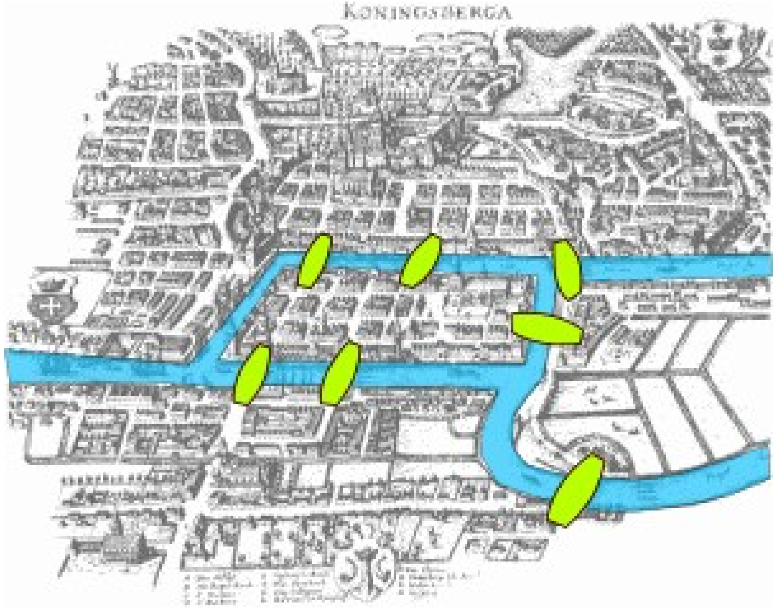
Topology with PostGIS 2.0



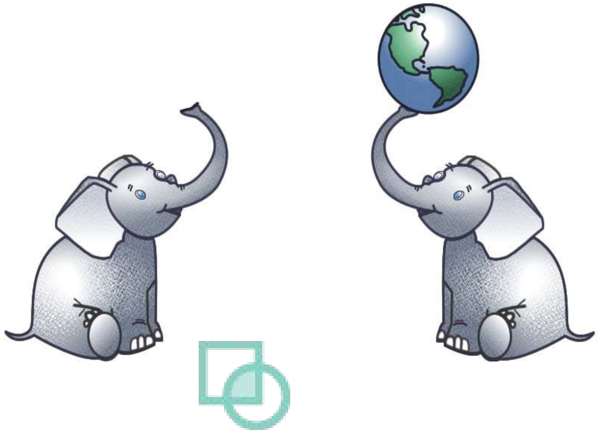
Sandro Santilli <strk@keybit.net>

- http://strk.keybit.net -

Paris 2011

PostGIS topology timeline

- Presented at FOSS4G 2006
  PostGIS: future developments
- Drafted in PostGIS 1.1.0
  Based on ISO SQL/MM
- Integrated in 2010
  Testsuite, build scripts, packaging
- Further Improved in 2011
  Full SQL/MM implementation,
  more utility functions
- Prime-time in PostGIS 2.0.0

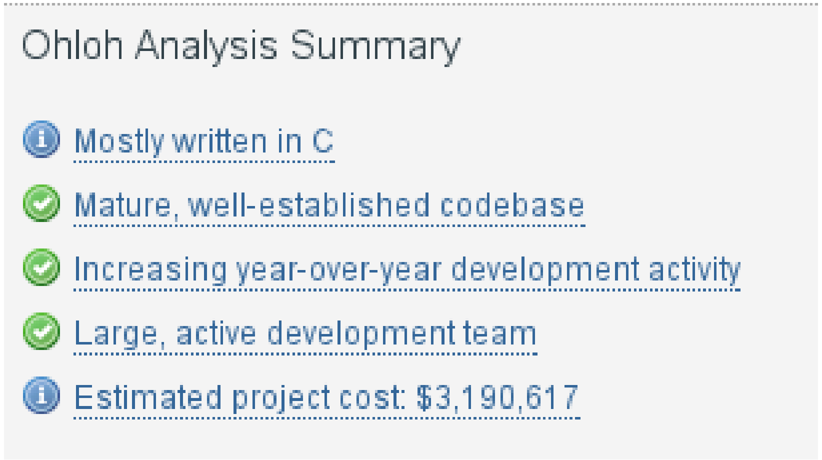


Faunalia

Regione Toscana

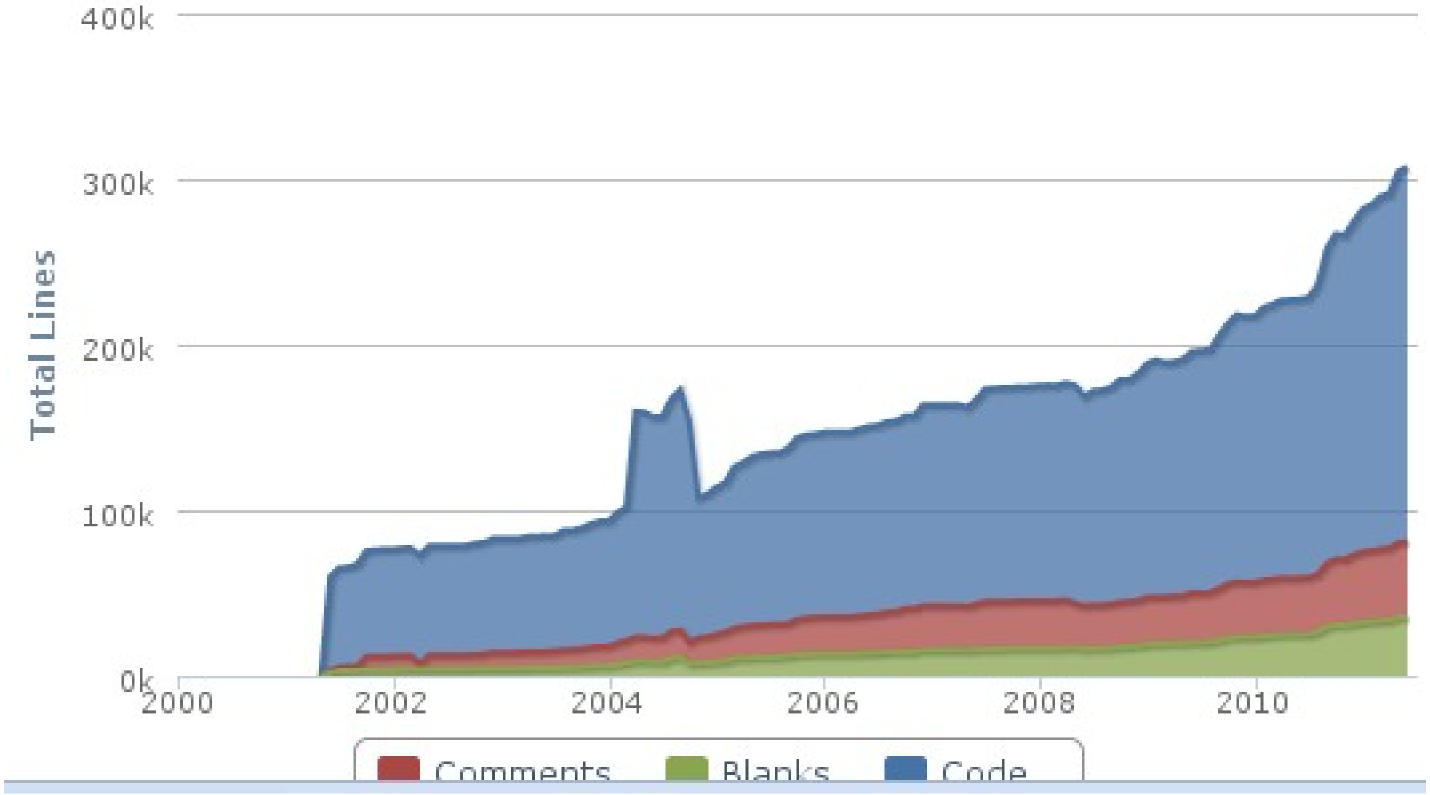
PostGIS development timeline and current state

stats by ohloh



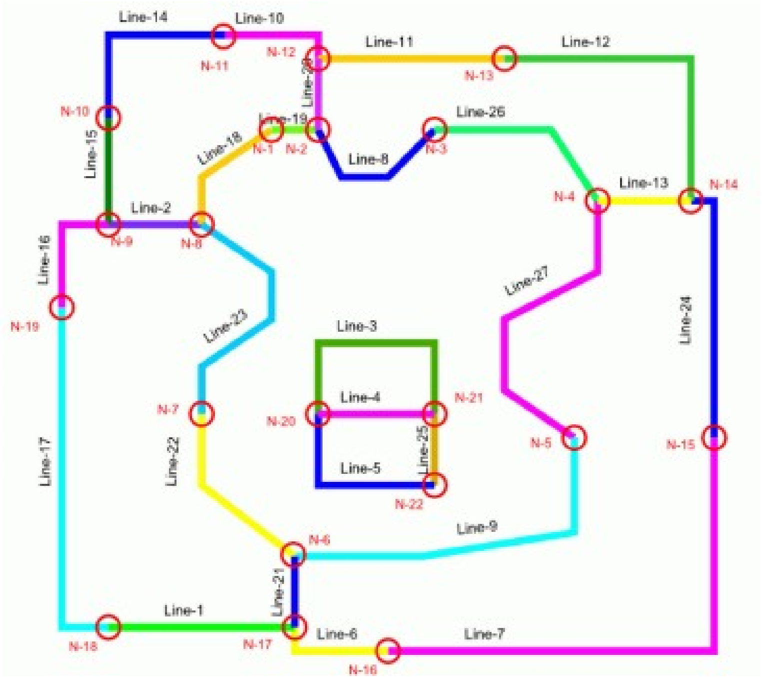
Ohloh Analysis Summary

- ⓘ Mostly written in C
- ✓ Mature, well-established codebase
- ✓ Increasing year-over-year development activity
- ✓ Large, active development team
- ⓘ Estimated project cost: $3,190,617



Why topology ?

- Normalized spatial data
- Standard interface
- Topological integrity
- Reduced storage size
- Explicit spatial relationships



Why topology ?

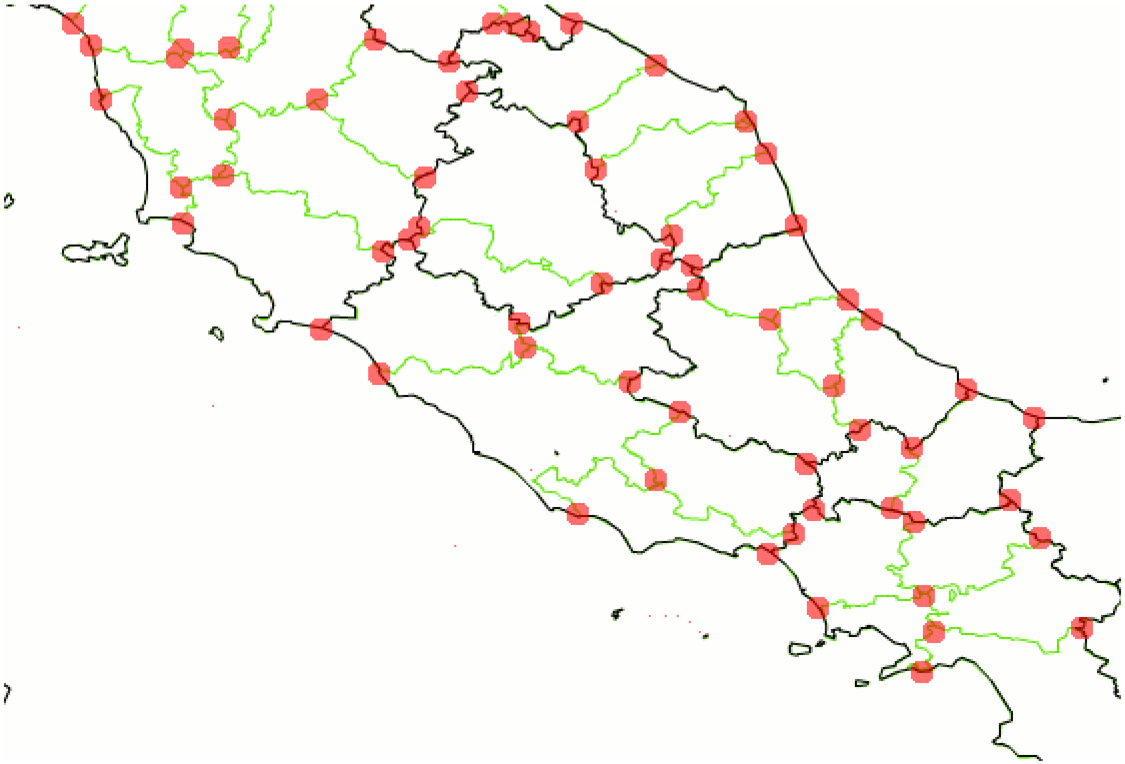
Topological integrity

- Every intersection is a node

Why topology ?

Topological integrity

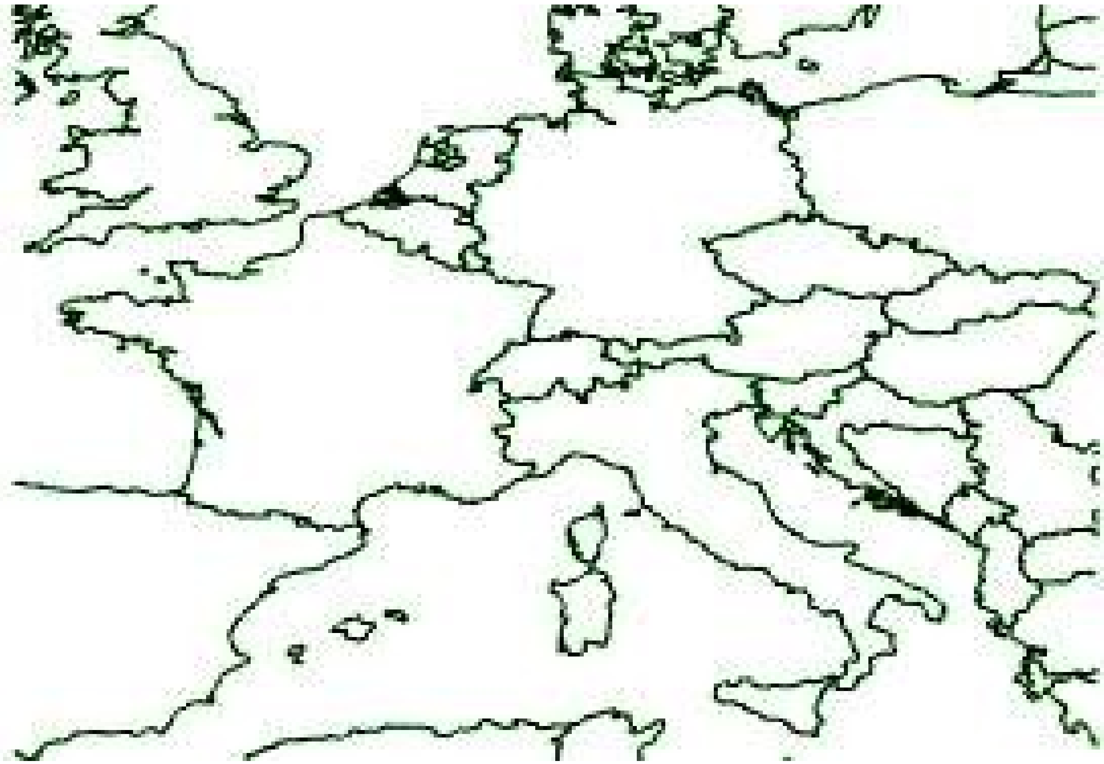
- Every intersection is a node



Why topology ?

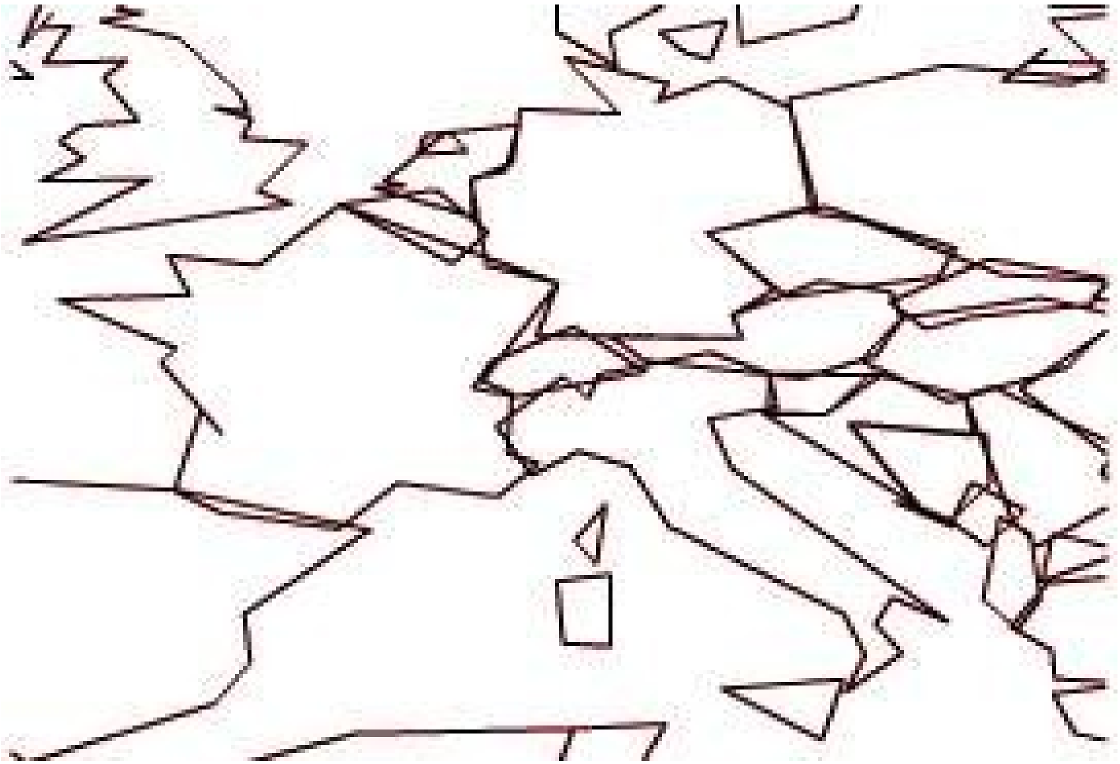
Topological integrity

- Edges are **shared** ...



Topological integrity

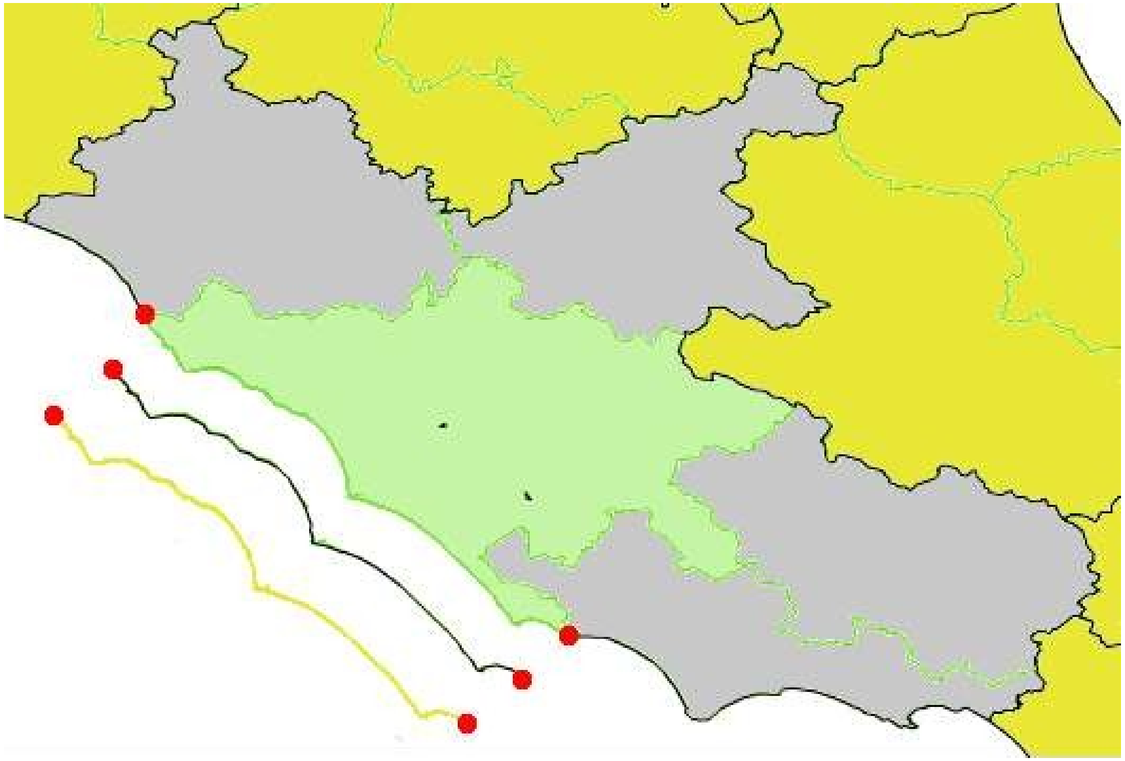
- … rather than **separate** entities



Why topology ?

Reduced storage size

- Every edge is stored only **once**



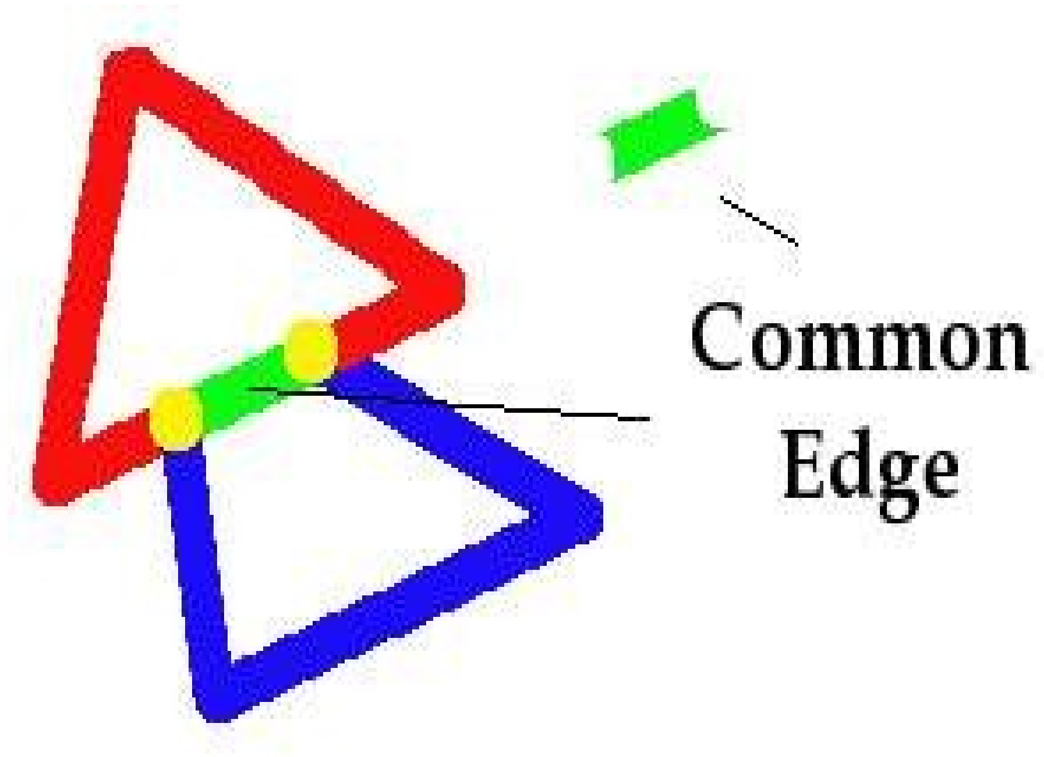
Reduced storage size

- Features in hierarchical layers can be defined by **composition**. For example:
  - A state is a collection of regions
  - A region is a collection of provinces
  - A province is a collection of municipalities
  - ....

Why topology ?

Explicit spatial relationships

- Do they touch ? YES !



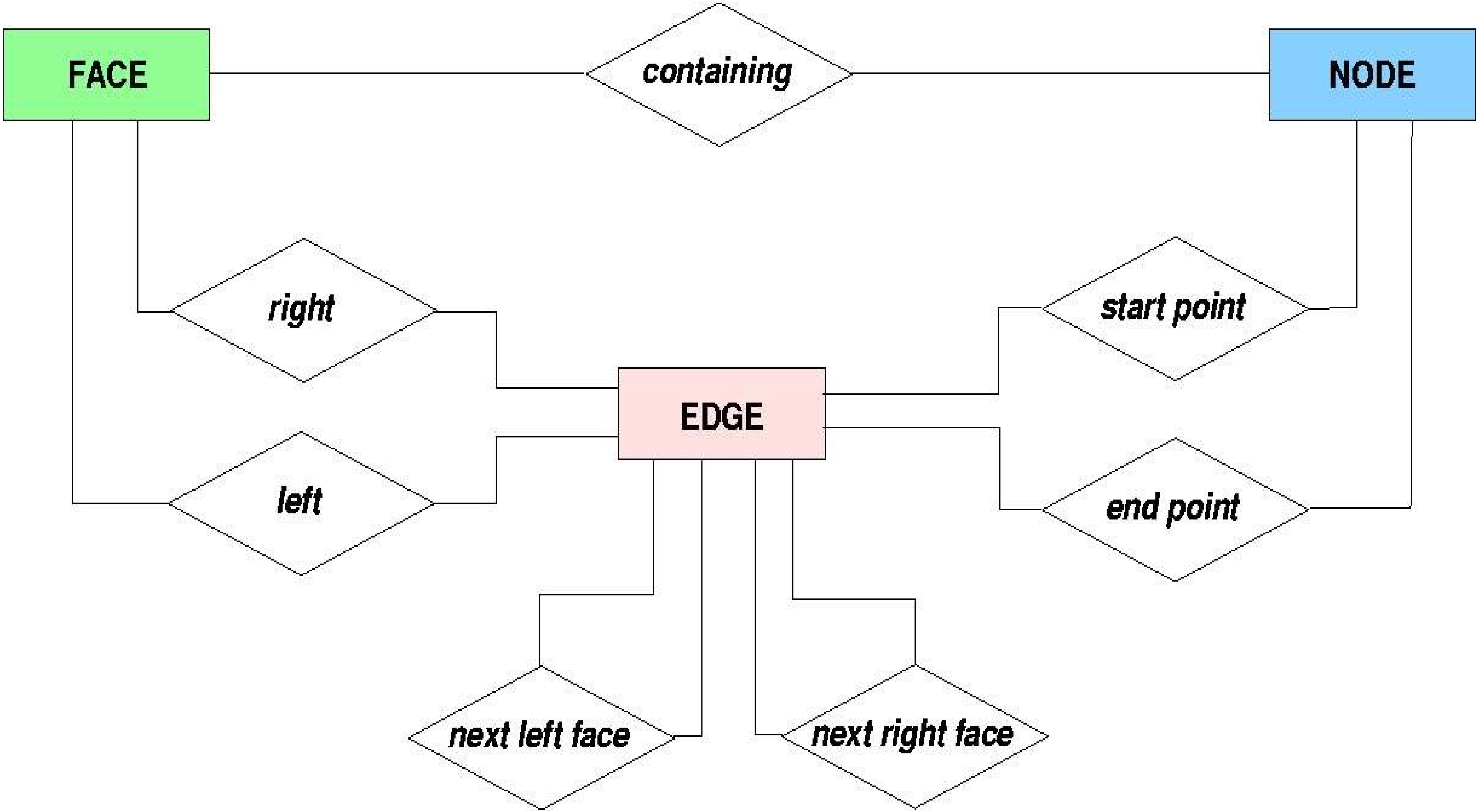
Explicit spatial relationships

- For each edge you know the right and left face
- For each isolated node you know the face it's in
- Every intersection is a node
- Nodes are shared

Conceptual Model

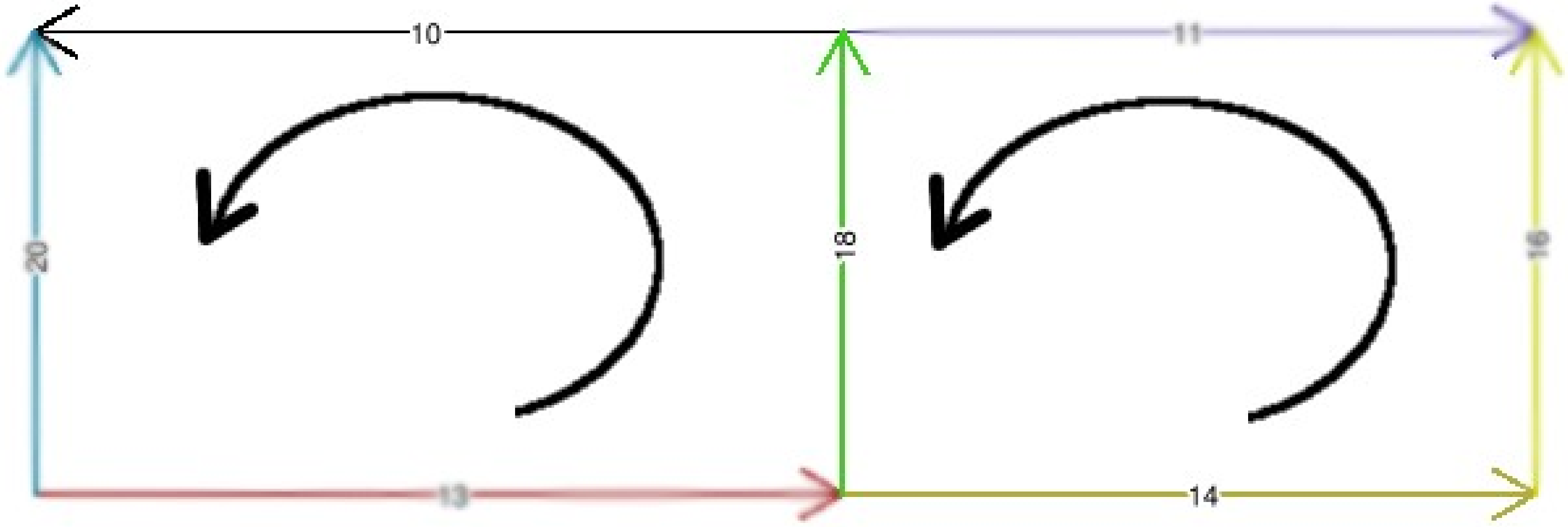
Conceptual model

Faces, Edges and Nodes

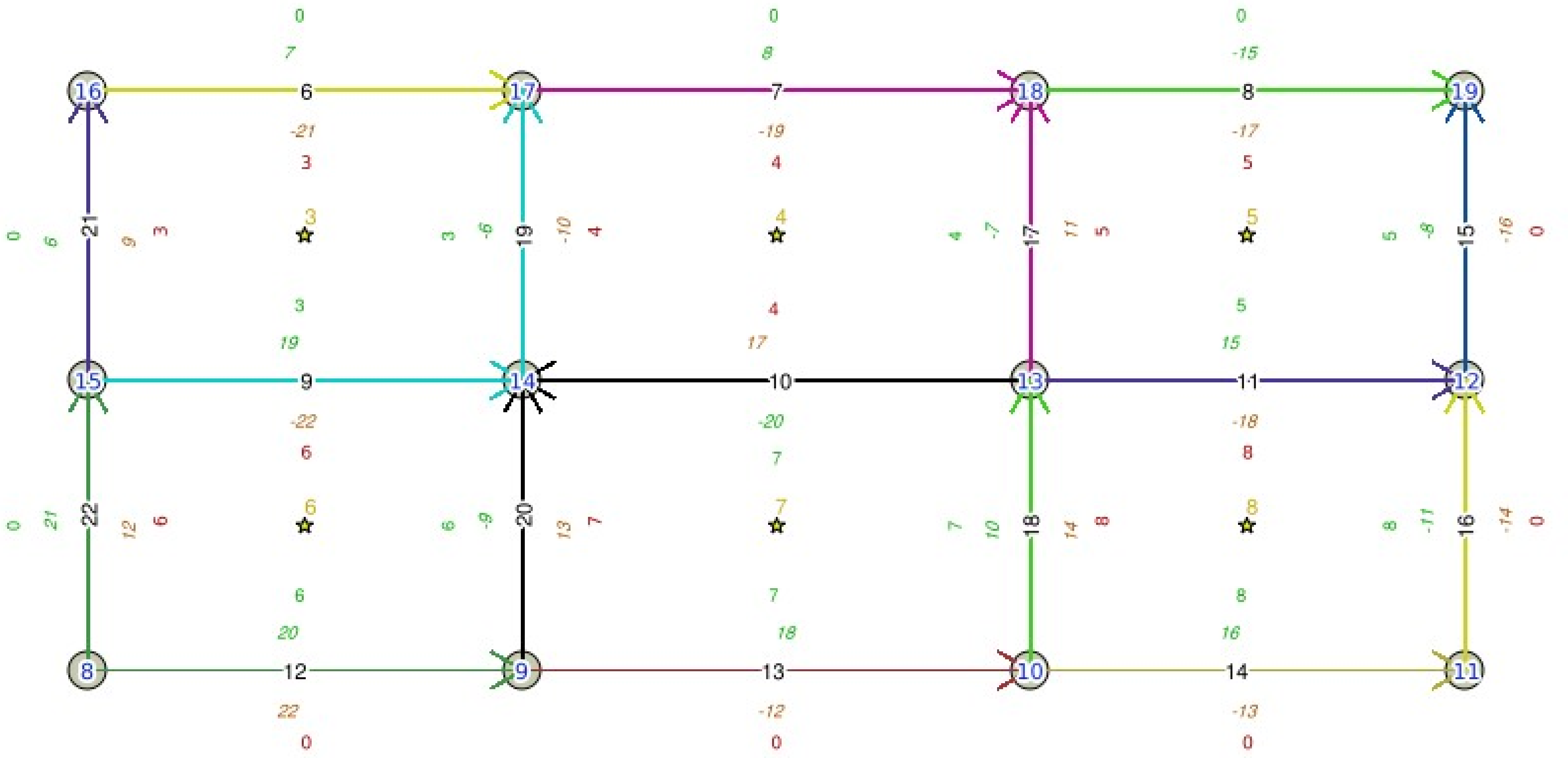


Conceptual model

Next right and left face edges

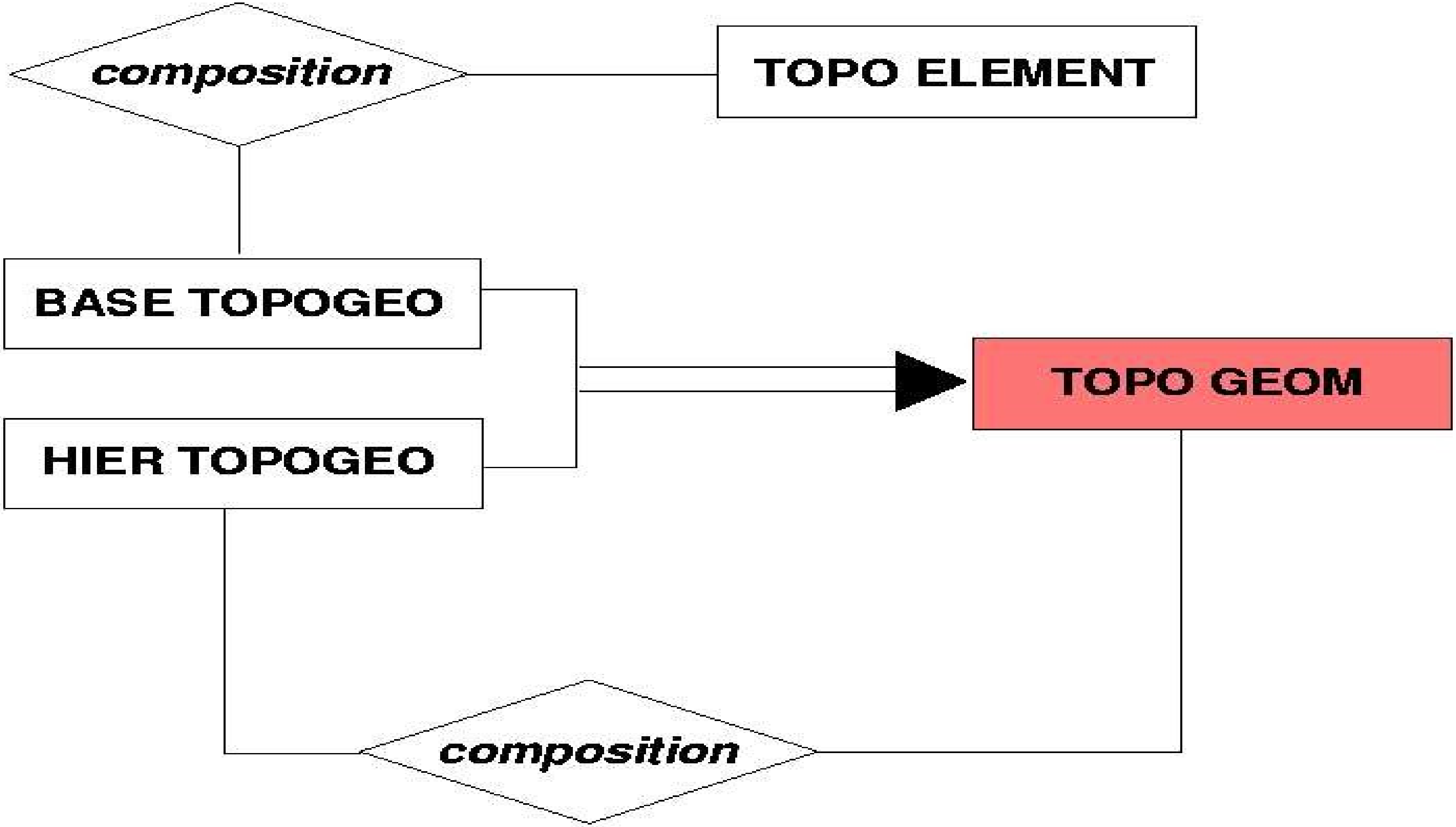


Conceptual model



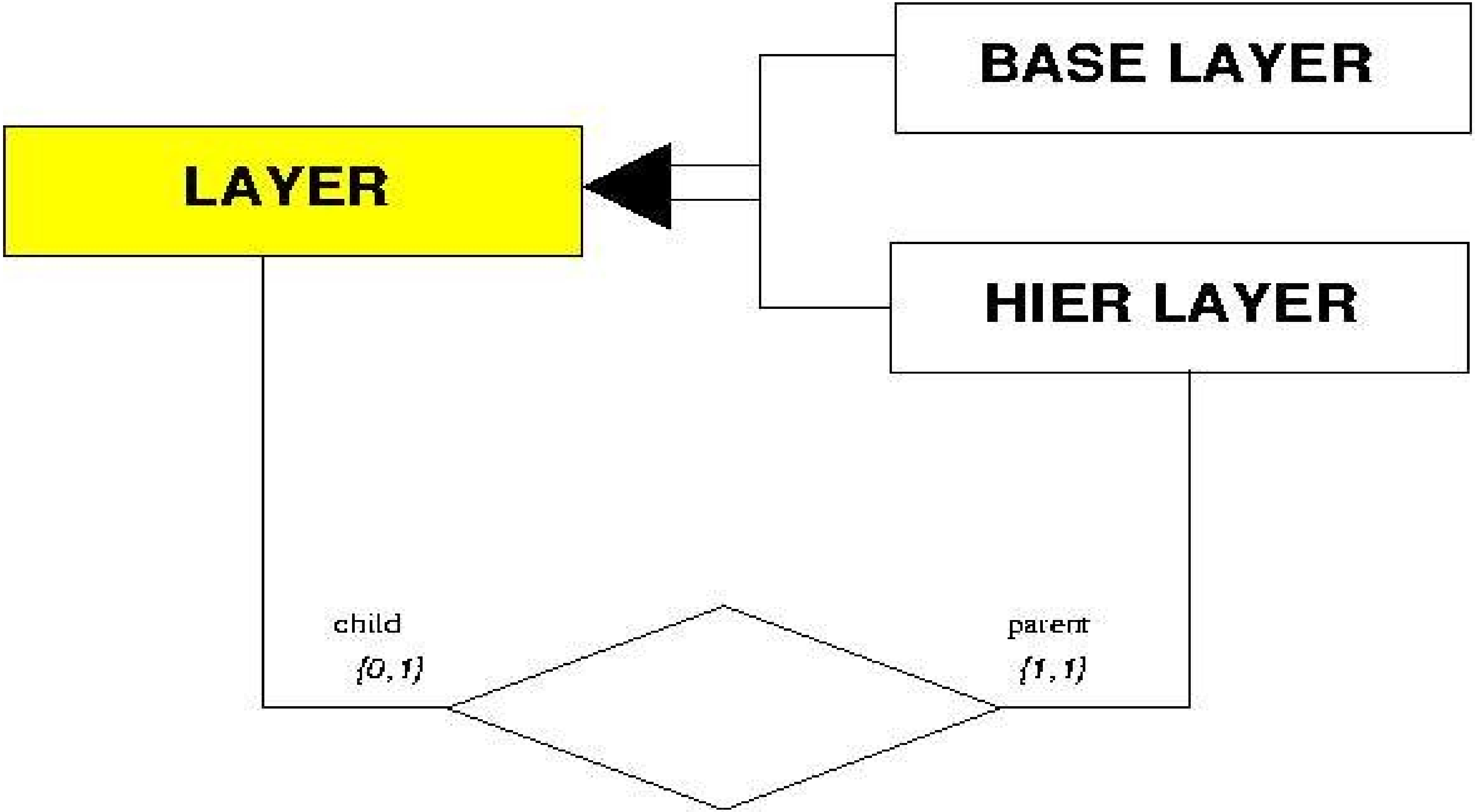
Conceptual model

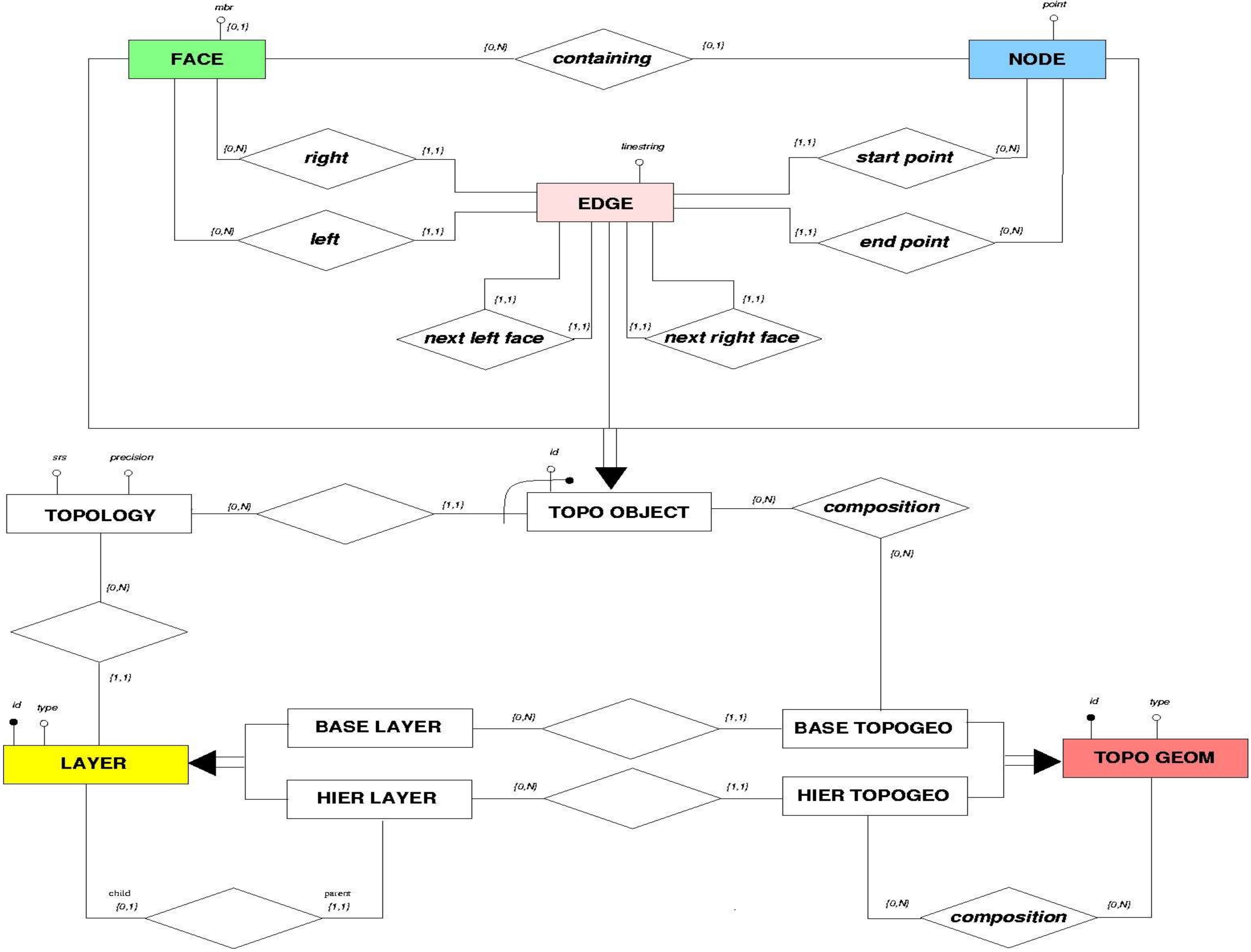
Topo-Geometries



Conceptual model

Layers




mbr
{0,1}
FACE
{0,N}
containing
{0,1}
point
NODE
{0,N}
right
{1,1}
linestring
{1,1}
start point
{0,N}
EDGE
{0,N}
left
{1,1}
{1,1}
end point
{0,N}
{1,1}
{1,1}
{1,1}
{1,1}
next left face
next right face
srs
precision
id
TOPOLOGY
{0,N}
{1,1}
TOPO OBJECT
{0,N}
composition
{0,N}
{0,N}
{1,1}
id
type
id
type
BASE LAYER
{0,N}
{1,1}
BASE TOPOGEO
LAYER
TOPO GEOM
HIER LAYER
{0,N}
{1,1}
HIER TOPOGEO
child
{0,1}
parent
{1,1}
{0,N}
composition
{0,N}

Physical Model

Physical model

- All routines, types and other management objects are stored in the ”topology” schema
- Each topology is stored in its own schema
- Metadata tables with informations about available topologies and topological layers
- TopoGeometry datatype

Topology schema

- <name>.edge
- <name>.face
- <name>.node
- <name>.relation (TopoGeometry composition)

Metadata tables

- topology.topology
- topology.layer

Functions

- Create, destroy, copy, summarize topologies
- Edit topologies (ISO SQL/MM and more)
- Validate topologies
- Define layers (simple and hierarchical)
- Define TopoGeometries (simple and hierarchical)
- Transform TopoGeometries to Geometries
- Topological GML output

What's new ?

Topology in PostGIS 2.0

Package integration

- ./configure --with-topology
- Automated regress testing
- Documentation

SQL/MM topology editing

- ST_AddEdgeNewFaces, ST_AddEdgeModFace
- ST_ModEdgeHeal, ST_NewEdgesHeal
- ST_GetFaceEdges

Topological GML output

- AsGML(TopoGeometry)
  - Xref support

Topology construction primitives

- AddNode
- AddEdge
- AddFace
- Polygonize

Topology management

- TopologySummary
- CopyTopology

Topology inspection

- GetNodeByPoint
- GetEdgeByPoint
- GetFaceByPoint
- GetRingEdges

Performance improvements

- Faster cast from TopoGeometry to Geometry
- Faster topology validation
- Over 30 bugfixes

Other cherries

(loosely related to topologies)

- ST_Split
- ST_Snap
- ST_UnaryUnion
- ST_MakeValid
- ST_SharedPaths

Examples

Topology creation and review

```
=> SET search_path TO topology,public;

=> SELECT CreateTopology('conf');

=> SELECT TopologySummary('conf');
```

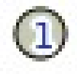
```
 Topology conf (1), SRID -1, precision 0
 0 nodes, 0 edges, 1 faces, 0 topogeoms in 0 layers
```

ISO SQL/MM Topology Population

```
=> SELECT ST_AddIsoNode(
 'conf', 0,
 'POINT(10 20)'
);
```

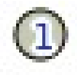
ISO SQL/MM Topology Population

```
=> SELECT ST_AddIsoNode(
 'conf', 0,
 'POINT(10 20)'
);
```



ISO SQL/MM Topology Population

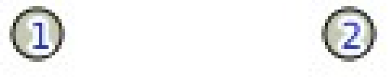
```
=> SELECT ST_AddIsoNode(
 'conf', 0,
 'POINT(10 20)'
);
```



```
=> SELECT ST_AddIsoNode(
 'conf', 0,
 'POINT(100 20)'
);
```

ISO SQL/MM Topology Population

```
=> SELECT ST_AddIsoNode(
 'conf', 0,
 'POINT(10 20)'
);
```



```
=> SELECT ST_AddIsoNode(
 'conf', 0,
 'POINT(100 20)'
);
```

ISO SQL/MM Topology Population

```
=> SELECT ST_AddIsoNode(
 'conf', 0,
 'POINT(10 20)'
);
```

```
=> SELECT ST_AddIsoNode(
 'conf', 0,
 'POINT(100 20)'
);
```

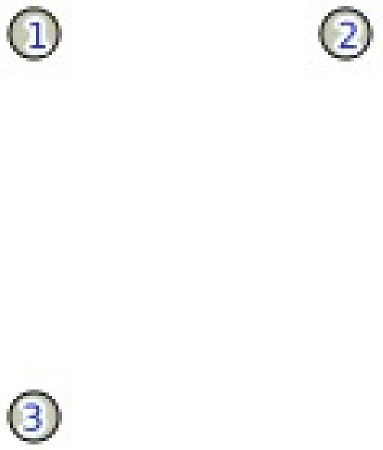
```
=> SELECT ST_AddIsoNode(
 'conf', 0,
 'POINT(10 -90)'
);
```

ISO SQL/MM Topology Population

```
=> SELECT ST_AddIsoNode(
 'conf', 0,
 'POINT(10 20)'
);
```

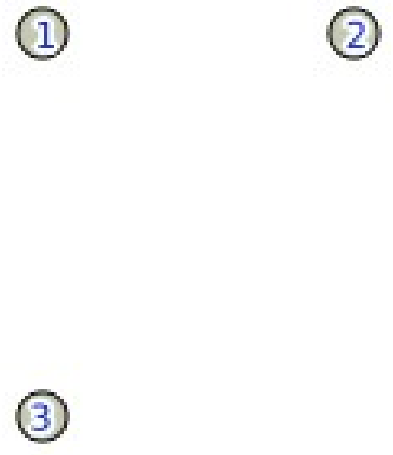
```
=> SELECT ST_AddIsoNode(
 'conf', 0,
 'POINT(100 20)'
);
```

```
=> SELECT ST_AddIsoNode(
 'conf', 0,
 'POINT(10 -90)'
);
```



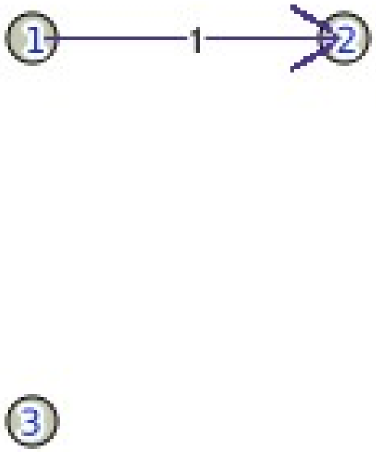
ISO SQL/MM Topology Population

```
=> SELECT
ST_AddEdgeModFace(
 'conf', 1, 2,
 'LINESTRING(10 20,
  100 20)'
);
```



ISO SQL/MM Topology Population

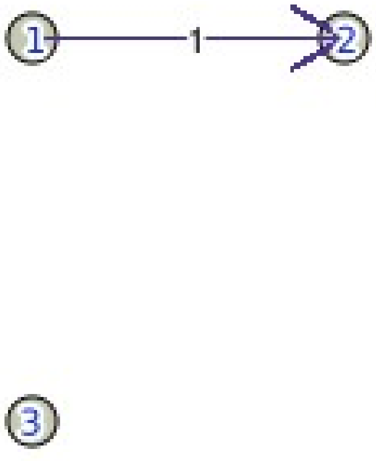
```
=> SELECT
ST_AddEdgeModFace(
 'conf', 1, 2,
 'LINESTRING(10 20,
  100 20)'
);
```



ISO SQL/MM Topology Population

```
=> SELECT
ST_AddEdgeModFace(
 'conf', 1, 2,
 'LINESTRING(10 20,
  100 20)'
);
```

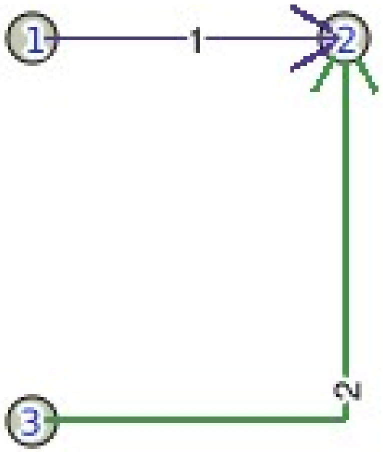
```
=> SELECT
ST_AddEdgeModFace(
 'conf', 3, 2,
 'LINESTRING(10 -90,
 100 -90, 100 20)'
);
```



ISO SQL/MM Topology Population

```
=> SELECT
ST_AddEdgeModFace(
 'conf', 1, 2,
 'LINESTRING(10 20,
  100 20)'
);
```

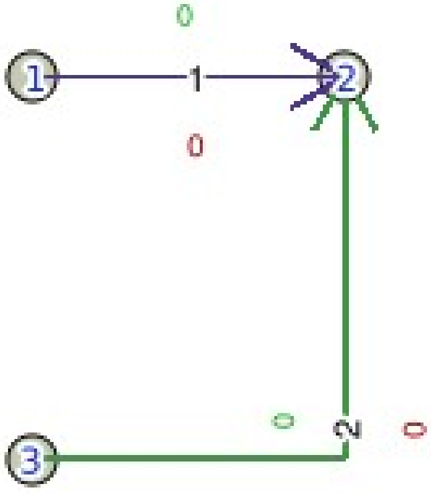
```
=> SELECT
ST_AddEdgeModFace(
 'conf', 3, 2,
 'LINESTRING(10 -90,
 100 -90, 100 20)'
);
```



ISO SQL/MM Topology Population

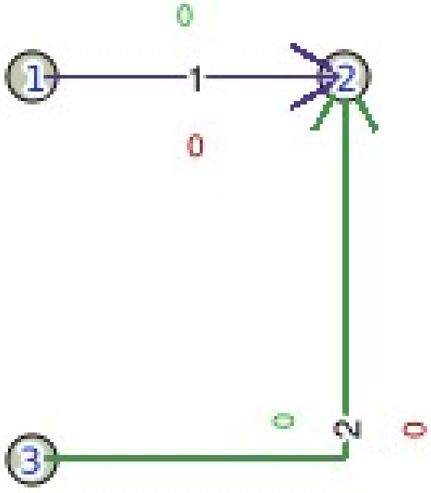
```
=> SELECT
ST_AddEdgeModFace(
 'conf', 1, 2,
 'LINESTRING(10 20,
  100 20)'
);
```

```
=> SELECT
ST_AddEdgeModFace(
 'conf', 3, 2,
 'LINESTRING(10 -90,
 100 -90, 100 20)'
);
```



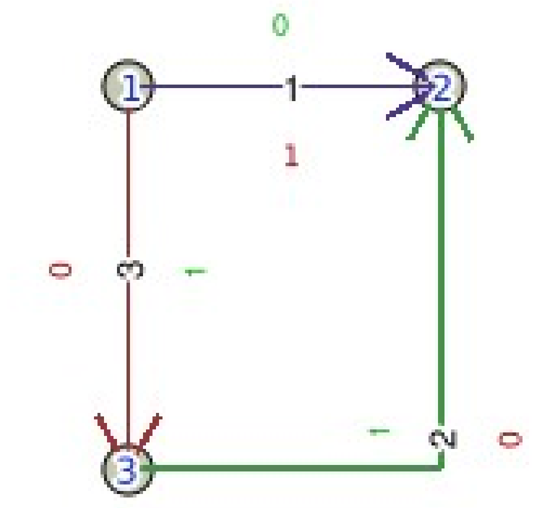
ISO SQL/MM Topology Population

```
=> SELECT
ST_AddEdgeModFace(
 'conf', 1, 3,
 'LINESTRING(10 20,
 10 -90)'
);
```



ISO SQL/MM Topology Population

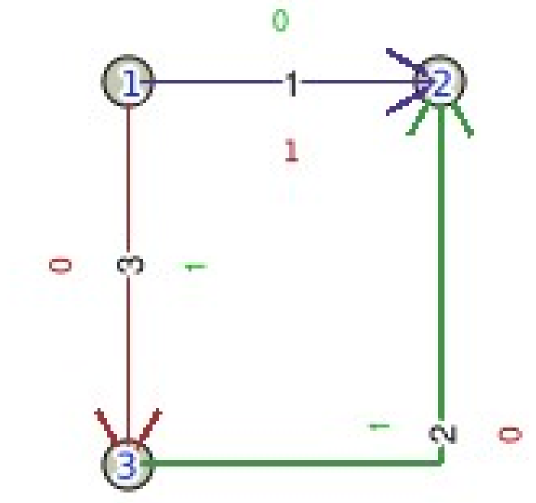
```
=> SELECT
ST_AddEdgeModFace(
 'conf', 1, 3,
 'LINESTRING(10 20,
 10 -90)'
);
```



ISO SQL/MM Topology Population

```
=> SELECT
ST_AddEdgeModFace(
 'conf', 1, 3,
 'LINESTRING(10 20,
 10 -90)'
);
```

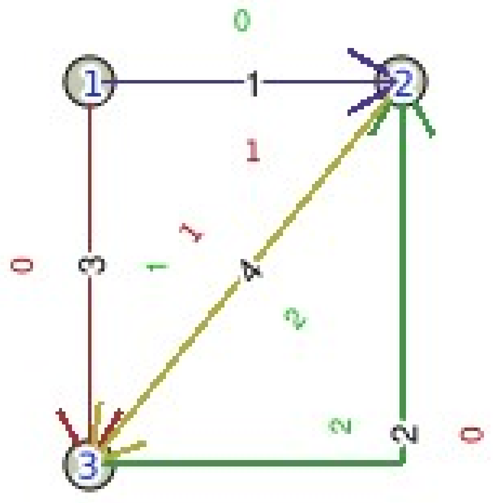
```
=> SELECT
ST_AddEdgeModFace(
 'conf', 2, 3,
 'LINESTRING(100 20,
 10 -90)'
);
```



ISO SQL/MM Topology Population

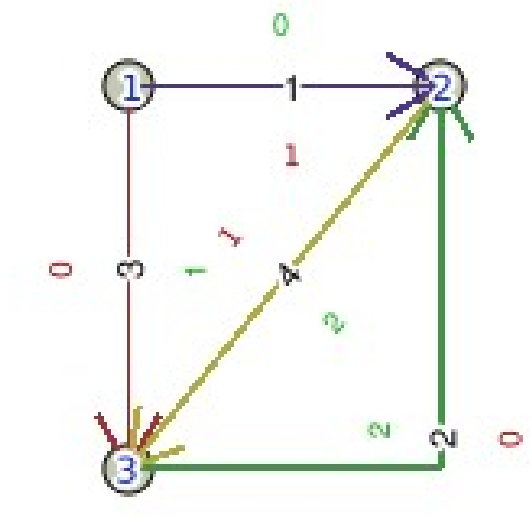
```
=> SELECT
ST_AddEdgeModFace(
 'conf', 1, 3,
 'LINESTRING(10 20,
 10 -90)'
);
```

```
=> SELECT
ST_AddEdgeModFace(
 'conf', 2, 3,
 'LINESTRING(100 20,
 10 -90)'
);
```



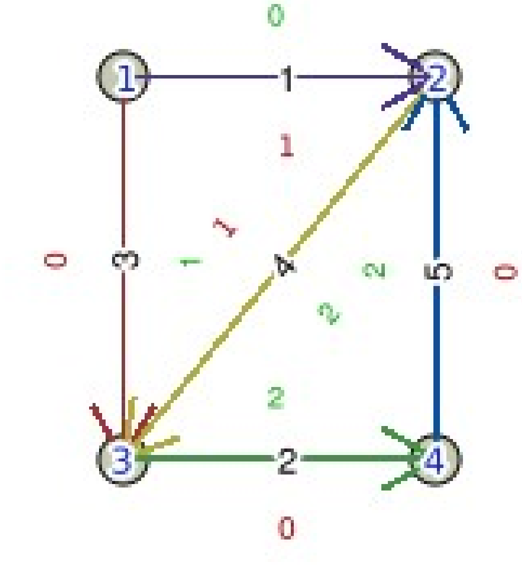
ISO SQL/MM Topology Population

```
=> SELECT
ST_ModEdgeSplit(
 'conf', 2,
 'POINT(100 -90)'
);
```



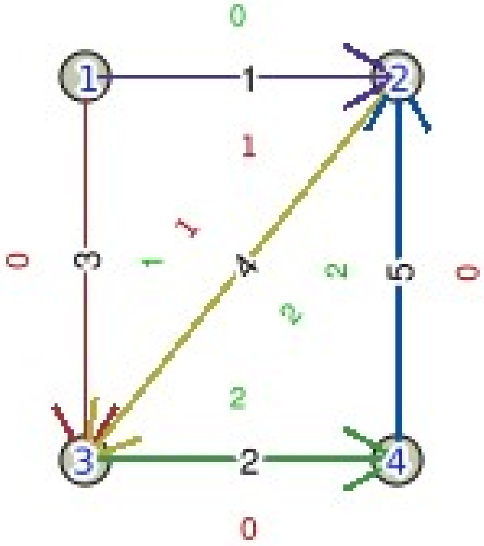
ISO SQL/MM Topology Population

```
=> SELECT
ST_ModEdgeSplit(
 'conf', 2,
 'POINT(100 -90)'
);
```



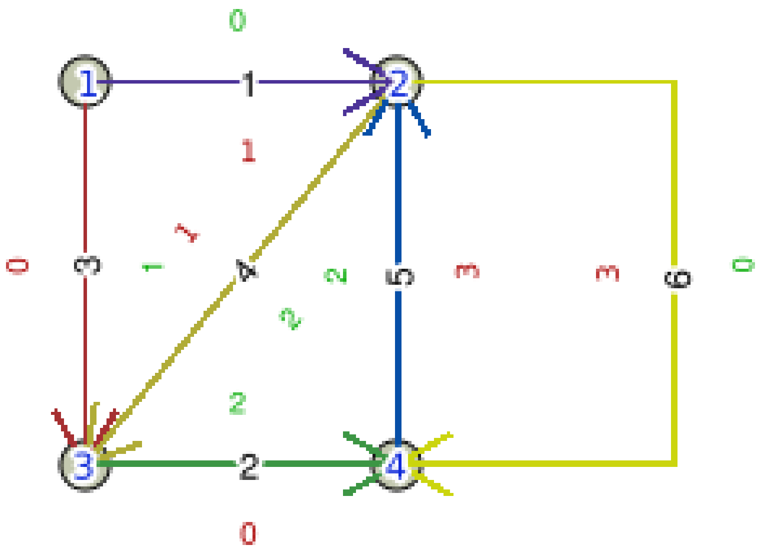
ISO SQL/MM Topology Population

```
=> SELECT
ST_ModEdgeSplit(
 'conf', 2,
 'POINT(100 -90)'
);


=> SELECT
ST_AddEdgeModFace(
 'conf', 2, 4,
 'LINESTRING(100 20,
 180 20, 180 -90,
 100 -90)');
```



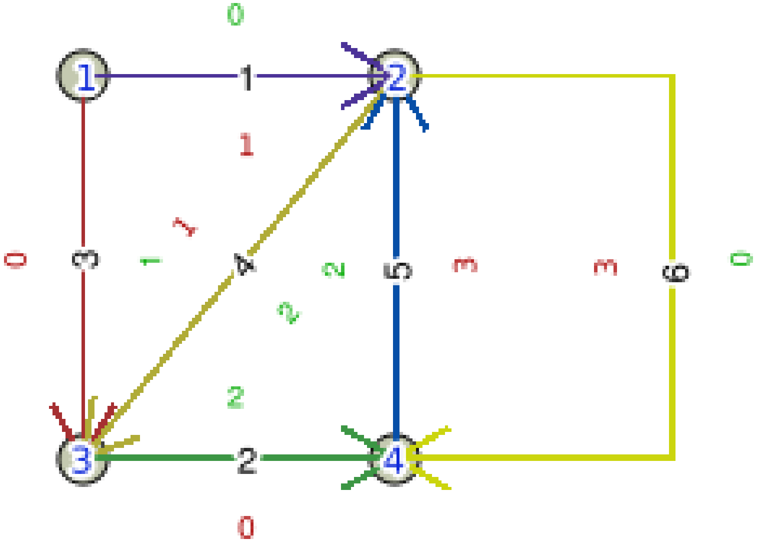
ISO SQL/MM Topology Population

```
=> SELECT
ST_ModEdgeSplit(
 'conf', 2,
 'POINT(100 -90)'
);


=> SELECT
ST_AddEdgeModFace(
 'conf', 2, 4,
 'LINESTRING(100 20,
 180 20, 180 -90,
 100 -90)');
```



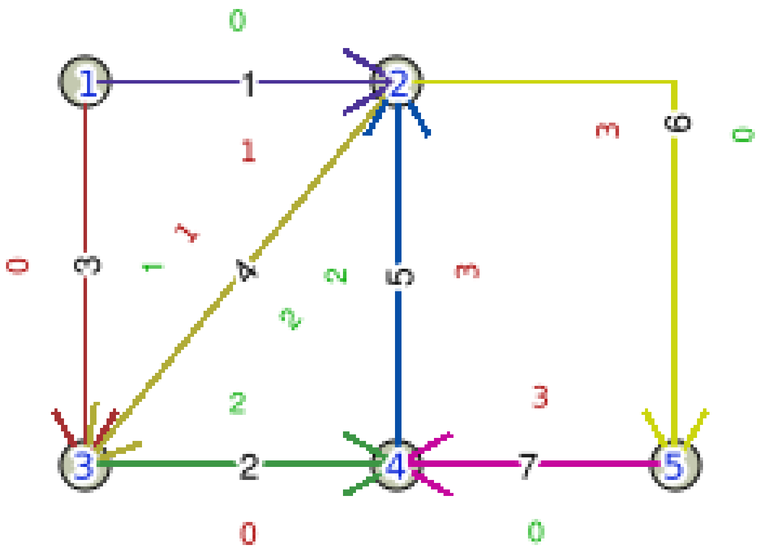
ISO SQL/MM Topology Population

```
=> SELECT
ST_ModEdgeSplit(
 'conf', 6,
 'POINT(180 -90)'
);
```



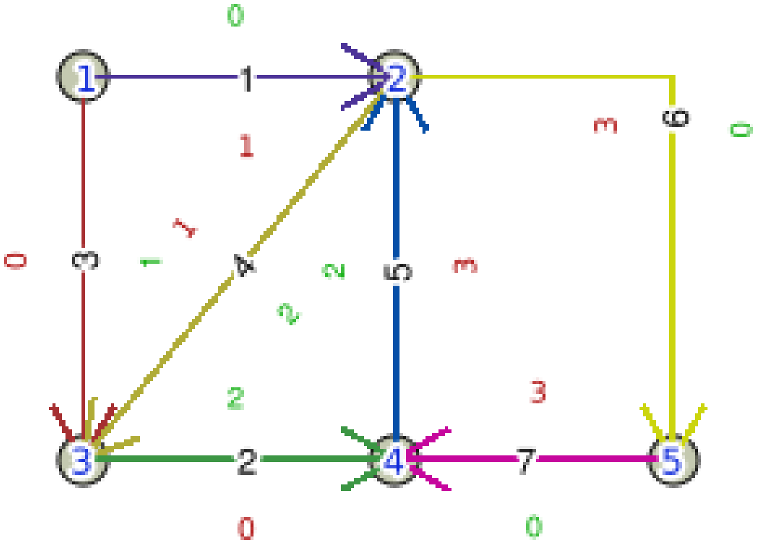
ISO SQL/MM Topology Population

```
=> SELECT
ST_ModEdgeSplit(
 'conf', 6,
 'POINT(180 -90)'
);
```



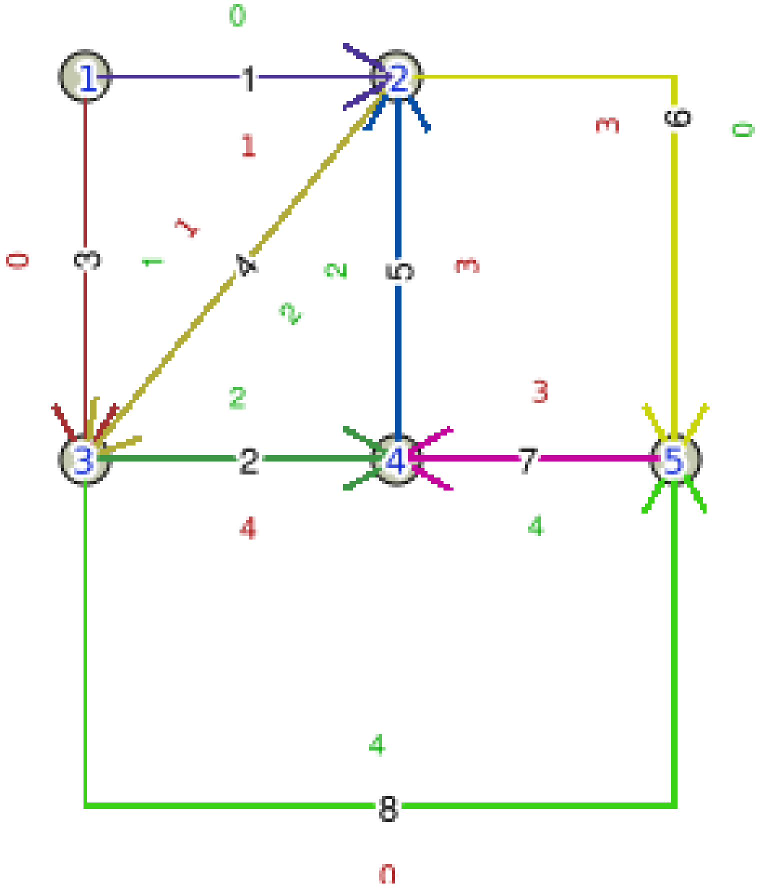
ISO SQL/MM Topology Population

```
=> SELECT
ST_ModEdgeSplit(
 'conf', 6,
 'POINT(180 -90)'
);


=> SELECT
ST_AddEdgeModFace(
 'conf', 3, 5,
 'LINESTRING(10 -90,
 10 -190, 180 -190,
 180 -90)');
```



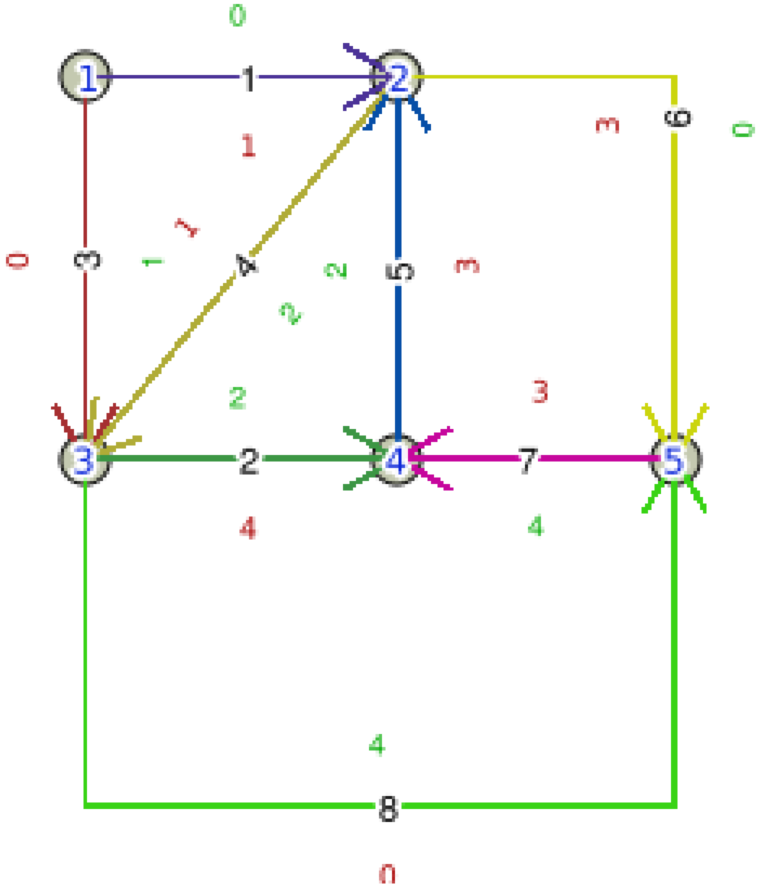
ISO SQL/MM Topology Population

```
=> SELECT
ST_ModEdgeSplit(
 'conf', 6,
 'POINT(180 -90)'
);


=> SELECT
ST_AddEdgeModFace(
 'conf', 3, 5,
 'LINESTRING(10 -90,
 10 -190, 180 -190,
 180 -90)');
```



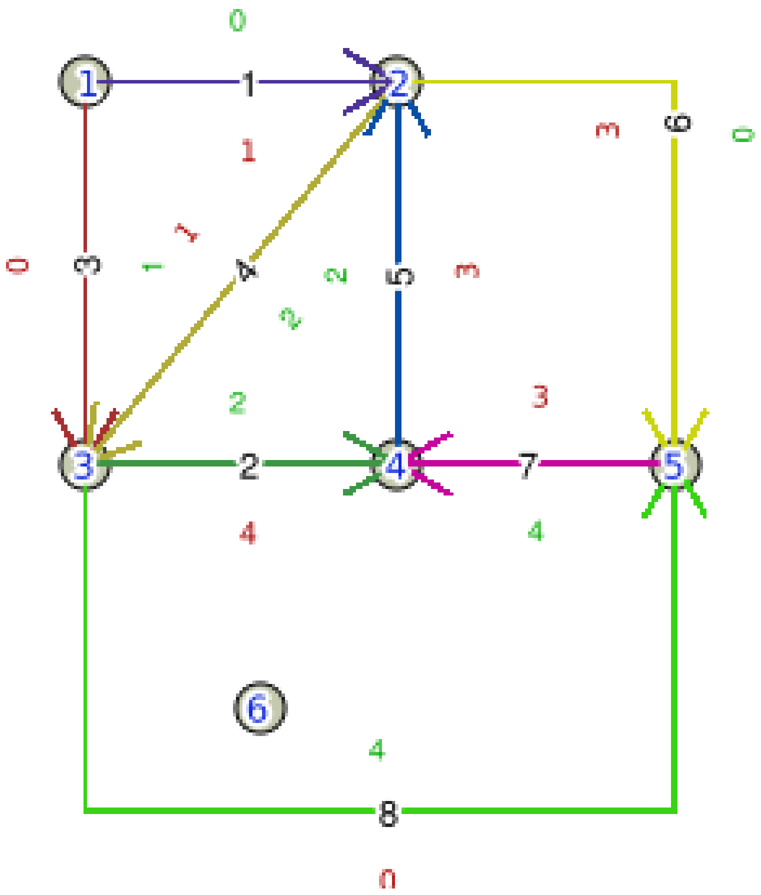
ISO SQL/MM Topology Population

```
=> SELECT ST_AddIsoNode(
 'conf', 4,
 'POINT(60 -160)'
);
```



ISO SQL/MM Topology Population

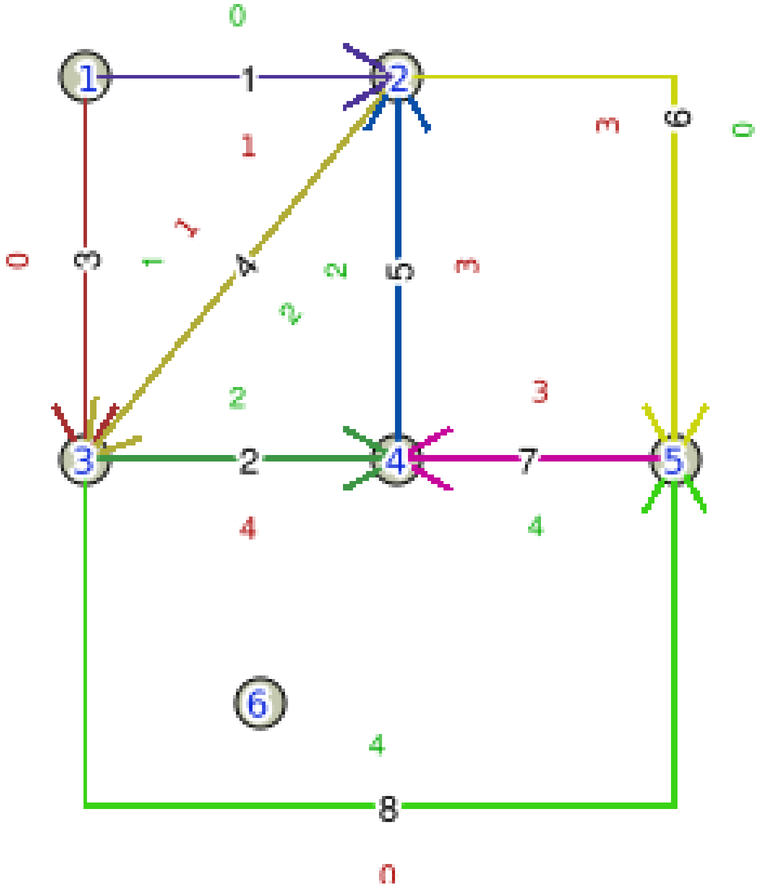
```
=> SELECT ST_AddIsoNode(
 'conf', 4,
 'POINT(60 -160)'
);
```



ISO SQL/MM Topology Population

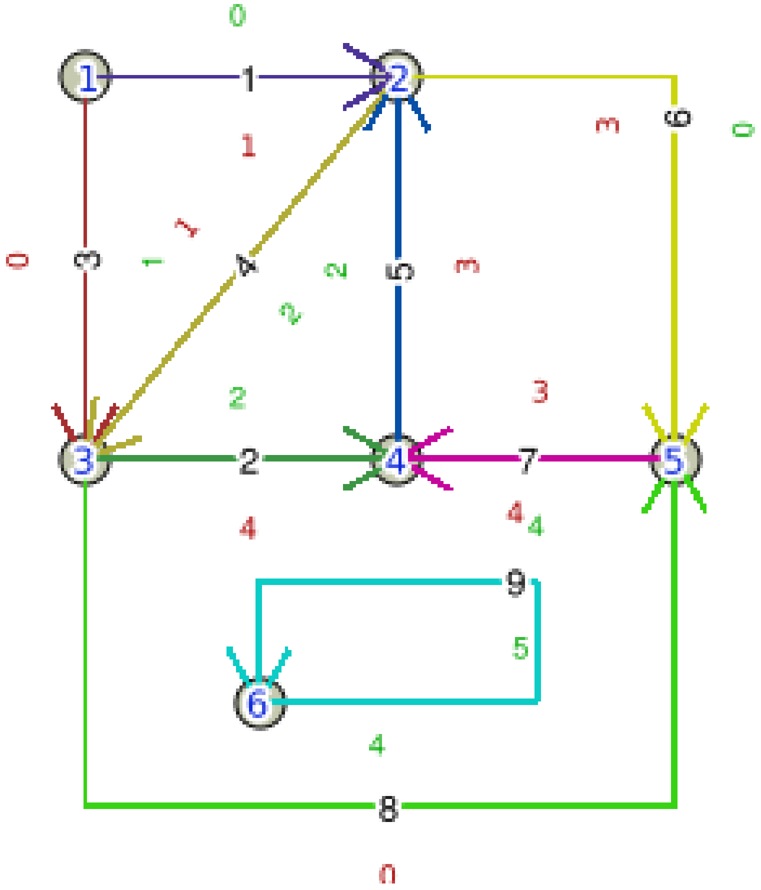
```
=> SELECT ST_AddIsoNode(
 'conf', 4,
 'POINT(60 -160)'
);
```

```
=> SELECT
ST_AddEdgeModFace(
 'conf', 6, 6,
 'LINESTRING(60 -160,
 140 -160, 140 -125,
 60 -125, 60 -160)');
```



ISO SQL/MM Topology Population

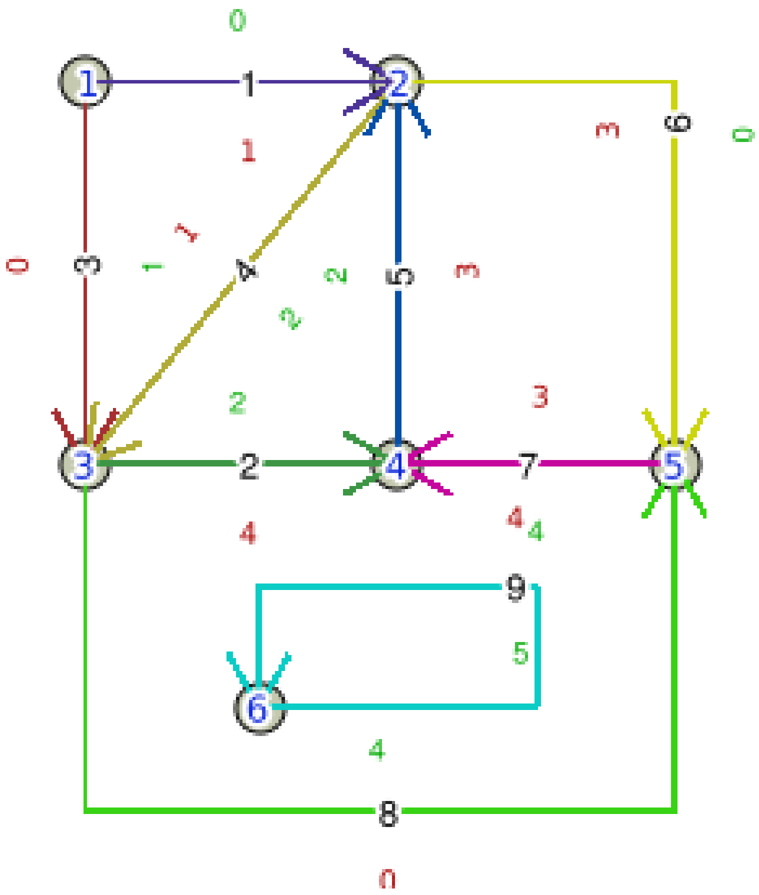
```
=> SELECT ST_AddIsoNode(
 'conf', 4,
 'POINT(60 -160)'
);


=> SELECT
ST_AddEdgeModFace(
 'conf', 6, 6,
 'LINESTRING(60 -160,
 140 -160, 140 -125,
 60 -125, 60 -160)');
```



ISO SQL/MM Topology Population

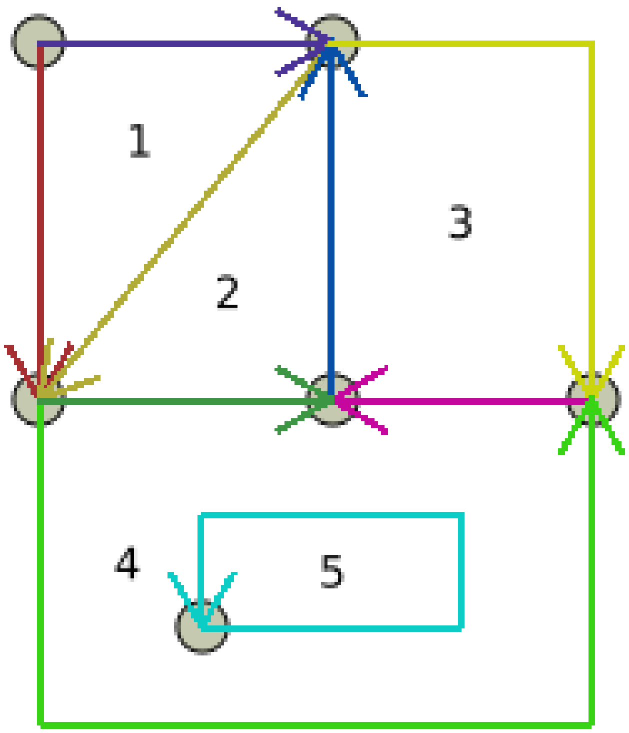
```
=> SELECT
TopologySummary('conf');
```

```
 Topology conf (1),
 SRID -1, precision 0

 6 nodes, 9 edges, 6 faces,
 0 topogeoms in 0 layers
```



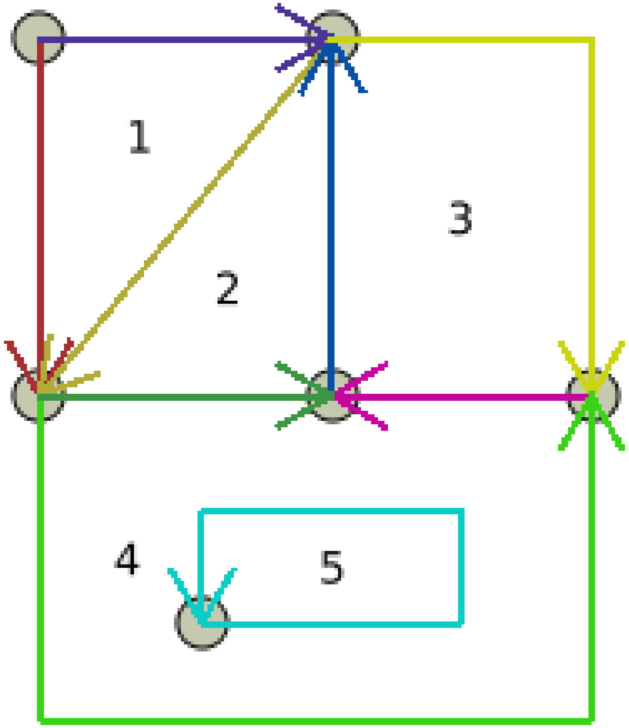
TopoGeometry: areal

```
=> CREATE TABLE
 conf.fa(id SERIAL
 PRIMARY KEY);


=> SELECT
AddTopoGeometryColumn(
'conf', 'conf', 'fa',
'g', 'POLYGON');
```



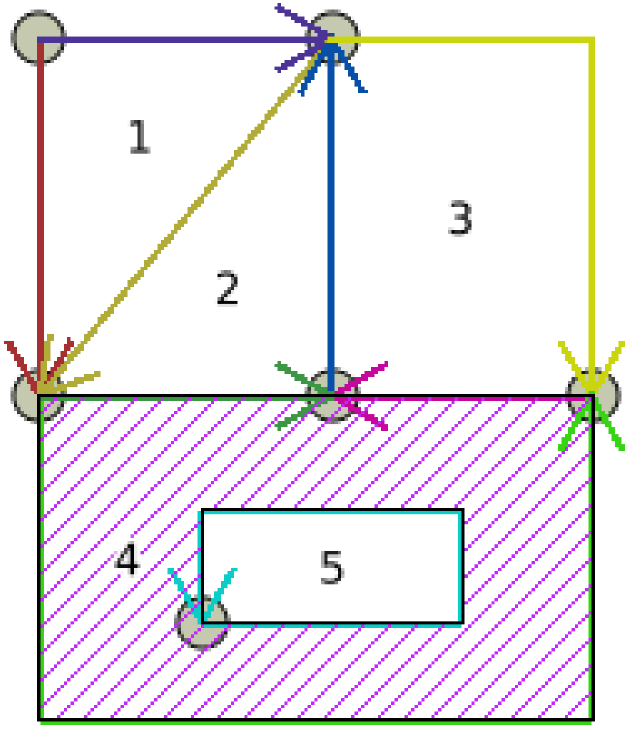
TopoGeometry: areal

```
=> INSERT
 INTO conf.fa (g)
 VALUES (
  CreateTopoGeom(
   'conf', -- Topo name
   3, -- type (areal)
   1, -- layer id
   '{{4,3}}' -- face 4
  )
 );
```



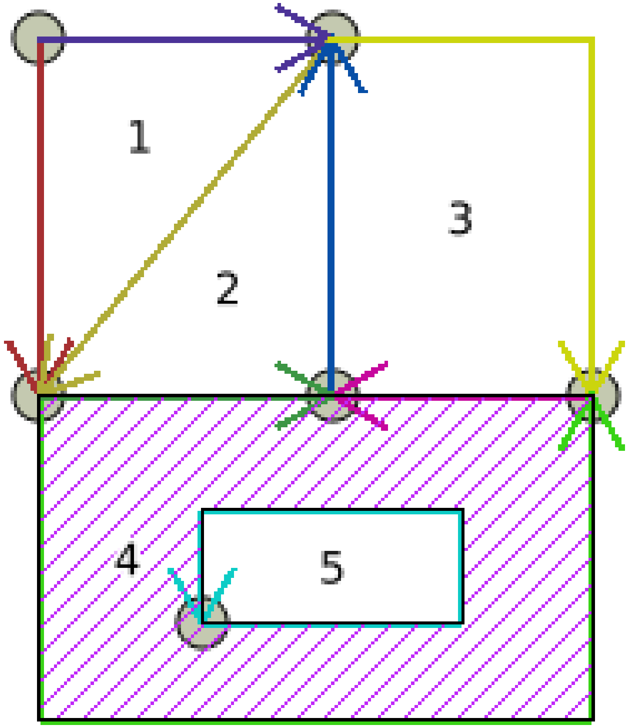
TopoGeometry: areal

```
=> INSERT
 INTO conf.fa (g)
 VALUES (
  CreateTopoGeom(
   'conf', -- Topo name
   3, -- type (areal)
   1, -- layer id
   '{{4,3}}' -- face 4
  )
 );
```



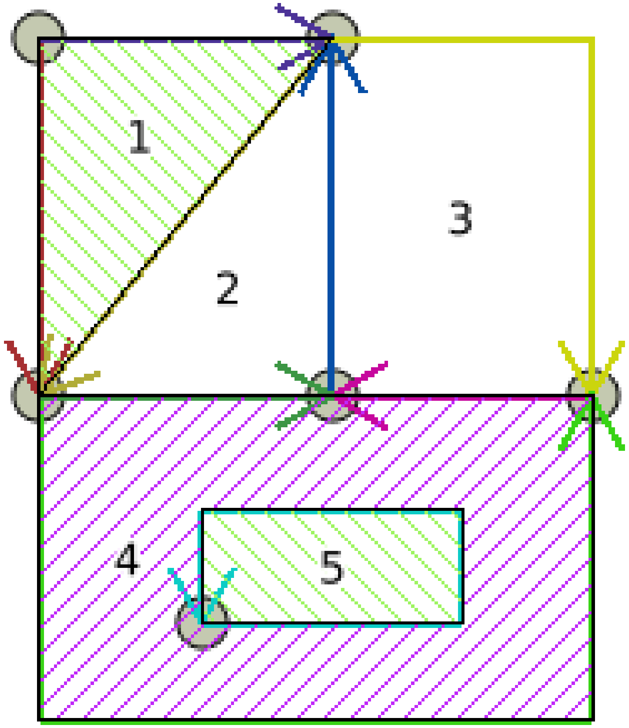
TopoGeometry: areal

```
=> INSERT
 INTO conf.fa (g)
 VALUES (
  CreateTopoGeom(
   'conf', 3, 1,
   -- faces 1 and 5
   '{{1,3},{5,3}}'
  )
 );
```



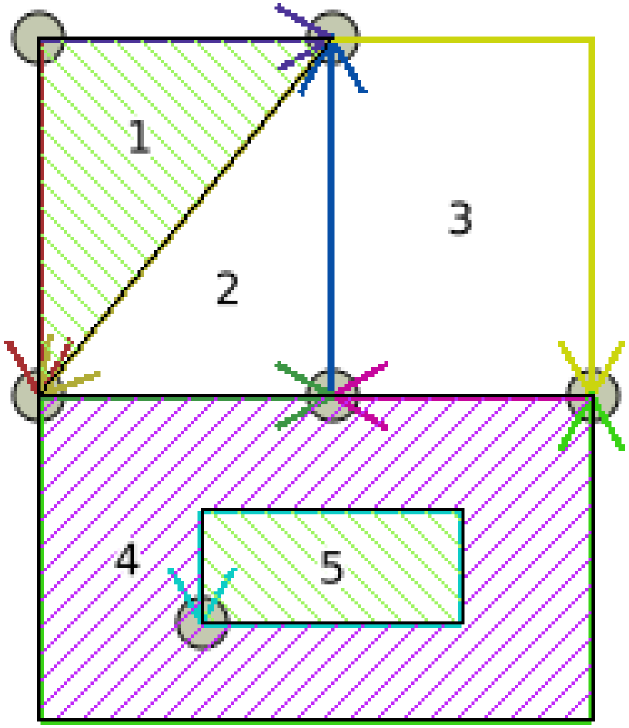
TopoGeometry: areal

```
=> INSERT
 INTO conf.fa (g)
 VALUES (
  CreateTopoGeom(
   'conf', 3, 1,
   -- faces 1 and 5
   '{{1,3},{5,3}}'
  )
 );
```



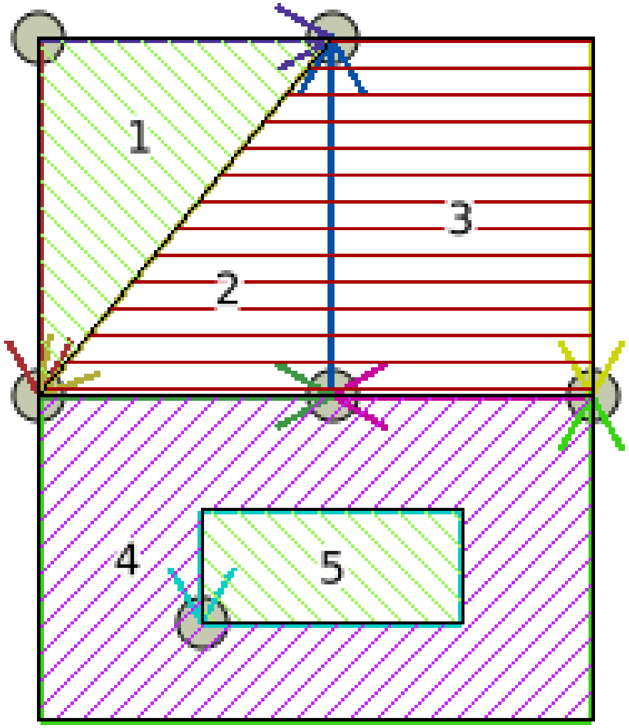
TopoGeometry: areal

```
=> INSERT
 INTO conf.fa (g)
 VALUES (
  CreateTopoGeom(
   'conf', 3, 1,
   -- faces 2 and 3
   '{{2,3},{3,3}}'
  )
 );
```



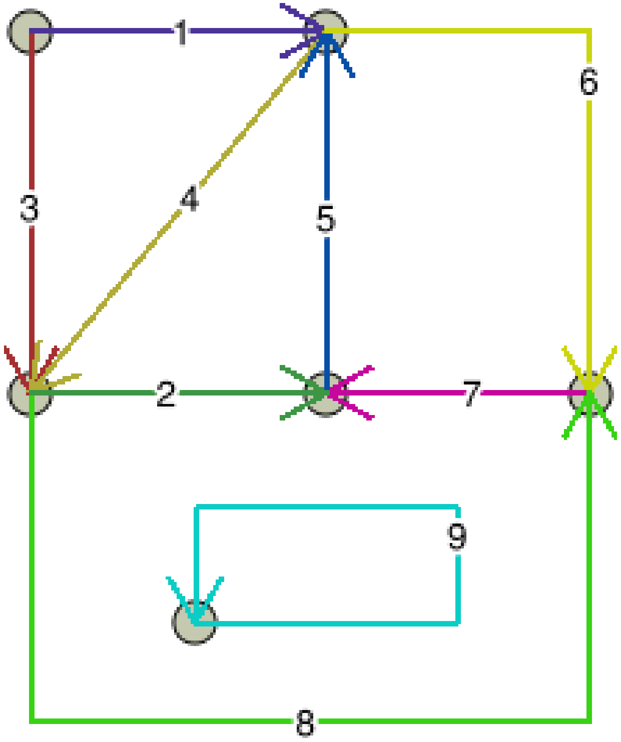
TopoGeometry: areal

```
=> INSERT
 INTO conf.fa (g)
 VALUES (
  CreateTopoGeom(
   'conf', 3, 1,
   -- faces 2 and 3
   '{{2,3},{3,3}}'
  )
 );
```



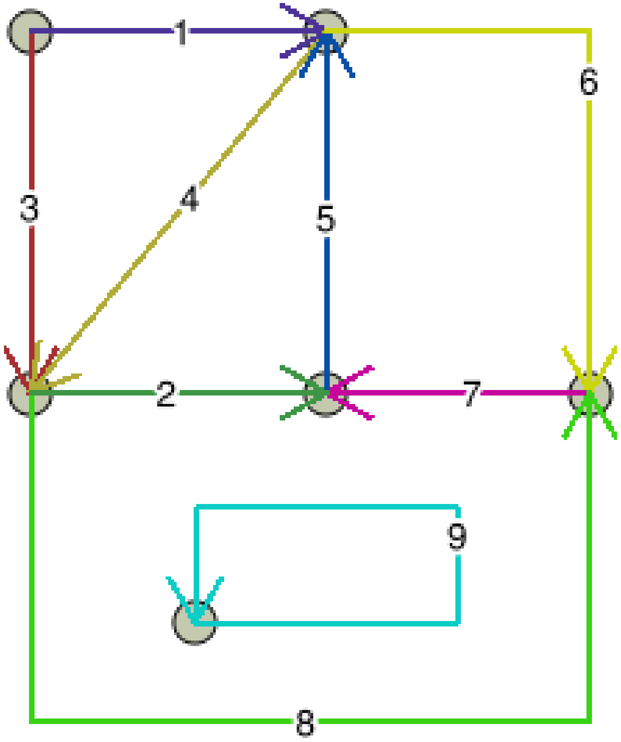
TopoGeometry: lineal

```
=> CREATE TABLE
 conf.fl(id SERIAL
 PRIMARY KEY);


=> SELECT
AddTopoGeometryColumn(
'conf', 'conf', 'fl',
'g', 'LINE');
```



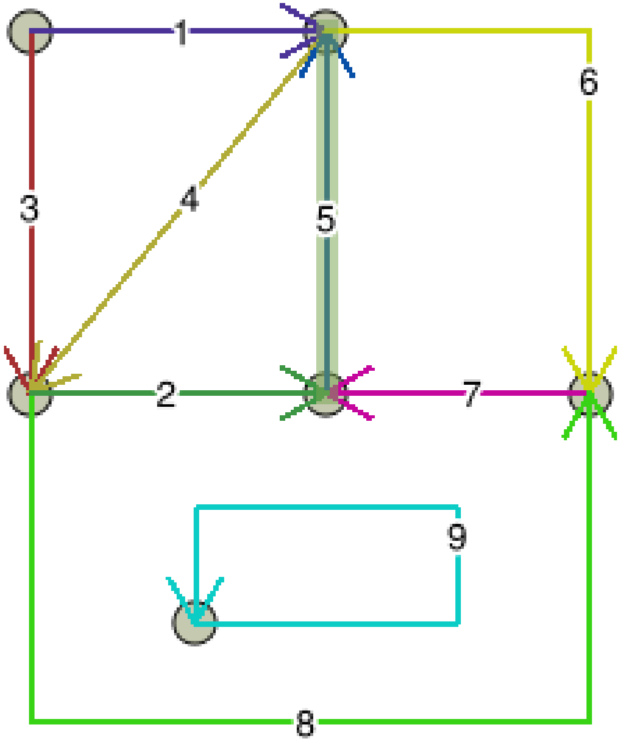
TopoGeometry: lineal

```
=> INSERT
 INTO conf.fl (g)
 VALUES (
  CreateTopoGeom(
   'conf', -- Topo name
   2, -- type (lineal)
   2, -- layer id
   '{{5,2}}' -- edge 5
  )
 );
```



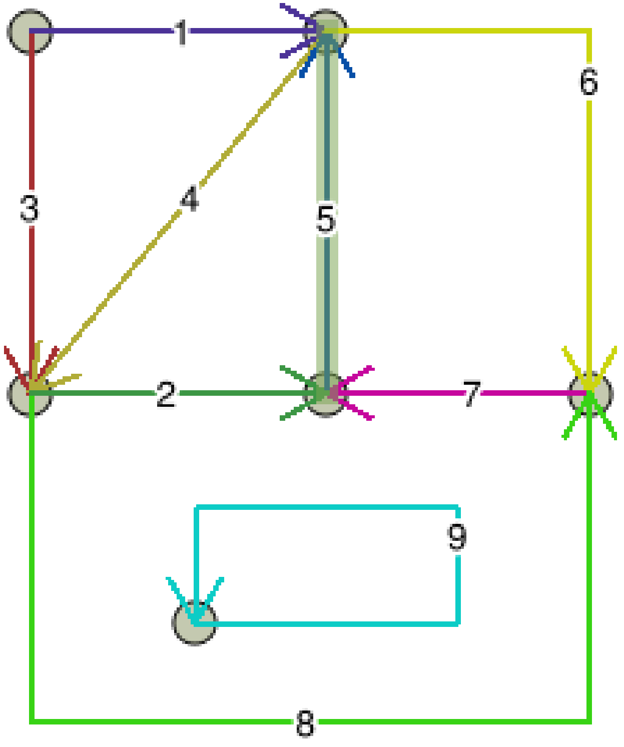
TopoGeometry: lineal

```
=> INSERT
 INTO conf.fl (g)
 VALUES (
  CreateTopoGeom(
   'conf', -- Topo name
   2, -- type (lineal)
   2, -- layer id
   '{{5,2}}' -- edge 5
  )
 );
```



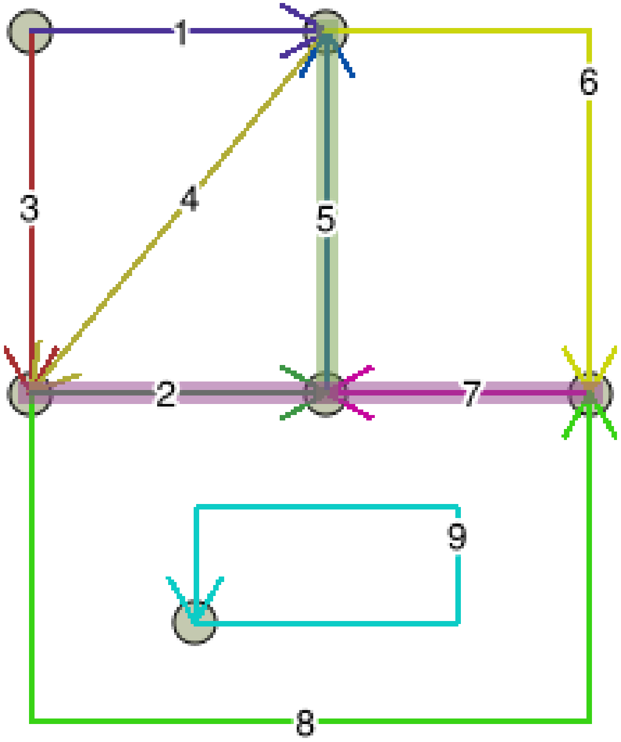
TopoGeometry: lineal

```
=> INSERT
 INTO conf.fl (g)
 VALUES (
  CreateTopoGeom(
   'conf', 2, 2,
   -- edges 2 and 7
   '{{2,2},{7,2}}'
  )
 );
```



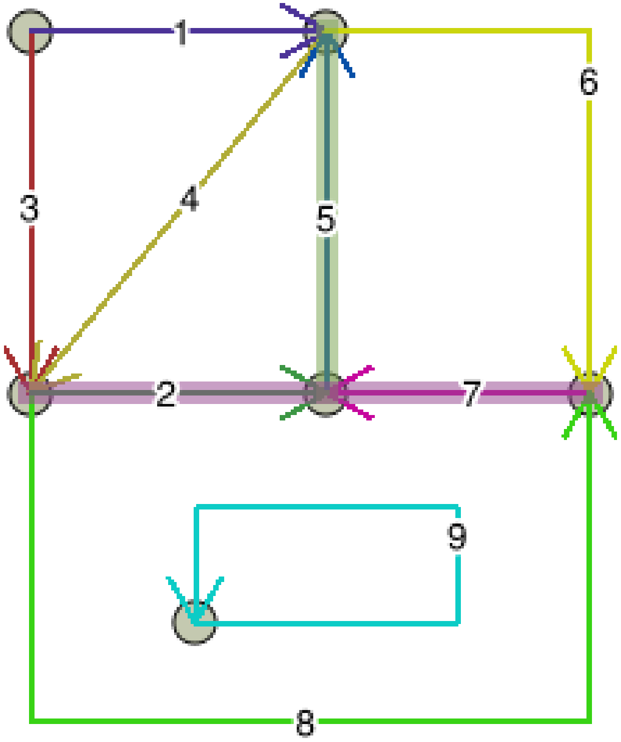
TopoGeometry: lineal

```
=> INSERT
 INTO conf.fl (g)
 VALUES (
  CreateTopoGeom(
   'conf', 2, 2,
   -- edges 2 and 7
   '{{2,2},{7,2}}'
  )
 );
```



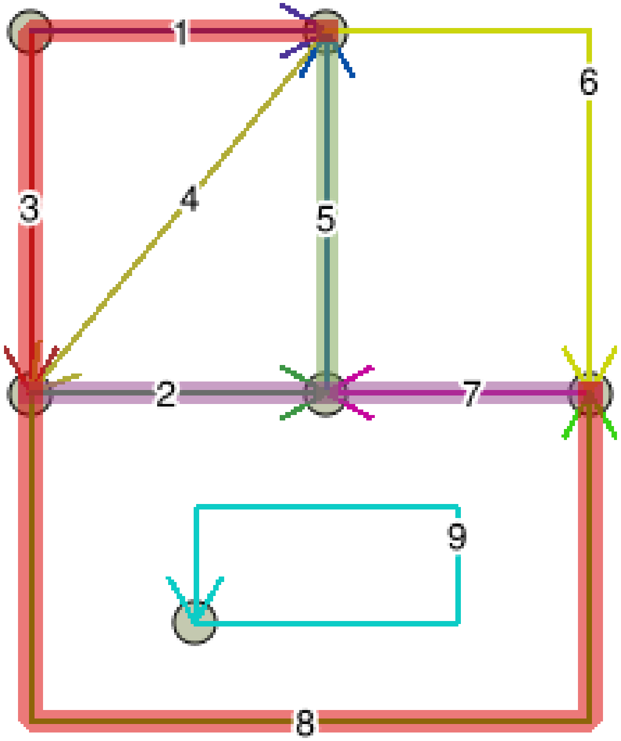
TopoGeometry: lineal

```
=> INSERT
 INTO conf.fl (g)
 VALUES (
  CreateTopoGeom(
   'conf', 2, 2,
   -- edges 3, 1 and 8
   '{{3,2},{1,2},{8,2}}'
  )
 );
```



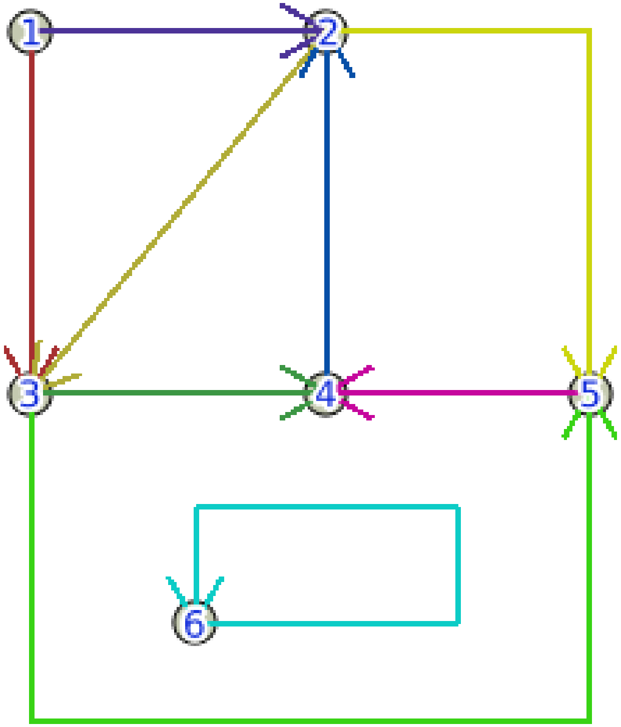
TopoGeometry: lineal

```
=> INSERT
 INTO conf.fl (g)
 VALUES (
  CreateTopoGeom(
   'conf', 2, 2,
   -- edges 3, 1 and 8
   '{{3,2},{1,2},{8,2}}'
  )
 );
```



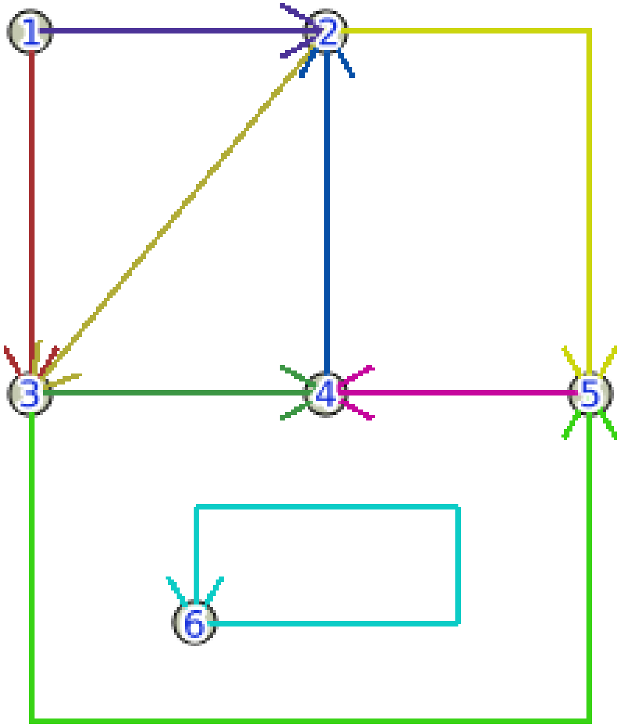
TopoGeometry: puntal

```
=> CREATE TABLE
 conf.fp(id SERIAL
 PRIMARY KEY);


=> SELECT
AddTopoGeometryColumn(
'conf', 'conf', 'fp',
'g', 'POINT');
```



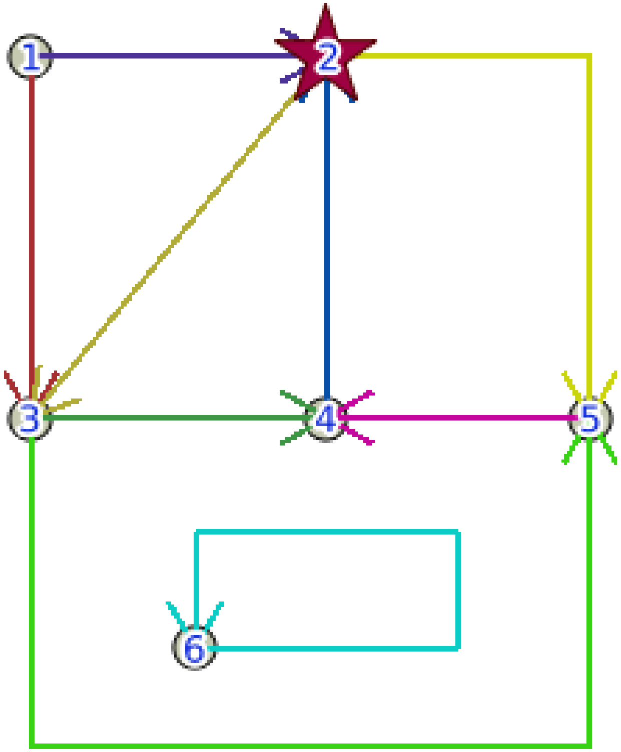
TopoGeometry: puntal

```
=> INSERT
 INTO conf.fp (g)
 VALUES (
  CreateTopoGeom(
   'conf', -- Topo name
   1, -- type (puntal)
   3, -- layer id
   '{{2,1}}' -- node 2
  )
 );
```



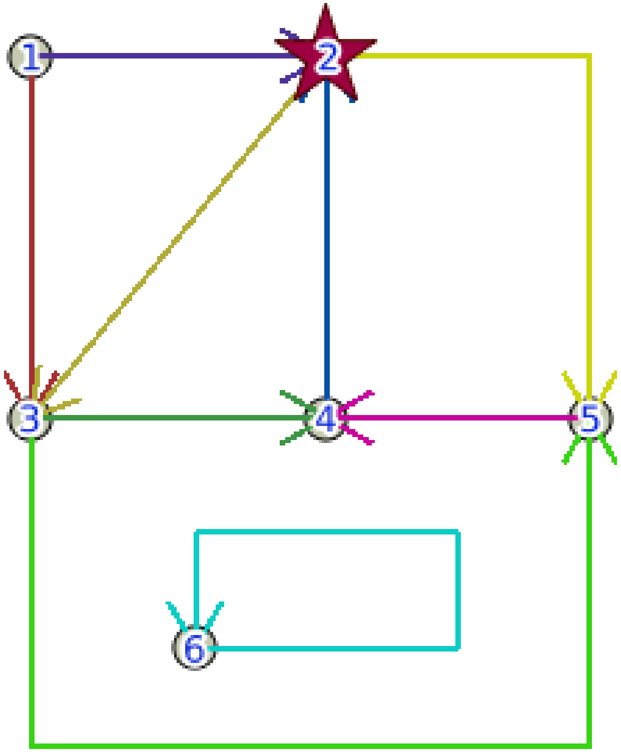
TopoGeometry: puntal

```
=> INSERT
 INTO conf.fp (g)
 VALUES (
  CreateTopoGeom(
   'conf', -- Topo name
   1, -- type (puntal)
   3, -- layer id
   '{{2,1}}' -- node 2
  )
 );
```



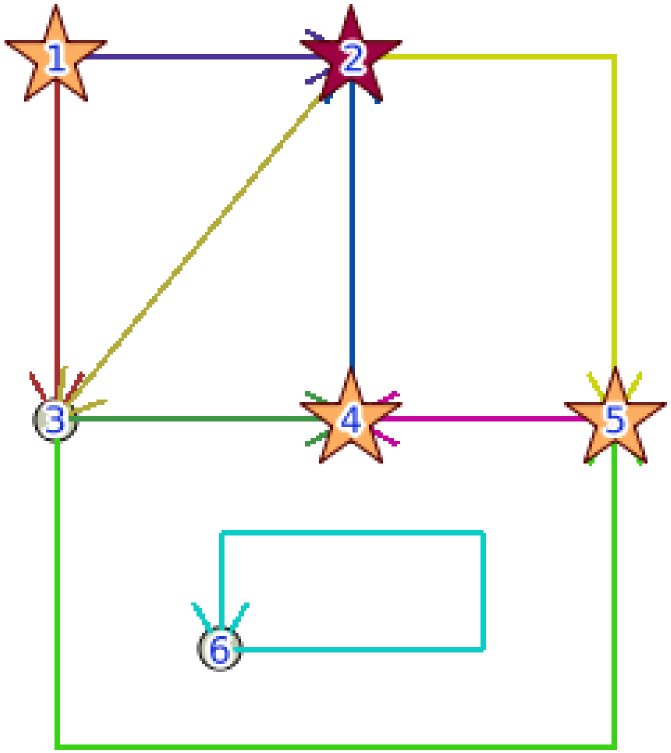
TopoGeometry: puntal

```
=> INSERT
 INTO conf.fp (g)
 VALUES (
  CreateTopoGeom(
   'conf', 1, 3,
   -- nodes 1, 4 and 5
   '{{1,1},{4,1},{5,1}}'
  )
 );
```



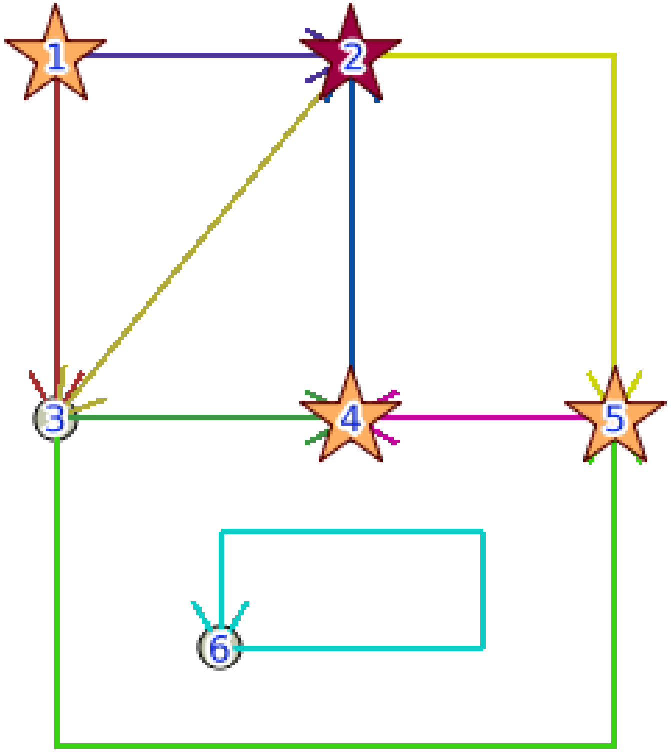
TopoGeometry: puntal

```
=> INSERT
 INTO conf.fp (g)
 VALUES (
  CreateTopoGeom(
   'conf', 1, 3,
   -- nodes 1, 4 and 5
   '{{1,1},{4,1},{5,1}}'
  )
 );
```



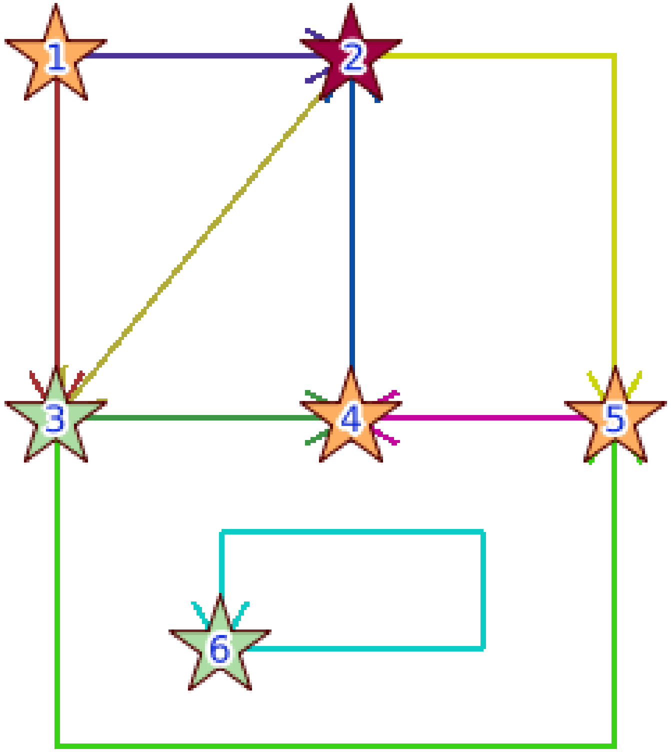
TopoGeometry: puntal

```
=> INSERT
 INTO conf.fp (g)
 VALUES (
  CreateTopoGeom(
   'conf', 1, 3,
   -- nodes 3 and 6
   '{{3,1},{6,1}}'
  )
 );
```



TopoGeometry: puntal

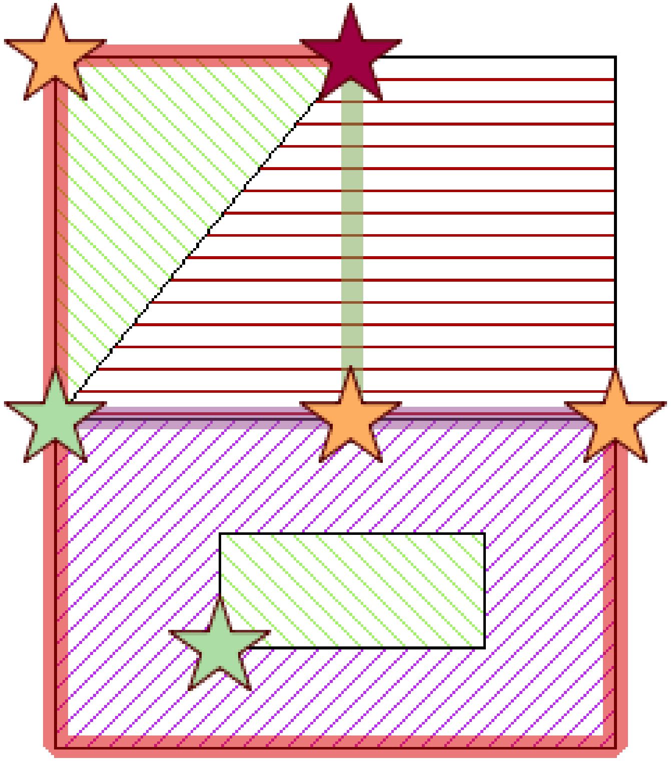
```
=> INSERT
 INTO conf.fp (g)
 VALUES (
  CreateTopoGeom(
   'conf', 1, 3,
   -- nodes 3 and 6
   '{{3,1},{6,1}}'
  )
 );
```



TopoGeometry layers summary

```
=> SELECT
TopologySummary('conf');
```

```
 Topology conf (1),
 SRID -1, precision 0

 6 nodes, 9 edges, 6 faces,
 9 topogeoms in 3 layers

 Layer 1, type Polygonal (3),
 3 topogeoms Deploy: conf.fa.g

 Layer 2, type Lineal (2),
 3 topogeoms Deploy: conf.fl.g

 Layer 3, type Puntal (1),
 3 topogeoms Deploy: conf.fp.g
```



Future developments

(funding opportunities)

Future developments
(funding opportunities)

- Missing ISO SQL/MM editing functions:
  - ST_CreateTopoGeo
  - ST_RemEdgeNewFace, ST_RemEdgeModFace
  - ST_ValidateTopoGeo (wrapper)

Future developments
(funding opportunities)

- Missing ISO SQL/MM editing functions:
  - ST_CreateTopoGeo
  - ST_RemEdgeNewFace, ST_RemEdgeModFace
  - ST_ValidateTopoGeo (wrapper)
- Geometry => TopoGeometry

Future developments
(funding opportunities)

- Missing ISO SQL/MM editing functions:
  - ST_CreateTopoGeo
  - ST_RemEdgeNewFace, ST_RemEdgeModFace
  - ST_ValidateTopoGeo (wrapper)
- Geometry => TopoGeometry
- TIGER/Line loader dumper

Future developments
(funding opportunities)

- Missing ISO SQL/MM editing functions:
  - ST_CreateTopoGeo
  - ST_RemEdgeNewFace, ST_RemEdgeModFace
  - ST_ValidateTopoGeo (wrapper)
- Geometry => TopoGeometry
- TIGER/Line loader dumper
- OGR driver

Future developments
(funding opportunities)

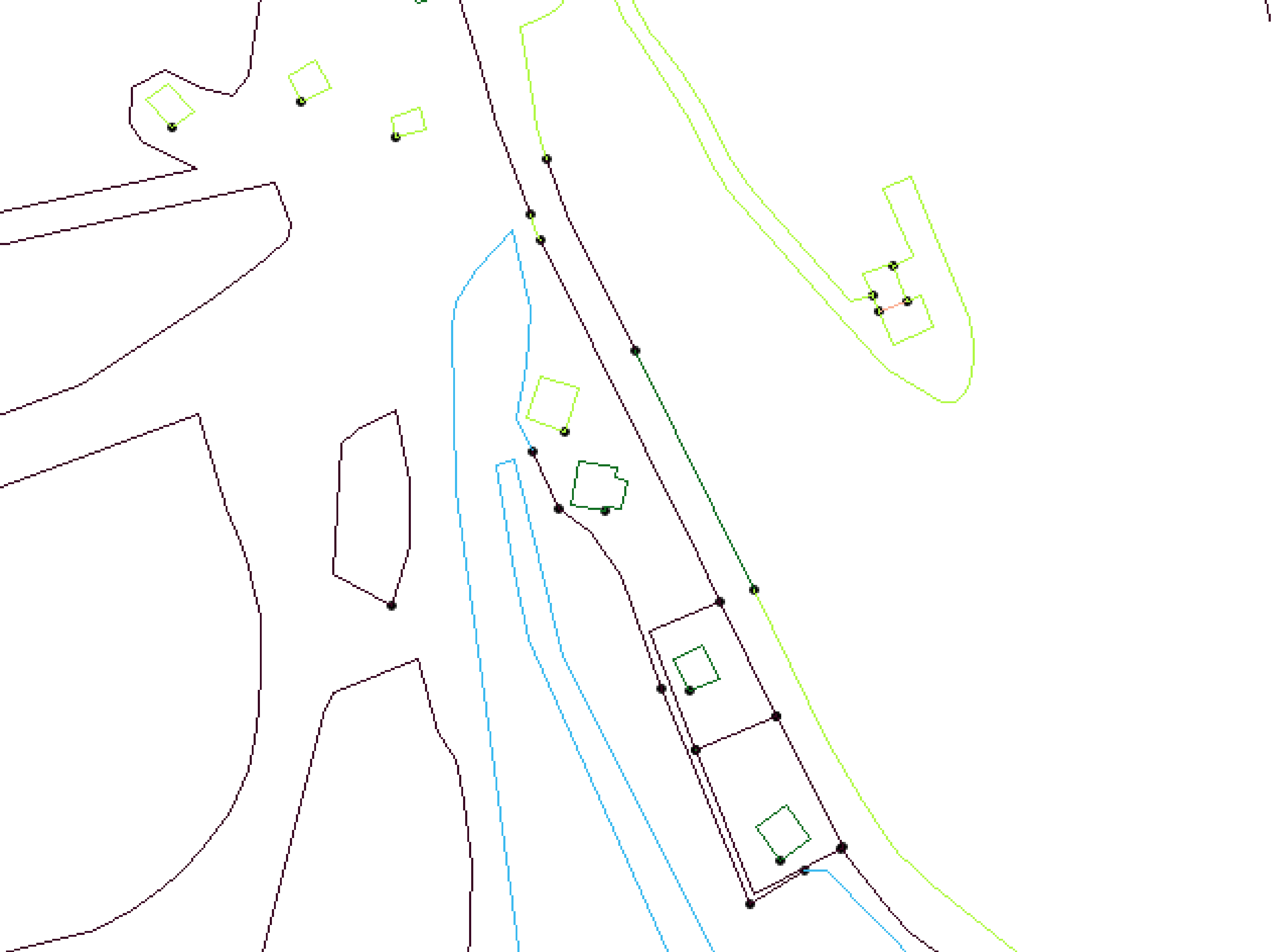
- Missing ISO SQL/MM editing functions:
  - ST_CreateTopoGeo
  - ST_RemEdgeNewFace, ST_RemEdgeModFace
  - ST_ValidateTopoGeo (wrapper)
- Geometry => TopoGeometry
- TIGER/Line loader dumper
- OGR driver
- Specialized spatial functions

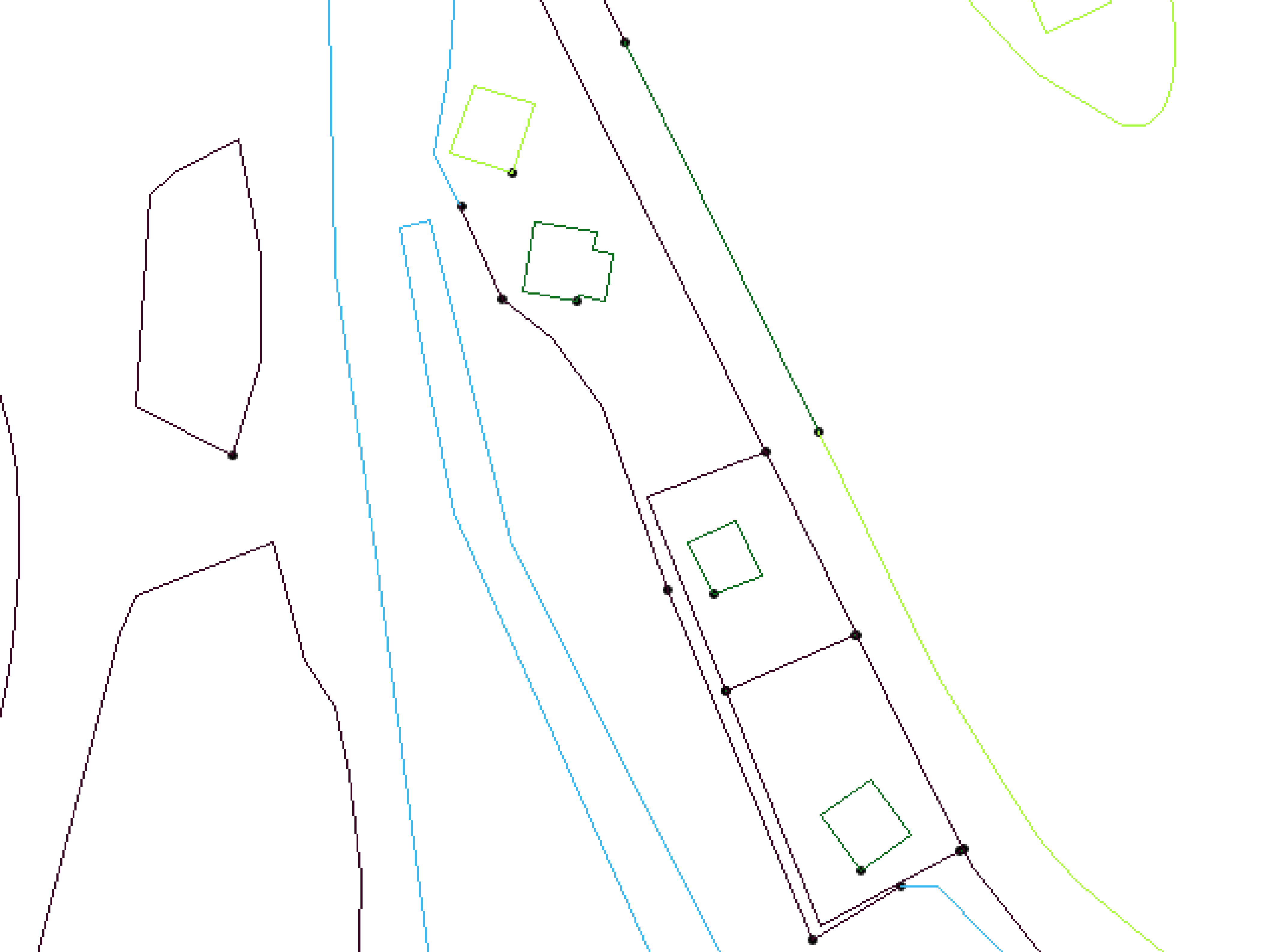
Future developments
(funding opportunities)

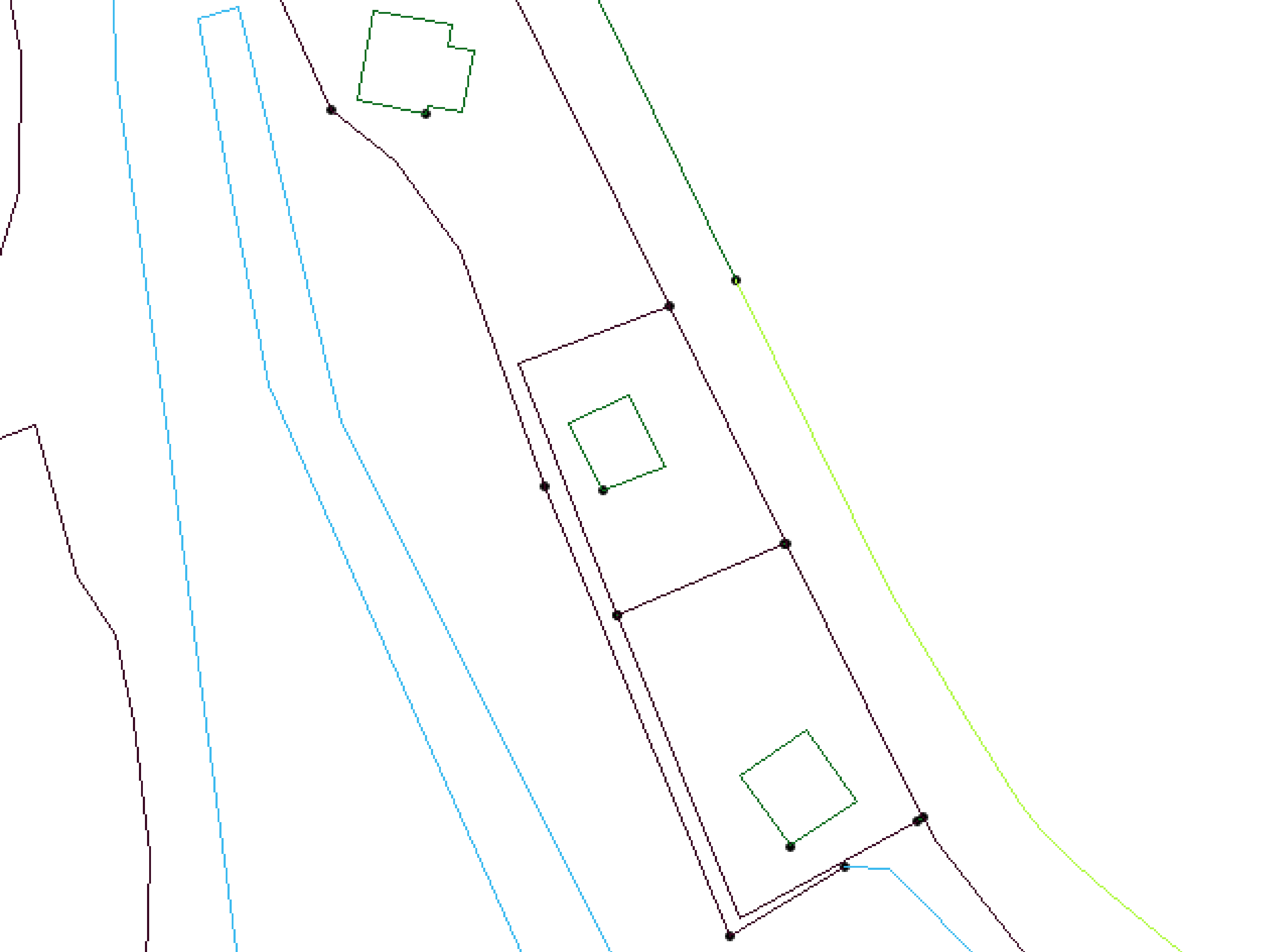
- Missing ISO SQL/MM editing functions:
  - ST_CreateTopoGeo
  - ST_RemEdgeNewFace, ST_RemEdgeModFace
  - ST_ValidateTopoGeo (wrapper)
- Geometry => TopoGeometry
- TIGER/Line loader dumper
- OGR driver
- Specialized spatial functions
- Your wish !

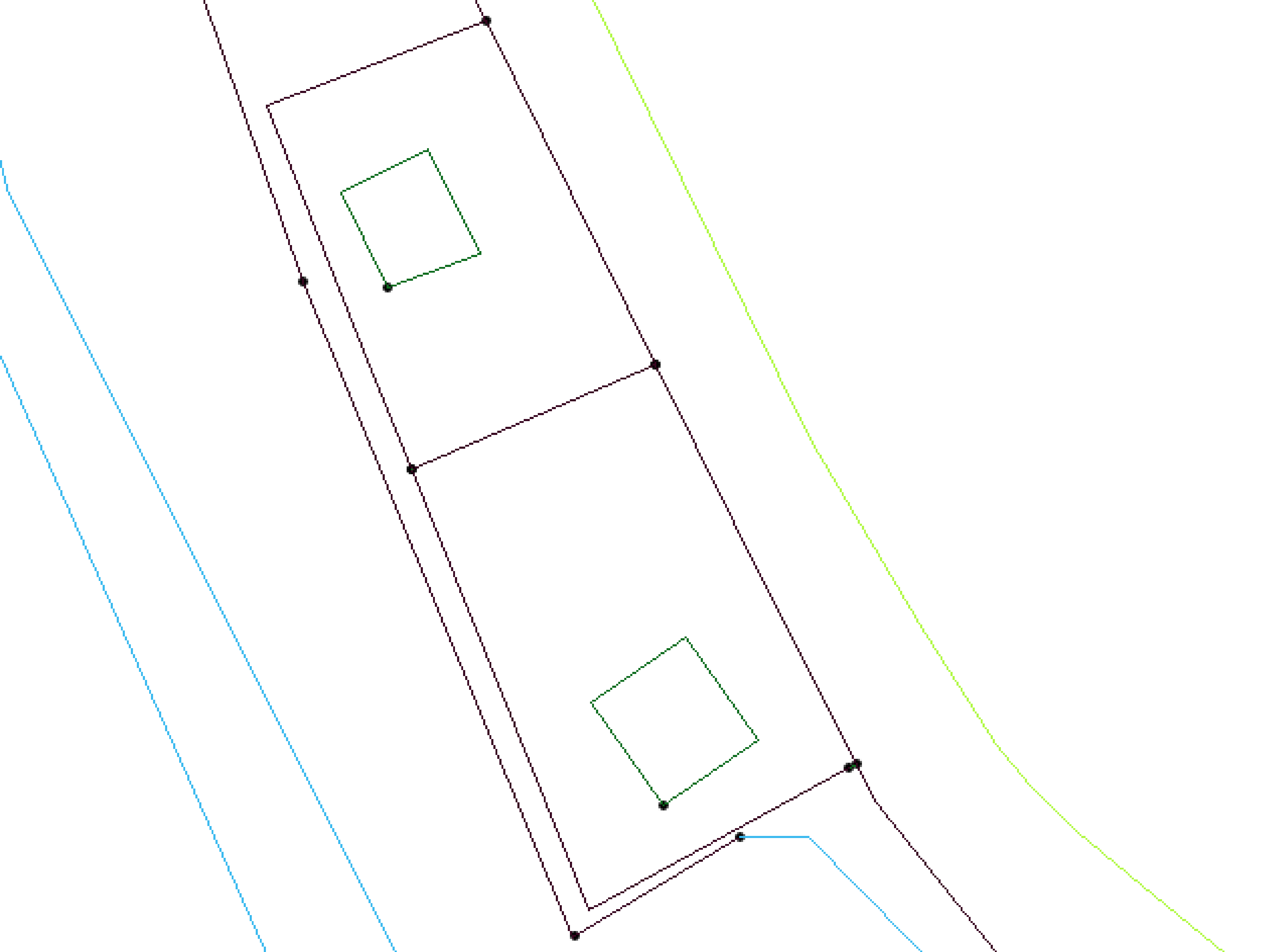
Question time

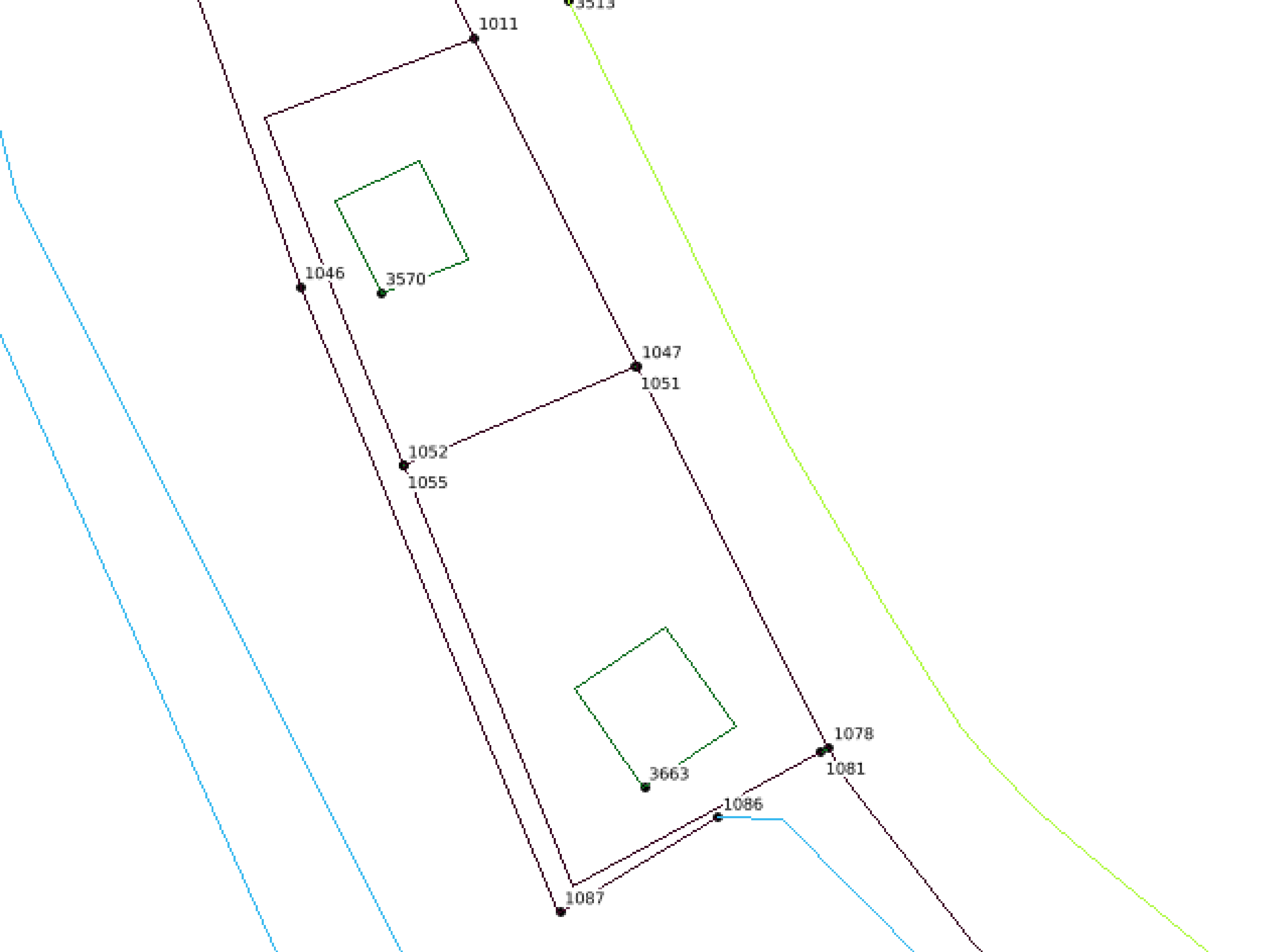
?










3513
1011
1046
3570
1047
1051
1052
1055
1078
1081
3663
1086
1087