

Introduction à PostGIS

Daniel FAIVRE

daniel.faivre@camptocamp.com

<http://www.camptocamp.com>

Introduction - sites web de référence

- PostGIS est un module spatial pour le sgbdr "o" PostgreSQL.
- PostGIS été crée par Refrations Research Inc.
- PostGIS respecte les spécifications de l'Open GIS Consortium pour l'intégration de données géographiques dans des bases de données:
 - "Simple Features Specification for SQL"
 - cf. <http://www.opengis.org/docs/99-049.pdf>
- Sites web de référence:
 - <http://postgis.refrations.net/>
 - <http://postgis.refrations.net/docs/>

PostGis

- PostGIS est une base de données spatiales libre. Ses fonctions la rende comparable aux couples Oracle + Spatial Cartridge ou Oracle – ArcSDE.
- PostGIS permet les requetes spatiales et/ou attributaires complexes, gère les changements de projections, les données 2, 3 ou 4d, les opérations "topologiques" telles que calculs de buffers, d'intersections, d'unions, d'enveloppes, etc.
- PostGIS se connecte à Mapserver, Jump, Grass, ..., et peut émuler un serveur ArcIMS.

PostGIS

- <http://refractions.net/postgis>
- PostGIS ajoute le support d'objets géographiques à la base de données relationnelle (et objet) PostgreSQL.
- PostGIS, comme PostgreSQL, est un logiciel open-source freeware.
- PostGIS respecte les spécifications OGC:
 - POINT(800000 2200000 755)
 - LINESTRING(0 0, 1 1, 1 2)
 - POLYGON((0 0 0,4 0 0,4 4 0,0 4 0,0 0 0),(1 1 0,2 1 0,2 2 0,1 2 0,1 1 0))

PostGIS

- `MULTIPOINT(0 0 0,1 2 1)`
 - `MULTILINESTRING((0 0 0,1 1 0,1 2 1),(2 3 1,3 2 1,5 4 1))`
 - `MULTIPOLYGON(((0 0 0,4 0 0,4 4 0,0 4 0,0 0 0),(1 1 0,2 1 0,2 2 0,1 2 0,1 1 0)),((-1 -1 0,-1 -2 0,-2 -2 0,-2 -1 0,-1 -1 0)))`
 - `GEOMETRYCOLLECTION(POINT(2 3 9),LINESTRING((2 3 4,3 4 5)))`
- Ces formats de description des objets géographiques en base de données sont normalisés par l'OGC: ce sont des formats « WKT » (Well Known Text).

PostGIS: introduire les données

- Pour insérer des données géographiques dans une base PostGIS, il existe deux méthodes:
- Requête SQL:

```
BEGIN;  
INSERT INTO ROADS (ID,GEOM,NAME ) VALUES (1,GeometryFromText('LINESTRING(191232  
243118,191108 243242)',-1),'Jeff Rd');  
INSERT INTO ROADS (ID,GEOM,NAME ) VALUES (2,GeometryFromText('LINESTRING(189141  
244158,189265 244817)',-1),'Geordie Rd');  
INSERT INTO ROADS (ID,GEOM,NAME ) VALUES (3,GeometryFromText('LINESTRING(192783  
228138,192612 229814)',-1),'Paul St');  
INSERT INTO ROADS (ID,GEOM,NAME ) VALUES (4,GeometryFromText('LINESTRING(189412  
252431,189631 259122)',-1),'Graeme Ave');  
INSERT INTO ROADS (ID,GEOM,NAME ) VALUES (5,GeometryFromText('LINESTRING(190131  
224148,190871 228134)',-1),'Phil Tce');  
INSERT INTO ROADS (ID,GEOM,NAME ) VALUES (6,GeometryFromText('LINESTRING(198231  
263418,198213 268322)',-1),'Dave Cres');  
COMMIT;
```

- Shp2pgsql:

```
shp2pgsql shaperoads roadstable roadsdb | psql -d  
roadsdb
```

PostGIS: extraire les données

- Là aussi, il y a deux méthodes:
- Requête SQL:

```
SELECT id, nom, the_geom FROM roads;
```

```
id | geom | nom
---+-----+-----
 1 | LINESTRING(191232 243118,191108 243242) | Jeff Rd
 2 | LINESTRING(189141 244158,189265 244817) | Geordie Rd
```

- Pgsq2shp:

```
pgsq2shp [<options>] <database> <table>
```

- La principale option est le nom du fichier (shapefile) dans lequel on va écrire les données. Les autres options sont détaillées dans la documentation.

PostGIS: indexation spatiale

- Les performances dépendent grandement de la qualité de la structuration des données, de l'indexation, et de l'utilisation de requêtes adéquates.
- PostGIS utilise un mécanisme spécifique d'indexation des données spatiales (GIST) permettant d'optimiser les recherches:

```
CREATE INDEX [indexname] ON [tablename]
```

```
USING GIST ( [geometryfield] GIST_GEOMETRY_OPS );
```

- Après avoir créé un index spatial, il est important de forcer PostgreSQL à recalculer ses statistiques, utilisées pour optimiser les requêtes:

```
VACUUM ANALYZE;
```

PostGIS: requêtes complexes

- Exemple:

```
SELECT the_geom FROM geom_table
```

```
WHERE distance( the_geom, GeometryFromText( 'POINT(100000  
200000)', -1 ) ) < 100
```

- Cette requête ne sera pas très rapide, car elle calcule une distance sur chaque enregistrement de la table.
- On peut optimiser ceci ainsi:

```
SELECT the_geom FROM geom_table
```

```
WHERE the_geom && 'BOX3D(90900 190900, 100100  
200100)::box3d
```

```
AND distance( the_geom, GeometryFromText( 'POINT(100000  
200000)', -1 ) ) < 100
```

- L'opérateur `&&` est « l'opérateur de superposition »: il teste la superposition des bounding boxes.

PostGIS

- Quelques fonctions de PostGIS:
- Mesures de longueurs: `length2d()`, `length3d()`, `length_spheroid()`, `perimeter2d()`, `perimeter3d()`, ...
- Mesures de surface: `area3d()`
- Topologie: `truly_inside(geomA, geomB)`
- Bbox: `extent(geometry)`
- Mesure de distance: `Distance(geomA, geomB)`
- Tests: `IsEmpty()`, `IsClosed()`, `IsSimple()`, ...
- Pour plus d'information, voir la documentation complète (en Anglais) sur le site <http://postgis.refractory.net>

PostGIS: formats de sortie

- AsBinary(geometry)
 - Retourne une géométrie sous forme binaire.
- AsSVG(geometry, [rel], [precision])
 - Retourne une géométrie sous forme de path SVG.
- AsGML(geometry, [precision])
 - Retourne une géométrie en tant qu'élément GML.
- ...

PostGIS + MapServer

- Les couches PostGIS peuvent se lire directement dans Cartoweb (fonctionnalité de Mapserver).

LAYER

CONNECTIONTYPE POSTGIS

CONNECTION « user=www-data dbname=geo_ifn »

DATA « the_geom from tfifn »

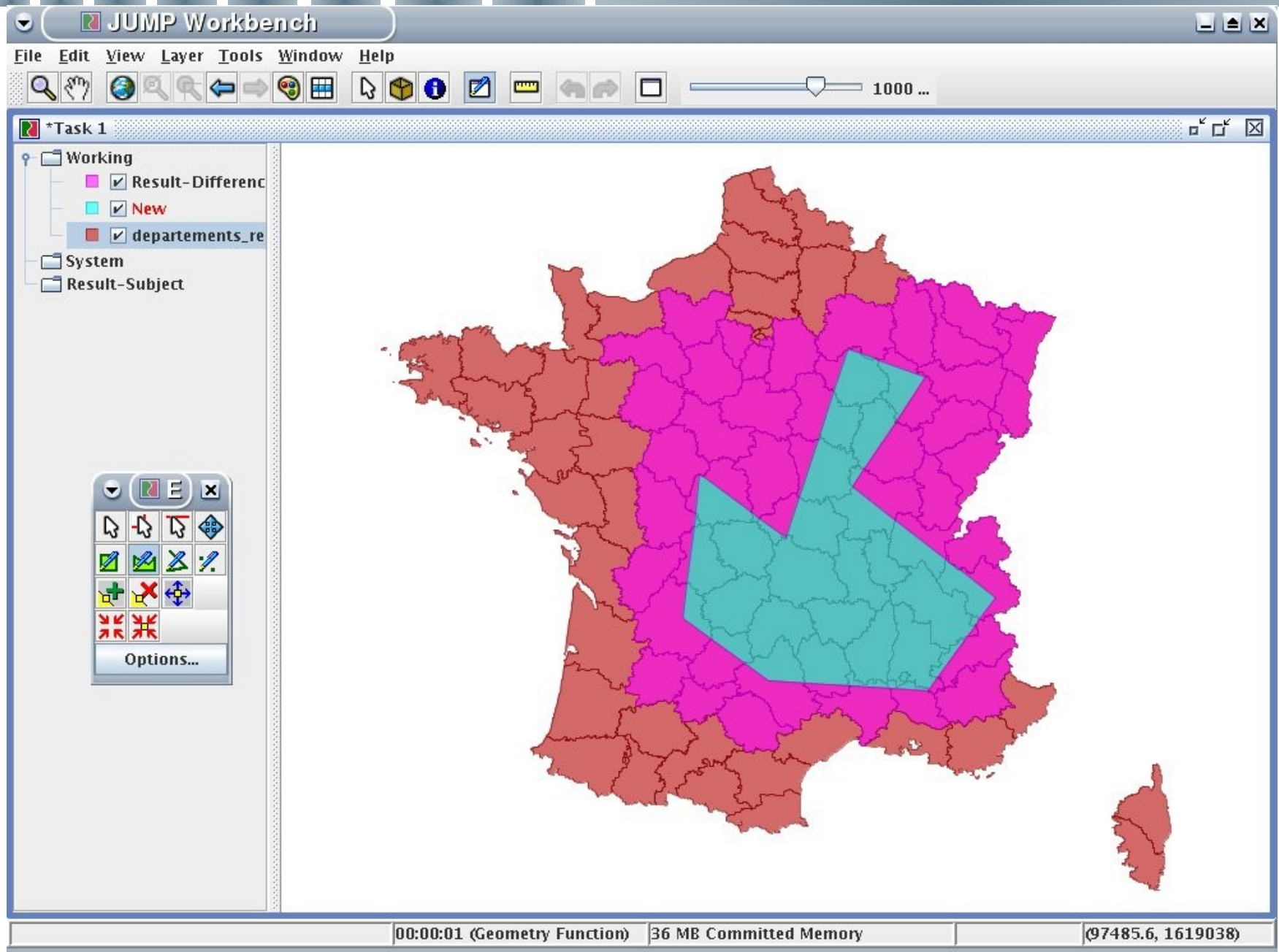
CLASS

[...]

END

END

Jump



PostGIS - Grass

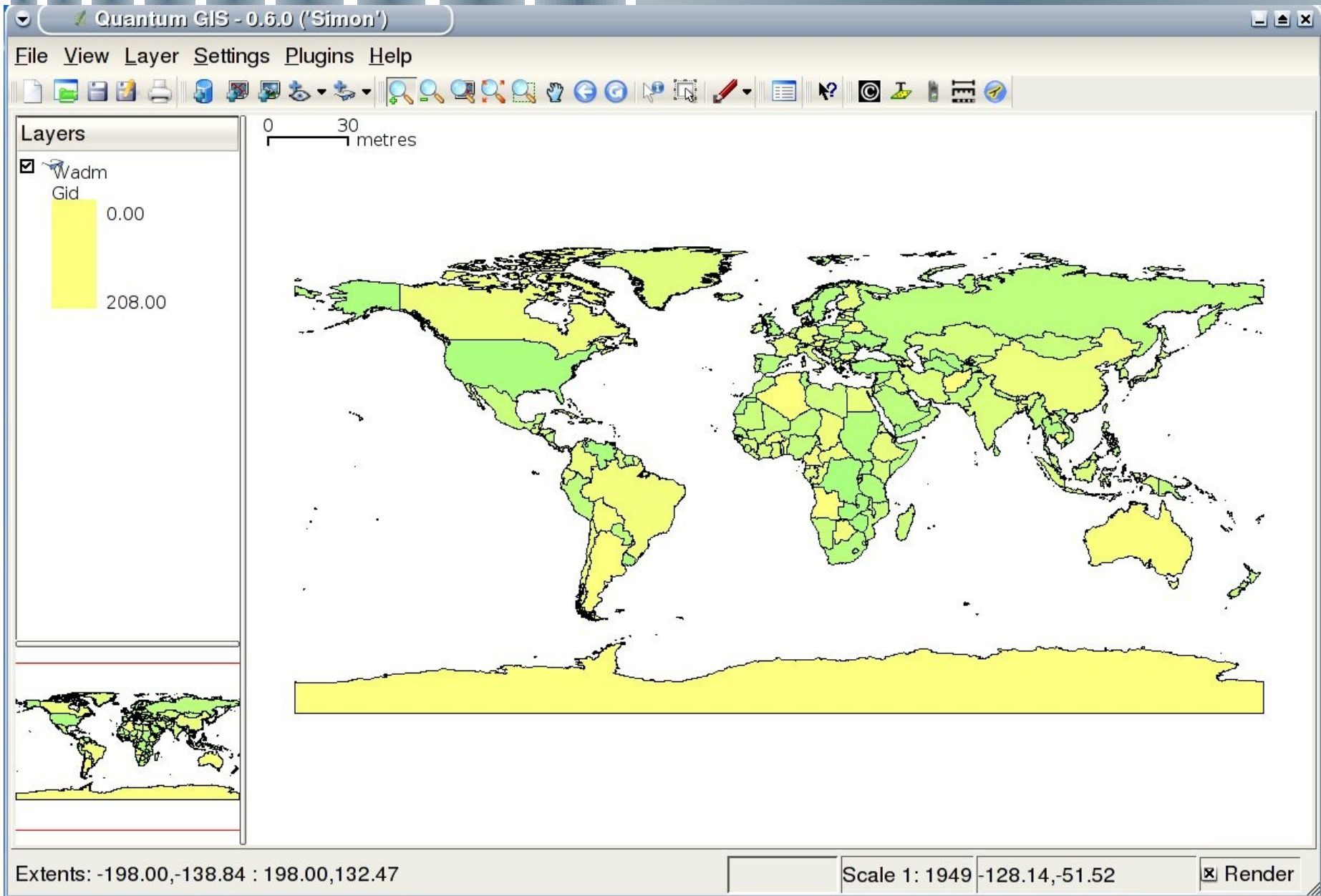
The image shows two windows from the GRASS GIS 6.0.0beta1 interface. The left window is the 'GIS Manager' and the right is the 'Monitor: x0'.

GIS Manager Window:

- Menu: File, GIS, Display, Raster, Vector, Image, Grid3D, Databases, Help
- Toolbar: Includes icons for map, search, zoom, pan, help, layers, vector, raster, image, grid3d, databases, and help.
- Layer list: +/- vector 1
- Vector name: dep_fr
- Display: shape category topology direction label
- Type: point line boundary centroid area face
- Symbol: basic/x Size: 5
- Line color: show lines Fill color: fill areas
- Random fill colors for each cat value GRASSRGB column for fill color
- Label color: Label size: 8 Label xpos: left Label ypos: center
- Layer for labels: 1 Attribute col for labels:
- Layer for query: 1 Category query:
- SQL query: use query
- Query with mouse: results as text in terminal edit attributes (form mode)
- Display when avg. region dimension is more than or less than
- Width (print): 1

Monitor: x0 Window:

- Map display showing the outline of France with internal department boundaries. Each department is marked with a small 'x' symbol.

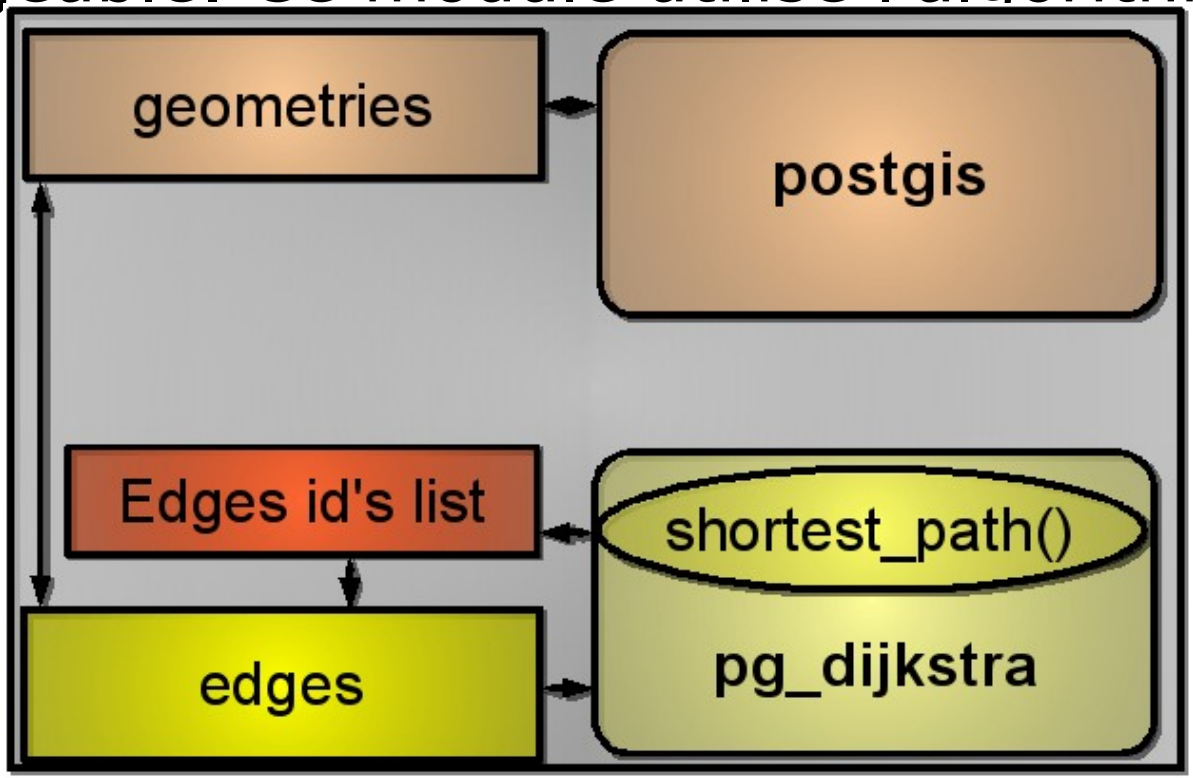


PostGIS

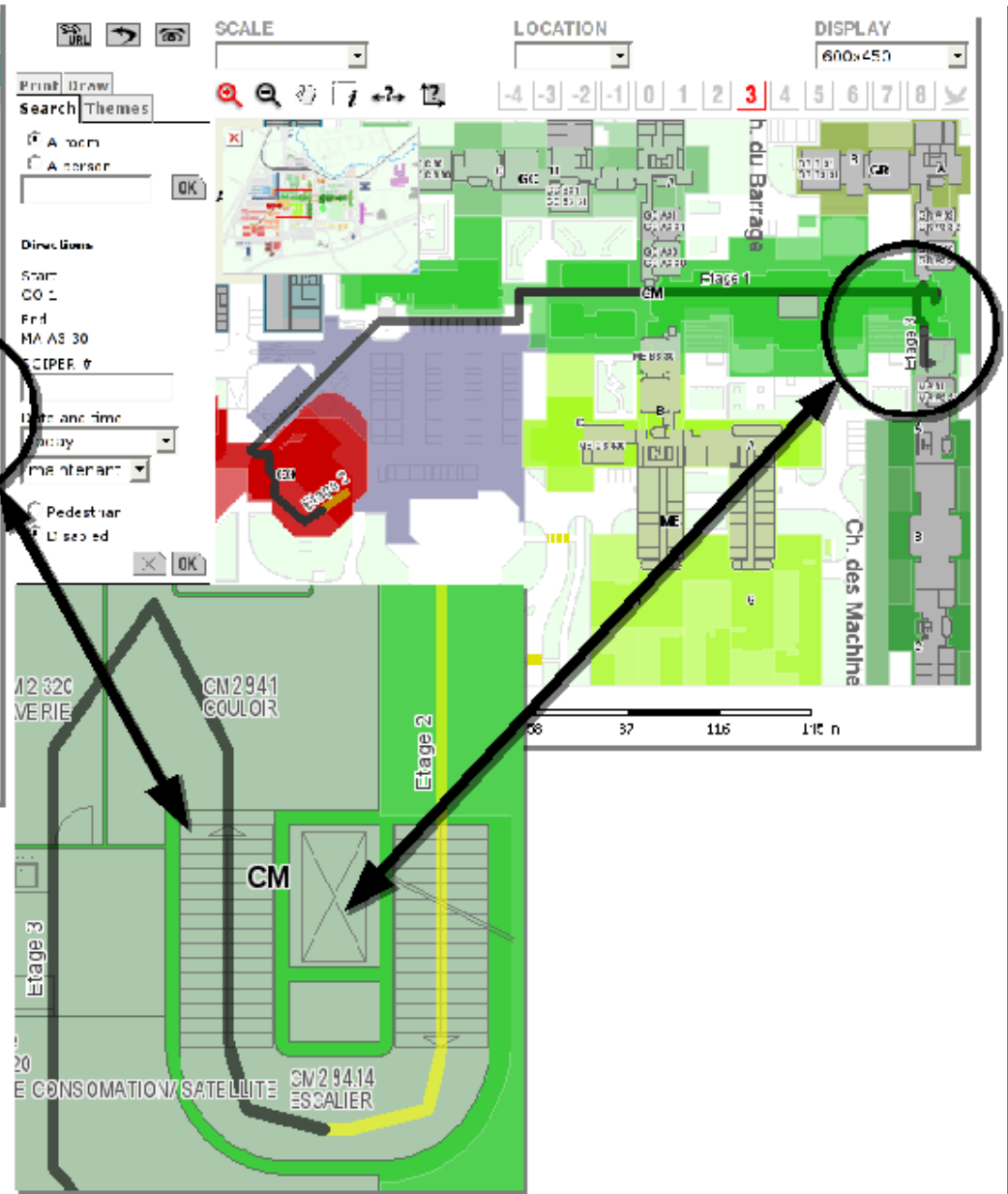
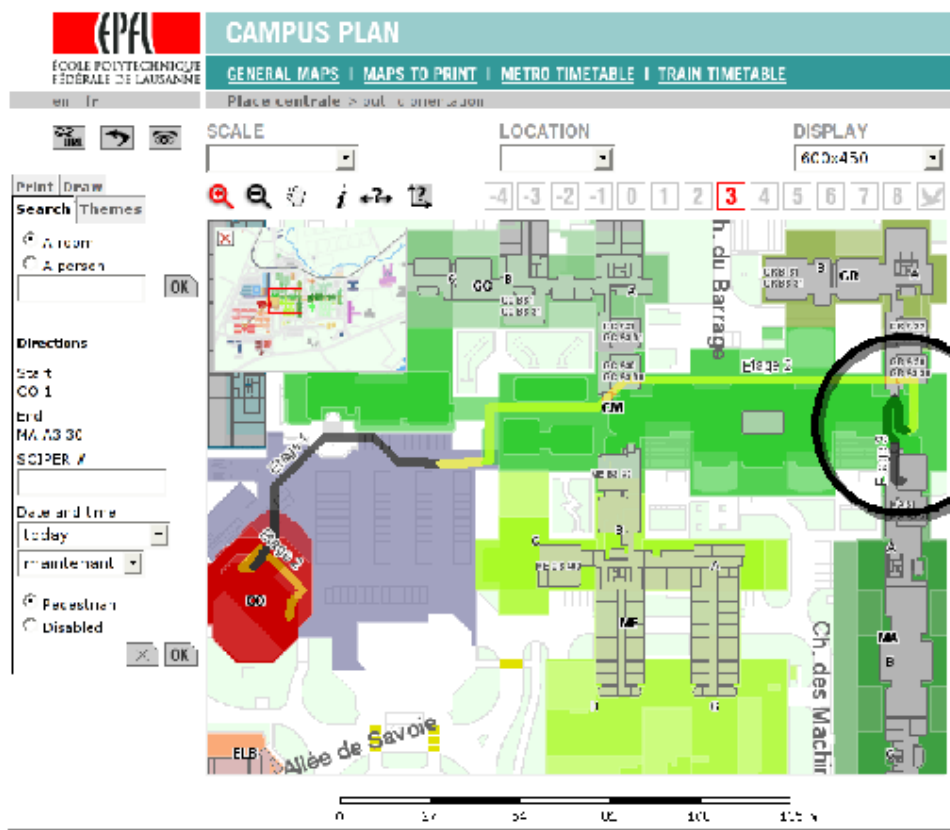
- Autres logiciels connectables à PostGIS:
 - UDIG
 - GeoTools
 - Geoserver
 - OGR
 - FME
 - Red spider (ionic)
 - Cadcorp sis
 - Esri (en cours)

Une extension: le routing avec pgdijkstra

- Camptocamp a développé une extension libre pour le calcul d'itinéraire avec PostgreSQL, **Pgdijkstra**. Cette extension calcule le chemin le plus court dans un graphe, qu'il s'agisse d'un réseau routier ou d'un réseau d'hyperliens entre pages web. Sous licence GPL, elle est librement téléchargeable. Ce module utilise l'algorithme Dijkstra.



Pgdijkstra - exemple



Questions ?