PostGIS 2.0 ...

... and beyond

Vincent Picavet – Oslandia

https://github.com/Oslandia/presentations/tree/master/fosdem_2013

OSLANDIA

PostGIS 2.0

PostGIS 2.0.0 : April 3, 2012

After 26 months !

Major version

Breaks compatibility

Loads of new features

Performance
improvement

OSLANDIA

What's new ?

OSLANDIA

Internals

- New serialization format
  - New geometry types (3D)
  - Fix 2D only bounding boxes
  - Fix bytes alignment
- New parsers
  - WKB
  - WKT

Features

Management fonctions

ISO SQL/MM compliancy

New functions for analysis

Topology (SQL/MM)

Real 3D storage

Raster / geometry functions

KNN indexed search

TIGER (geocoder / reverse...)

Install

Easier installation (PG >= 9.1)

CREATE EXTENSION postgis ;

CREATE EXTENSION postgis_topology ;

Manage

geometry_columns → view

Typmod usage

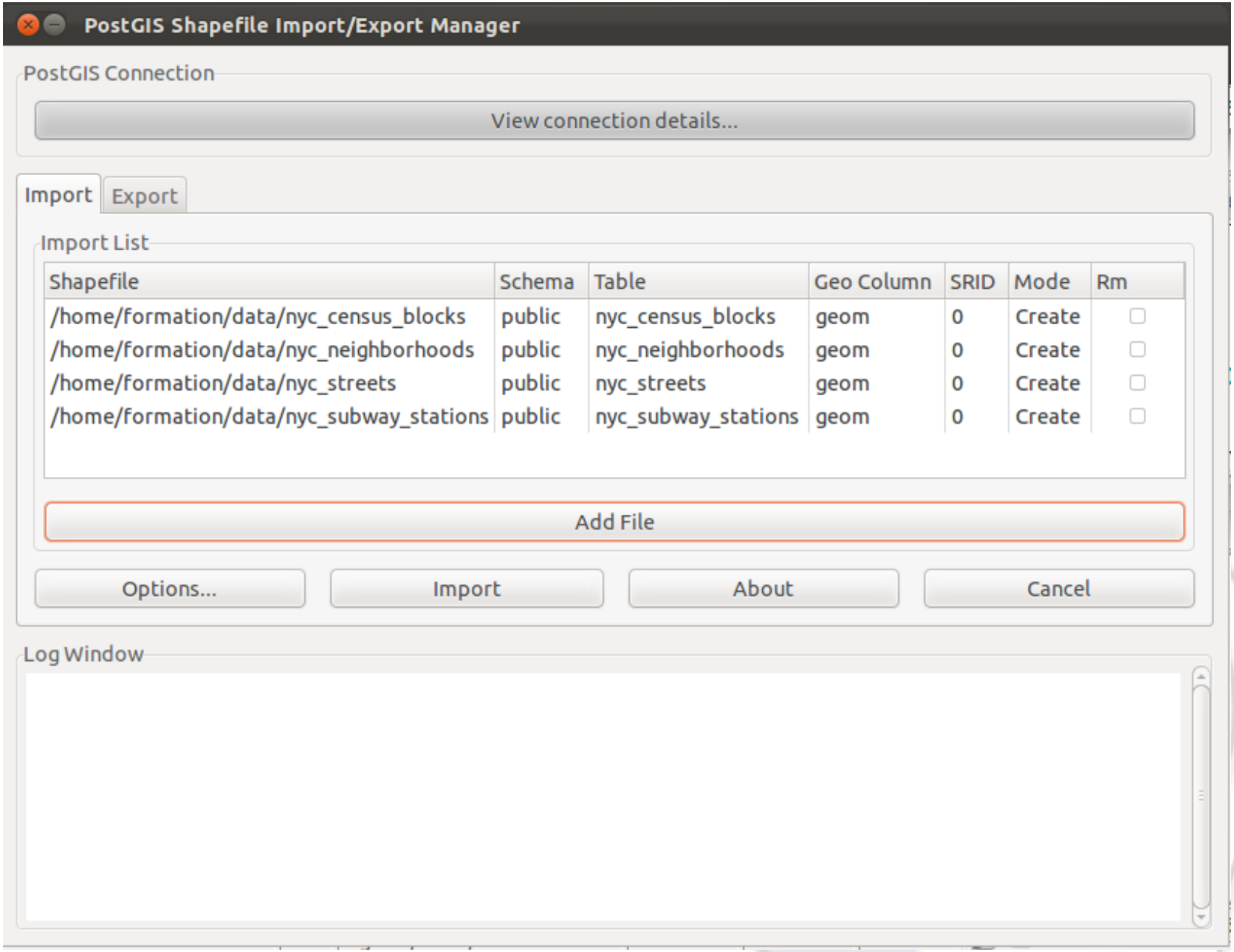
Old way still available

```
CREATE TABLE buildings (
        gid SERIAL PRIMARY KEY
        , geom geometry(MultiPolygon, 26986)
);

alter table buildings
        alter column geom
                type geometry(MultiPolygon, 2154)
                using st_setsrid(geom, 2154);
```

OSLANDIA

Load



OSLANDIA

Functions

ST_ConcaveHull

ST_Snap

ST_Split

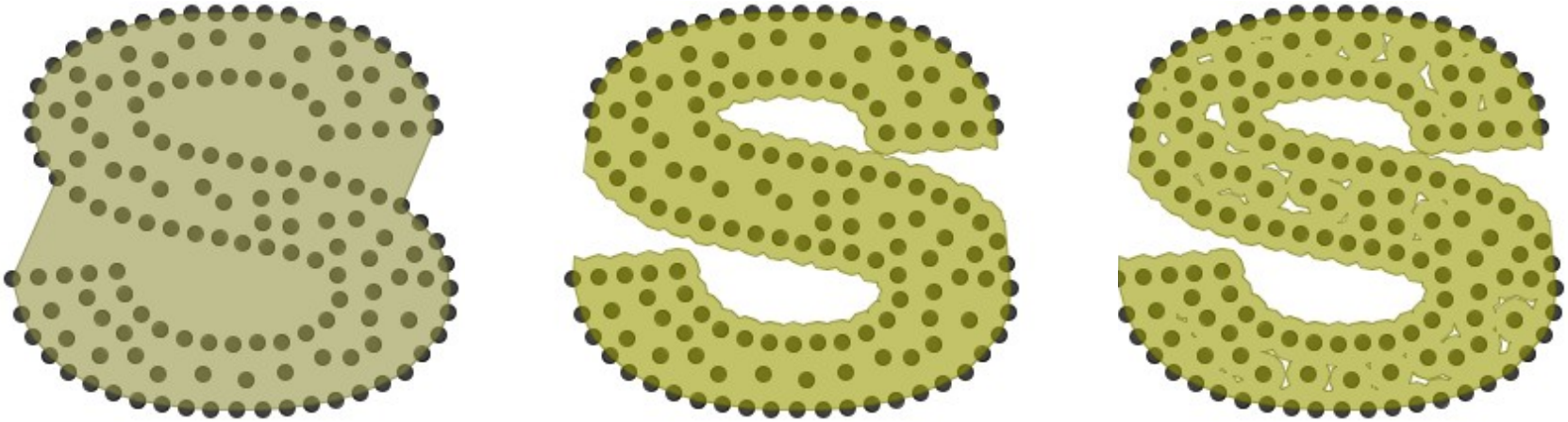
ST_MakeValid

ST_IsValidDetail

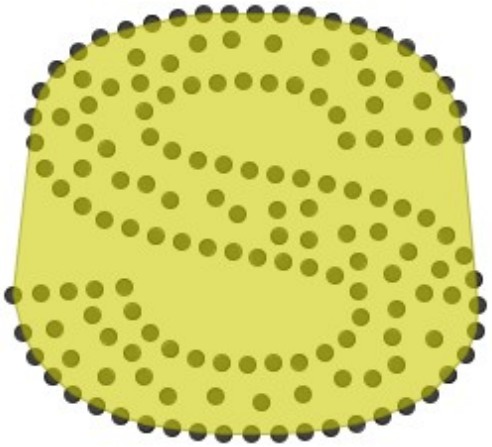
ST_OffsetCurve

...

hulls and curves



Concave hulls with various settings



Convex hull

Offset curves

OSLANDIA

cleaning data

Before : ST_Buffer(the_geom, 0)

After :

- ST_MakeValid()
- ST_RemoveRepeatedPoints()
- ST_isValidReason()
- ST_IsValidDetail()

```
SELECT ST_IsValid(geom),ST_IsValidReason(geom) FROM
 (SELECT ST_GeomFromText('POLYGON ((0 0, 0 10, 10 10, 10 0, 0 0),(20 20, 20 30, 30 30, 30 20, 20 20))') as geom) as foo;
 st_isvalid |                      st_isvalidreason
------------+-------------------------------------------------------------
 f          | Hole lies outside shell at or near point (20.0, 20.0, NaN)
```

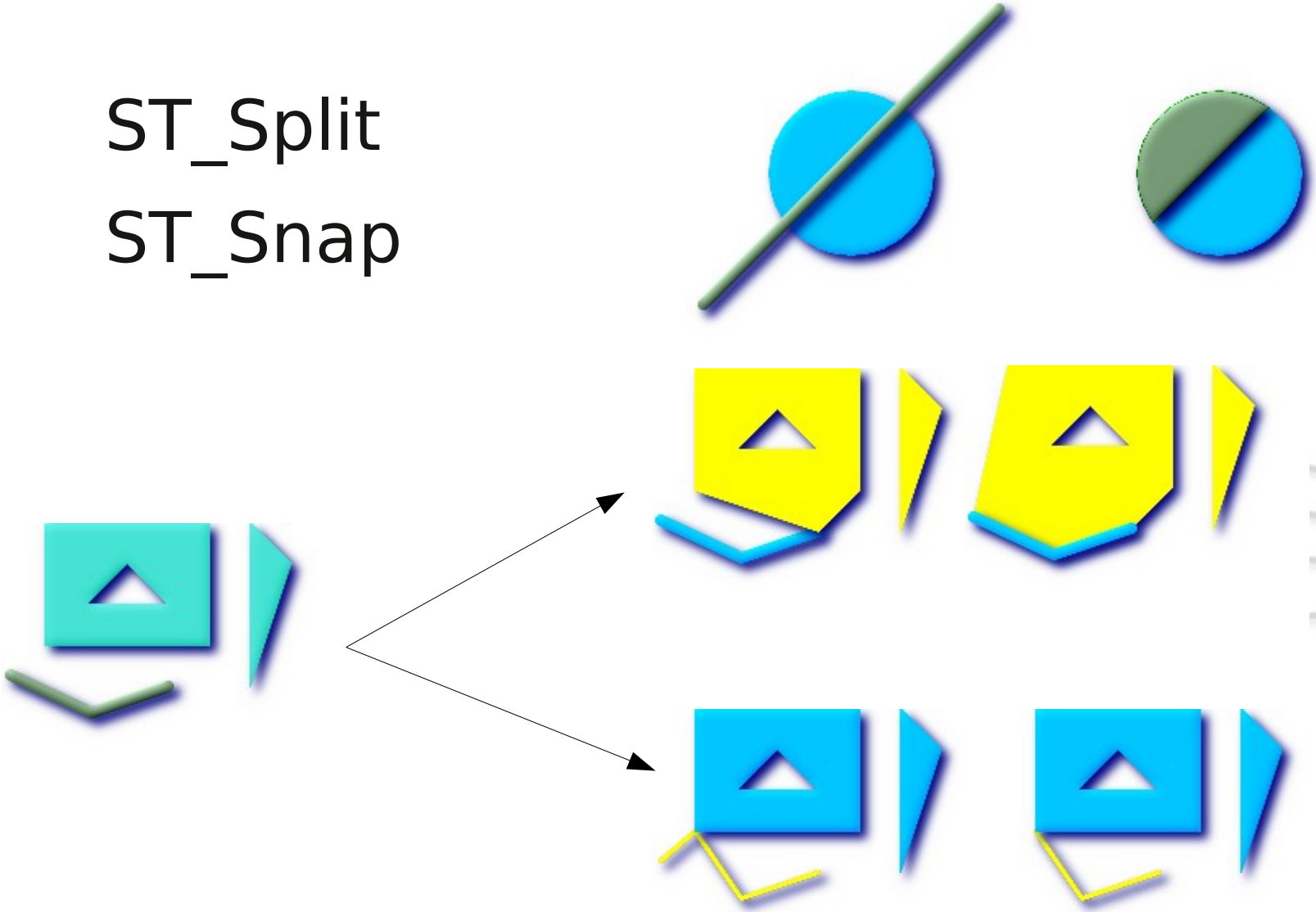
```
SELECT * FROM ST_IsValidDetail('LINESTRING(...)');
 gid  |      reason       |  location
------+-------------------+-------------
 5330 | Self-intersection | POINT(32 5)
```

OSLANDIA

Splitting and snapping

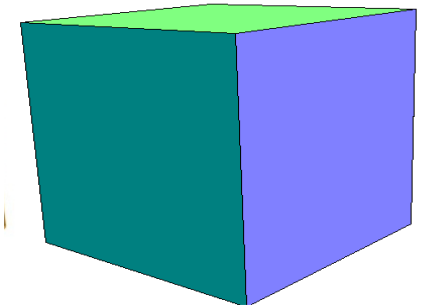
ST_Split

ST_Snap



OSLANDIA

Real 3D



«real» 3D inside PostGIS

ISO and OGC standards

- ISO 19125, SQL/MM, SFS 1.2.0

First step of implementation

New data types & functions

... Much more coming soon ...

OSLANDIA

3D

New types :

- TRIANGLE, POLYHEDRALSURFACE, TIN

New functions :

- ST_3DDistance, ST_3DIntersects, ST_3DDWithin, ST_3DClosestPoint...
- Input/Output : ST_AsGML, ST_AsX3D...

New operators

- &&&

Spatial index : nd-indexes

TIGER

Geocoder

Reverse geocoder

TIGER to PostGIS topology loader

Updated loader

Topology

Beware of the spaghetti monster !

OSLANDIA

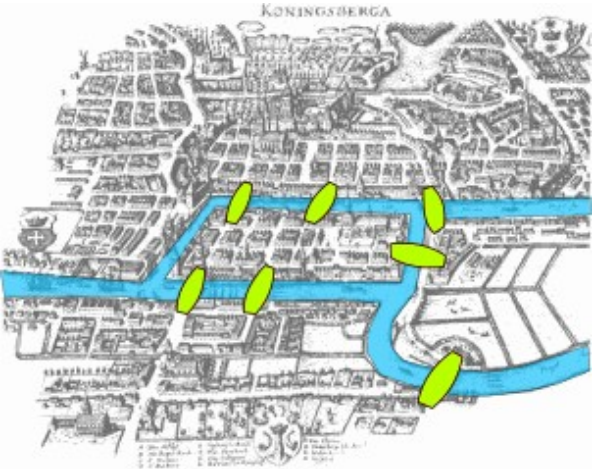
Topology - Graphs

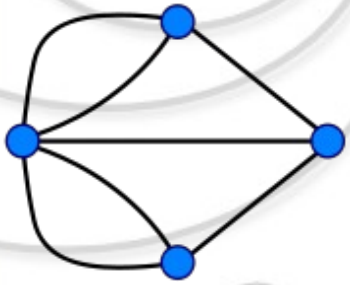
Explicit relations between objects

Graph representation

Various models

- OGC : Node / edge / face





Topology

Node/Edge/Face model

TopoGeometry Datatype

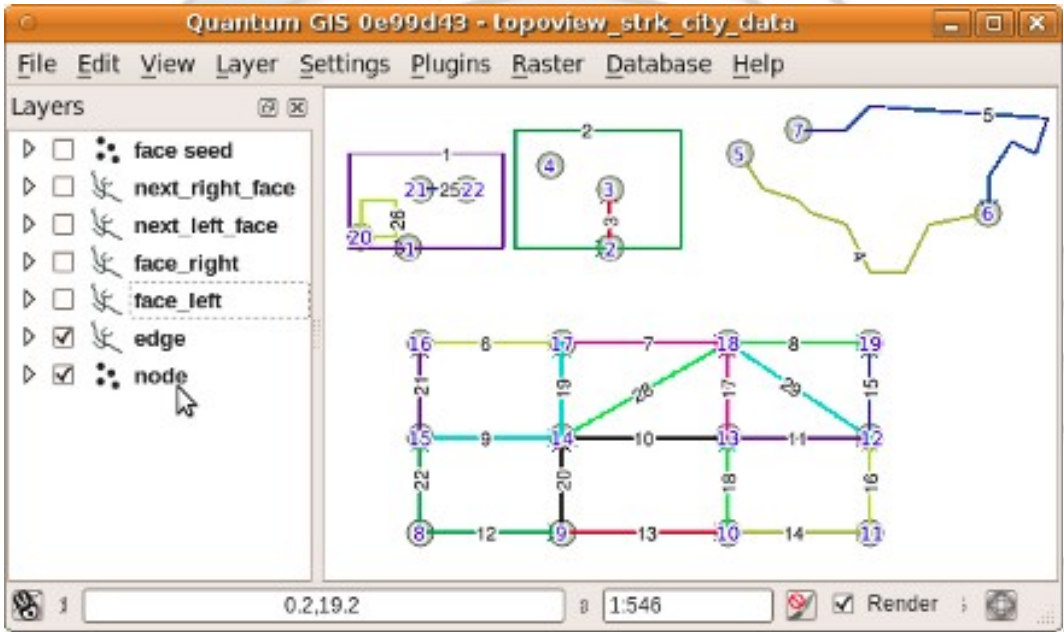
Use schemas

- «topology» for functions and others
- Each topology in its own schema

Full SQL/MM topology support

Integrated in 2.0

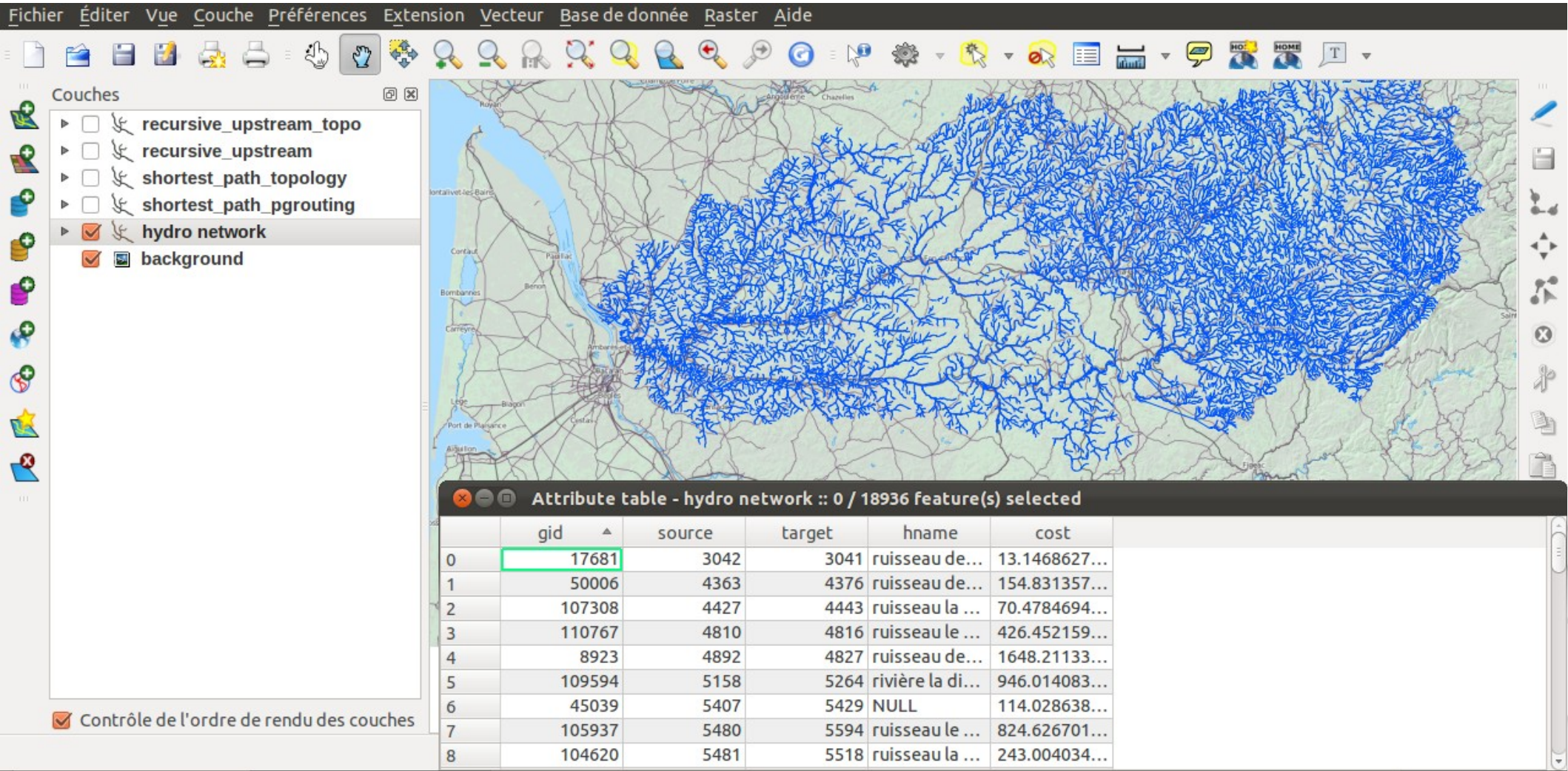
- Sandro Santilli
- Toscane Region



OSLANDIA

Topology use case

Table name : **tr**



	gid	source	target	hname	cost
0	17681	3042	3041	ruisseau de...	13.1468627...
1	50006	4363	4376	ruisseau de...	154.831357...
2	107308	4427	4443	ruisseau la ...	70.4784694...
3	110767	4810	4816	ruisseau le ...	426.452159...
4	8923	4892	4827	ruisseau de...	1648.21133...
5	109594	5158	5264	rivière la di...	946.014083...
6	45039	5407	5429	NULL	114.028638...
7	105937	5480	5594	ruisseau le ...	824.626701...
8	104620	5481	5518	ruisseau la ...	243.004034...

```sql
-- Create a topology
SELECT topology.CreateTopology('hydro', 2154);
-- 1

-- we put the postgis topology features for hydro network in another table
CREATE TABLE tr_topo (gid integer);

-- Add a layer
SELECT topology.AddTopoGeometryColumn('hydro', 'public',
        'tr_topo', 'topogeom', 'MULTILINESTRING');
-- 1

-- Populate the layer and the topology from tr geometry features
INSERT into tr_topo (gid, topogeom)
        SELECT gid, topology.toTopoGeom(geom, 'hydro', 1) FROM tr;
```

- Schémas (3)
  - hydro
    - Collationnements (0)
    - Domaines (0)
    - Configurations FTS (0)
    - Dictionnaires FTS (0)
    - Analyseurs FTS (0)
    - Modèles FTS (0)
    - Fonctions (0)
    - Séquences (5)
    - Tables (4)
      - edge_data
      - face
      - node
      - relation
    - Fonctions trigger (0)
    - Types (0)
    - Vues (1)
      - edge

```sql
select * from hydro.edge limit 10;
```

neau sortie

ortie de données | Expliquer (Explain) | Messages | Historique

	edge_id integer	start_node integer	end_node integer	next_left_edge integer	next_right_edge integer	left_face integer	right_face integer	geom geometry(LineStrin
1	175256	190369	190361	175230	-175243	0	0	01020000206A08
2	167356	183762	181917	166725	167356	0	0	01020000206A08

```sql
select * from tr_topo limit 10;
```

au sortie

rtie de données | Expliquer (Explain) | Messages

	gid integer	topogeom topology.topogeometry
	116768	(1,1,163704,2)
	116767	(1,1,163705,2)

Recursive CTE

```
create table
        rec_res2 as
with recursive
        search_graph(edge_id, start_node, depth, path, length, cycle) as (
                select
                        g.edge_id, g.start_node, 1 as depth, ARRAY[g.edge_id] as path
                        , st_length(geom) as length, false as cycle
                from
                        hydro.edge as g
                where
                        edge_id = 173832
                union all
                select
                        g.edge_id
                        , g.start_node
                        , sg.depth + 1 as depth
                        , path || g.edge_id as path
                        , sg.length + st_length(g.geom) as length
                        , g.edge_id = ANY(path) as cycle
                from
                        hydro.edge as g
                join
                        search_graph as sg
                on
                        sg.start_node = g.end_node
                where
                        not cycle
                )
```

1

2

```
select
        sg.*
        , edge.geom as geom
from
        search_graph as sg
join
        hydro.edge as edge
on
        sg.edge_id = edge.edge_id
limit 1000;
```

3

OSLANDIA

1 : init

```
select
        g.edge_id, g.start_node, 1 as depth, ARRAY[g.edge_id] as path
        , st_length(geom) as length, false as cycle
from
        hydro.edge as g
where
        edge_id = 173832
union all
```

OSLANDIA

2 : recursive part

```
select
        g.edge_id
        , g.start_node
        , sg.depth + 1 as depth
        , path || g.edge_id as path
        , sg.length + st_length(g.geom) as length
        , g.edge_id = ANY(path) as cycle
from
        hydro.edge as g
join
        search_graph as sg
on
        sg.start_node = g.end_node
where
        not cycle
)
```

Stack the gid to the path for this record

Sum up the cost (it's the length here)

If the record gid is already in the path, we have a cycle

Join result set from previous iteration to connected upstream edges

Do not take elements which make a cycle

OSLANDIA

3 : Get results

```
select
        sg.*
        , edge.geom as geom
from
        search_graph as sg
join
        hydro.edge as edge
on
        sg.edge_id = edge.edge_id
limit 1000;
```

Join CTE results to original table to get geometries

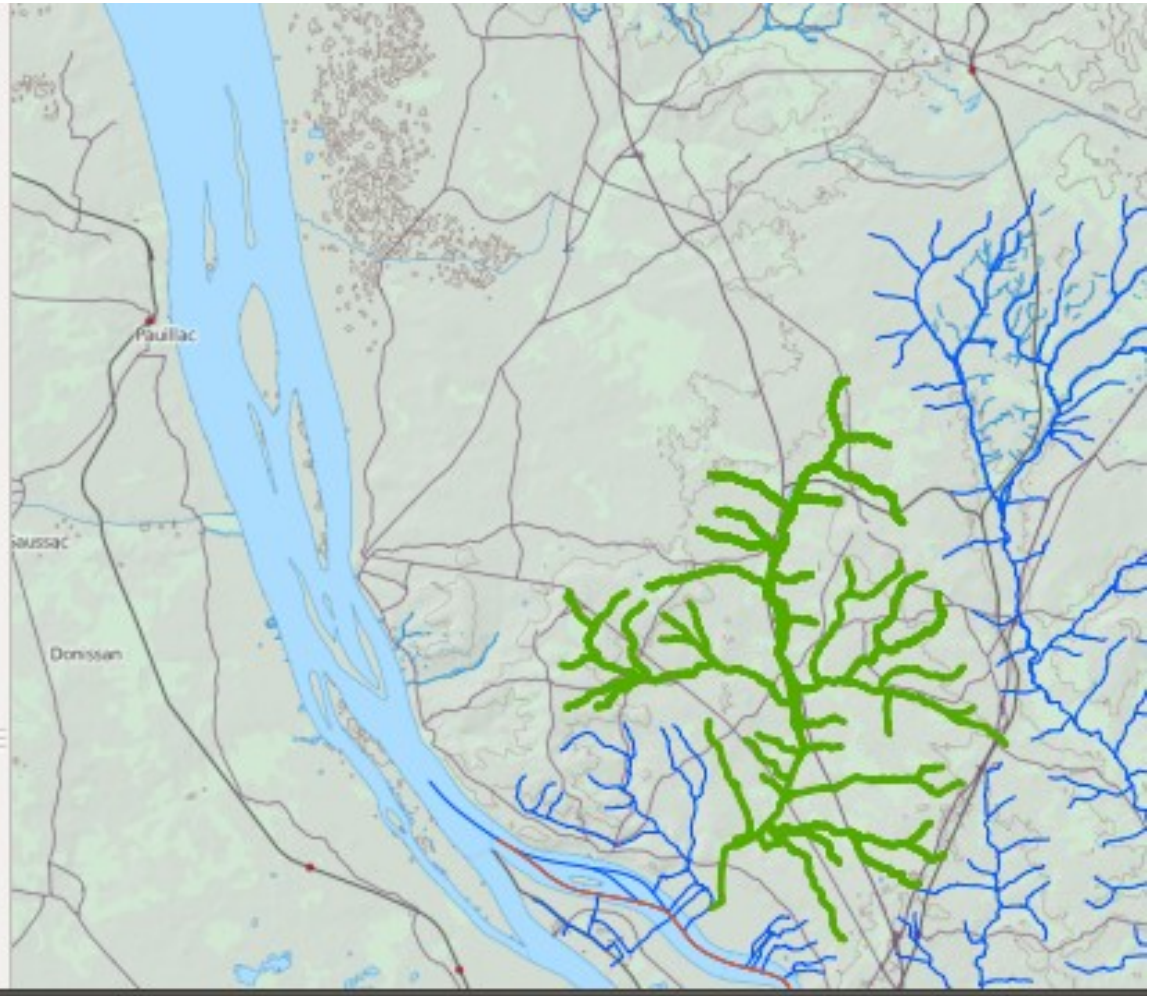
Better limit recursive queries to avoid unfinite loops

gid integer	source integer	depth integer	path integer[]	length double precision	cycle boolean	geom geometry(MultiLineString,2154)
31913	20850	1	{31913}	2666.0523017	f	01050000206A08000001000
33855	20735	2	{31913,	3473.3086319	f	01050000206A08000001000
32477	20845	2	{31913,	2725.7640259	f	01050000206A08000001000
33854	19909	3	{31913,	7183.7295195	f	01050000206A08000001000

OSLANDIA

Couches

- recursive_upstream_topo
- recursive_upstream
- shortest_path_topology
- shortest_path_pgrouting
- hydro network
- background



Attribute table - recursive_upstream_topo :: 0 / 478 feature(s) selected

	edge_id	start_node	depth	path	length	cycle
0	173832	189333	1	{173832}	2666.05230...	f
1	173452	189332	2	{173832,17...	3473.30863...	f

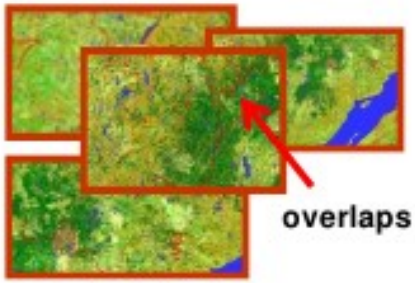
OSLANDIA

PostGIS Raster

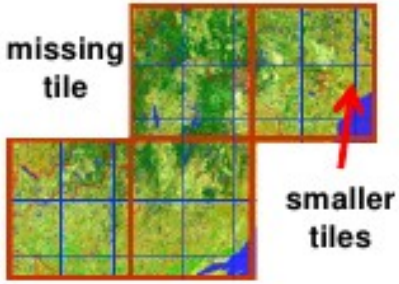
- Raster / vector analysis
- New datatype
  - Looks like geometry
  - But for rasters
- Multiresolution, multiband, tile coverage
- Import/export (GDAL)
- Functions
  - Statistics, reprojection, edit, compute
  - Vector/raster functions

OSLANDIA

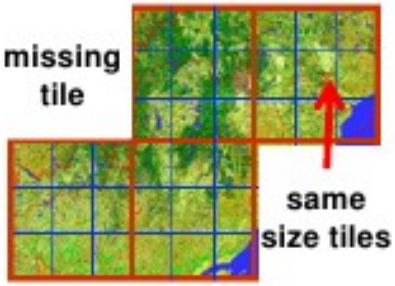
PostGIS Raster



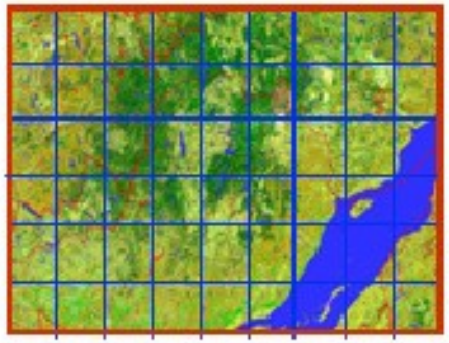
a) warehouse of untiled and unrelated images (4 images)



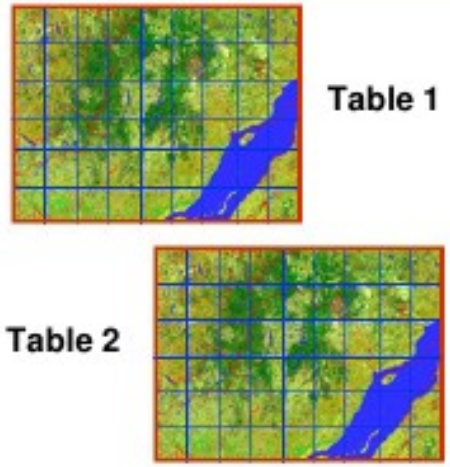
b) irregularly tiled raster coverage (36 tiles)



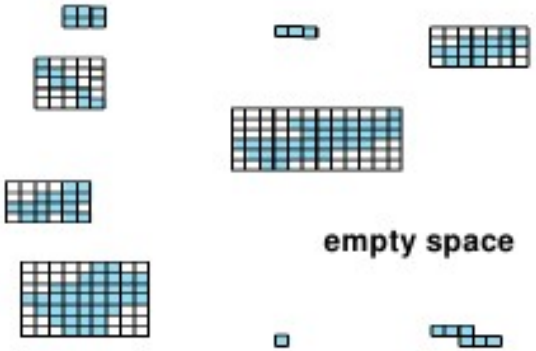
c) regularly tiled raster coverage (36 tiles)



d) rectangular regularly tiled raster coverage (54 tiles)



e) tiled images (2 tables of 54 tiles)

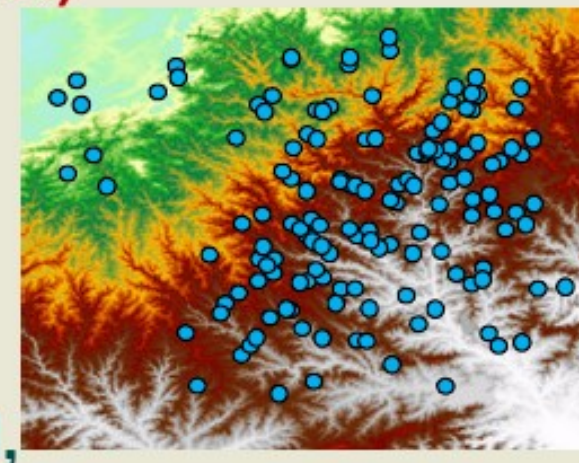


f) rasterized geometries coverage (9 lines in the table)

PostGIS 2.0 : PostGIS Raster

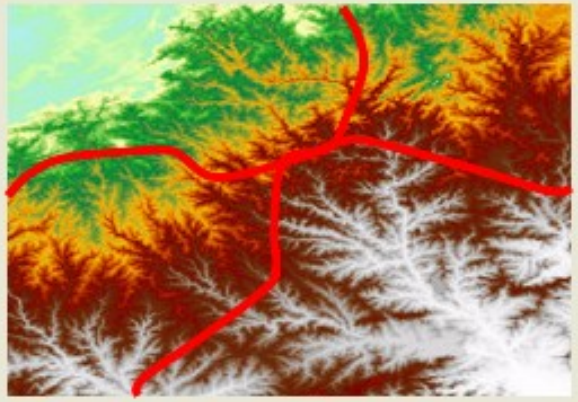
Extract ground elevation values for lidar points…

- SELECT pointID, ST_Value(rast, geom) elevation
  FROM lidar, srtm WHERE ST_Intersects(geom, rast)



Intersect a road network to extract elevation values for each road segment

- SELECT roadID,
  (ST_Intersection(geom, rast)).geom road,
  (ST_Intersection(geom, rast)).val elevation
  FROM roadNetwork, srtm WHERE ST_Intersects(geom, rast)



PostGIS 2.0 : nearest neighbours

KNN-GIST search in PostgreSQL 9.1

Use indexes !

Spatial nearest neighbors

```
SELECT name, gid FROM geonames
ORDER BY
    geom <-> st_setsrid(st_makepoint(-90,40),4326)
LIMIT 10;
```

Distance operator

- <-> or <#> : center or bbox
- Need to refine for non-point geometries

OSLANDIA

PostGIS 2.1 ?

PostGIS 2.1

Currently under work :

Arc-geometry distance

Distance with cached tree

R-Tree index improvement (pick-split)

SP-Gist Index

- New in PG 9.1
- Up to 3x faster construction
- Faster to read

OSLANDIA

PostGIS 2.1

Under work, raster part :

ST_MapAlgebra with n rasters

Complete St_SetValues

Bands to array function

Better « isnodata» management

Raster tiling

(proprietary) support in FME ETL

PostGIS 2.1

And more...

Topology improvement

Tiger geocoder as PG extension

PgRouting as PG extension

+ windows support

PostGIS 2.1

And even more...

Development platform improvement

Meet Debbie and Winnie !

Jenkins

search

log in | sign up

Jenkins

ENABLE AUTO REFRESH

People

Build History

Build Queue

No builds in the queue.

Build Executor Status

#	Status
1	Idle
2	Idle
3	Idle
4	Idle

All | GDAL | GEOS | PostGIS | PostgreSQL

S	W	Name ↓	Last Success	Last Failure	Last Duration
		GDAL_PostGIS_Regress	1 day 17 hr (#113)	1 mo 1 day (#2)	5 min 2 sec
		GDAL_Regress	1 day 17 hr (#133)	5 days 15 hr (#127)	2 min 19 sec
		GDAL_Trunk	1 day 17 hr (#159)	20 days (#89)	17 min
		GEOS_Trunk	1 mo 7 days (#13)	N/A	2 min 28 sec
		PG_Version	11 days (#9)	1 mo 13 days (#6)	4 min 7 sec
		PG_Version_Dev	2 days 10 hr (#11)	1 mo 1 day (#4)	4 min 40 sec
		PostGIS_2.0	16 hr (#34)	4 days 20 hr (#29)	9 min 24 sec
		PostGIS_2.0_docs	16 hr (#48)	7 days 3 hr (#32)	10 min
		PostGIS_2.1	11 hr (#230)	2 days 21 hr (#220)	14 min
		PostGIS_2.1_docs	11 hr (#148)	16 hr (#145)	10 min
		PostGIS_2.1_doxygen	20 hr (#45)	8 hr 18 min (#47)	23 min

PostGIS 2.1

```
[00:40] <debbie> Project PostGIS_2.0 build #34: SUCCESS in 9 min 21 sec: http://debbie.postgis.net:8080/job/Post
[00:40] <debbie> Paul Ramsey: (#2026) fix performance regression in geography distance calculation
[00:40] <sigq> Title: PostGIS_2.0 #34 [Jenkins] (at debbie.postgis.net:8080)
[00:44] <pramsey> done!
[00:47] <debbie> Project PostGIS_2.1 build #227: SUCCESS in 26 min: http://debbie.postgis.net:8080/job/PostGIS_
[00:47] <debbie> Paul Ramsey: (#2026) fix performance regression in geography distance calculation
[00:47] <sigq> Title: PostGIS_2.1 #227 [Jenkins] (at debbie.postgis.net:8080)
[00:51] --> tomkralidis a rejoint ce canal (~tomkralid@CPE0013ce450e14-CM001692413c80.cpe.net.cable.rogers.co
[00:51] <-- tomkralidis a quitté ce serveur (Changing host).
[00:51] --> tomkralidis a rejoint ce canal (~tomkralid@osgeo/member/tomkralidis).
[00:51] <debbie> Project PostGIS_2.1_docs build #145: FAILURE in 4 min 0 sec: http://debbie.postgis.net:8080/job/P
[00:51] <debbie> * Bborie Park: Added news and docs for ST_Tile(raster). Additional regression tests for
[00:51] <debbie> one additional variant of ST_Tile(raster)
[00:51] <sigq> Title: PostGIS_2.1_docs #145 [Jenkins] (at debbie.postgis.net:8080)
[00:51] <debbie> * Bborie Park: Added ST_Tile() and regression tests. The circle is complete.
[00:51] <debbie> * Bborie Park: Added rt_band_get_pixel_line() and regression tests
[00:51] <debbie> * Paul Ramsey: (#2063) fix the vertex-crossing logic in the circular tree code to use the new edge
[00:51] <debbie> * Paul Ramsey: (#2026) fix performance regression in geography distance calculation
[00:52] --> tbowden a rejoint ce canal (~tim@124-148-118-242.dyn.iinet.net.au).
[00:54] <-- epifanio a quitté ce serveur (Read error: Operation timed out).
[00:58] <winnie> Project PostGIS 2.1 mingW regress build #456: STILL FAILING in 2 min 30 sec: http://winnie.postg
[00:58] <sigq> Title: PostGIS 2.1 mingW regress #456 [Jenkins] (at winnie.postgis.net:1500)
```

+ Hallie :
documentation bot (PostgreSQL FTS & more)

OSLANDIA

And beyond ?

OSLANDIA

PostGIS 3.0 ?

Paris codesprint and barcamp May 2012

Find directions for future

- Git, build system (mainly Windows)
- Geometry backend (GEOS vs BGL vs ?)
- Raster improvement
- 3D topology & processing (CGAL?)
- Point clouds
- Performance, performance, performance

Let's go 3D !

PostGIS 3D

2.5D already in

3D storage is in

We want analysis !

- ST_3Dintersects
- ST_3Dintersection
- ST_extrude (2D → 3D)
- ST_3Dconvexhull
- ST_DelaunayTriangles...

OSLANDIA

PostGIS 3D

2D & 3D geometric computation

C++

Exact computation

Efficient, generic, extensible...

...and now GPL !

PostGIS 3D

Some European funding

Cooperation with IGN & others

Creation of SFCGAL framework (OGC-SF)

Use it in PostGIS

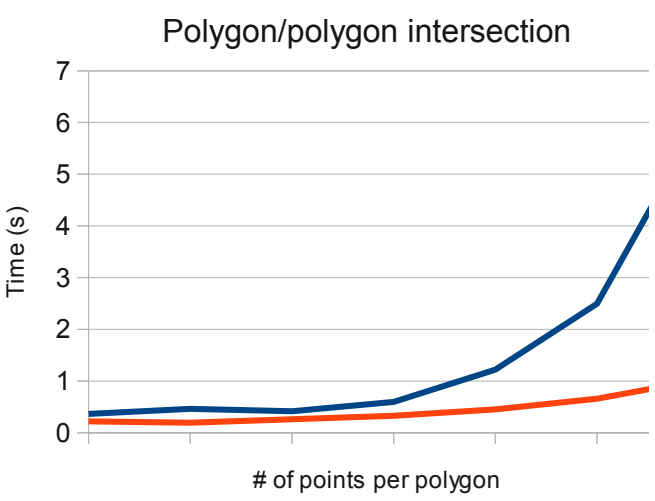
Compare with GEOS (for 2D)

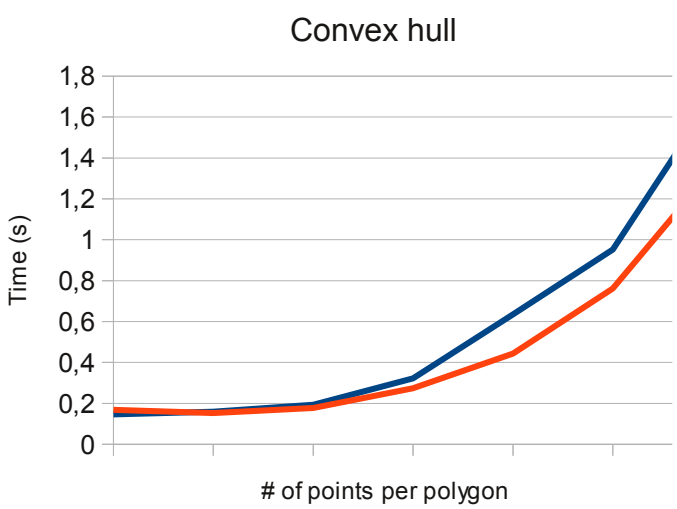
OSLANDIA

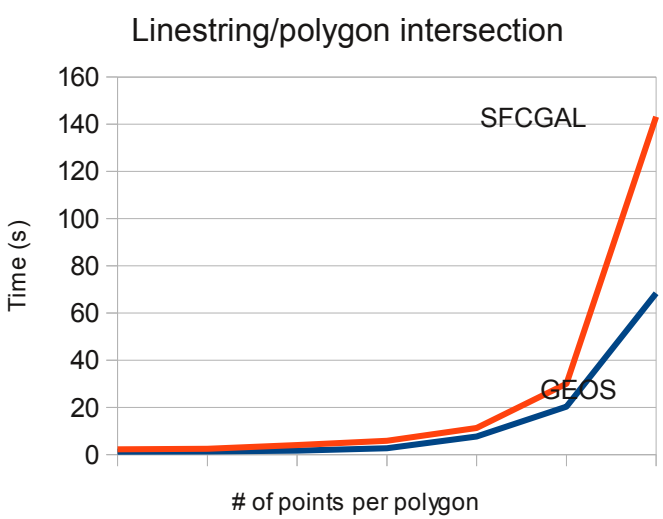
PostGIS 3D

Most operations faster

Some operations to optimize







OSLANDIA

PostGIS 3D

Issues

CGAL : exact predicates & constructions

PostGIS / GEOS : snap bounding (2D)

```
SELECT
  ST_Intersects(
    ST_Intersection(
      'LINESTRING(0 0,2 1)'::geometry,
      'LINESTRING(1 0,0 1)'::geometry),
    'LINESTRING(0 0,2 1)'::geometry);
 st_intersects
---------------
 f
(1 row)
```

ry reference

See Hugo's talk on GitHub

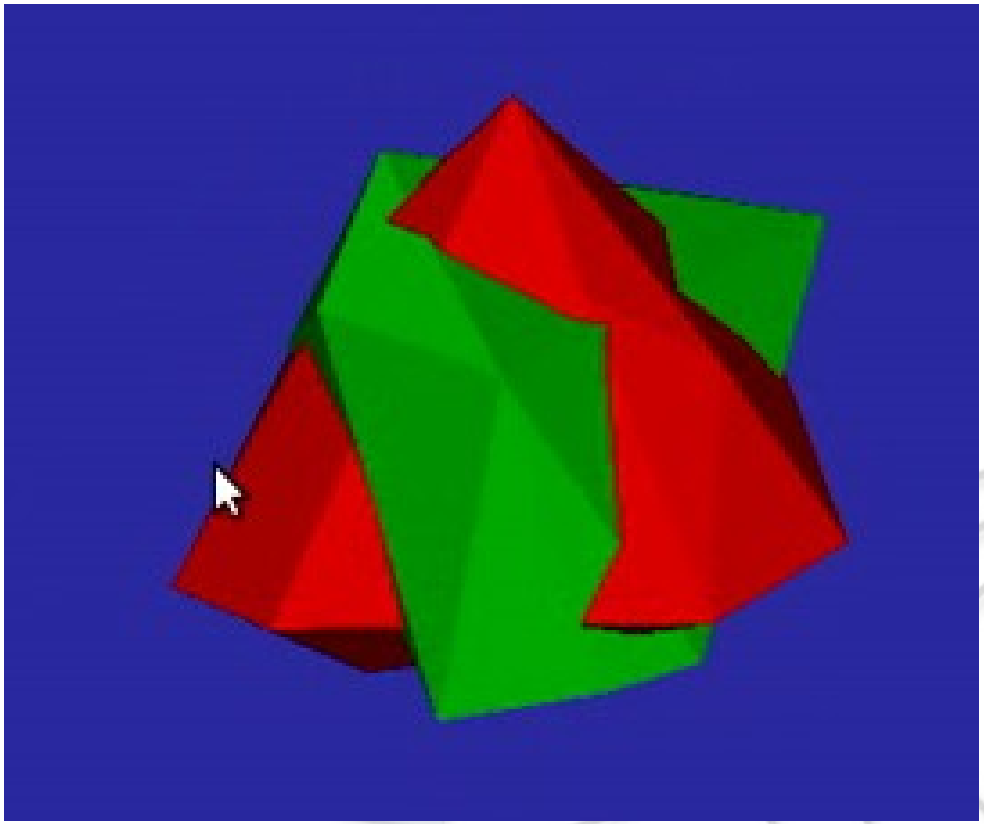
OSLANDIA

(very) Recent progress

Quantum GIS client (Globe)

Some analysis functions

- ST_Extrude
- ST_3DConvexhull
- ST_3DIntersection
  - Surfaces
  - Solids



OSLANDIA

Want to see ?

3D Next steps

Debug

More features from CGAL

Better QGIS support

CityGML & Collada loaders / exporters

PostGIS core integration

- To be discussed at Boston codesprint 2013

TinyOWS for 3D webservices

Find €€€€€ to speed up development

OSLANDIA

That's it...

Questions ?

vincent.picavet@oslandia.com

Twitter : @vpicavet

http://www.oslandia.com

OSLANDIA