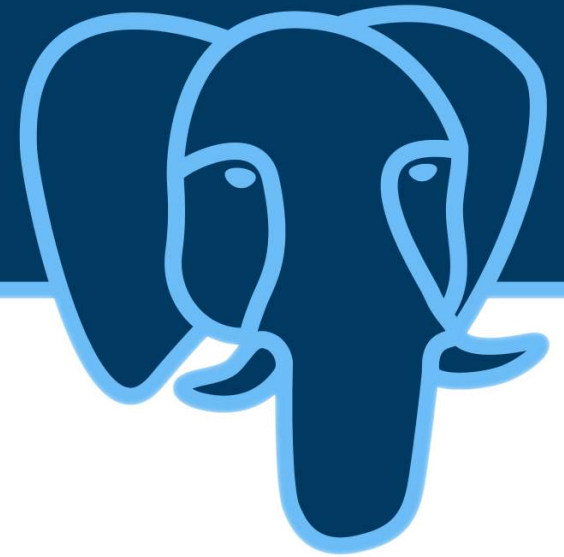


PGSession10



DAO et PostGIS



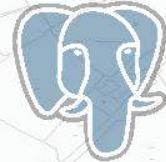
Paris

22 novembre 2018

Jean-Marie Arzac

Société de services géomatiques

- Développement
- Conseil
- Formation
- Bentley, QGIS
- PostgreSQL/PostGIS



Accompagner clients publics et privés

```
if((etype=mdlElement_getType(e1P)) == CELL_HEADER_ELM || etype == SHARED_CELL_ELM)
{
    if(etype == CELL_HEADER_ELM)
    {
        ie = ownCell_extract((DPoint3d *)&origine, (DPoint3d *)NULL, (RotMatrix *)NULL, (DPoint3d *)NULL, (OwnMSWChar *)NULL, MAX_CELLNAME_LENGTH, (MSElement *)e1P);
    }
    else if(etype == SHARED_CELL_ELM)
    {
        ie = ownSharedCell_extract((DPoint3d *)&origine, (DPoint3d *)NULL, (RotMatrix *)NULL, (DPoint3d *)NULL, NULL, (OwnMSWChar *)NULL, MAX_CELLNAME_LENGTH, (MSElement *)e1P, ACTIVEMODEL);
    }
    CopyO(&ori, &origine, DPoint3d);
    SetElementAttributesDatabase(AZDB_USER_LINK);
    SetElementAttributesApplicationId(SIG_APPLICATION_DATA_ID);
    ExtractElementAttributes(e1P, (long *)&id, &name, &attr_str);
    SetElementAttributesDatabase(AZDB_MICROSTATION);
    _SIGGA.Get(SIG_US, US_LEVNAME, levelname);
    LevelIdFromLevelName(levelname, (long *)&level, MASTERFILE);
    _SIGGA.Get(SIG_US, US_CELLID, lib);
    if(*lib) _SIGGA.Activate(lib);
    _SIGGA.Activate(SIG_US, US_CELL);
    _SIGGA.Get(SIG_US, US_CELLNAME, w_name);
    ConvertMultibyteToUnicode(w_name, w_name, MAX_CELLNAME_LENGTH);
    if(etype == CELL_HEADER_ELM)
    {
        ie = ownCell_getImDscr(&ed2P->NULL, NULL, NULL, TRUE, NULL, NULL, 0, TRUE, w_name, NULL);
    }
    else if(etype == SHARED_CELL_ELM)
    {
        //on cherche dans toutes les bibliothèques, celle attachée en premier
        if(mdlCell_findCell(&libfo, NULL, w_name, TRUE) == SUCCESS)
        {
            ie = ownCell_getImDscr(&ed2P->NULL, NULL, NULL, TRUE, NULL, NULL, 0, TRUE, w_name, NULL);
        }
        else
        {
            ie = ERROR;
            sprintf(msg, TXT_fmt_CelluleIntrouvable, nom);
            WriteError(msg);
        }
    }
    if(ie == SUCCESS)
    {
        newetype=mdlElement_getType(&ed2P->e1);
        if(newetype == CELL_HEADER_ELM)
        {
            ie = ownCell_extract((DPoint3d *)NULL, (DPoint3d *)&shape, (RotMatrix *)NULL, (DPoint3d *)NULL, (OwnMSWChar *)NULL, MAX_CELLNAME_LENGTH, (MSElement *)&ed2P->e1);
        }
        else if(newetype == SHARED_CELL_ELM || newetype == SHAREDCELL_DEF_ELM)
    }
}
```

- Outils métier spécifiques
- Assistance/support/administration
- Intégrer outils libres et propriétaires

Bruxelles Environnement

Anciennement connu sous l'appellation IBGE c'est l'administration de l'environnement et de l'énergie de la Région de Bruxelles-Capitale (Belgique).

Créée par Arrêté royal du 8 mars 1989 (MB 24/03/1989).



L'Institut Bruxellois pour la Gestion de l'Environnement

Dans le cadre de la mise à jour de la carte
UrbIS, IBGE a besoin

- de pouvoir afficher et récupérer /fusionner
les WFS dans l'outil CAD
- mais aussi de lire et écrire des données
directement depuis ou vers PostGis avec
l'outil CAD

DAO

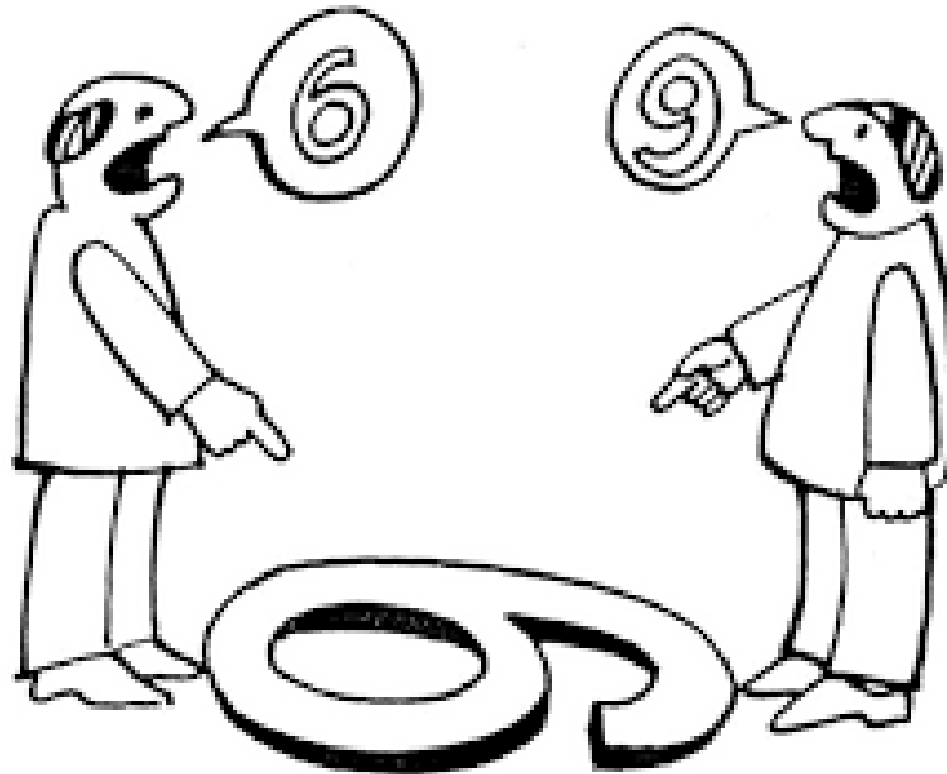
- Dessin Assisté par Ordinateur (CAD)

PostGIS

- Base de données spatiales (SIG)

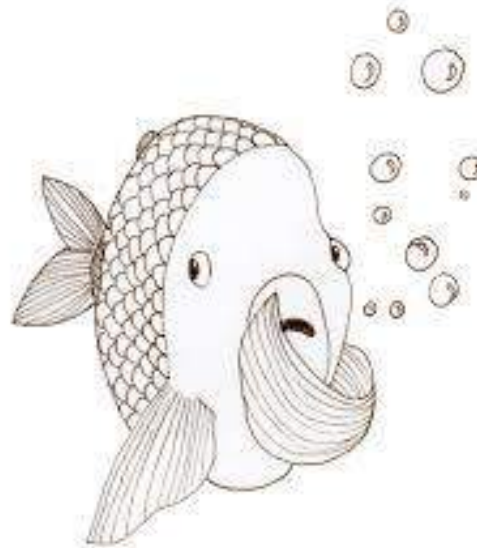


Des visions différentes



Faire communiquer DAO et SIG

- DAO ajoute une fiche BD à un élément graphique
- PostGIS ajoute une géométrie à un enregistrement
- Dialogue propriétaire libre



Modèle de données DAO

- Points (cellules), polygones, arcs, courbes, polygones,
- Géométries linéaires complexes (ouvertes ou fermées)
- Possibilité de lier chaque élément graphique à un ou plusieurs enregistrements BD

Liens BD dans fichier DAO

- Élément graphique lié à un enregistrement BD grâce à deux attributs de type entier

=> entitynum pointe sur la table liée

=> mslink pointe sur l'enregistrement dans la table liée

Modèle de données PostGIS

- Une table par primitive graphique,
- Géométries 3D (xyz)
- Tables « polyglottes »

Tables spatiales

polygon

gid integer NOT NULL,
site_id character(100),
geom_type character(100),
cell_orientation character(20),
type character(100),
the_geom geometry(PointZ,31370),
mslink integer,
mapid integer

line

gid integer NOT NULL,
site_id character(100),
geom_type character(100),
cell_orientation character(20),
type character(100),
the_geom geometry(PointZ,31370),
mslink integer,
mapid integer

point

gid integer NOT NULL,
site_id character(100),
geom_type character(100),
cell_orientation character(20),
type character(100),
the_geom geometry(PointZ,31370),
mslink integer,
mapid integer

labels

gid integer NOT NULL,
site_id character(100),
label character(200),
orientation character(20),
size real,
type character(100),
the_geom geometry(PointZ,31370),
mslink integer,
mapid integer



Tables « système »

MicroStation

maps

mapid integer,
filename character(256),

mscatalog

entitynum integer,
tablename character(64),

IBGE

description

id serial NOT NULL,
cell character varying,
cell_orientation character varying,
layer character varying,
table_id character varying,
abbr_fr character varying,
descr_fr character varying,
abbr_du character varying,
descr_du character varying,
obj_geom_type character varying,
obj_color character varying,
klasse character varying,
classe character varying,

Modèle de données pgconnector

polygon

gid integer NOT NULL,
site_id character(100),
geom_type character(100),
cell_orientation character(20),
the_geom geometry(PointZ,31370),
mslink integer,
mapid integer

line

gid integer NOT NULL,
site_id character(100),
geom_type character(100),
cell_orientation character(20),
the_geom geometry(PointZ,31370),
mslink integer,
mapid integer

point

gid integer NOT NULL,
site_id character(100),
geom_type character(100),
cell_orientation character(20),
the_geom geometry(PointZ,31370),
mslink integer,
mapid integer

maps

mapid integer,
filename character(256),

mscatalog

entitynum integer,
tablename character(64),

labels

gid integer NOT NULL,
site_id character(100),
label character(200),
orientation character(20),
size real,
type character(100),
the_geom geometry(PointZ,31370),
mslink integer,
mapid integer

description

id serial NOT NULL,
cell character varying,
cell_orientation character varying,
layer character varying,
table_id character varying,
abbr_fr character varying,
descr_fr character varying,
abbr_du character varying,
descr_du character varying,
obj_geom_type character varying,
obj_color character varying,
klasse character varying,
classe character varying,



Outils DAO (Bentley)

- MicroStation :
 - client ORACLE et ODBC
 - client WMS
- Bentley Map :
 - client Oracle et ODBC
 - client Oracle Spatial
 - client WMS et WFS

Environnement de développement

- MDL (Microstation Development Language) pseudo code en syntaxe C
- VBA (Visual Basic for Applications) + classes Bentley



PostgreSQL / PostGIS

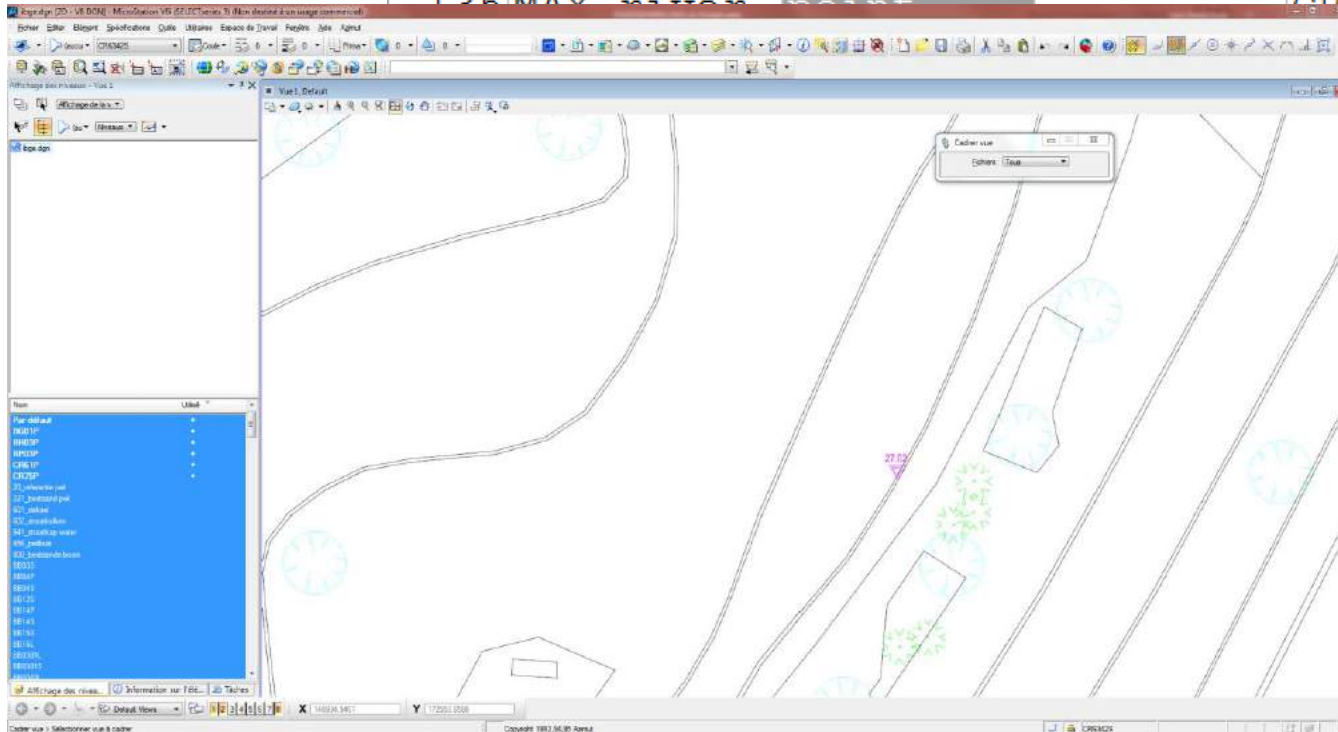
- ODBC
- Libpq
- SQL

Commandes

- Créer/supprimer connexion
- Connecter
- Check in/Check out
- Attacher en référence
- Synchroniser

Check out

gid integer	site_id character(100)	geom_type character(100)	cell_orientation character(20)	type character(100)	mmlink integer	mapid integer
135	MAX_river	point		CR6105WP	3	2
135	MAX_river	point		CR6105WP	4	2
135	MAX_river	point		CR6105WP	5	2
135	MAX_river	point		CR6105WP	6	2
				5105WP	7	2
				5105WP	8	2
				5105WP	9	2
				5105WP	10	2
				5105WP	3	2
				5105WP	4	2



Synchroniser

- Utilise un système équivalent aux triggers
- Intercepte toutes les entrées dans le « undo » buffer
- Si niveau correspond à une couche PG, INSERT, UPDATE ou DELETE
- Désactive le système « undo/redo »

Limitations

- Conversion des linéaires en polygones
- Dépendant de l'organisation des données
- Pas vraiment « open source » puisque sur outil DAO propriétaire
- Nécessite SDK de Bentley et une licence MicroStation

<https://github.com/jmarsac/pgconnector>



Merci de votre attention.

jmarsac@azimut.fr

<https://azimut.fr>

+33 6 11 05 88 23

