

## Raster to Postgis

--locate postgis bin folder and navigate to the same (run in terminal)

```
cd /opt/PostgreSQL/9.6/bin/
```

-- get your postgres version (run in terminal)

```
postgres -V
```

-- Configure your device with RASTER2PGSQL

raster2pgsql (try by enter this text: if it is configured such, continue with it)

(else try to install raster2pgsql in your system)

-- integration of gdal with postgis

```
SET postgis.gdal_enabled_drivers = 'ENABLE_ALL';
```

-- Create a new\_database and add an extension POSTGIS in your PostgresDB:

(try this by creating a DB and extensions by step-by-step using postgres/pgadmin)

-- Create a new database

```
CREATE DATABASE rast_test --give a name for db eg: rast_test
```

```
WITH OWNER = postgres
```

```
ENCODING = 'UTF8'
```

```
TABLESPACE = pg_default
```

```
LC_COLLATE = 'en_US.UTF-8'
```

```
LC_CTYPE = 'en_US.UTF-8'
```

```
CONNECTION LIMIT = -1;
```

-- Add an extension postgis

```
CREATE EXTENSION postgis
```

```
SCHEMA public
```

```
VERSION "2.4.2";
```

-- raster to postgis Syntax: (to be run in terminal)

```
raster2pgsql -s <srid> -l -C -M <filepath> <table_name> | psql -U <user>
```

```
PGPASSWORD=<P@$$WORD> -d <database> -h <host> -p <port>
```

## Queries to run on Raster

-- get pixel values of raster

```
SELECT ST_DumpValues(rast) FROM <rast> ;
```

-- get pixel values of raster by no of pixel intervals (eg; 2x2)

```
SELECT ST_Envelope(ST_Tile(rast, 2, 2, TRUE)), ST_DumpValues(ST_Tile(rast, 2, 2, TRUE))  
As rast  
FROM <rast> WHERE rid=1 limit 10;
```

-- to show tiled\_raster and its corresponding pixel values as a dump

```
SELECT ST_DumpValues(ST_Tile(rast, 255, 255, TRUE)) As rast FROM <rast> WHERE rid=1  
limit 10;
```

-- Count of distinct raster values with in a given polygon

-- ref: [https://postgis.net/2014/09/26/tip\\_count\\_of\\_pixel\\_values/](https://postgis.net/2014/09/26/tip_count_of_pixel_values/)

```
SELECT (pvc).value, SUM((pvc).count) AS tot_pix,  
FROM <rast>  
INNER JOIN  
(SELECT ST_UNION(the_geom) as the_geom, RANDOM()< 0.01 from grid_1km where  
RANDOM() < 0.01) AS geom  
ON ST_INTERSECTS(<rast>.rast, the_geom),  
ST_ValueCount(ST_Clip(<rast>.rast,the_geom),1) AS pvc  
GROUP BY (pvc).value  
ORDER BY (pvc).value ;
```

-- tiled\_raster:

-- ref: [https://postgis.net/docs/RT\\_ST\\_Tile.html](https://postgis.net/docs/RT_ST_Tile.html)

```
WITH foo AS (  
SELECT ST_AddBand(ST_AddBand(ST_MakeEmptyRaster(3, 3, 0, 0, 1, -1, 0, 0, 0), 1,  
'8BUI', 1, 0), 2, '8BUI', 10, 0) AS rast UNION ALL  
SELECT ST_AddBand(ST_AddBand(ST_MakeEmptyRaster(3, 3, 3, 0, 1, -1, 0, 0, 0), 1,  
'8BUI', 2, 0), 2, '8BUI', 20, 0) AS rast UNION ALL  
SELECT ST_AddBand(ST_AddBand(ST_MakeEmptyRaster(3, 3, 6, 0, 1, -1, 0, 0, 0), 1,  
'8BUI', 3, 0), 2, '8BUI', 30, 0) AS rast UNION ALL  
  
SELECT ST_AddBand(ST_AddBand(ST_MakeEmptyRaster(3, 3, 0, -3, 1, -1, 0, 0, 0), 1,  
'8BUI', 4, 0), 2, '8BUI', 40, 0) AS rast UNION ALL  
SELECT ST_AddBand(ST_AddBand(ST_MakeEmptyRaster(3, 3, 3, -3, 1, -1, 0, 0, 0), 1,  
'8BUI', 5, 0), 2, '8BUI', 50, 0) AS rast UNION ALL  
SELECT ST_AddBand(ST_AddBand(ST_MakeEmptyRaster(3, 3, 6, -3, 1, -1, 0, 0, 0), 1,  
'8BUI', 6, 0), 2, '8BUI', 60, 0) AS rast UNION ALL
```

```
SELECT ST_AddBand(ST_AddBand(ST_MakeEmptyRaster(3, 3, 0, -6, 1, -1, 0, 0, 0), 1,
'8BUI', 7, 0), 2, '8BUI', 70, 0) AS rast UNION ALL
SELECT ST_AddBand(ST_AddBand(ST_MakeEmptyRaster(3, 3, 3, -6, 1, -1, 0, 0, 0), 1,
'8BUI', 8, 0), 2, '8BUI', 80, 0) AS rast UNION ALL
SELECT ST_AddBand(ST_AddBand(ST_MakeEmptyRaster(3, 3, 6, -6, 1, -1, 0, 0, 0), 1,
'8BUI', 9, 0), 2, '8BUI', 90, 0) AS rast
), bar AS (
SELECT ST_Union(rast) AS rast FROM foo
), baz AS (
SELECT ST_Tile(rast, 3, 3, TRUE) AS rast FROM bar
)
SELECT
ST_DumpValues(rast)
FROM baz;
```