-...	เชียงราย สกช.	01040002...
4	48310	12.7	23	2019-10-...	พะเยา	01040002...
5	48302	0	16.9	2019-10-...	ดอยล่างข้าง	01040002...
6	48327	15.5	22.1	2019-10-...	เชียงใหม่	01040002...
7	48328	1.8	24.5	2019-10-...	สapaง	01040002...
8	48324	5.6	24.5	2019-10-...	เชียง	01040002...
9	48334	0	23	2019-10-...	สapaง สกช.	01040002...
10	48329	2	22.5	2019-10-...	สapa	01040002...
11	48330	0.01	24.4	2019-10-...	แมร์	01040002...
12	48331	0	23.3	2019-10-...	น่าน	01040002...
13	48333	0.5	22.4	2019-10-...	น่าน สกช.	01040002...
14	48315	0.2	22.3	2019-10-...	ท่าวังผา	01040002...
15	48307	0	21	2019-10-...	ทุ่งช้าง	01040002...
16	48352	0	22.3	2019-10-...	หนองคาย	01040002...
17	48353	0	24	2019-10-...	เลย	01040002...
18	48350	0	23	2019-10-...	เลย สกช.	01040002...
19	48354	0	21.2	2019-10-...	อุดรธานี	01040002...
20	48356	0	20	2019-10-...	สกลนคร	01040002...
21	48355	0	18.2	2019-10-...	สกลนคร สกช.	01040002...
22	48357	0	20.2	2019-10-...	นครพนม	01040002...
23	48358	0	23.5	2019-10-...	นครพนม สกช.	01040002...
24	48360	0	23.4	2019-10-...	หนองบัวลำภู	01040002...
25	48372	13	24.6	2019-10-...	สจทัย	01040002...

Prepare Data in PostgreSQL for support layer of GeoServer

1. Convert date date to format date
2. Create view “TMD\_today”
3. Joint data TMD\_today and TMDStation
4. Create view “TMD\_join”

# Management layer in GeoServer



**Edit SQL view**

Update the definition of the SQL view and its metadata

**View Name:** RainToday

**SQL statement:**

```
Select "name", "to_date", "rain", "geom" from "TMD_join" WHERE "to_date" = CURRENT_DATE
```

**SQL view parameters:**

Guess parameters from SQL   Add new parameter   Remove selected

<input type="checkbox"/>	Name	Default value	Validation regular expression
<input type="checkbox"/>	Escape special SQL characters		
<input checked="" type="checkbox"/>	Guess geometry type and srid		
	Name	Type	SRID
	name	String	
	to_date	Date	
	rain	String	
	geom	Geometry	4326

**Attributes:**

Refresh    Guess geometry type and srid

**Security:**

Settings    Authentication    Passwords    Users, Groups, Roles    Data    Services    WPS security

**Monitor:**

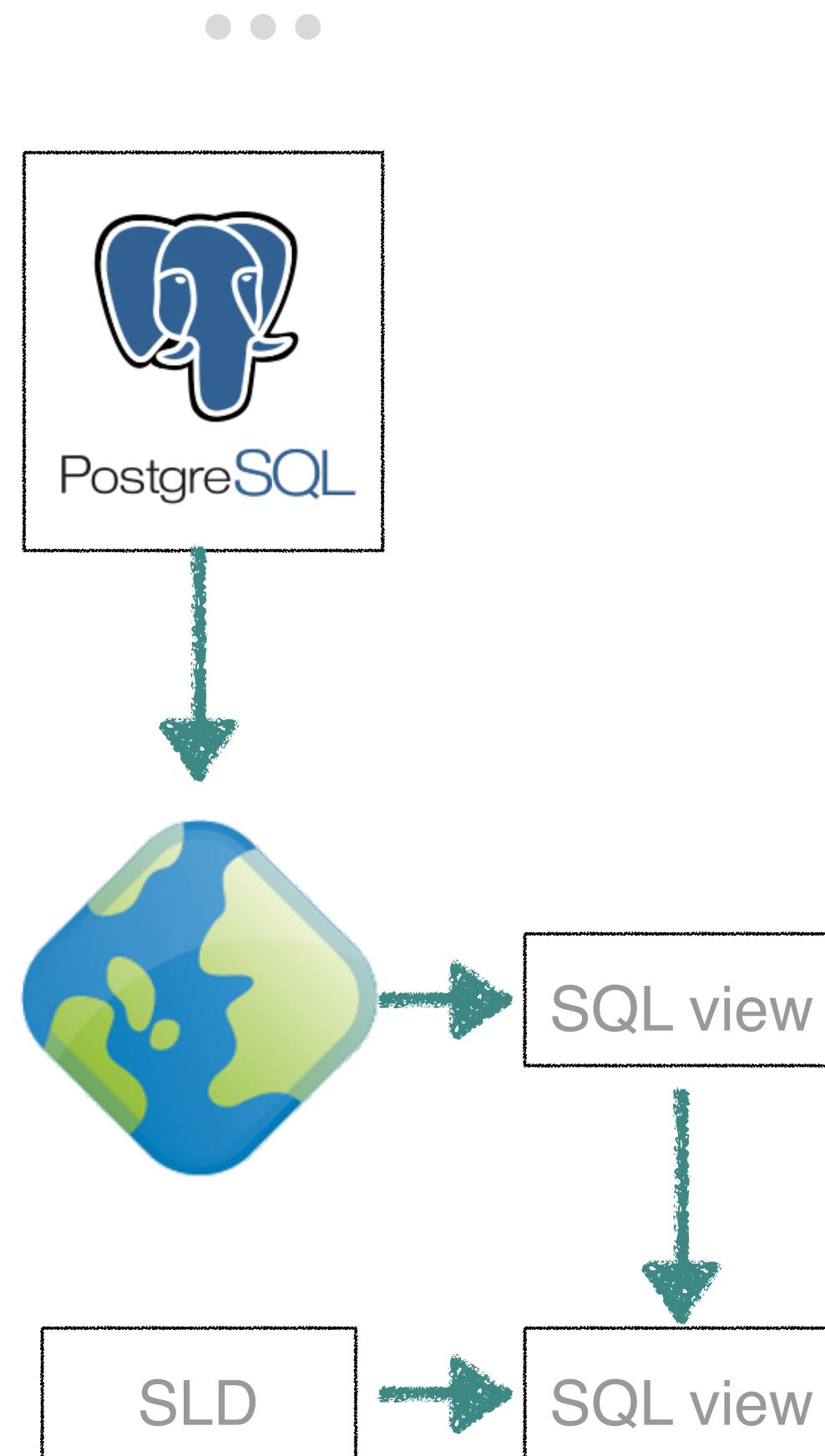
Activity

**Buttons:**

**Save**   **Cancel**

1. Create layer in GeoServer using SQL view
2. Create style for layer RainToday

# Management layer in GeoServer

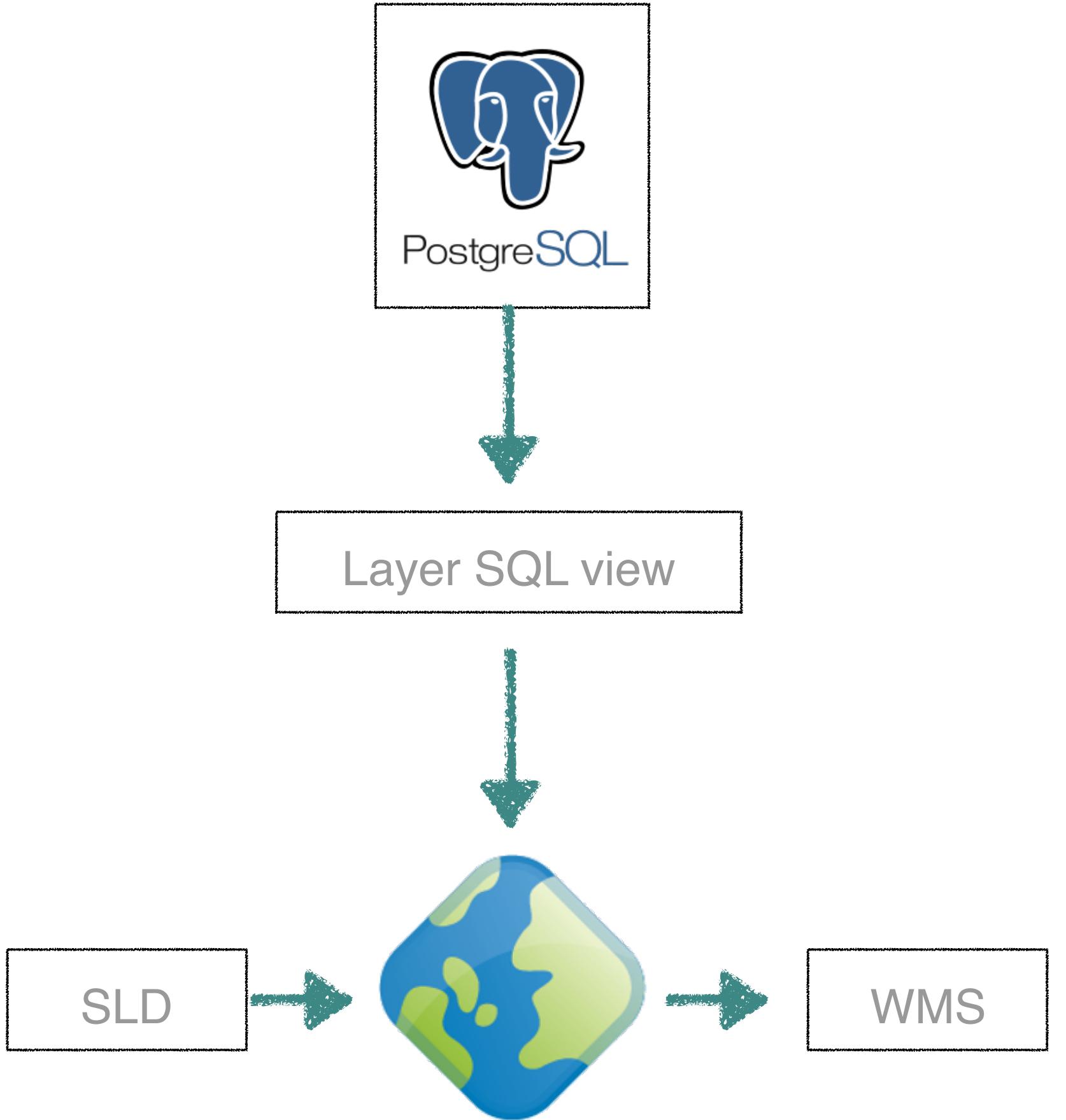


1. Rain today
2. Rain yesterday
3. Rain last 3 day



# Data service WMS

•••



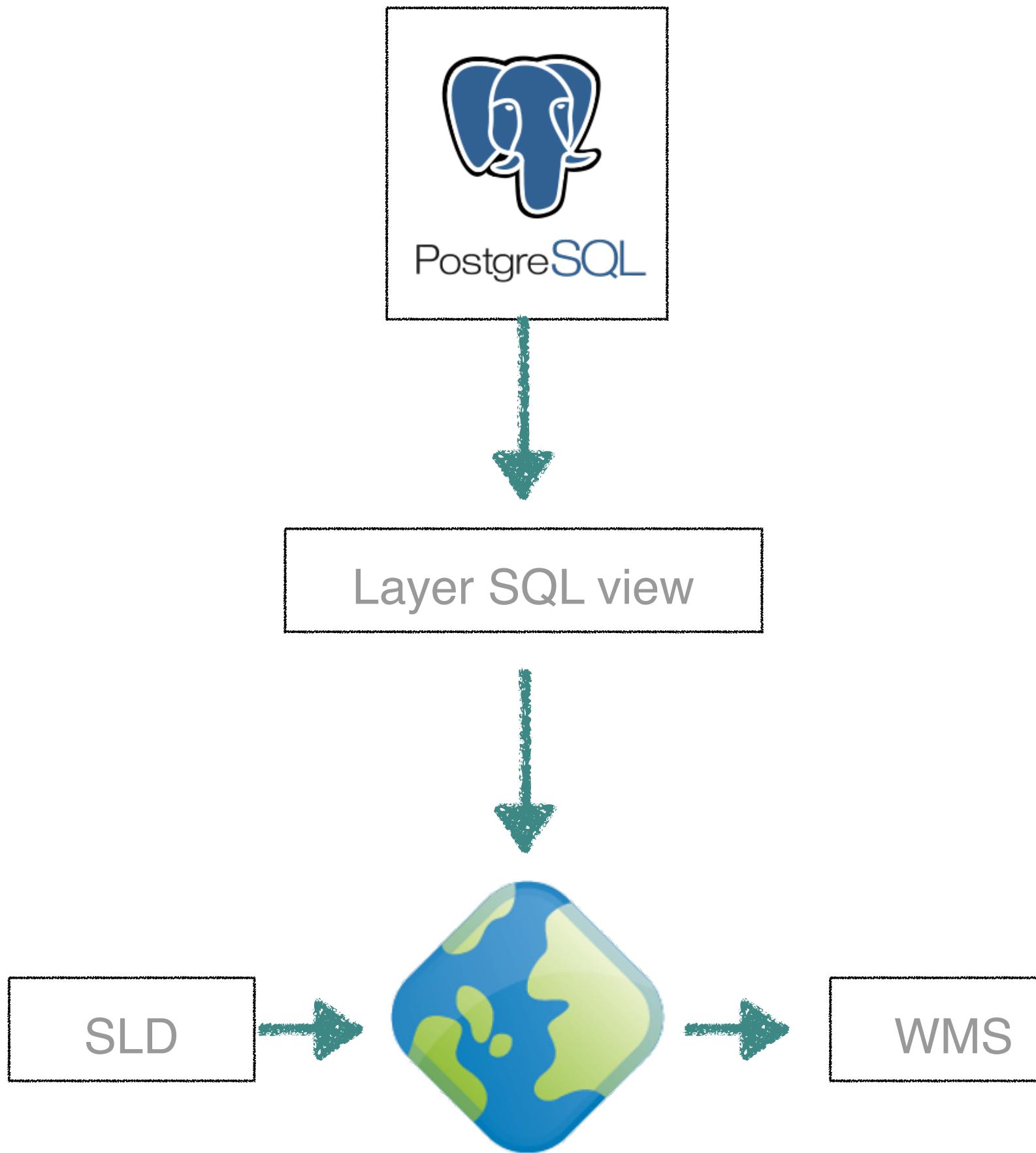
การใช้งานร้องขอข้อมูลลักษณะของแผนที่ต่าง ๆ ที่มีค่าพิกัดของข้อมูลแผนที่อยู่ด้วย Format ต่าง ๆ ประกอบด้วย PNG, GIF หรือ JPEG

## มาตรฐาน WMS: Web Map Service รองรับการร้องขอบริการจากผู้ใช้

1. GetCapabilities จะส่งค่าการให้บริการ ในส่วนของ Metadata ซึ่งเป็นตัวอธิบายเกี่ยวกับ รายละเอียดของ ข้อมูลที่ให้บริการและการยอมรับค่าตัวแปรต่าง ๆ
2. GetMap จะเป็นการส่งภาพแผนที่ซึ่งสามารถระบุชั้นข้อมูล ขนาดของภาพแผนที่และลักษณะ ของภาพ แผนที่ได้ ซึ่งรูปแผนที่แสดงภาพในรูปแบบ PNG, GIF หรือ JPEG
3. GetFeatureInfo มาตรฐานตัวนี้จะเป็น Option ใน การร้องขอข้อมูลเกี่ยวกับรายละเอียดของ ข้อมูล ในแผนที่ การร้องขอการให้บริการต้องระบุเงื่อนไขทาง ตำแหน่ง เพื่อทำการสืบค้นข้อมูลของ แผนที่มาแสดงผล
4. GetLegendGraphic มาตรฐานตัวนี้จะเป็น Option ใน การร้องขอข้อมูลเกี่ยวกับรายละเอียดของค่าอธิบายสัญลักษณ์แผนที่

# Data service WMS

• • •



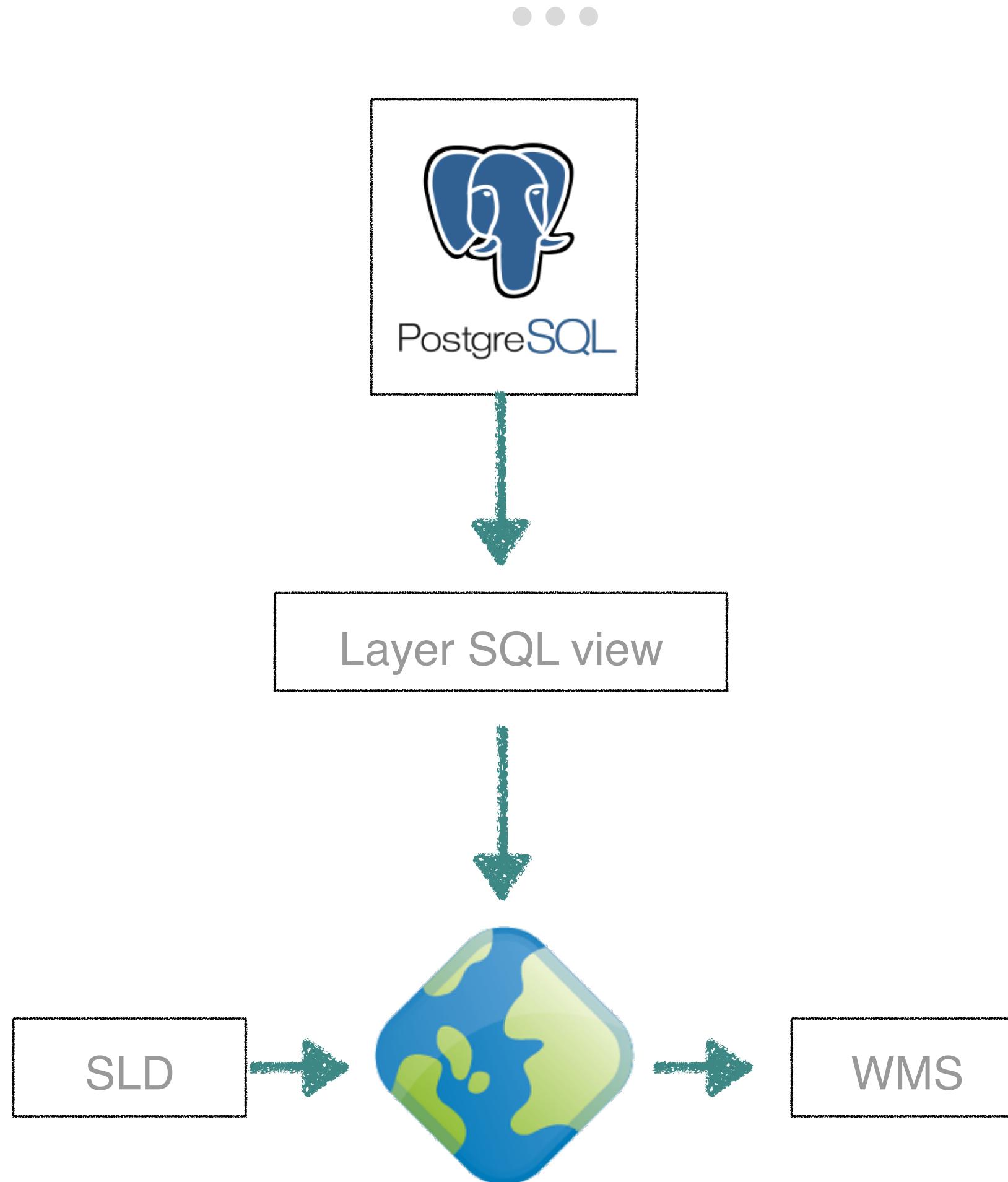
## WMS: Requests GetCapabilities

`http://localhost:8082/geoserver/ows?service=wms &version=1.1.1 &request=GetCapabilities`

```

<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE WMT_MS_Capabilities SYSTEM "http://localhost:8082/geoserver/schemas/wms/1.1.1/WMS_MS_Capabilities.dtd">
<WMT_MS_Capabilities updateSequence="261" version="1.1.1">
  - <Service>
    <Name>OGC:WMS</Name>
    <Title>GeoServer Web Map Service</Title>
    <Abstract>A compliant implementation of WMS plus most of the SLD extension (dynamic styling). Can also gene
  - <KeywordList>
    <Keyword>WFS</Keyword>
    <Keyword>WMS</Keyword>
    <Keyword>GEO SERVER</Keyword>
  </KeywordList>
  <OnlineResource xlink:href="http://geoserver.org" xlink:type="simple" xmlns:xlink="http://www.w3.org/1999/xli
  - <ContactInformation>
    - <ContactPersonPrimary>
      <ContactPerson>sattawat Arab</ContactPerson>
      <ContactOrganization>The Ancient Geographers</ContactOrganization>
    </ContactPersonPrimary>
    <ContactPosition>GIS Analyst</ContactPosition>
    - <ContactAddress>
      <AddressType>Work</AddressType>
      <Address/>
      <City>Bangkok</City>
      <StateOrProvince/>
      <PostCode/>
      <Country>Thailand</Country>
    </ContactAddress>
    <ContactVoiceTelephone/>
    <ContactFacsimileTelephone/>
    <ContactElectronicEmailAddress>sattawatarab@gmail.com</ContactElectronicEmailAddress>
  </ContactInformation>
  <Fees>NONE</Fees>
  <AccessConstraints>NONE</AccessConstraints>
</Service>
- <Capability>
  
```

# Data service WMS

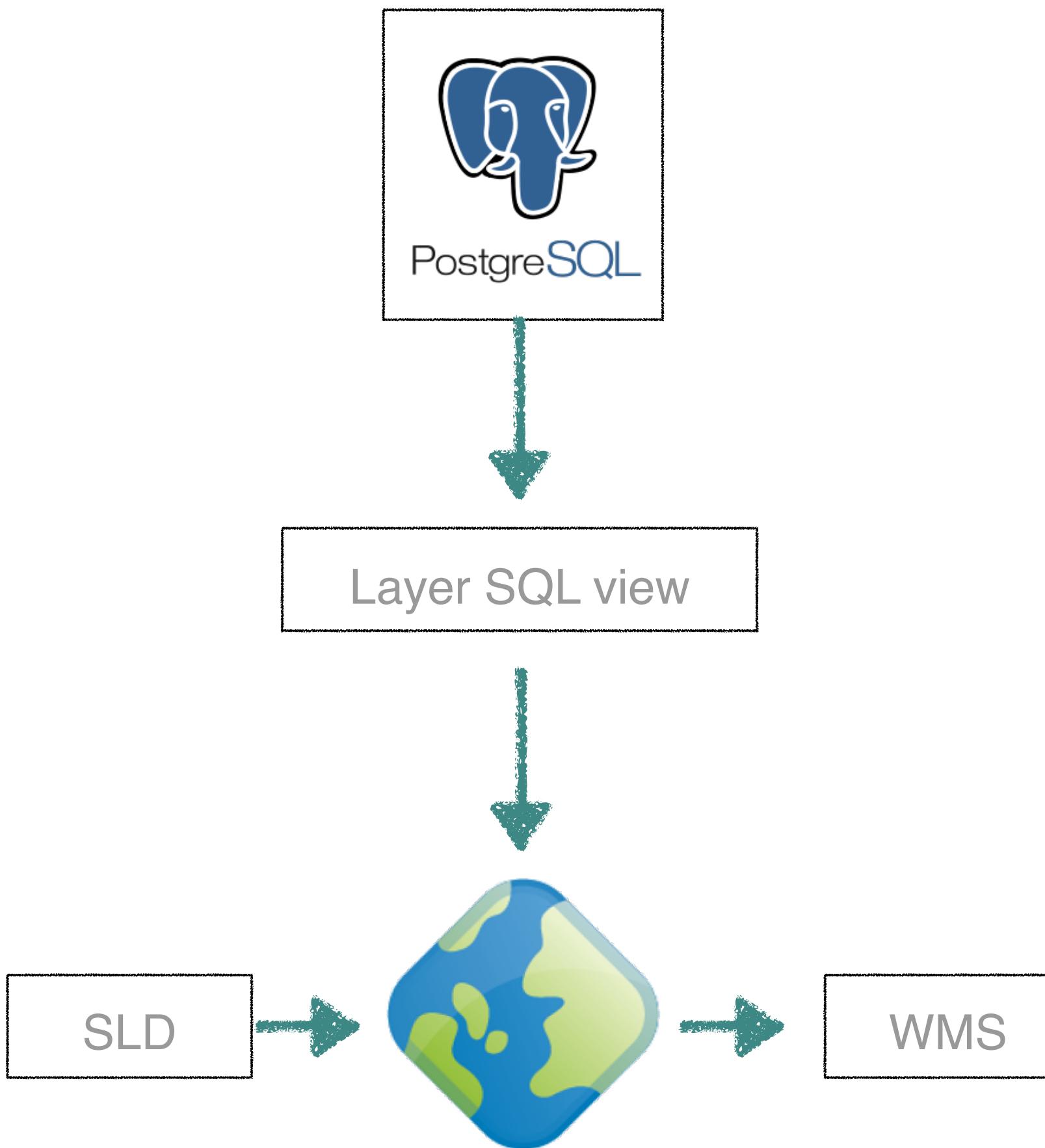


## WMS: Reques GetMap

```
http://localhost:8082/geoserver/foss4g-2019/wms?  
service=WMS  
&version=1.1.0  
&request=GetMap  
&layers=foss4g-2019:RainToday  
&styles=  
&bbox=97.93333333,6.416666667,105.0235,19.96138889  
&width=402&height=768  
&srs=EPSG:4326  
&format=image/png
```

# Data service WMS

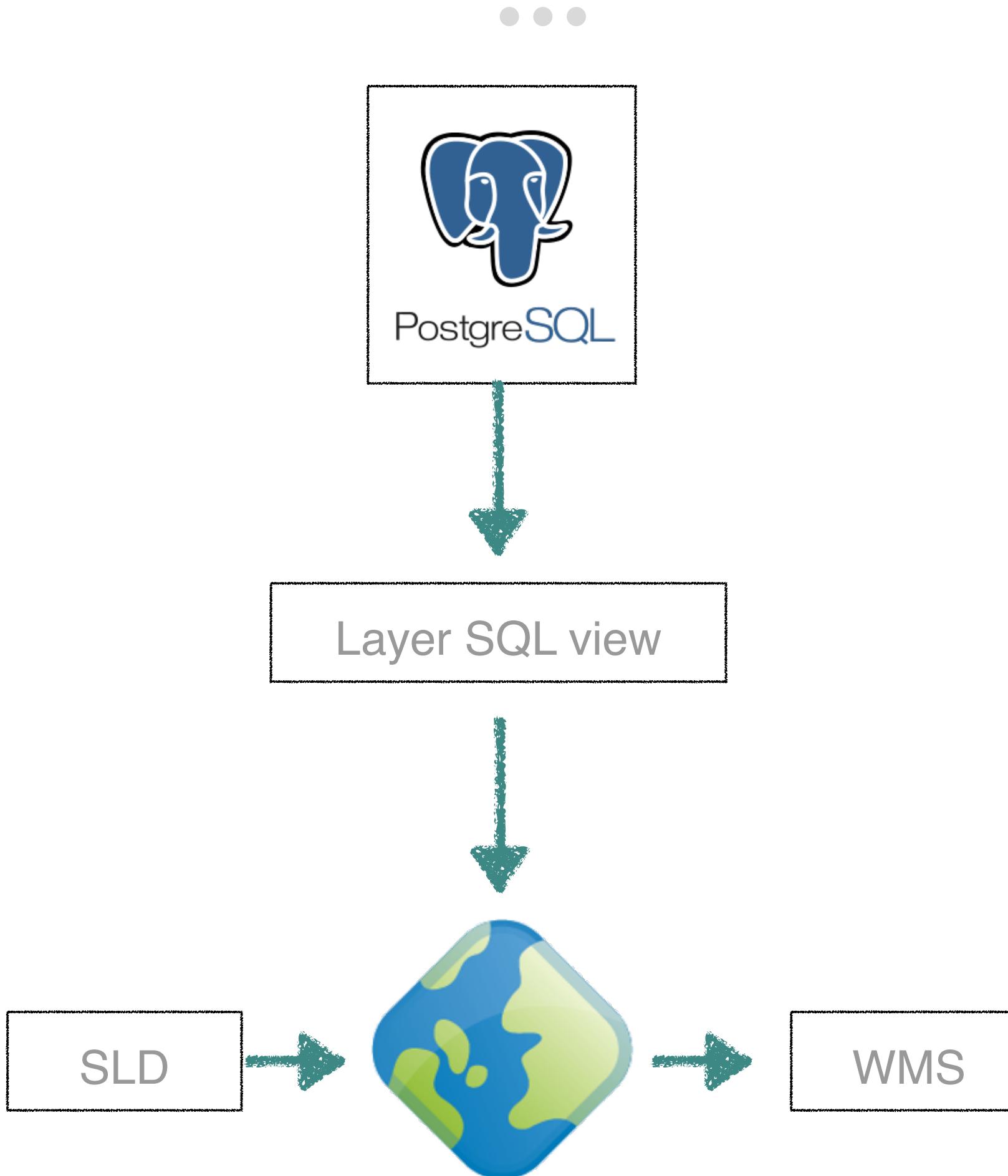
• • •



## WMS: Request GetFeatureInfo

```
http://localhost:8082/geoserver/foss4g-2019/wms?SERVICE=WMS
&VERSION=1.1.1
&REQUEST=GetFeatureInfo
&FORMAT=image/png
&TRANSPARENT=true&QUERY_LAYERS=foss4g-2019:RainToday
&STYLES
&LAYERS=foss4g-2019:RainToday
&info_format=application/json
&X=50&Y=50
&SRS=EPSG:4326
&WIDTH=101&HEIGHT=101
&BBOX=101.72241210937501,10.733642578125,103.9416503906250
1,12.952880859375
```

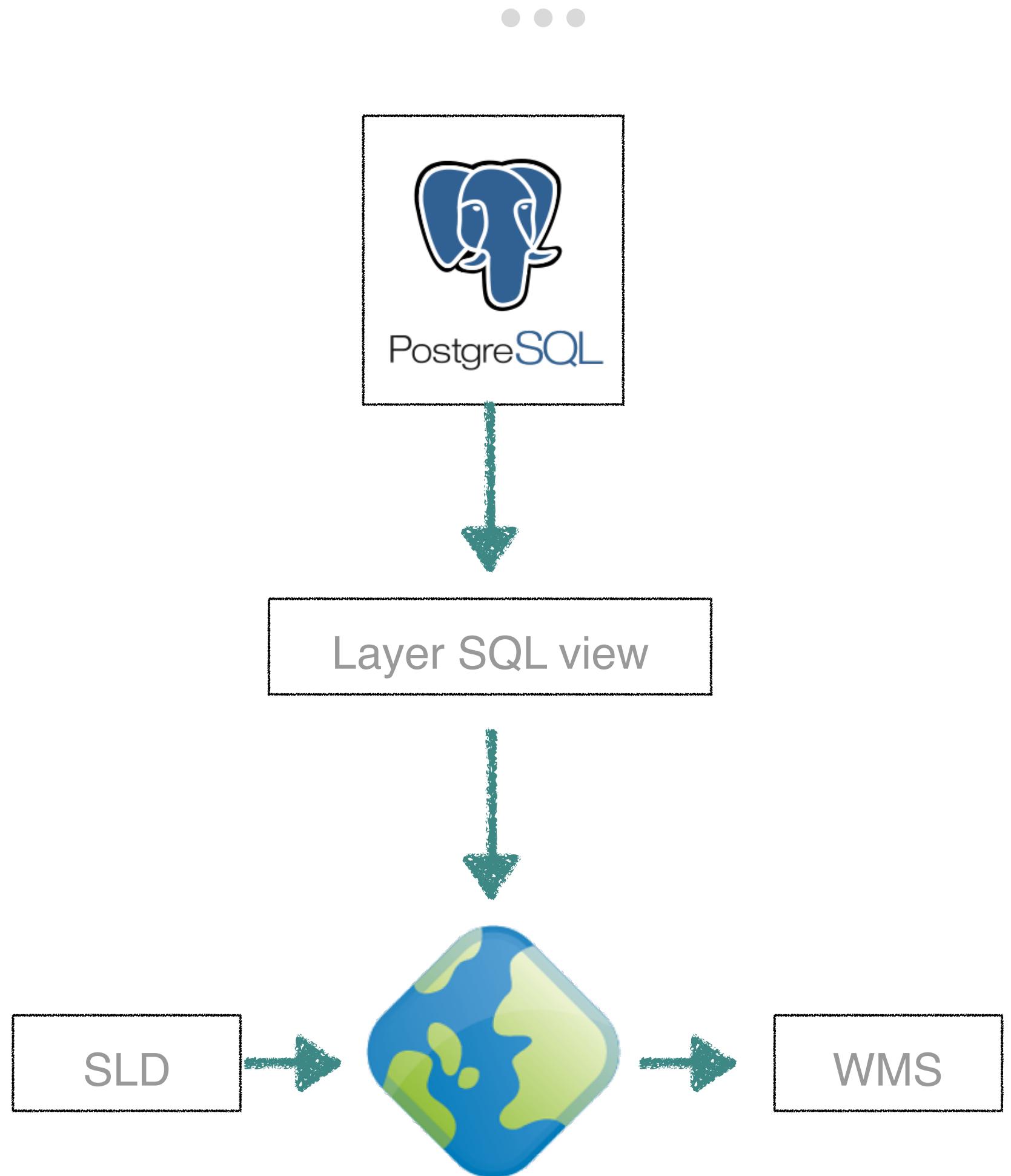
# Data service WMS



## WMS: Request GetLegendGraphic

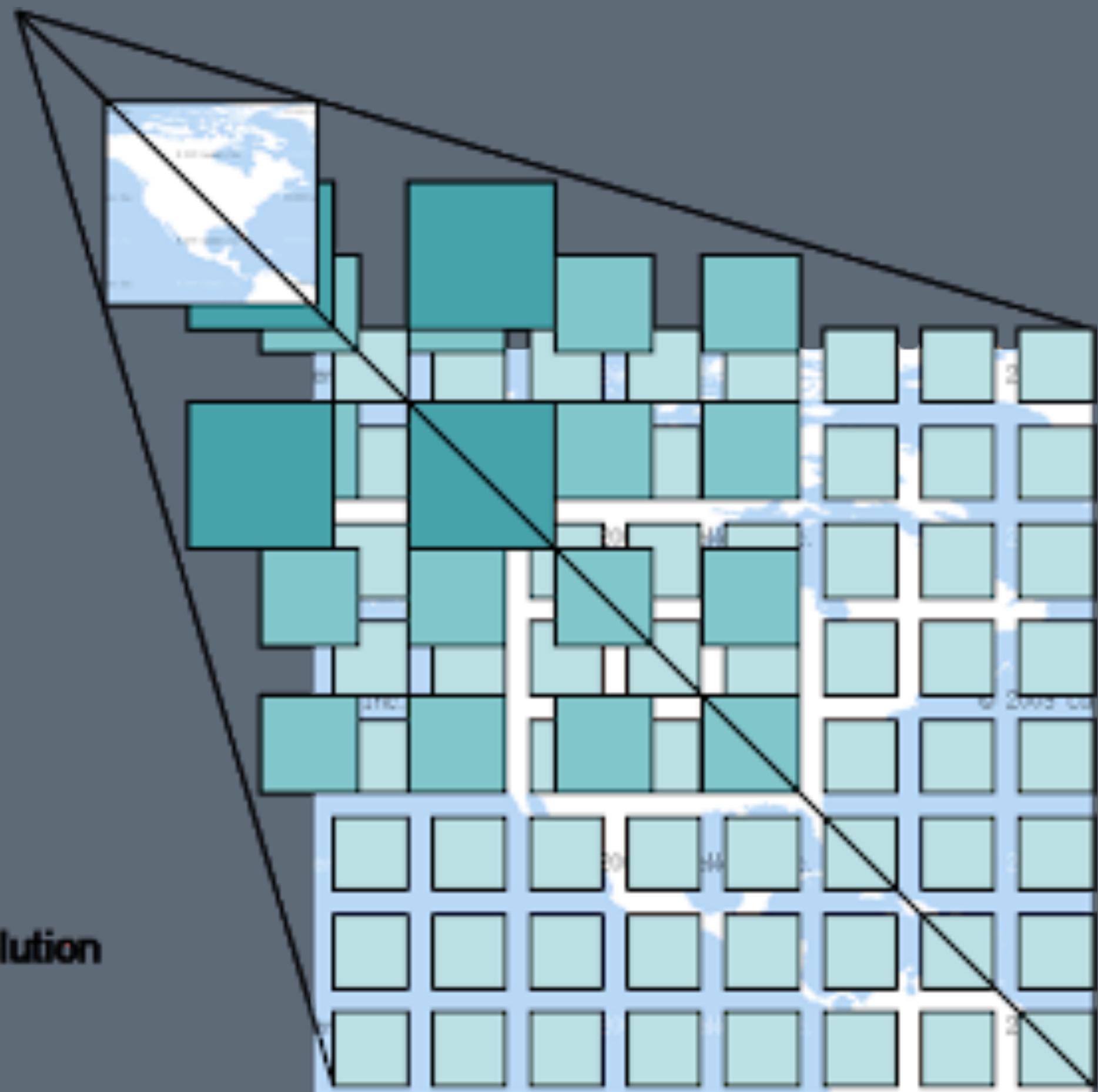
```
http://localhost:8082/geoserver/foss4g-2019/wms?service=WMS  
&request=GetLegendGraphic  
&version=1.3.0  
&FORMAT=image/png  
&WIDTH=25  
&HEIGHT=30  
&Layer=RainToday
```

# Data service WMTS



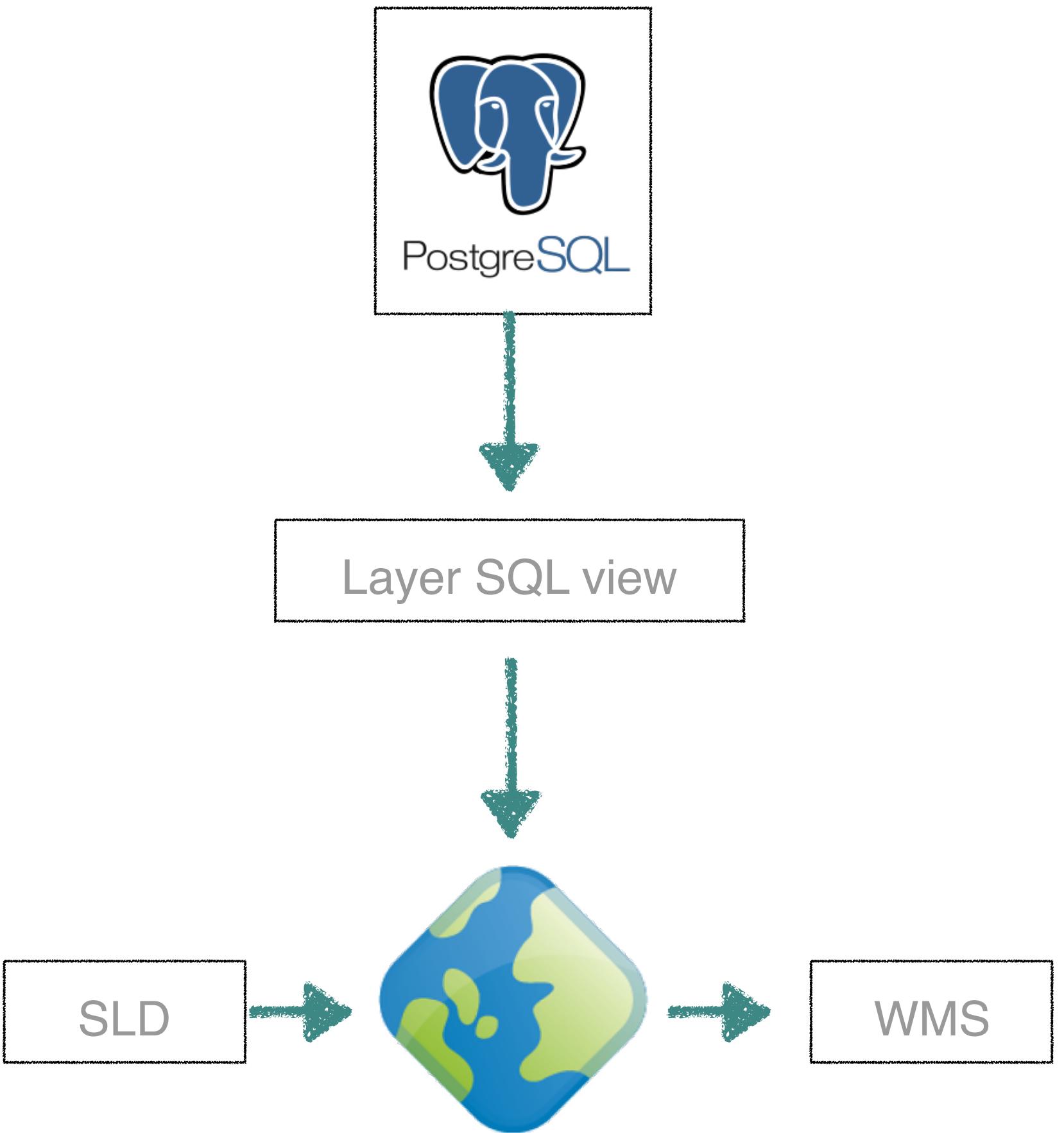
Coarse resolution  
Highest scale denominator

Detailed resolution  
Lowest scale denominator



# Data service WMTS

•••



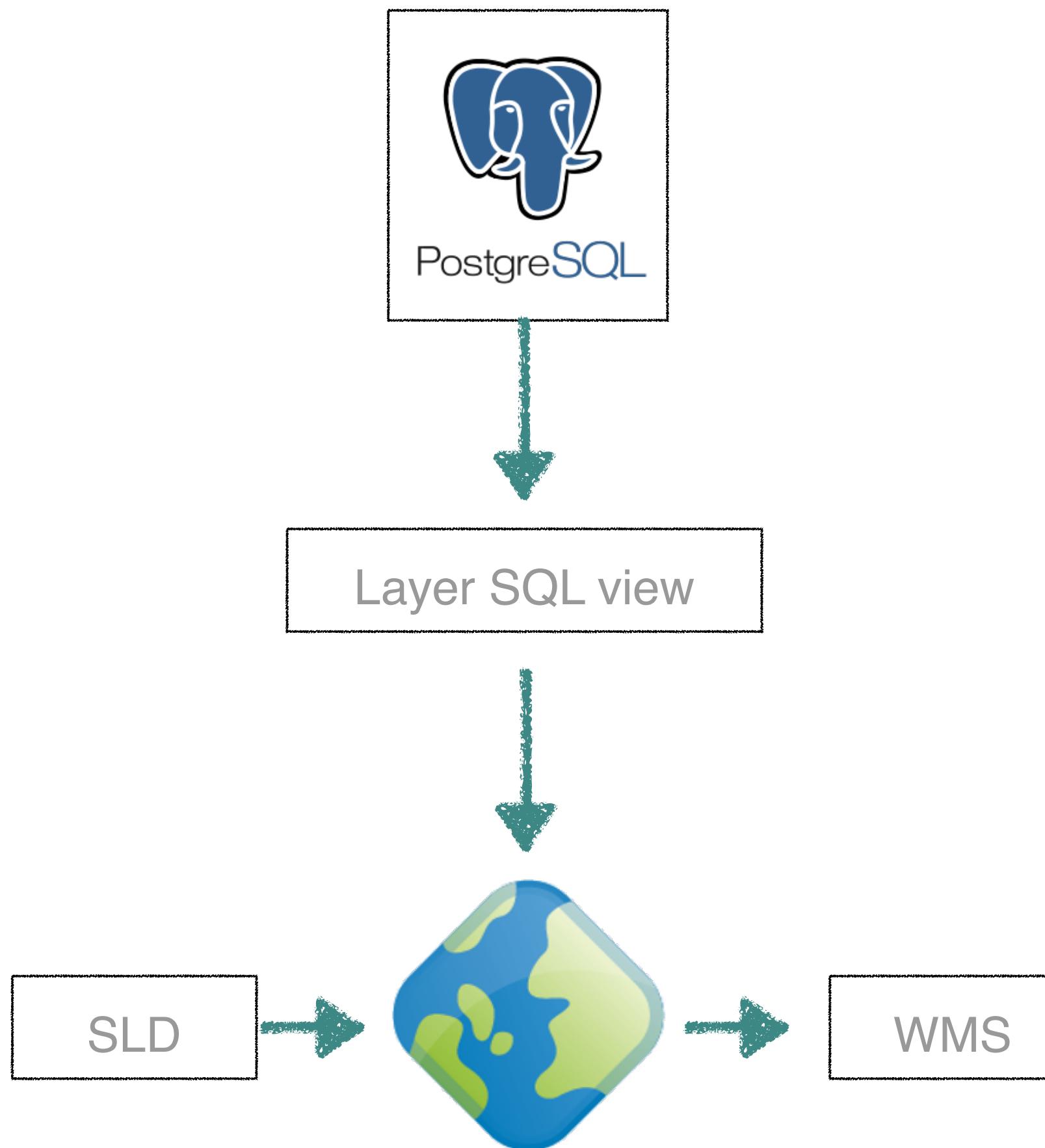
การใช้งานร้องขอข้อมูลลักษณะของแผนที่ต่าง ๆ โดยข้อมูลแผนที่จะถูกแบ่งเป็น Tile (ชิ้นของรูปภาพ) โดยผู้ร้องขอจะไม่ได้ขอมาจริงๆ ใน Layer แต่จะได้ข้อมูลที่ถูกทำ Tile ไว้แล้ว ทำให้การให้บริการรูปแบบนี้เร็วกว่าการให้บริการแบบ WMS

**WMTS: Web Map Tile Service รองรับการร้องขอบริการจากผู้ใช้**

1. GetCapabilities จะส่งค่าการให้บริการ ในส่วนของ Metadata ซึ่งเป็นตัวอธิบายเกี่ยวกับ รายละเอียดของ ข้อมูลที่ให้บริการและการยอมรับค่าตัวแปรต่าง ๆ
2. GetMap จะเป็นการส่งภาพแผนที่ซึ่งสามารถบุชั้นข้อมูล ชิ้นของ Tile แผนที่ที่ต้องการแสดงได้ ข้อมูลที่ได้จะอยู่ในรูปของไฟล์รูปภาพ เช่น PNG

# Data service WMTS

• • •



## WMTS: Reques GetCapabilities

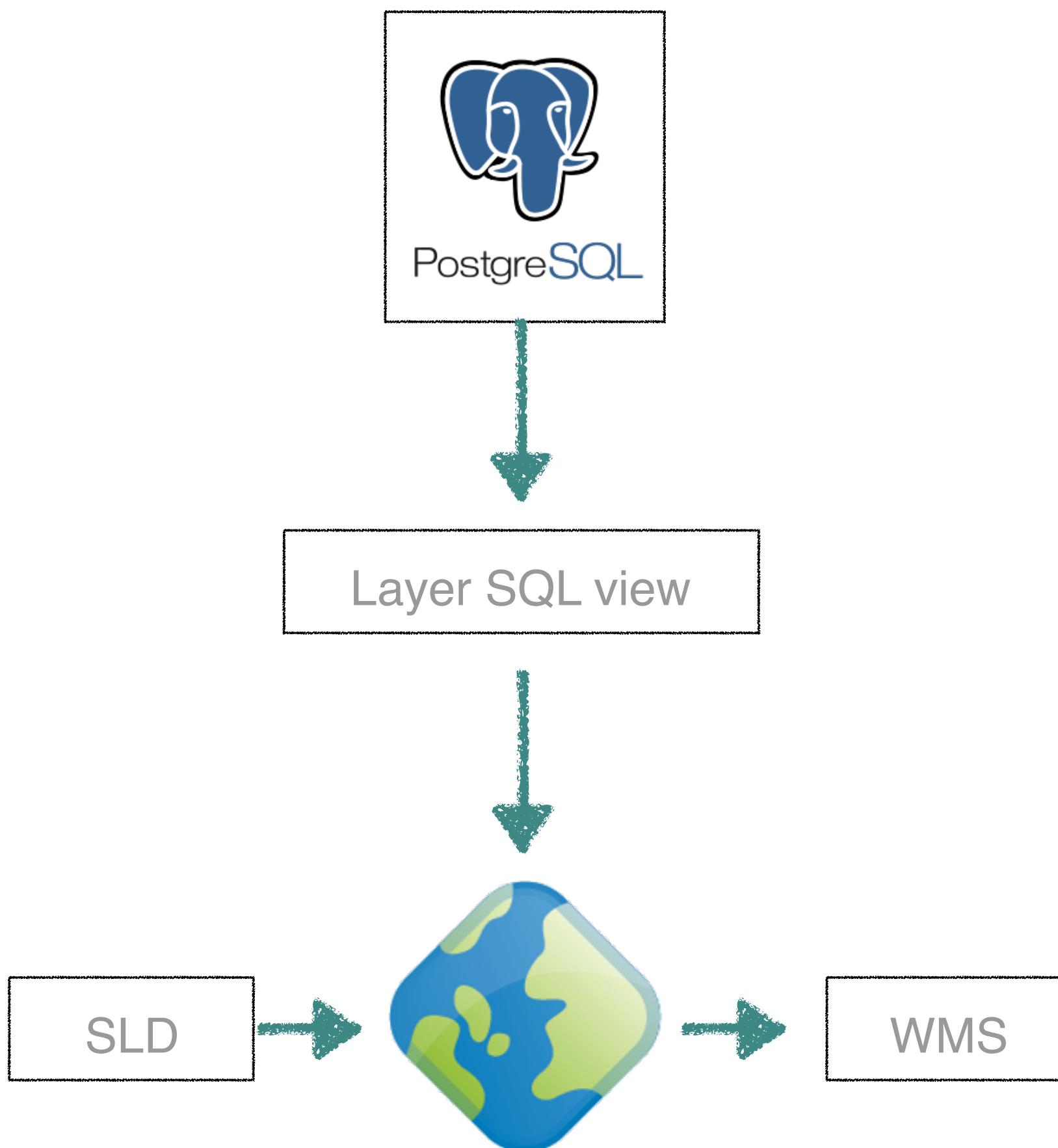
[http://localhost:8082/geoserver/ows?service=wms &version=1.1.1 &request=GetCapabilities](http://localhost:8082/geoserver/ows?service=wms&version=1.1.1&request=GetCapabilities)

```

</ows:WGS84BoundingBox>
<ows:Identifier>sattawat:Centroid</ows:Identifier>
<Style isDefault="true">
    <ows:Identifier/>
    <LegendURL width="20" height="20" format="image/png" xlink:href="http://localhost:8082/geoserver/ows?service=WMS&request=GetLegendGraphic&layer=sattawat:Centroid"/>
</Style>
<Format>image/png</Format>
<Format>image/jpeg</Format>
<InfoFormat>text/plain</InfoFormat>
<InfoFormat>application/vnd.ogc.gml</InfoFormat>
<InfoFormat>text/xml</InfoFormat>
<InfoFormat>application/vnd.ogc.gml/3.1.1</InfoFormat>
<InfoFormat>text/xml</InfoFormat>
<InfoFormat>text/html</InfoFormat>
<InfoFormat>application/json</InfoFormat>
<TileMatrixSetLink>
    <TileMatrixSet>EPSG:4326</TileMatrixSet>
    <TileMatrixSetLimits>
        <TileMatrixLimits>
            <TileMatrix>EPSG:4326:0</TileMatrix>
            <MinTileRow>0</MinTileRow>
            <MaxTileRow>0</MaxTileRow>
            <MinTileCol>1</MinTileCol>
            <MaxTileCol>1</MaxTileCol>
        </TileMatrixLimits>
    </TileMatrixSetLimits>
</TileMatrixSetLink>
  
```

# Data service WMTS

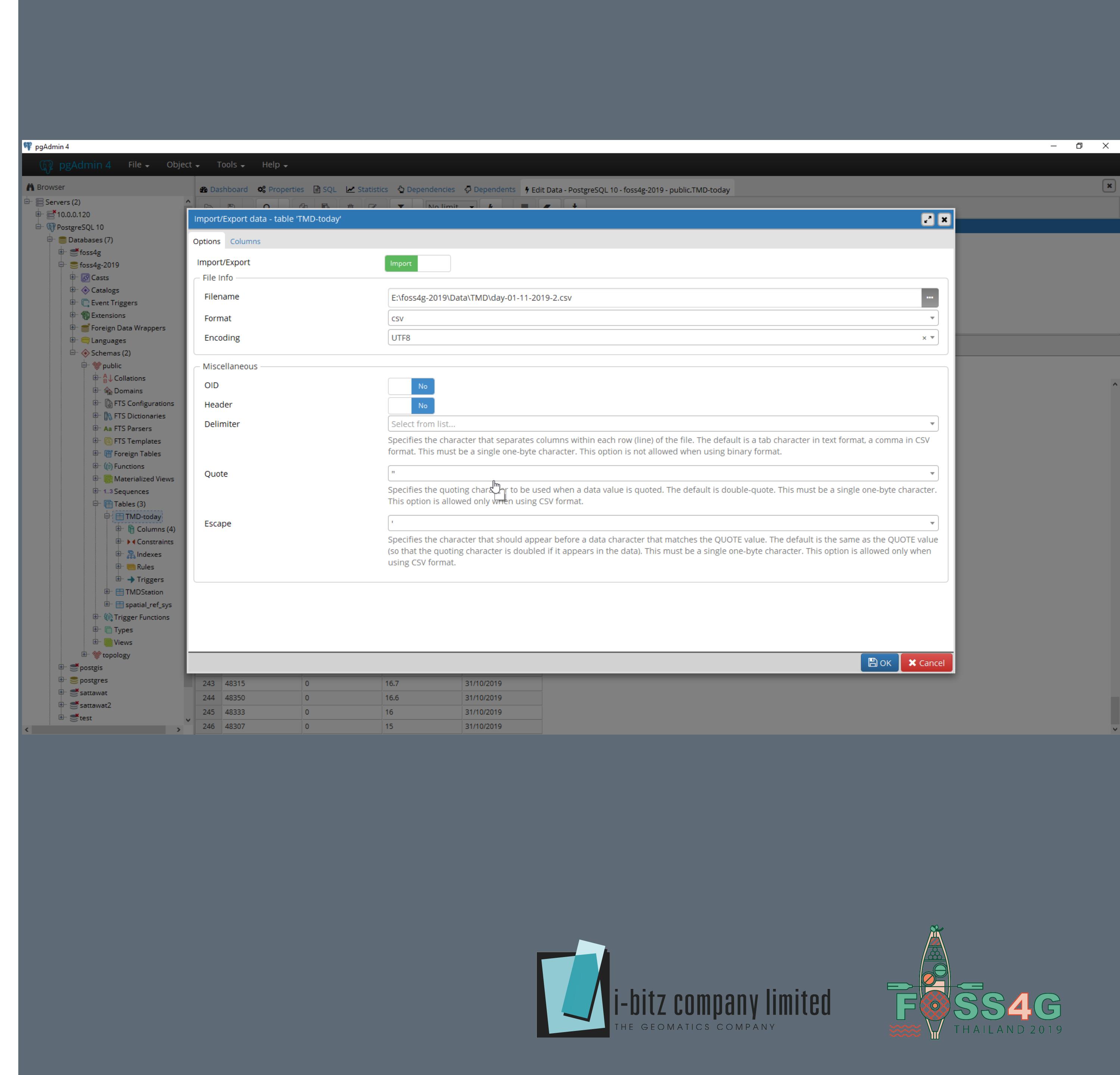
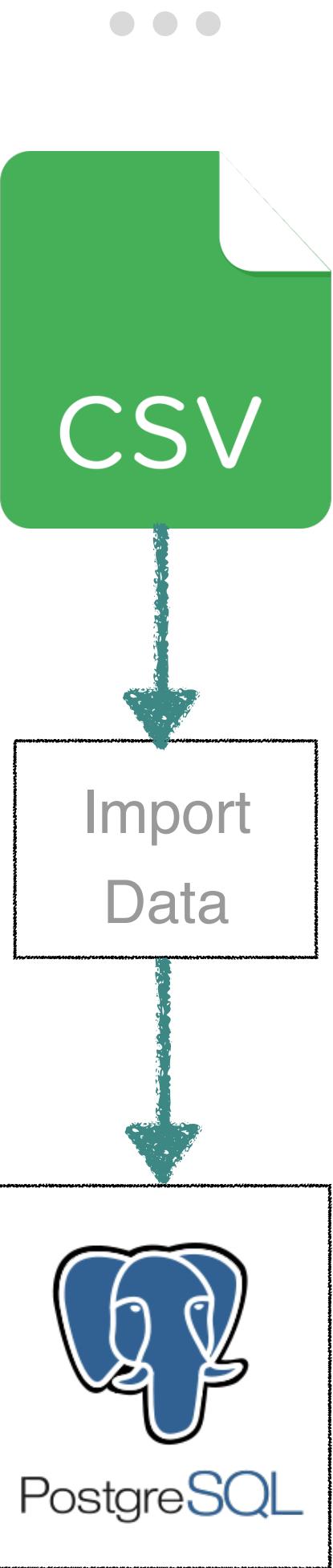
...



## WMTS: Request GetMap

```
http://localhost:8082/geoserver/gwc/service/wmts?
layer=foss4g-2019:RainToday
&style=
&tilematrixset=EPSG:4326
&Service=WMTS
&Request=GetTile
&Version=1.0.0
&Format=image/png
&TileMatrix=EPSG:4326:0
&TileCol=1&TileRow=0
```

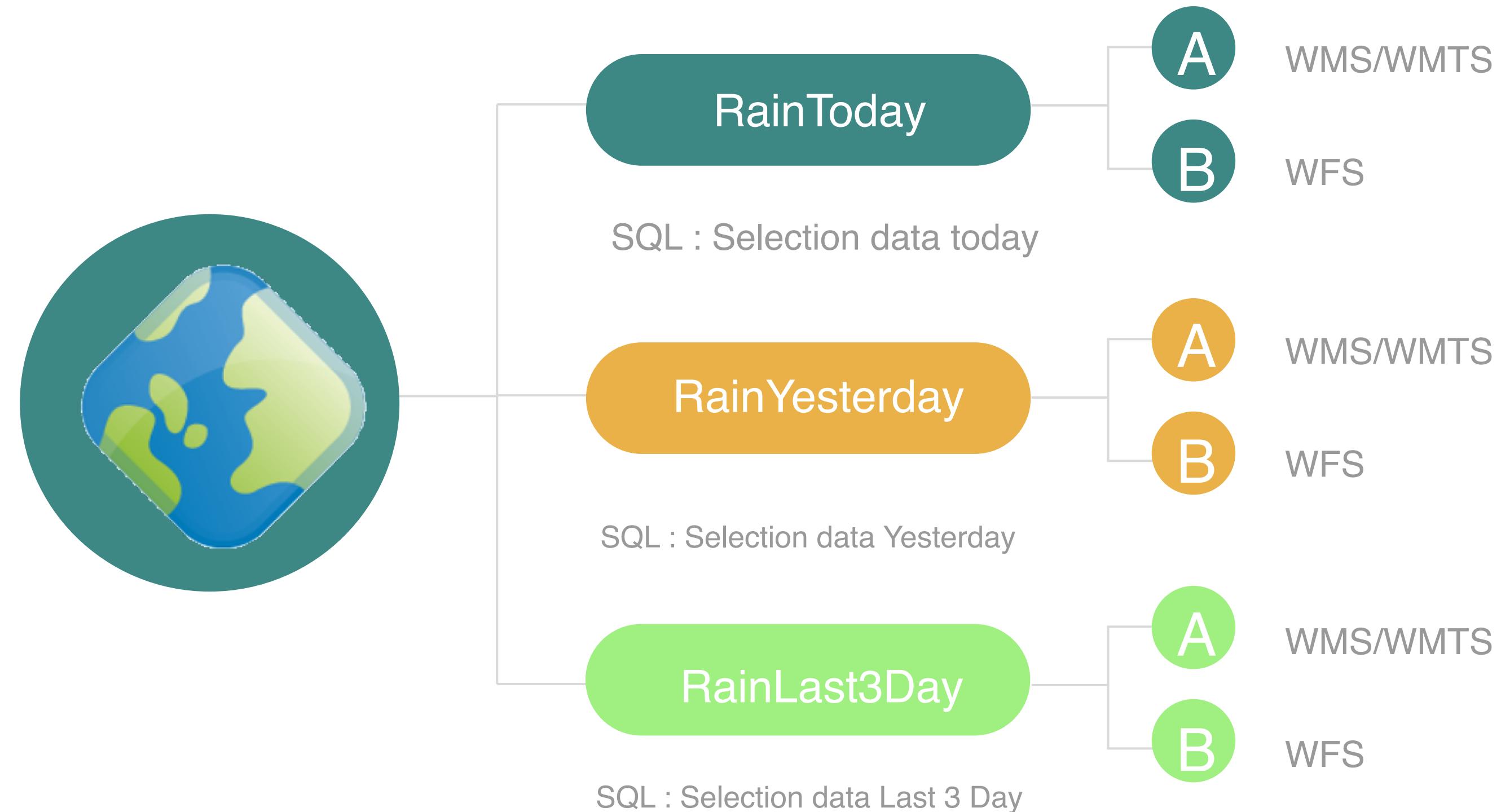
# Import CSV To PostgreSQL



Basic to Advance GeoServer

# Create Vector Layer

• • •



**Edit SQL view**

Update the definition of the SQL view and its metadata

**View Name:** RainToday

**SQL statement:**

```
Select "name","to_date","rain","geom" from "TMD_join" WHERE "to_date" = CURRENT_DATE
```

**SQL view parameters:**

Name	Type	Default value	Validation regular expression
name	String		
to_date	Date		
rain	String		
geom	Geometry	4326	

**Settings:**

- Global
- Image Processing
- Raster Access

**Tile Caching:**

- Tile Layers
- Caching Defaults
- Gridsets
- Disk Quota
- BlobStores

**Security:**

- Settings
- Authentication
- Passwords
- Users, Groups, Roles
- Data
- Services
- WPS security

**Monitor:**

- Activity

**Layers**

Manage the layers being published by GeoServer

**Add a new layer**

**Remove selected layers**

**Data:**

Type	Title	Name	Store	Enabled	Native SRS
Buffer	sattawat:Buffer	postgis	✓	✓	EPSG:4326
Centroid	sattawat:Centroid	postgis	✓	✓	EPSG:4326
Contains	sattawat:Contains	postgis	✓	✓	EPSG:4326
buffer_viewparam	sattawat:buffer_viewparam	postgis	✓	✓	EPSG:4326
test	sattawat:test	postgis	✓	✓	EPSG:4326
Rain Today	foss4g-2019:Rain Today	postgis-foss4g	✓	✓	EPSG:4326

**Services:**

- CSW
- WMTS
- WCS
- WFS
- WMS
- WPS

**Settings:**

- Global
- Image Processing
- Raster Access

**Tile Caching:**

- Tile Layers
- Caching Defaults
- Gridsets
- Disk Quota
- BlobStores

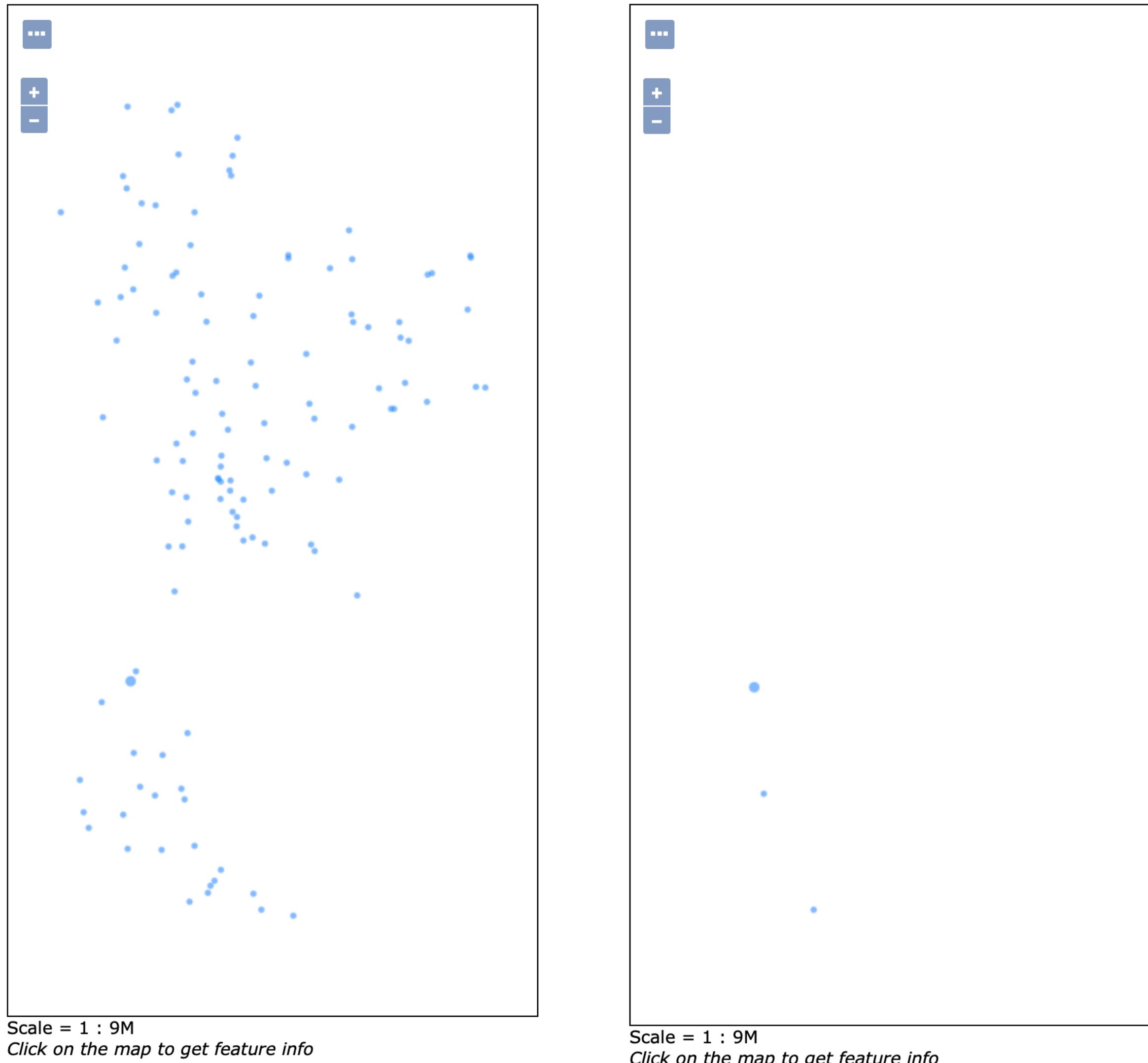
**Security:**

- Settings
- Authentication
- Passwords
- Users, Groups, Roles
- Data
- Services
- WPS security

**Monitor:**

- Activity

# CQL (Common Query Language)

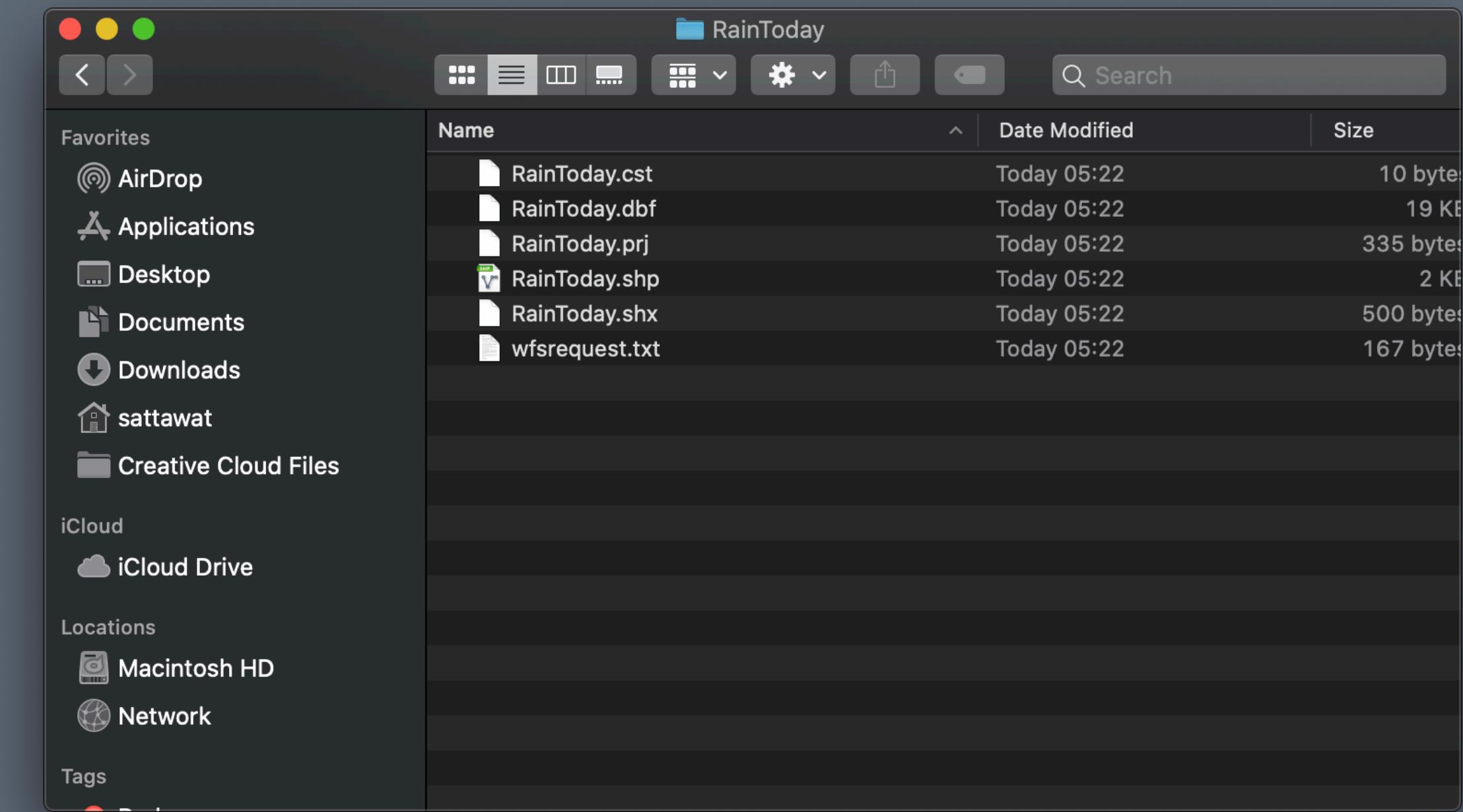
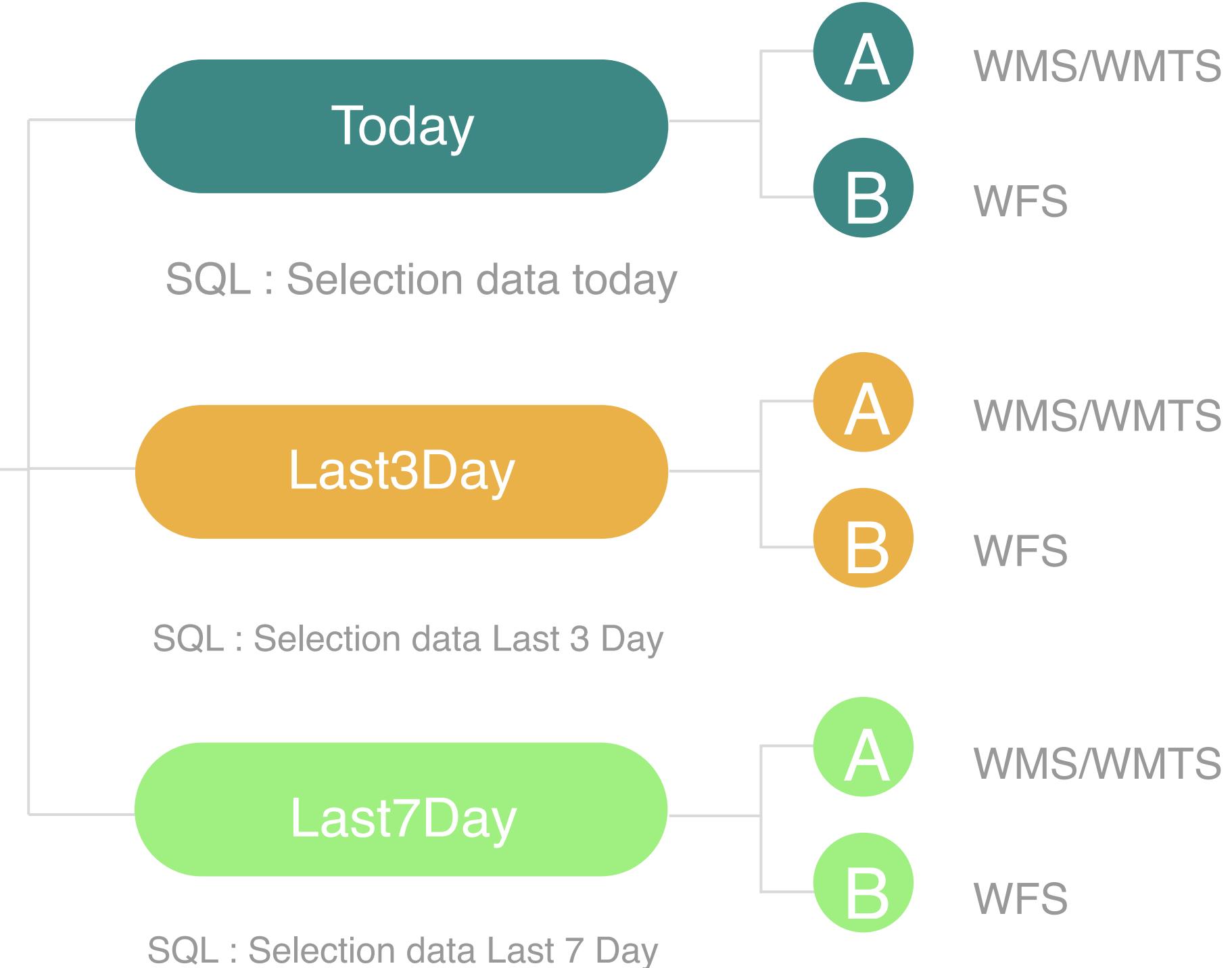
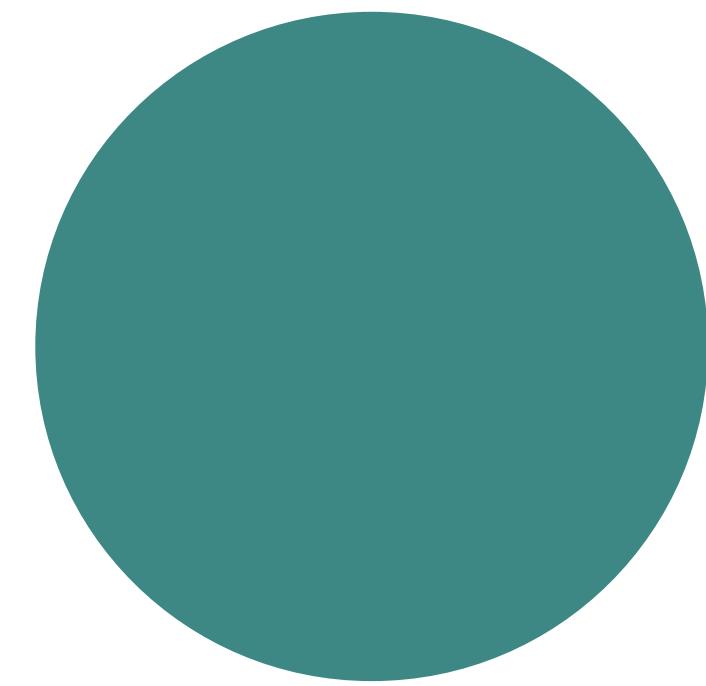


CQL ==> Rain > 5 mm

```
http://localhost:8082/geoserver/foss4g-2029/wms?  
service=WMS  
&version=1.1.0  
&request=GetMap  
&layers=foss4g-2029:RainToday  
&bbox=97.93333333,6.41666667,105.0235,19.96138889  
&width=402&height=768  
&srs=EPSG:4326  
&format=application/openlayers  
&CQL_FILTER=rain>5
```

# Data service WFS

• • •



```
http://localhost:8082/geoserver/foss4g-2019/ows?
service=WFS
&version=1.0.0
&request=GetFeature
&format_options=CHARSET=UTF-8
&typeName=foss4g-2029:RainToday
&outputFormat=SHAPE-ZIP
```

# WFS View params

• • •

Create new SQL view

Define a new SQL view and configure its identified and geometry columns

View Name: RainBydate

SQL statement:

```
Select "name", "to_date", "rain", "geom"
from "TMD_join"
WHERE "to_date" = %to_date%
```

SQL view parameters

Guess parameters from SQL   Add new parameter   Remove selected

Name	Default value	Validation regular expression
to_date	'2019-10-30'	^[w\d\s]+\$

Escape special SQL characters

Attributes

Refresh    Guess geometry type and srid

Name	Type	SRID	Identifier
name	String		<input type="checkbox"/>
to_date	Date		<input type="checkbox"/>
rain	Double		<input type="checkbox"/>
geom	Geometry	4326	<input type="checkbox"/>

**Save** **Cancel**

## WFS Provide by date

[http://localhost:8082/geoserver/foss4g-2019/ows?  
service=WFS  
&version=1.0.0  
&request=GetFeature  
&format\\_options=CHARSET=UTF-8  
&typeName=foss4g-2019:RainBydate  
&outputFormat=SHAPE-ZIP  
&viewparams=date:2019-10-30](http://localhost:8082/geoserver/foss4g-2019/ows?service=WFS&version=1.0.0&request=GetFeature&format_options=CHARSET=UTF-8&typeName=foss4g-2019:RainBydate&outputFormat=SHAPE-ZIP&viewparams=date:2019-10-30)

# WFS View params

• • •

## Create new SQL view

Define a new SQL view and configure its identified and geometry columns

View Name  
RainBydate-date

SQL statement

```
Select "name","to_date","rain","geom" from
"TM_D_join"
WHERE "to_date" = '%start_date%' or "to_date"
<= %end_date%'
```

SQL view parameters

Guess parameters from SQL Add new parameter Remove selected

Name	Default value	Validation regular expression
<input type="checkbox"/> end_date	2019-10-30	^[w\d\\$]+\$
<input type="checkbox"/> start_date	2019-10-31	^[w\d\\$]+\$

Escape special SQL characters

Attributes

Refresh  Guess geometry type and srid

Name	Type	SRID	Identifier
name	String		<input type="checkbox"/>
to_date	Date		<input type="checkbox"/>
rain	Double		<input type="checkbox"/>
geom	Geometry	4326	<input type="checkbox"/>

**Save** **Cancel**

## WFS Provide by Start date to End date

[http://localhost:8082/geoserver/foss4g-2019/ows?  
service=WFS  
&version=1.0.0  
&request=GetFeature  
&format\\_options=CHARSET=UTF-8  
&typeName=foss4g-2019:RainBydate  
&outputFormat=SHAPE-ZIP  
&viewparams=start\\_date:2019-10-30;  
end\\_date:2019-11-01](http://localhost:8082/geoserver/foss4g-2019/ows?service=WFS&version=1.0.0&request=GetFeature&format_options=CHARSET=UTF-8&typeName=foss4g-2019:RainBydate&outputFormat=SHAPE-ZIP&viewparams=start_date:2019-10-30;end_date:2019-11-01)

# Processing Buffer Using PostGIS

• • •

The screenshot shows the pgAdmin 4 interface. On the left is the 'Browser' tree, which includes 'Extensions', 'Foreign Data Wrappers', 'Languages' (with 'public' selected), 'Schemas (2)', 'Tables (4)' (including 'Centroids', 'TMD-today', 'TMDStation', 'spatial\_ref\_sys', 'Trigger Functions', 'Types', 'Views (6)' containing 'TMD\_data', 'TMD\_join', 'geography\_columns', 'geometry\_columns', 'raster\_columns', and 'raster\_overviews'), and 'topology'. Other schemas listed are 'postgis', 'postgres', 'sattawat', 'sattawat2', and 'test'. The main area shows a SQL query window with the following code:

```

1 SELECT "WmoNumber", "rain",
2 ST_Buffer("TMD_join".geom,0.05::double precision) AS the_geom
3 FROM public."TMD_join"
4 where "rain" > 10
5
6
7

```

Below the query is a 'Create - View' dialog box. It has tabs for 'General', 'Definition', 'Security', and 'SQL'. The 'Definition' tab is active, showing the same SQL code. The 'Data Output' tab displays a table with 20 rows of data, each containing 'WmoNumber' (character varying), 'rain' (double precision), and 'the\_geom' (geometry). The 'Definition' tab also contains the same SQL code.

สถานที่ที่อยู่ในรัศมีที่มีปริมาณน้ำฝนมากกว่า 10 มม. ใช้การอ้างอิงจากสถานีตรวจวัด โดยใช้ความสามารถของ PostGIS

## Using function

1. ST\_Buffer for create buffer
2. ST\_Contains for create point in polygon

# Processing Buffer Using PostGIS

• • •

The screenshot shows the pgAdmin 4 interface. On the left is the 'Browser' pane, which lists various database objects like extensions, schemas, tables, and views. In the center, a 'Query' window displays a SQL script for creating a view:

```

1 SELECT "WmoNumber", "rain",
2 ST_Buffer("TMD_join".geom,0.05::double precision) AS the_geom
3 FROM public."TMD_join"
4 where "rain" > 10
5
6
7

```

A 'Create - View' dialog is open over the query window. It contains tabs for 'General', 'Definition', 'Security', and 'SQL'. The 'Definition' tab shows the same SQL script. Below it is a table with columns 'WmoNumber', 'rain', and 'the\_geom'. The table has 20 rows of data. At the bottom of the dialog are buttons for 'Save', 'Cancel', and 'Reset'.

**SELECT**

**"Centroids"."amp\_code",  
"Centroids"."amp\_nam\_t",  
"Centroids"."geom"  
from "Centroids","rain\_10"**

**where**

**ST\_Contains("rain\_10".the\_geom  
, "Centroids".geom)**

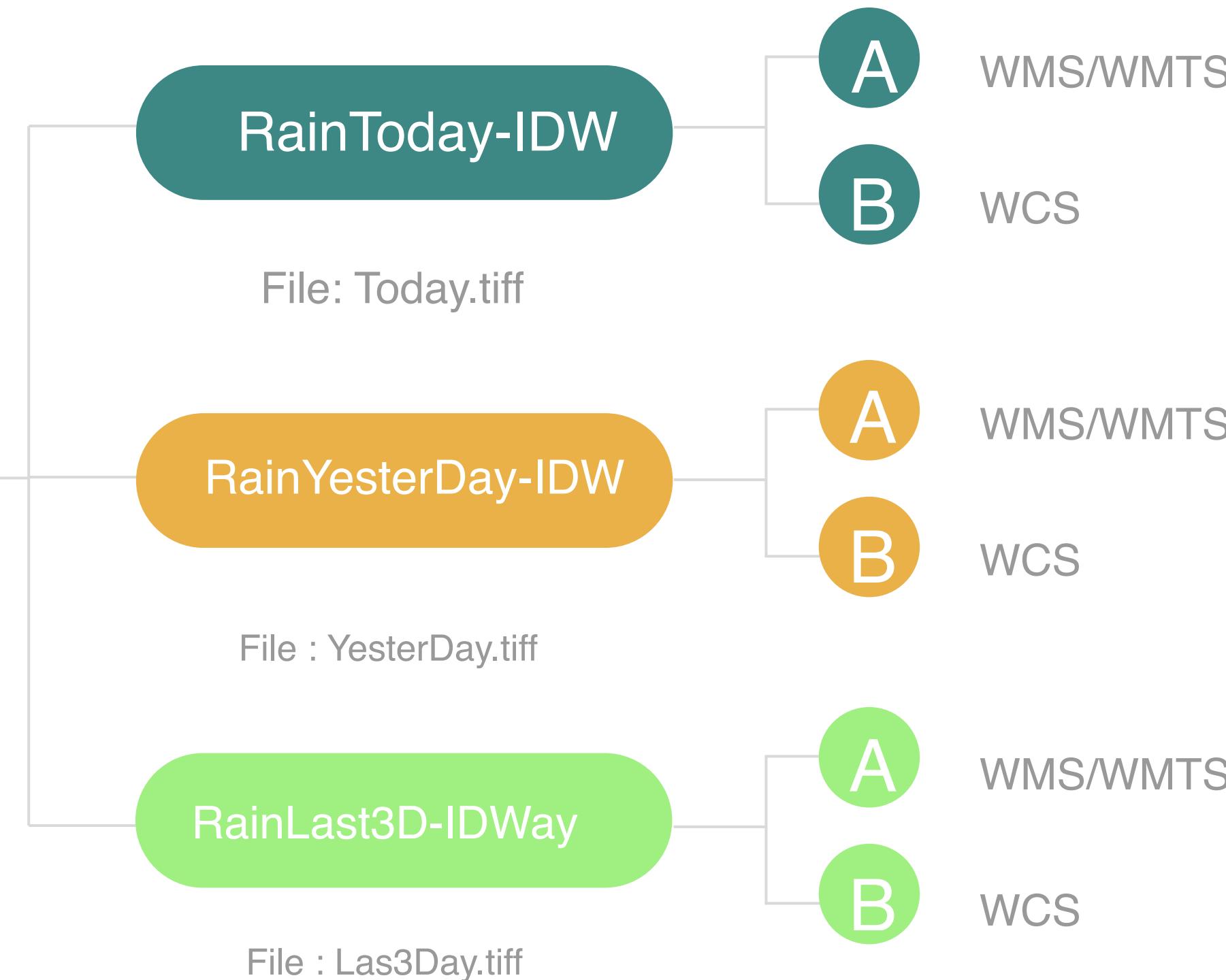
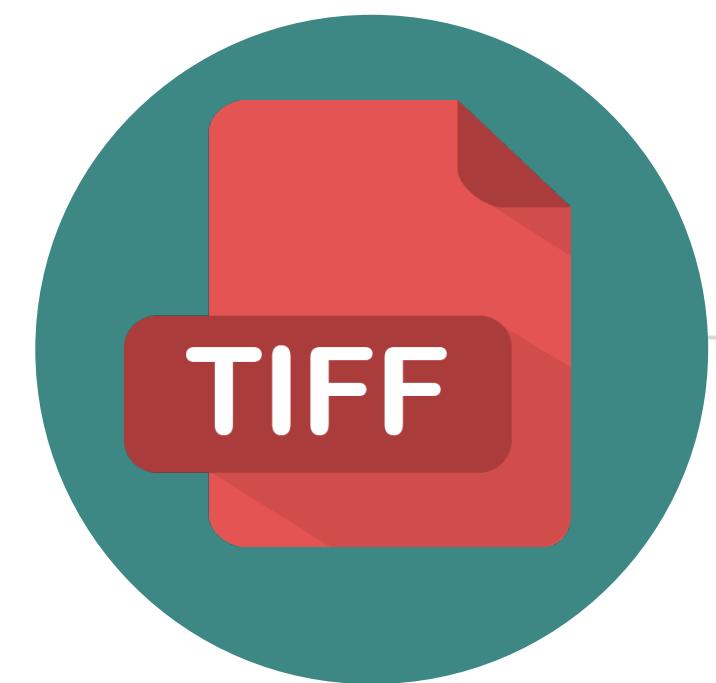


i-bitz company limited  
THE GEOMATICS COMPANY



# Create Raster Layer

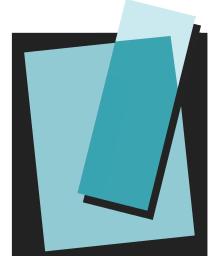
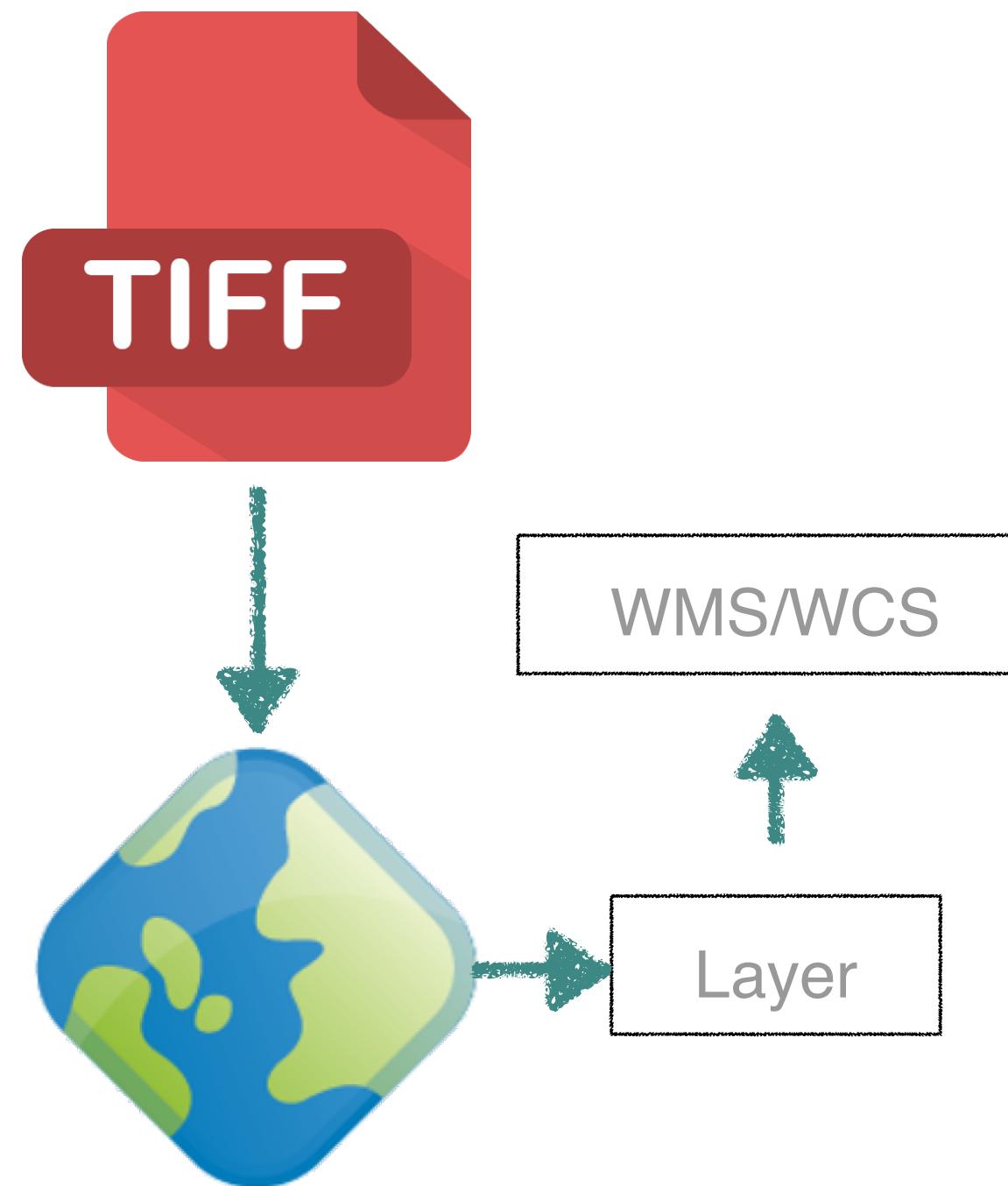
...



1. RainToday-IDW
2. RainResterday-IDW
3. RainLast3day-IDW

# Data service WCS

•••



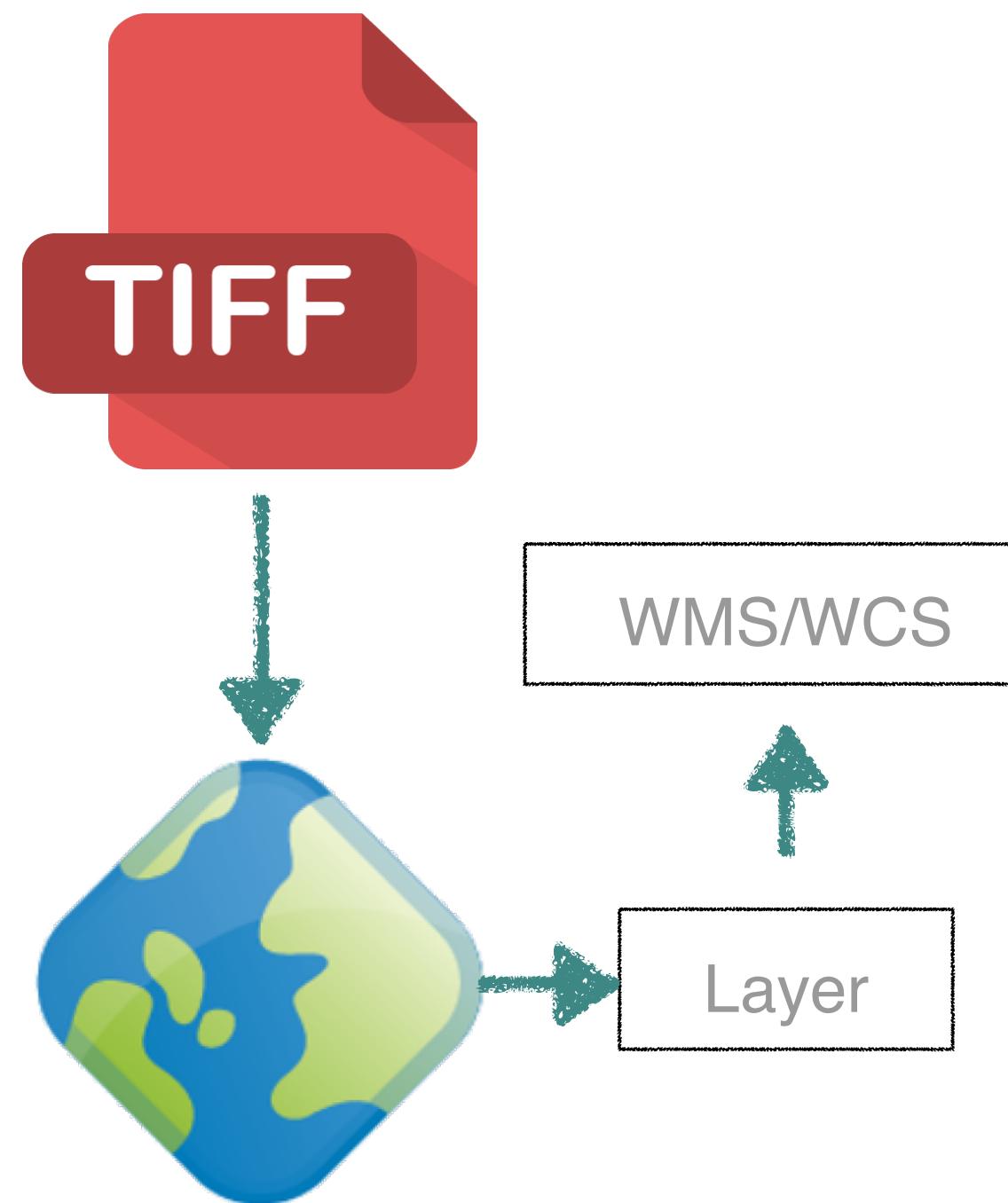
## Request : GetCapabilities

[http://localhost:8082/geoserver/ows?  
service=wcs&version=1.0.0&request=GetCapabilities](http://localhost:8082/geoserver/ows?service=wcs&version=1.0.0&request=GetCapabilities)

```
<wcs:ContentMetadata>
  <wcs:CoverageOfferingBrief>
    <wcs:description>Generated from GeoTIFF</wcs:description>
    <wcs:name>Phuket-Vichit:DEM</wcs:name>
    <wcs:label>DEM</wcs:label>
    <wcs:lonLatEnvelope srsName="urn:ogc:def:crs:OGC:1.3:CRS84">
      <gml:pos>98.22959174894233 7.450577754629645</gml:pos>
      <gml:pos>98.51158610417316 8.228418977091476</gml:pos>
    </wcs:lonLatEnvelope>
    <wcs:keywords>
      <wcs:keyword>TH83</wcs:keyword>
      <wcs:keyword>WCS</wcs:keyword>
      <wcs:keyword>GeoTIFF</wcs:keyword>
    </wcs:keywords>
  </wcs:CoverageOfferingBrief>
  <wcs:CoverageOfferingBrief>
    <wcs:description>Generated from GeoTIFF</wcs:description>
    <wcs:name>foss4g-2029:RainToday-IDW</wcs:name>
    <wcs:label>RainToday-IDW</wcs:label>
    <wcs:lonLatEnvelope srsName="urn:ogc:def:crs:OGC:1.3:CRS84">
      <gml:pos>97.2937 5.562737800000001</gml:pos>
      <gml:pos>105.686780115 20.514644399</gml:pos>
    </wcs:lonLatEnvelope>
    <wcs:keywords>
      <wcs:keyword>2019-10-31</wcs:keyword>
      <wcs:keyword>WCS</wcs:keyword>
      <wcs:keyword>GeoTIFF</wcs:keyword>
    </wcs:keywords>
  </wcs:CoverageOfferingBrief>
</wcs:ContentMetadata>
```

# Data service WCS

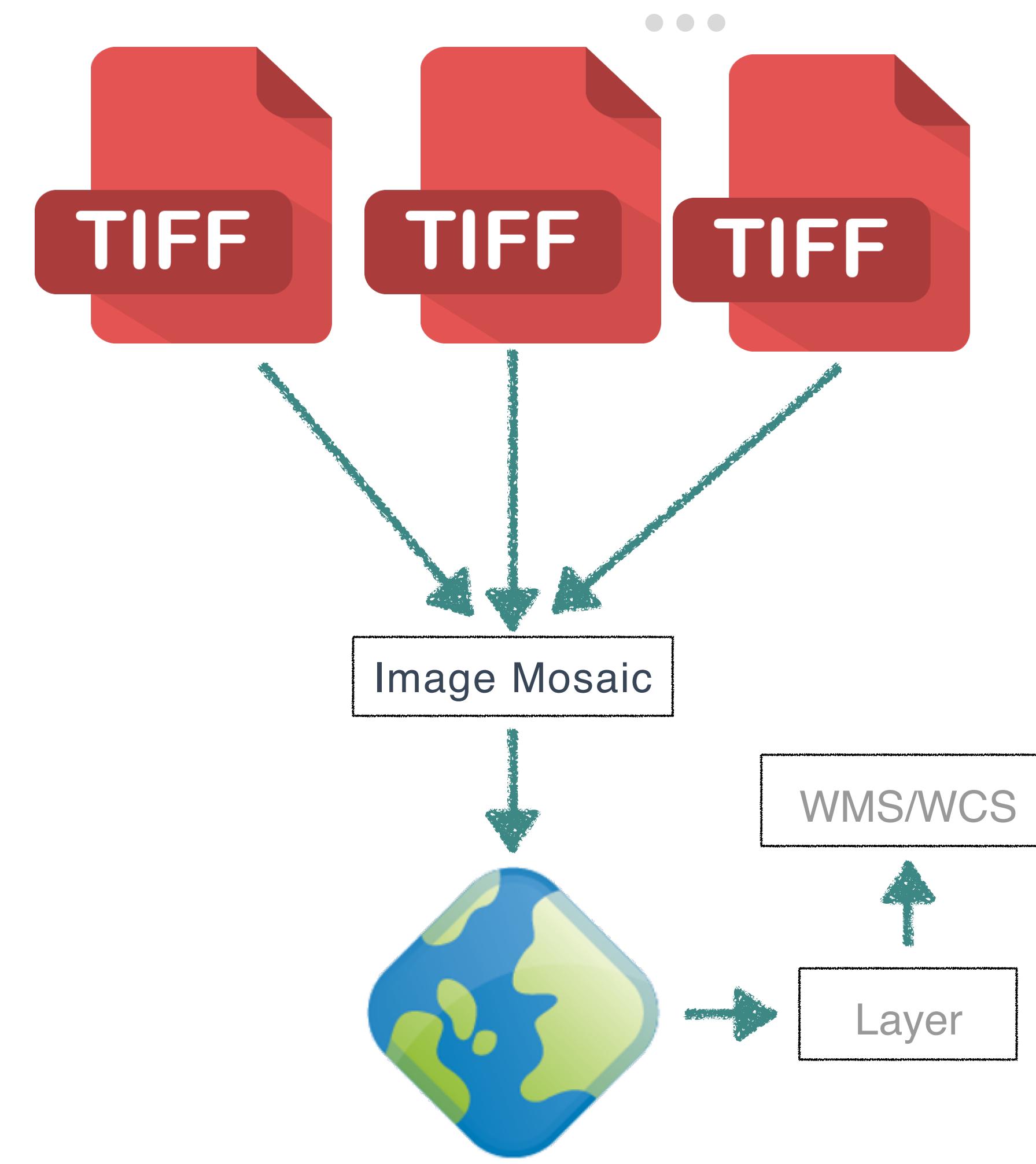
...



## Request : GetCoverage

```
http://localhost:8082/geoserver/ows?service=wcs  
&version=1.0.0  
&request=getcoverage  
&format=geotiff  
&bbox=97,5,106,21  
&width=1000&height=1800  
&crs=epsg:4326  
&interpolation=nearest:neighbor  
&coverage=foss4g-2019:RainToday-IDW
```

# ImageMosaic



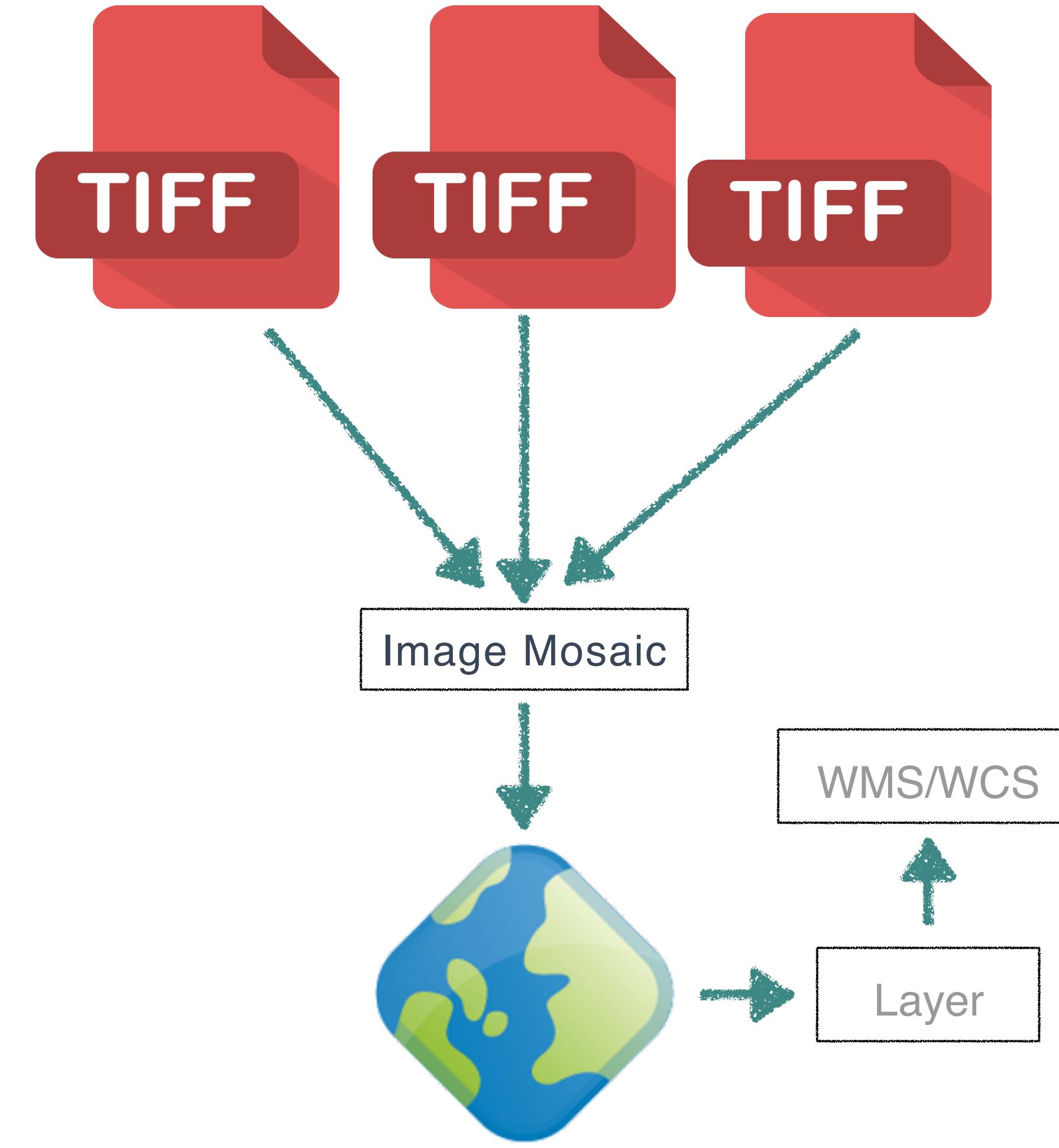
## Create file

Name	Date Modified	Size
im_2019-10-29.tif.aux.xml	Today 17:03	324 bytes
im_20191029.tif	Today 17:00	103 KB
im_20191029.tif.aux.xml	Today 17:03	324 bytes
im_20191030.tif	Today 17:00	103 KB
im_20191030.tif.aux.xml	Today 17:03	324 bytes
im_20191031.tif	Today 12:52	103 KB
im_20191031.tif.aux.xml	Today 17:03	325 bytes
indexer.properties	Today 16:44	160 bytes
timeregex.properties	Today 16:43	14 bytes

- indexer.properties
- timeregex.properties

# ImageMosaic

...

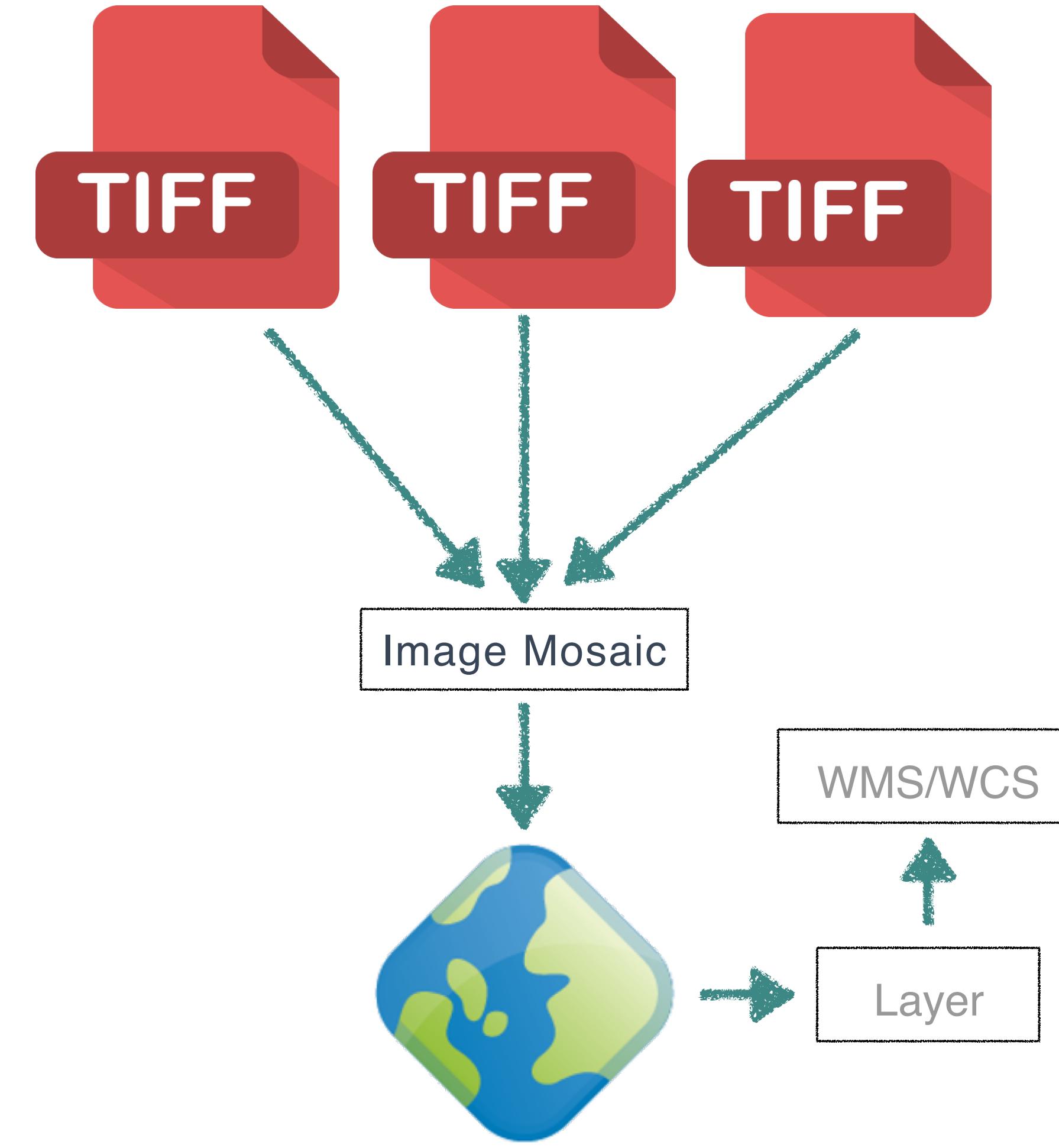


## WMS Provide by date

```
http://localhost:8082/geoserver/foss4g-2019/wms?service=WMS  
&version=1.1.0  
&request=GetMap  
&layers=foss4g-2019:ImageMosaic  
&bbox=97.2937,5.562737800000001,105.686780115,20.514644399  
&width=431&height=768  
&srs=EPSG%3A4326  
&format=application/openlayers  
&TIME=2019-10-30
```

# ImageMosaic

...

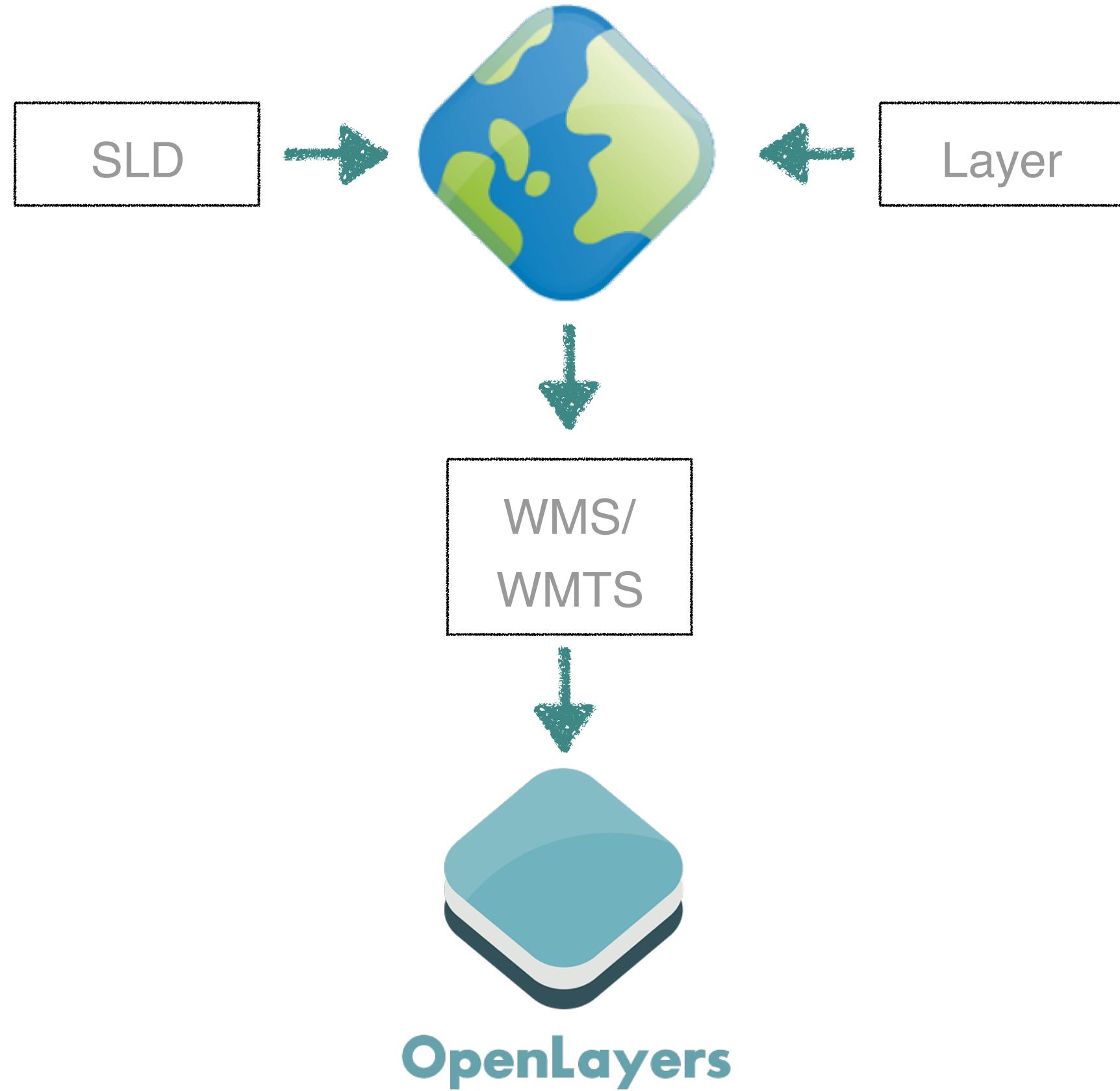


## WCS Provide by date

`http://localhost:8082/geoserver/ows?service=wcs  
&version=1.0.0  
&request=getcoverage  
&format=geotiff  
&bbox=97,5,106,21  
&width=1000&height=1800  
&crs=epsg:4326  
&interpolation=nearest:neighbor  
&coverage=foss4g-2019:ImageMosaic  
&TIME=2019-10-30`

# OpenLayers

• • •



```

view: new ol.View({
  center: ol.proj.fromLonLat([100, 13]),
  zoom: 6
});

var btnShowWms = document.getElementById('showWms');
var btnHideWms = document.getElementById('hideWms');
var showCountry = document.getElementById('showCountry');

btnShowWms.onclick = function() {
  removeExistingWms();
  var wmsSourceProvince = new ol.source.ImageWMS({
    url: 'http://localhost/geoserver/foss4g-2019/wms?',
    params: {
      'LAYERS': 'foss4g-2019:RainToday'
    },
  });

  wmsLayerProvince = new ol.layer.Image({source: wmsSourceProvince});
  map.addLayer(wmsLayerProvince);

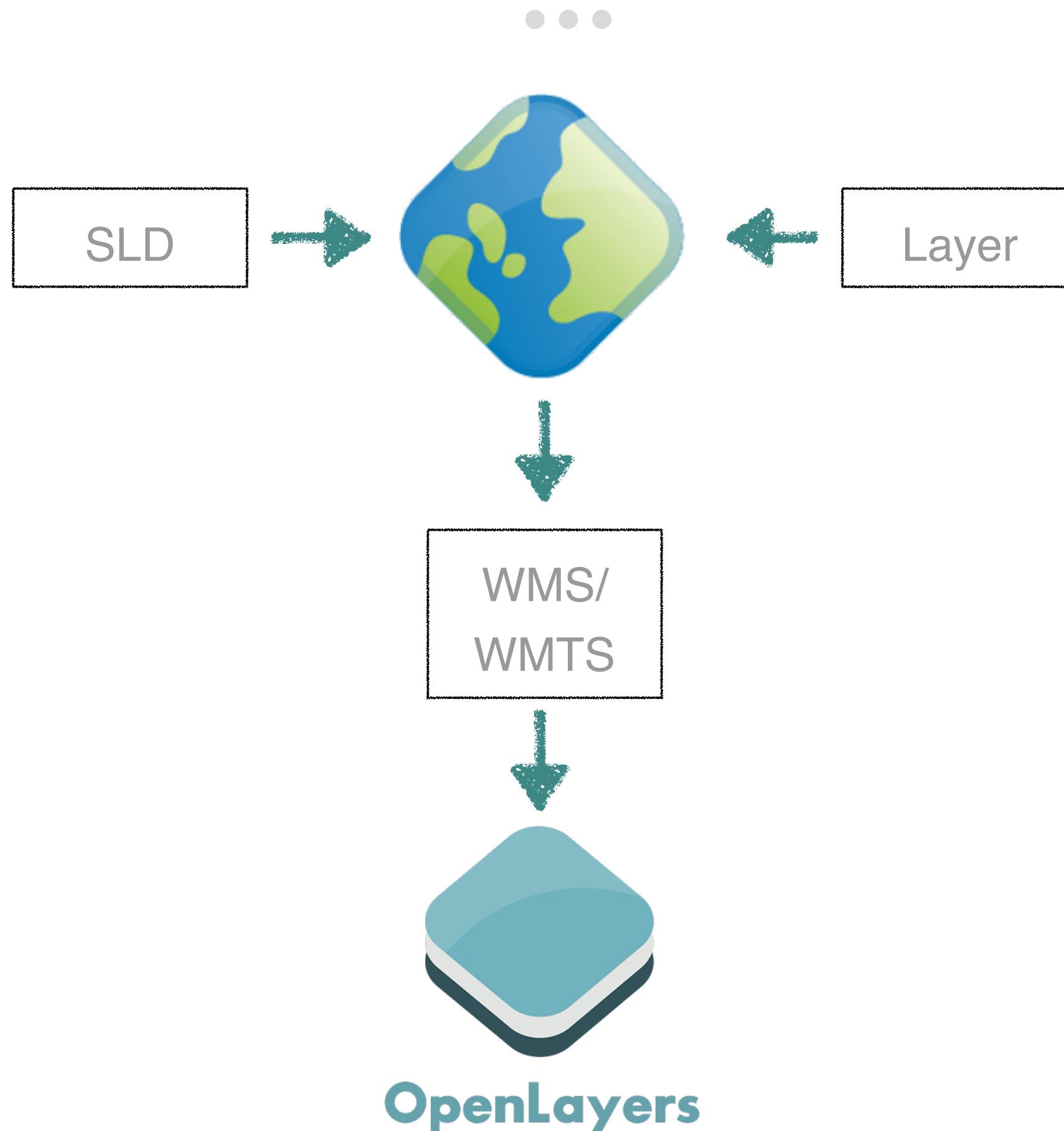
  map.getView().fit(ol.proj.transformExtent([98.464198, 8.182077, 98.250123, 7.723311], 'EPSG:4326')
}

btnHideWms.onclick = function() {
  removeExistingWms();
}

showCountry.onclick = function() {
  map.setView(
    new ol.View({
      center: ol.proj.fromLonLat([100.4833, 13.7500]),
      zoom: 6
    });
}

function removeExistingWms() {
  if (typeof wmsLayerProvince === 'object') {
    map.removeLayer(wmsLayerProvince);
  }
}
  
```

# OpenLayers Call Layer WMS

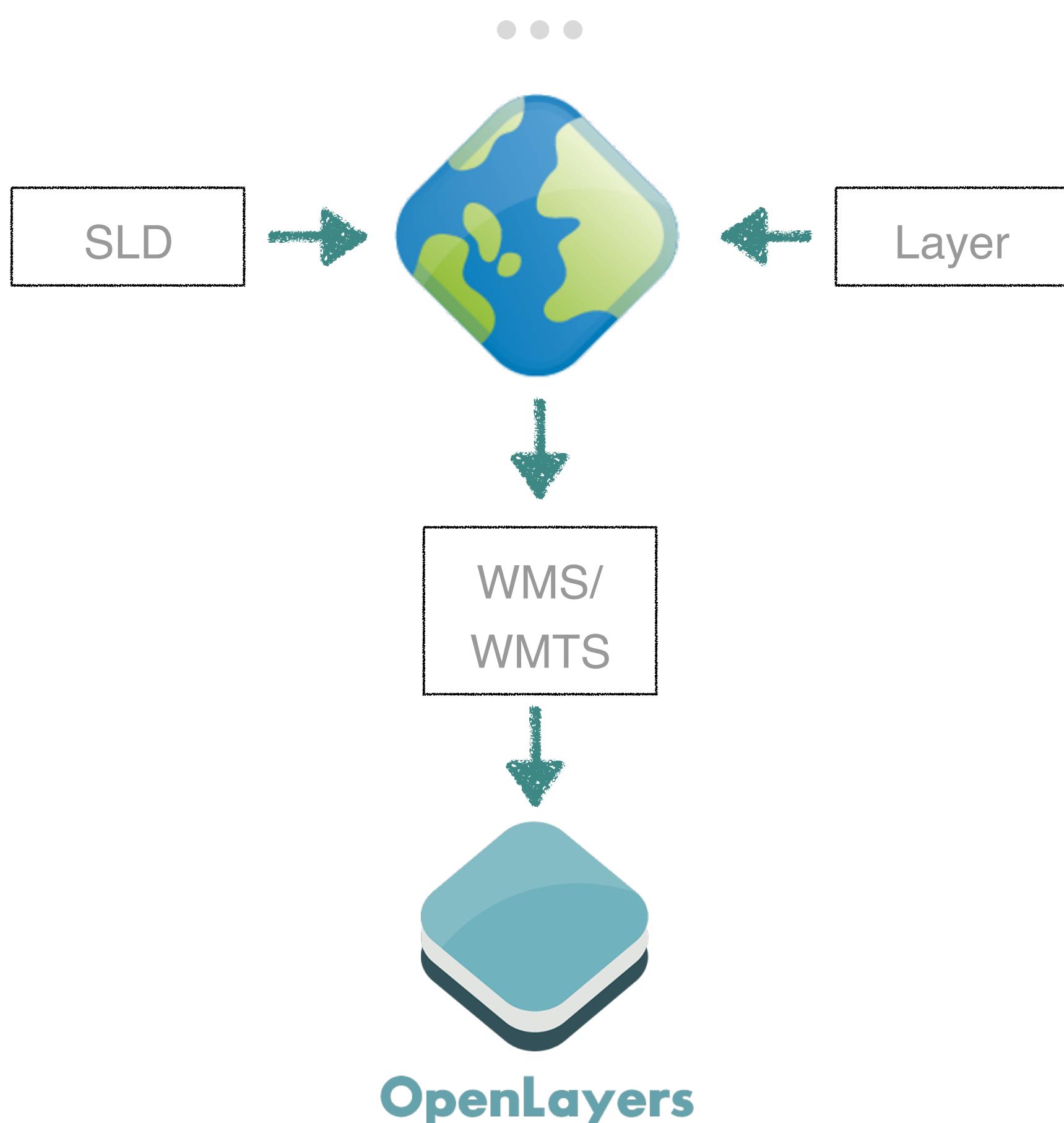


```

<!doctype html>
<html lang="en">
  <head>
    <link rel="stylesheet" href="https://openlayers.org/en/v4.6.5/css/ol.css" type="text/css"/>
    <script src="https://openlayers.org/en/v4.6.5/build/ol.js" type="text/javascript"></script>
    <title>Call WMS</title>
  </head>
  <body>
    <div id="map" class="map"></div>
    <script type="text/javascript">
      var wmsLayerProvince;
      var map = new ol.Map({
        target: 'map',
        layers: [
          new ol.layer.Tile({
            source: new ol.source.OSM()
          })
        ],
        view: new ol.View({
          center: ol.proj.fromLonLat([100, 13]),
          zoom: 6
        })
      });
      var wmsSourceProvince = new ol.source.ImageWMS({
        url: 'http://localhost/geoserver/foss4g-2019/wms?',
        params: {
          'LAYERS': 'RainToday'
        },
        crossOrigin: 'anonymous'
      });

      wmsLayerProvince = new ol.layer.Image({source: wmsSourceProvince});
      map.addLayer(wmsLayerProvince);
    </script>
  </body>
</html>
  
```

# OpenLayers Call Layer WMTS



```

matrixIds[z] = "EPSG:4326:" + z;;
}

resolutions = [
  0.703125, 0.3515625, 0.17578125, 0.087890625,
  0.0439453125, 0.02197265625, 0.010986328125,
  0.0054931640625, 0.00274658203125, 0.001373291015625,
  6.866455078125E-4, 3.4332275390625E-4, 1.71661376953125E-4,
  8.58306884765625E-5, 4.291534423828125E-5, 2.1457672119140625E-5,
  1.0728836059570312E-5, 5.364418029785156E-6, 2.682209014892578E-6,
  1.341104507446289E-6, 6.705522537231445E-7, 3.3527612686157227E-7
];

var map = new ol.Map({
  layers: [
    new ol.layer.Tile({
      source: new ol.source.OSM(),
      opacity: 0.7
    }),
    new ol.layer.Tile({
      opacity: 0.9,
      source: new ol.source.WMTS({
        url: 'http://localhost:8082/geoserver/gwc/service/wmts?',
        layer: 'foss4g-2019:RainToday',
        matrixSet: 'EPSG:4326',
        format: 'image/png',
        projection: projection,
        tileGrid: new ol.tilegrid.WMTS({
          origin: ol.extent.getTopLeft(projectionExtent),
          resolutions: resolutions,
          matrixIds: matrixIds
        }),
        style: '',
        wrapX: true
      })
    })
  ]
});

```

# OpenLayers call to Application

```

view: new ol.View({
  center: ol.proj.fromLonLat([100, 13]),
  zoom: 6
});

var btnShowWms = document.getElementById('showWms');
var btnHideWms = document.getElementById('hideWms');
var showCountry = document.getElementById('showCountry');

btnShowWms.onclick = function() {
  removeExistingWms();
  var wmsSourceProvince = new ol.source.ImageWMS({
    url: 'http://localhost/geoserver/foss4g-2019/wms?',
    params: {
      'LAYERS': 'foss4g-2019:RainToday'
    },
  });

  wmsLayerProvince = new ol.layer.Image({source: wmsSourceProvince});
  map.addLayer(wmsLayerProvince);

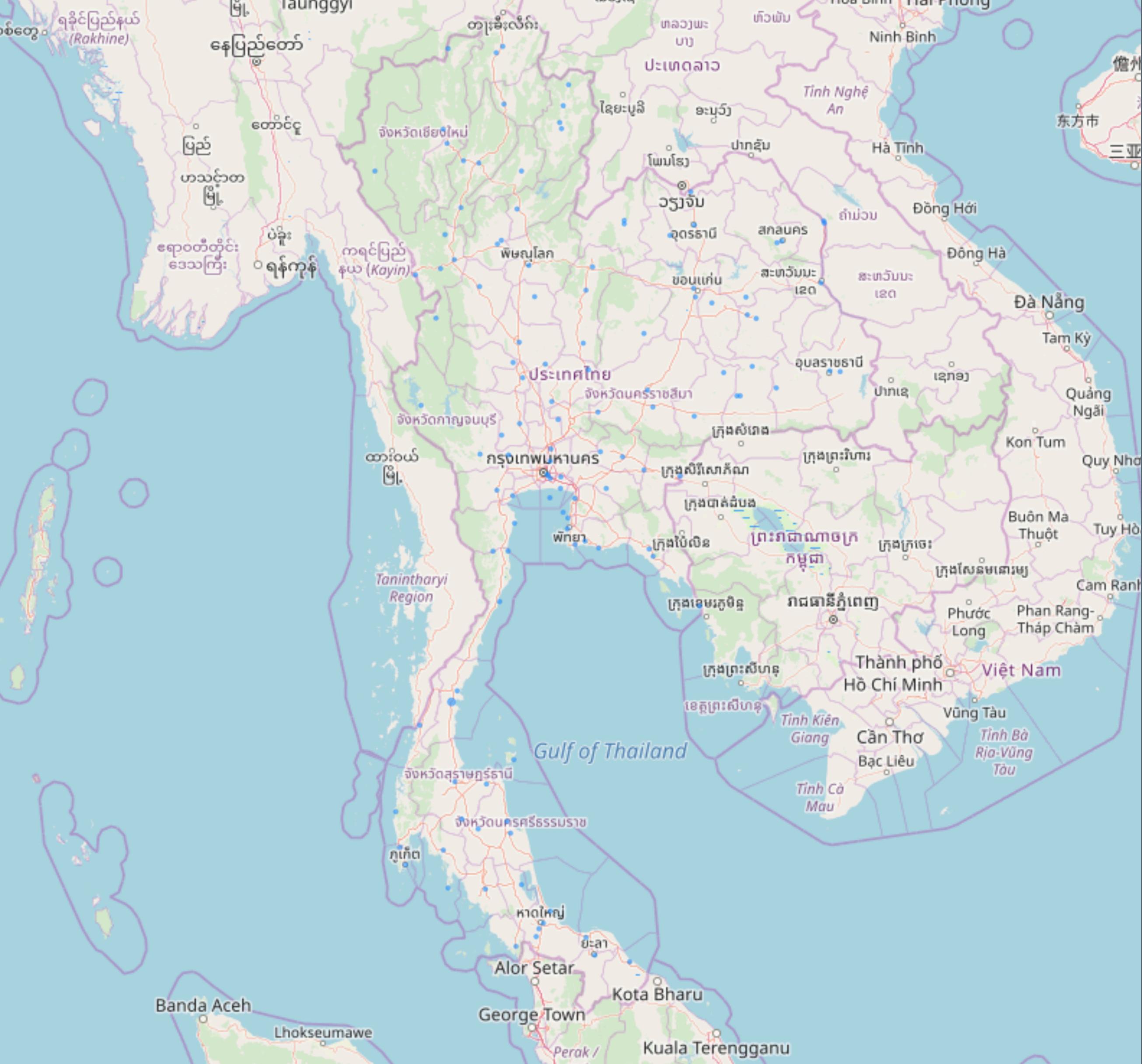
  map.getView().fit(ol.proj.transformExtent([98.464198, 8.182077, 98.250123, 7.723311], 'EPSG:4326')
}

btnHideWms.onclick = function() {
  removeExistingWms();
}

showCountry.onclick = function() {
  map.setView(
    new ol.View({
      center: ol.proj.fromLonLat([100.4833, 13.7500]),
      zoom: 6
    })
  );
}

function removeExistingWms() {
  if (typeof wmsLayerProvince === 'object') {
    map.removeLayer(wmsLayerProvince);
  }
}

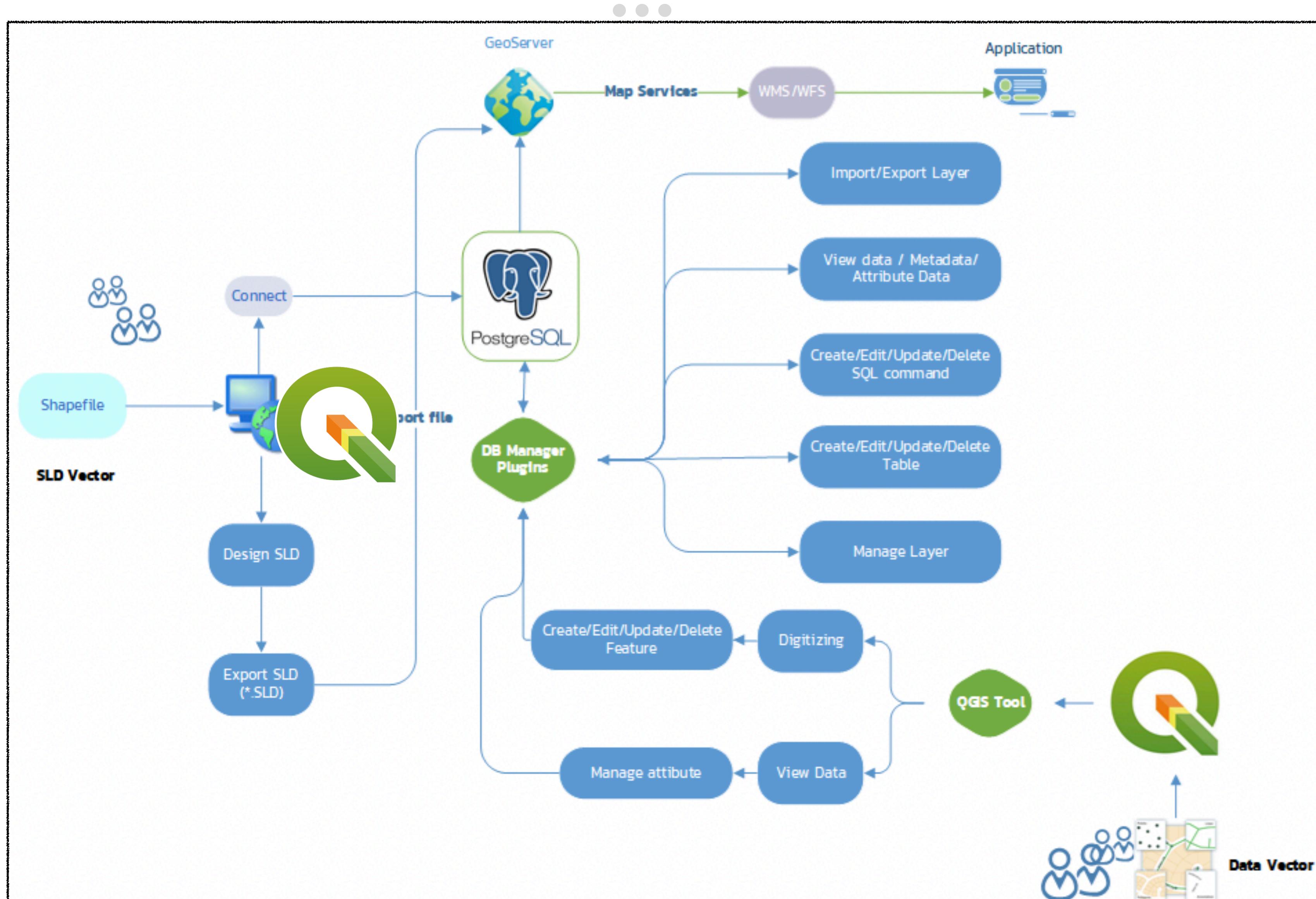
```



i-bitz company limited  
THE GEOMATICS COMPANY



# Concept data service using GeoServer



# Data service



Andaman - The City Dashboard

Display on Application



i-bitz company limited  
THE GEOMATICS COMPANY

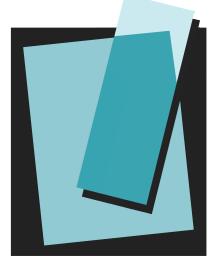


Basic to Advance GeoServer

# GeoServer User Manual

• • •

<https://docs.geoserver.org/stable/en/user/index.html>



i-bitz company limited  
THE GEOMATICS COMPANY



# Interest full course register at ...

● ● ●

The screenshot shows a grid of course offerings:

Course Type	Course Name	Level	Price (excluding vat)	Duration	Topics
Geoserver Basic	QGIS Basic	Basic	7,000 ₧	(9.00 AM - 4.00 PM)	GIS data standards, Shapefile Management, Tables Management, Create Shapefile, Georeferencing, Data management, Geoprocessing Tools, Terrain Analysis, Map Design
	QGIS (Customized Course)	Intermediate	7,500 ₧	(9.00 AM - 4.00 PM)	Shapefile Management, Tables Management, Spatial data analysis, Potential Surface Analysis (PSA), Create report, Create map, Mini project, Customized Course, 20 persons
	Geoserver Basic	Basic	8,500 ₧ (excluding vat)	(9.00 AM - 4.00 PM)	Internet GIS, Brief OGC Standard, Vector Data Management, Raster Data Management, Layers/Layer Group, Styled Layer Descriptor (SLD), Basic Html., Basic Openlayers API, 15 persons
	Docker for GIS Intermediate	Intermediate	13,500 ₧ (excluding vat)	(9.00 AM - 4.00 PM)	Best Practice Installation, Security, Tuning Docker, -
	GIS Web application Intermediate	Intermediate	22,500 ₧ (excluding vat)	( 8.30 AM - 5.30 PM)	HTML5, CSS3, jQuery, Bootstrap, Google Map API, Codeigniter framework (PHP), Create GIS web application, Database with MySQL, Web design Tools, Debugging and Testing, Online and Hosting, 10 persons

<https://gisbuildup.i-bitz.co.th>

