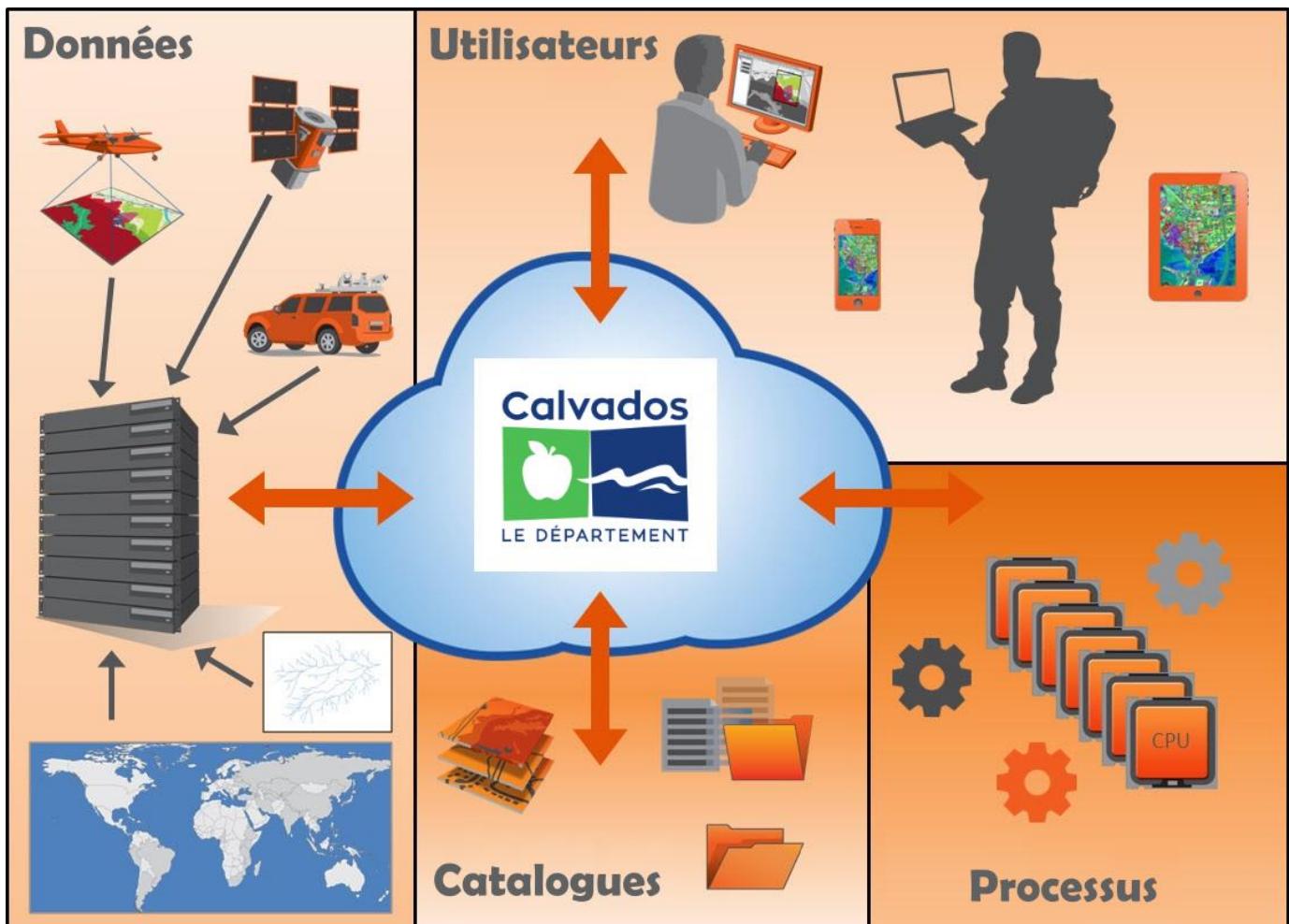


Conception de modèles et d'outils pour la production de données géographiques

Tuteur professionnel : Jérémie Ory, chef du pôle SIG

Tuteur universitaire : David Gaillard, maître de conférences



Théo Rousseau

Licence professionnelle SIG-DAT 2019-2020

REMERCIEMENTS

Je voudrais tout d'abord remercier Jérémie Ory, mon tuteur de stage, pour m'avoir donné l'opportunité d'effectuer ce stage, et de m'avoir transmis ses connaissances.

Je remercie aussi David Gaillard, mon professeur et tuteur de stage universitaire, pour avoir accepté de m'encadrer au cours de mon stage.

Je remercie toute l'équipe du pôle SIG et plus globalement l'ensemble de la direction du Département pour m'avoir accueilli au sein de leur service et avec qui ce fut un plaisir de travailler au cours de ces mois.

Je remercie aussi l'ensemble des professeurs de la licence SIG-DAT pour m'avoir transmis leurs connaissances avec pédagogie et m'avoir permis de m'améliorer dans les différents domaines de la géomatique.

TABLES DES MATIERES

INTRODUCTION	4
Le Département du Calvados.....	4
Les outils administrés par le Pôle SIG	5
1. Mapéo Calvados.....	6
2. SIR : Système d'Information Routier	7
3. Calvados Atlas	8
Objectifs du stage.....	10
Partie I : L'appariement automatique des itinéraires de randonnée sur la BDTopo de l'IGN	11
Contexte : Le relevé GPS des itinéraires de randonnée de Valdallière	12
Objectif : Apparier les relevés GPS sur la BDTopo.....	13
Solution retenue : Traitement automatique via PostGis.....	15
Conclusion	32
Partie II : Mise en place de géotraitements avec PGRouting.....	33
Contexte : Intégrer la bibliothèque de fonctions PGRouting au SIG du Calvados.....	34
Objectifs : Créer un itinéraire entre deux points et des isodistances depuis un point.....	34
1. Tester la recherche d'un itinéraire - Solution proposée : Utilisation de pg_dijkstra.....	34
Conclusion	39
2. Créer des isodistances – Solution proposée : Utilisation de « pgr_drivingdistance »	40
Conclusion	48
Partie III : Publier des données en open data sur data.calvados.fr.....	49
Contexte : Production d'un catalogue de données en ligne	50
Objectif : Automatiser la publication des données géographiques	50
Solution proposée : Publication automatique avec FME	50
Conclusion	60
CONCLUSION GÉNÉRALE.....	61
BIBLIOGRAPHIE ET WEBOGRAPHIE	62
TABLE DES FIGURES	63
ACRONYMES ET DÉFINITIONS.....	64

INTRODUCTION

Le Département du Calvados

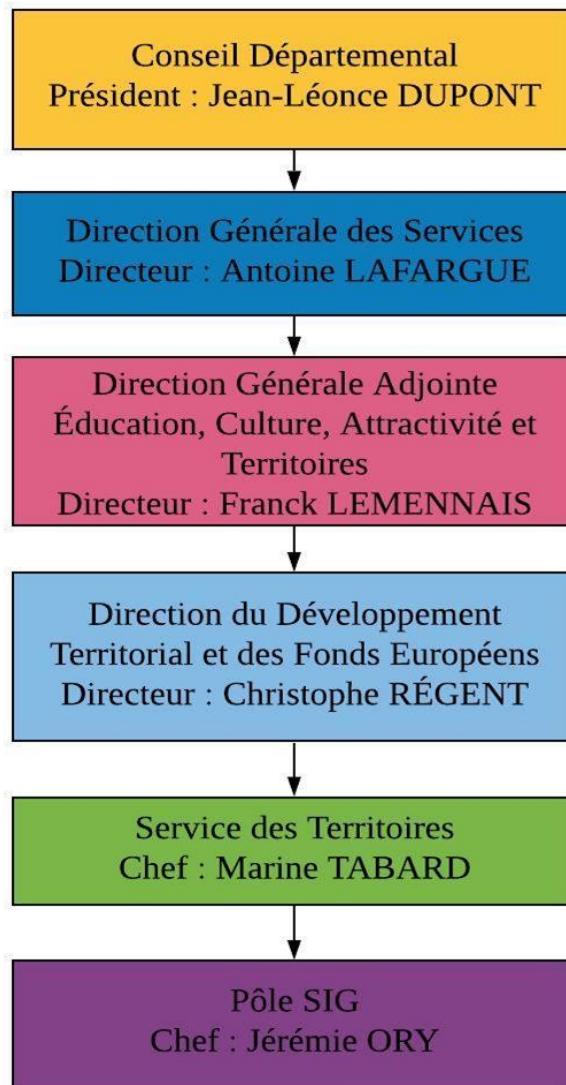
Que ce soit sur les bibliothèques des villes et villages, les collèges, les ports ou encore les routes, le Département du Calvados intervient dans une multitude de domaines impactant la vie quotidienne des 694 000 personnes vivant sur son territoire.

En effet, les compétences départementales relèvent aussi bien de l'action sociale (aides sociales, politiques d'hébergement et d'insertion sociale, gestion des maisons de retraite, ...), de l'éducation (gestion des collèges), de l'aménagement et du transport (gestion de l'eau et de la voirie rural, de l'équipement rural, de l'aménagement foncier, ...) ou encore de l'action culturelle et sportive (protection du patrimoine, gestion des musées, ...). Pour mener à bien les directives prises par les élus et veiller au bon fonctionnement des services publics, le Département dispose de 3 000 agents travaillant au sein des différents services.

Ces différents services sont répartis au sein de cinq directions générales : la Direction Générale Adjointe de la Solidarité, celle de l'Aménagement et de l'Environnement, celle des Finances et des Moyens, celle des Ressources Humaines et enfin celle de l'Éducation, de la Culture, de l'Attractivité et des Territoires. Ces différentes directions générales sont-elles même divisées en plusieurs directions. Ainsi, c'est au sein de la Direction Générale de l'Éducation, de la Culture, de l'Attractivité et des Territoires que l'on retrouve la Direction du Développement Territorial et des Fonds Européens. Cette direction porte la politique du Département en matière d'aides aux territoires, de fonds européens, et d'aides dans le domaine de l'économie. Elle est donc chargée de piloter l'ensemble des missions relatives aux fonds européens (FSE, LEADER, FEDER, ...), de favoriser l'émergence des projets européens sur le territoire ou encore de faciliter l'accès des services de la collectivité aux sources de financements européens.

Le Pôle SIG fait partie du Service des Territoires, qui est lui-même intégré à la Direction du Développement Territorial et des Fonds Européens.

Figure 1 : Organigramme de la hiérarchie des services au sein du Département

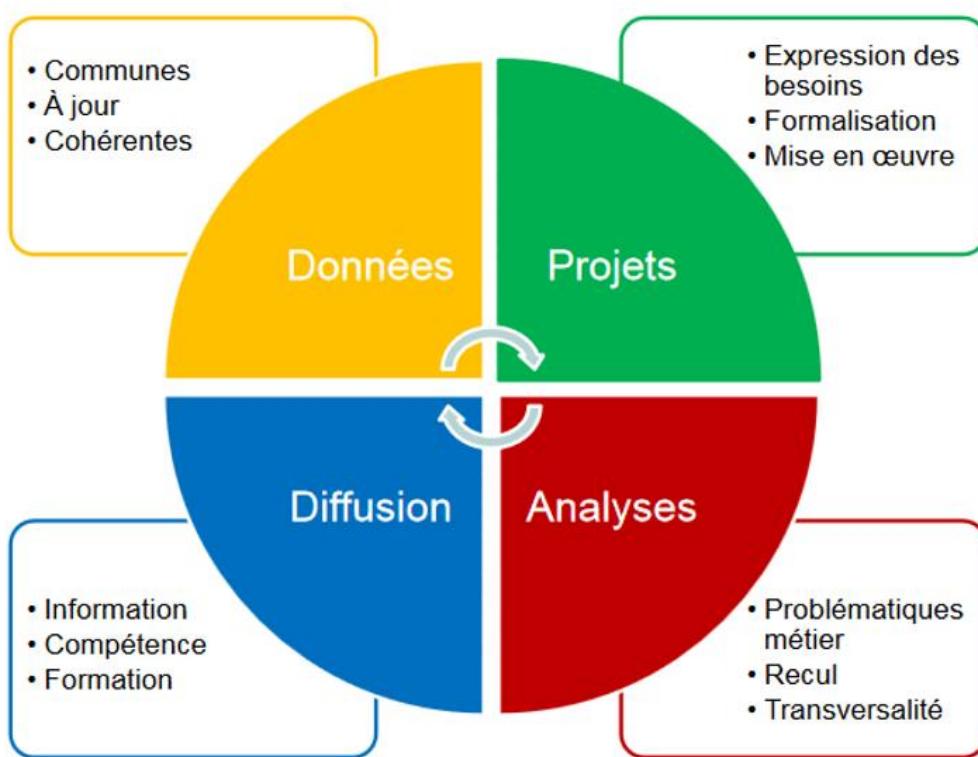


Les outils administrés par le Pôle SIG

D'une manière générale, le Pôle SIG s'articule autour de quatre missions au sein du Département :

1. Il veille à ce que les données géographiques départementales soient correctement mises à jour, à ce qu'elles soient cohérentes et communes entre les différents services du Département.
2. Il élabore des projets en fonction des besoins des différents services et les met en œuvre.
3. Il a un rôle d'analyse de diverses problématiques métier.
4. Il diffuse l'information géographique au sein du Département et forme les agents aux outils SIG qui sont à leur disposition.

Figure 2 : Schéma du fonctionnement du Pôle SIG

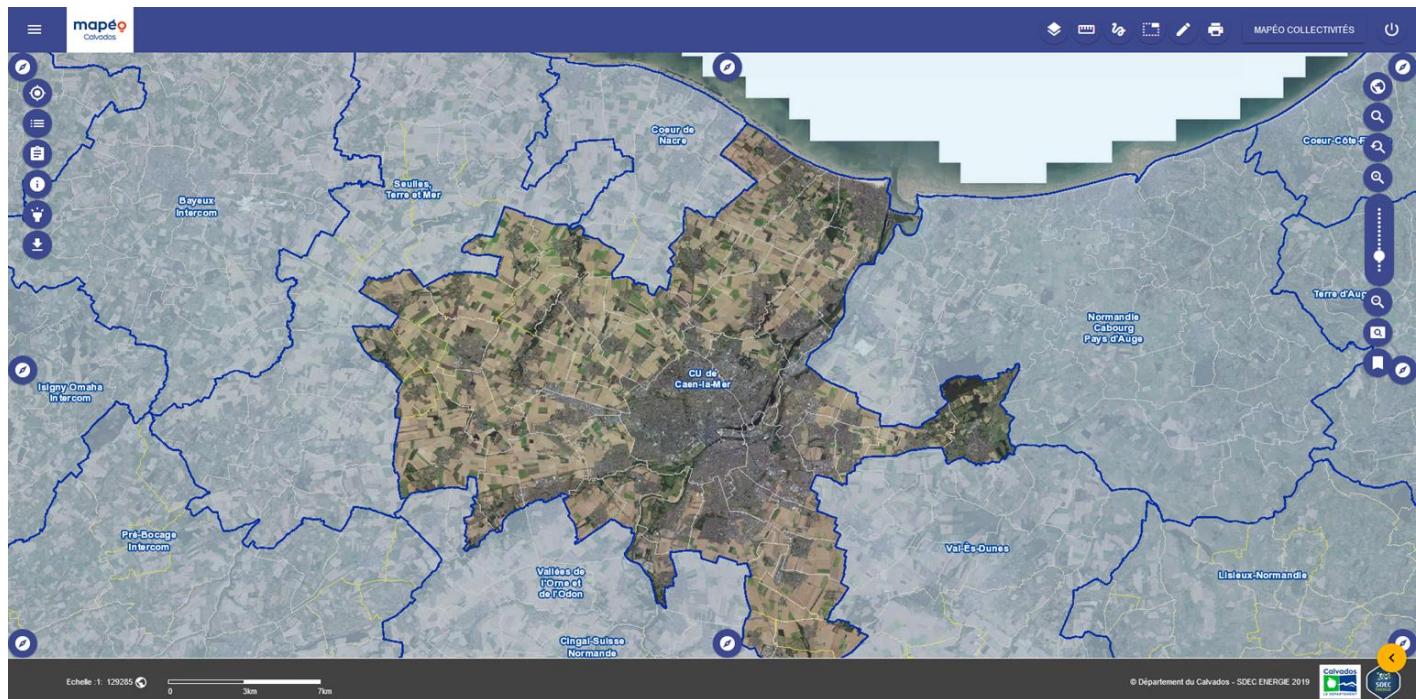


1. Mapéo Calvados

Le principal outil administré par le Pôle SIG est Mapéo Calvados, un portail cartographique en ligne à destination des collectivités : communes et EPCI. Le but de ce portail est de proposer une information cartographique de qualité, exhaustive et évolutive : cadastre, documents d'urbanisme, les différents réseaux (électricité, éclairage, gaz) ou encore les routes, les chemins de randonnées.

Le portail permet de consulter nombreuses informations géographiques de manière simple et efficace. Il permet également d'améliorer la gouvernance de l'information géographique à l'échelle départementale en regroupant l'ensemble de ces informations sur un même espace.

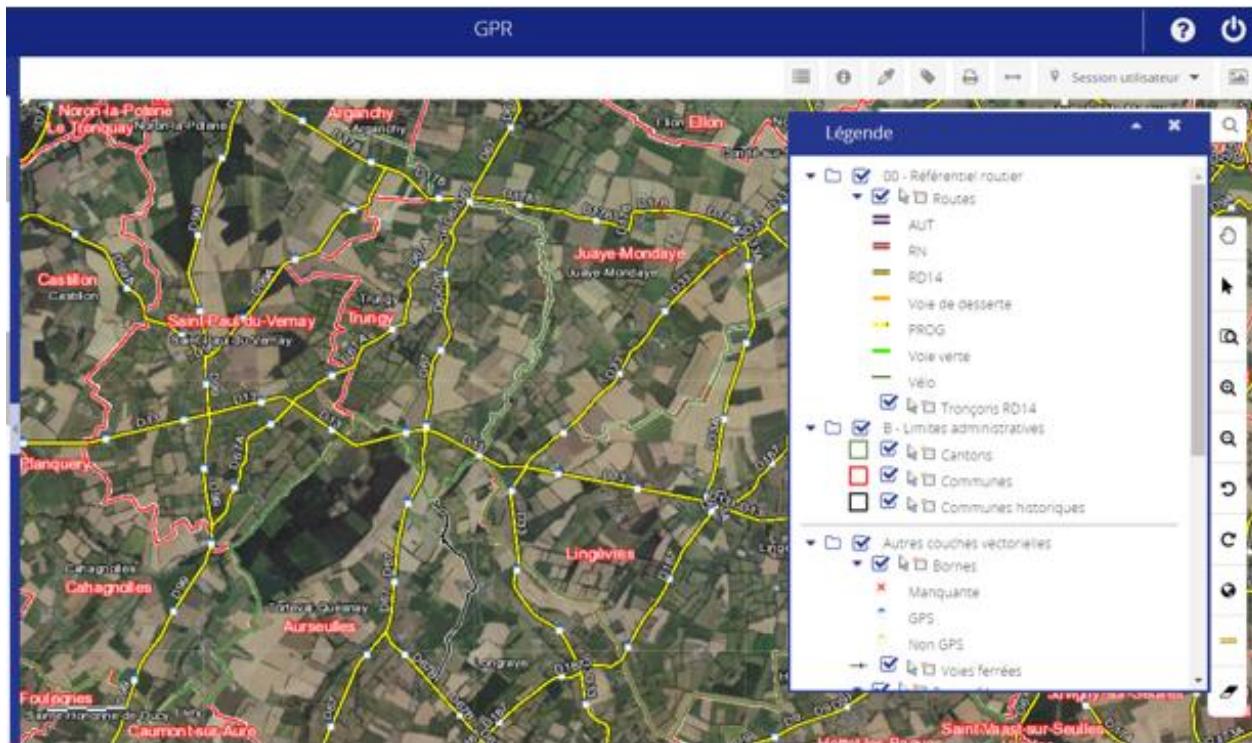
Figure 3 : Capture d'écran du portail Mapéo Calvados



2. SIR : Système d'Information Routier

Le SIR est un autre portail cartographique administré par le Pôle SIG, dont l'objectif est de permettre la mise à jour et la diffusion des données géographiques routières : le patrimoine routier départemental ainsi que le domaine public routier et maritime.

Figure 4 : Capture d'écran de SIR

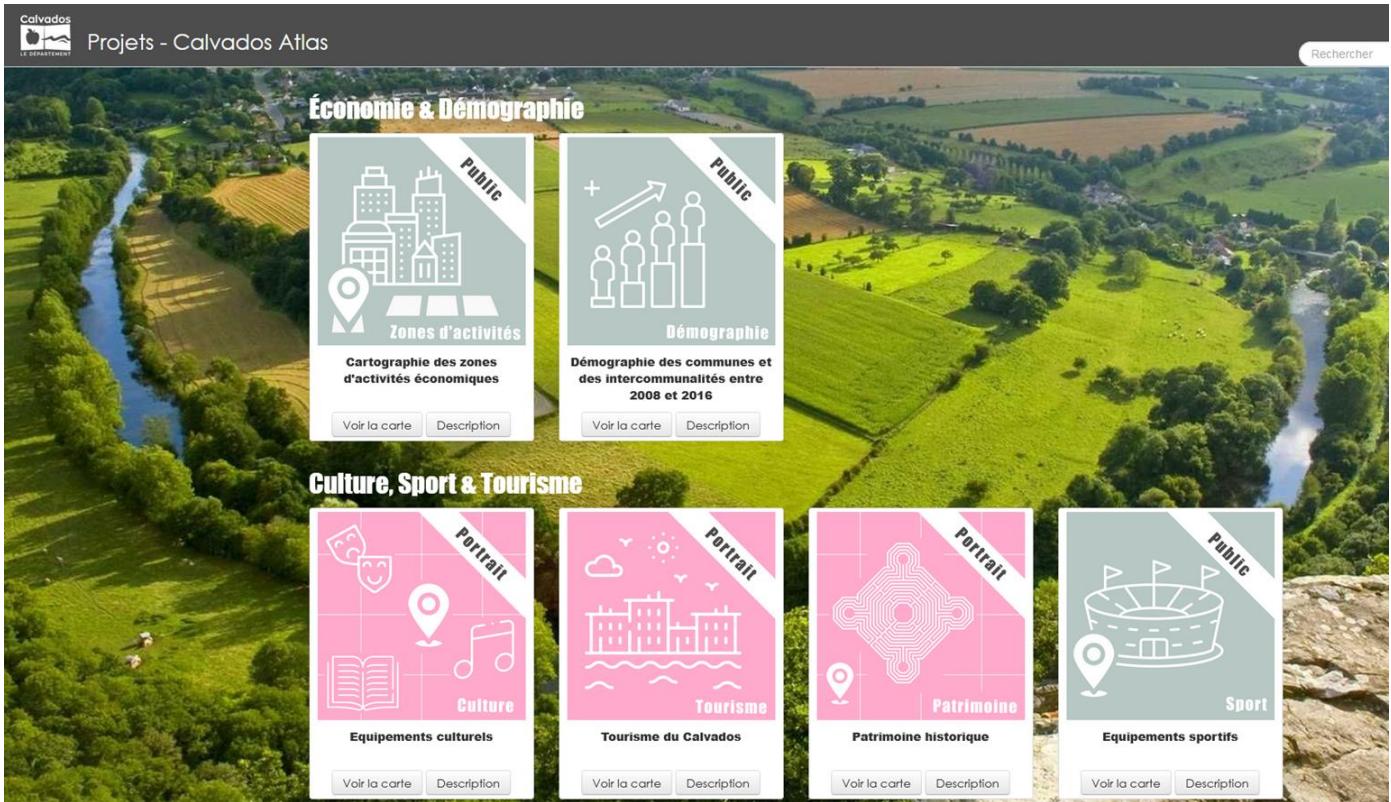


Son but est de répondre aux besoins métier de la Direction Générale Adjointe de l’Aménagement et de l’Environnement qui gère le réseau routier départemental ainsi que de faciliter le partage d’information entre la Direction et les agences routières départementales via un portail SIG.

3. Calvados Atlas

Calvados Atlas est un portail cartographique proposant des données géographiques en ligne, divisées selon plusieurs thématiques (solidarité, économie, culture/patrimoine, ...). Il propose ainsi plusieurs données métiers à destination des services du Département : réseau routier, ENS (espaces naturels sensibles), plan vélo, etc...

Figure 5 : Capture d'écran de Calvados Atlas



Ce projet a pour objectifs de subvenir aux besoins métiers des différents services du Département en matière d'information géographique mais aussi de centraliser cette information sur une même plateforme. Calvados Atlas permettra aussi un suivi des politiques territoriales départementales : cartographie des zones d'activités économiques, des espaces naturels sensibles, avancement de la fibre optique, ...

De plus, ce portail n'est pas seulement destiné aux services départementaux mais aussi à ses partenaires. Calvados Atlas leur donnera ainsi accès aux données départementales et hébergera aussi les leurs : études et diagnostics structurants du territoire avec l'Aucame (Agence d'Urbanisme Caen Normandie Métropole), projets et actions territoriales avec Calvados Attractivité (agence d'attractivité du Calvados), programmes et études sur l'habitat avec Inolya (premier bailleur social du département), ...

Objectifs du stage

Le Département du Calvados était à la recherche d'un stagiaire dans le but d'accompagner le Pôle SIG pour ses diverses tâches en cours. Le stage, initialement prévu pour un délai de quatre mois et demi, d'avril à mi-août 2020, a dû être raccourci à trois mois en raison de l'épidémie du Covid 19. Le stage s'articule autour de trois problématiques majeures :

- **L'appariement automatique des itinéraires de randonnées sur la BDTopo de l'IGN.**

Objectif : Automatiser la rectification des erreurs de tracé sur des polylignes obtenues via GPS.

- **Mise en place de géotraitements avec PGRouting.**

Objectif : Calculer automatiquement le chemin le plus court entre deux emplacements/Créer des isodistances autour d'un point.

- **Alimenter en données data.calvados.fr.**

Objectif : Diffuser des données en ligne en open data.

Le principe du stage d'occuper un poste de chef de projet junior en participant à l'ensemble des chaînes de traitement de l'information géographique du Calvados :

1. Mise en place d'outils d'analyse cartographique : créer des isodistances à partir de scripts attaquant directement la base de données ;
2. Vérifier et analyser la qualité des données géographiques : correction des erreurs géométriques sur les données GPS des itinéraires de randonnée ;
3. Partage de données : mise en place d'un catalogue open data

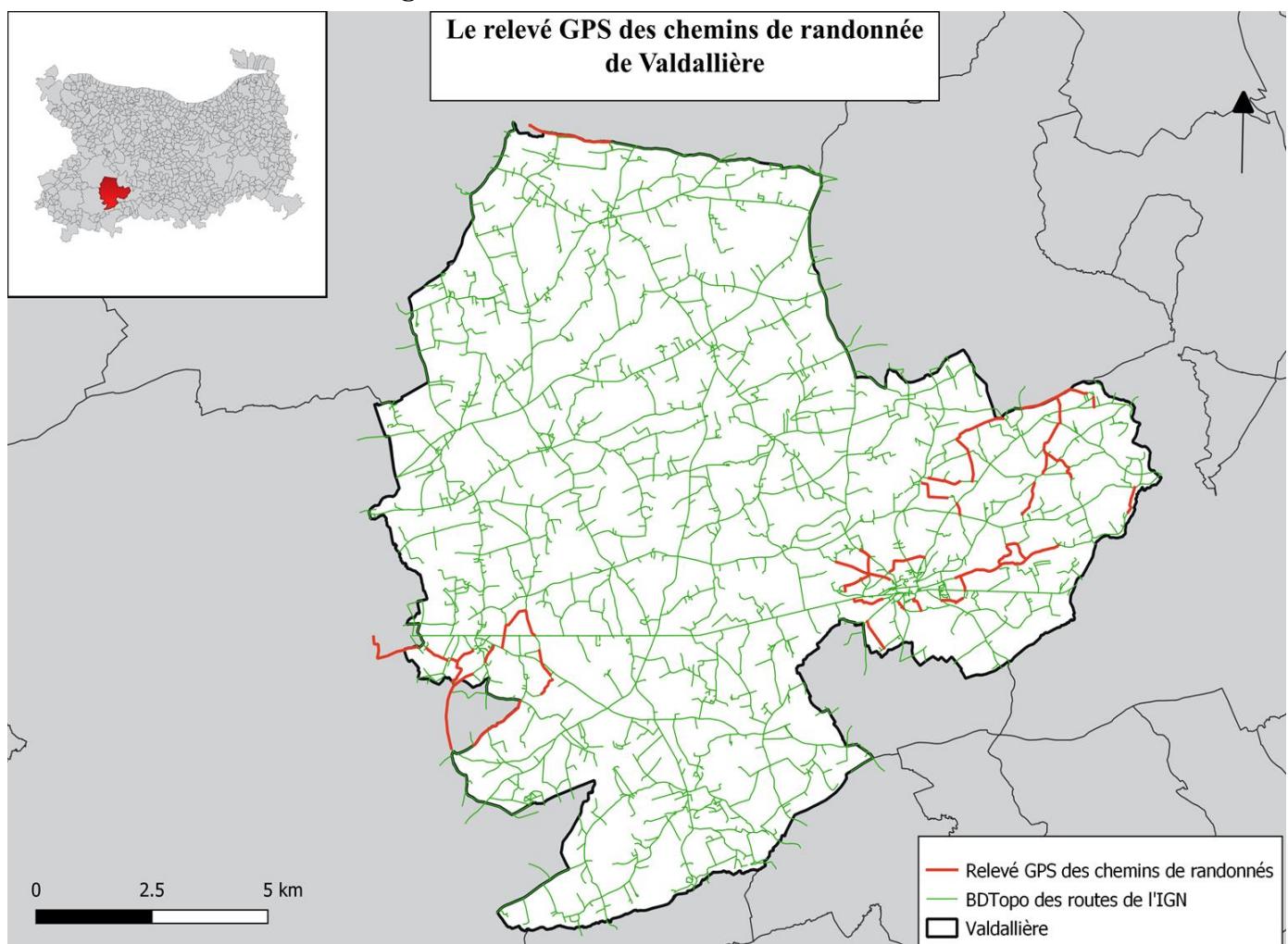
En tant que chef de projet junior, j'ai travaillé sur mis en place des outils que j'ai testé sur des jeux de données géographiques échantillon, le plus souvent à l'échelle communale. Les traitements ont été calibrées afin qu'ils puissent être mise en place au niveau départemental.

Partie I : L'appariement automatique des itinéraires de randonnée sur la BDTopo de l'IGN



Contexte : Le relevé GPS des itinéraires de randonnée de Valdallière

Figure 6 : Carte des relevés GPS à Valdallière



Calvados Attractivité, l'agence d'attractivité créée à l'initiative du Comité Départemental du Tourisme du Calvados, a effectué des relevés GPS sur le terrain afin d'obtenir le tracé des itinéraires de randonnées présents sur le territoire départemental, cela dans l'objectif final de les intégrer au portail cartographique Mapéo Calvados. Cependant, une fois le travail de collecte effectué, les données nécessitaient un post-traitement pour corriger la qualité géométrique des données qui étaient parfois mauvaises. En effet, les relevés GPS n'étaient pas précis et il arrive parfois que le tracé obtenu s'éloigne parfois de plusieurs mètres des coordonnées réelles de l'objet que l'on veut cartographier.

Ainsi, si l'on compare les données obtenues à la BD Topo du réseau routier de l'IGN, on peut observer ces erreurs de tracé.

Figure 7 : Exemple d'une erreur de tracé obtenue avec le relevé GPS



Objectif : Apparier les relevés GPS sur la BDTopo

Ainsi, l'objectif sera ici d'apparier les polylignes obtenus via relevé GPS sur le tracé de la BD Topo qui est géométriquement plus précis. Ce travail pourrait être réalisé manuellement, mais cela prendrait un temps considérable, la table contenant 749 objets. Il serait donc plus efficace d'automatiser le traitement.

Les outils à notre disposition ici seront le logiciel QGis 3.4, le système de gestion de base de données PostgreSQL et son extension PostGIS ainsi que l'application Open Jump.

Les deux données sur lesquelles nous travaillerons sont les tracés GPS des itinéraires de randonnée du Département ainsi que la BDTopo_routes du Département de l'IGN.

Zone géographique test :

Pour faire ce traitement, j'ai choisi de faire d'abord des essais sur une zone test : la commune de Valdallière. J'ai pris cette dernière car on y trouve plusieurs tracés des itinéraires de randonnées assez éloignés de ceux de la BDTopo.

Tests logiciels effectués :

La difficulté ici est que QGis ne dispose pas d'outils rapide pour faire correspondre une entité vectorielle à une autre. Cependant, une application permettrait de réaliser cette tâche rapidement : Open Jump. Cette dernière est une application SIG modulaire en licence libre et alimentée par la communauté internationale qui dispose d'un nombre d'outils varié, dont notamment celui de faire correspondre des points par rapport à des polylignes.

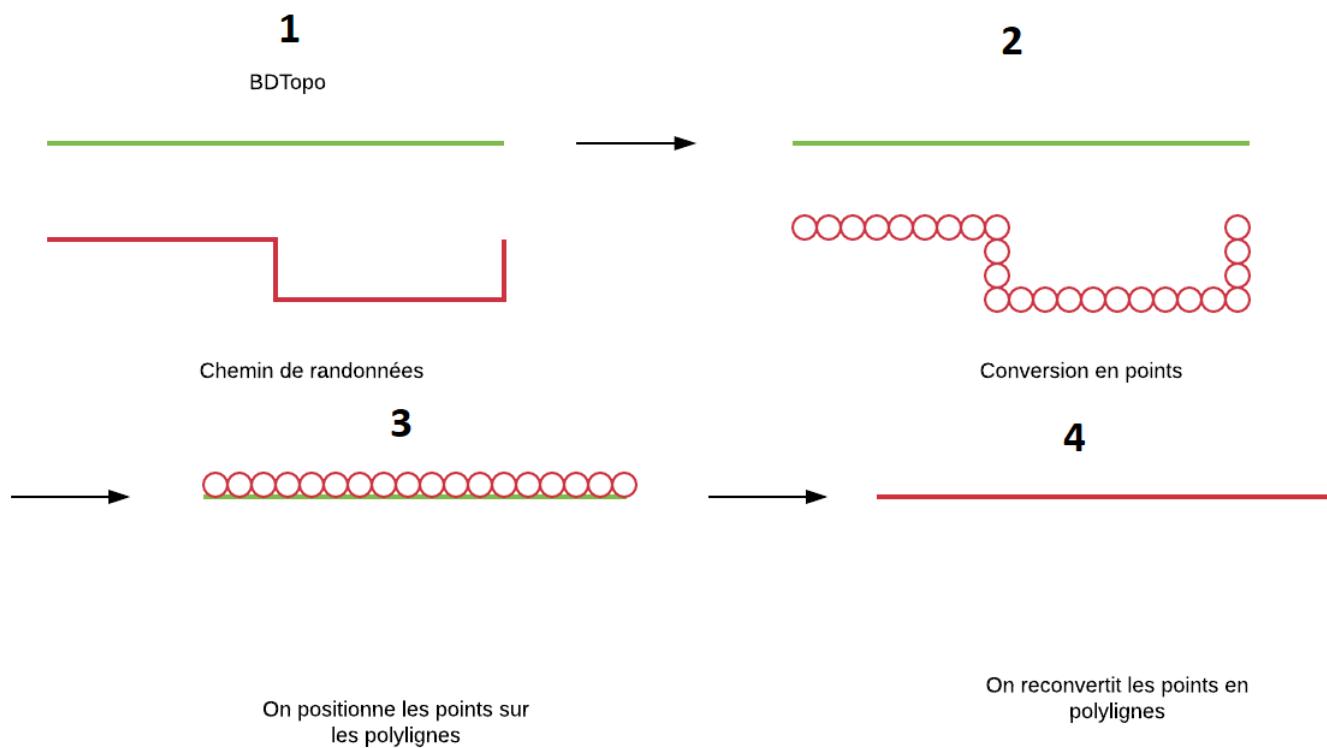
Cependant, pour ce qui est de notre cas, nous devons faire correspondre des polylignes entre elles et non des points avec des polylignes. L'idée ici sera donc, dans un premier temps, de convertir les polylignes des itinéraires de randonnées en points, de les repositionner sur les polylignes de la BDTopo puis de retrouver ces points en polylignes afin d'obtenir des itinéraires de randonnées au bon tracé.

Il existe un outil sur Open Jump censé effectuer ce géotraitement, « Fragmenter des lignes par des points », cependant il ne fonctionne pas sur les couches cibles. Après plusieurs recherches, il existe un moyen qui permet d'extraire les points des polylignes des itinéraires de randonnées et de les projeter sur les polylignes de la BDTopo directement sur QGIS. Cela nous permettrait donc de nous passer d'Open Jump et de réaliser l'ensemble de l'opération sur le même logiciel, ce qui est un avantage non négligeable lorsqu'on cherche à automatiser au maximum les tâches.

Après plusieurs recherches, il semble qu'il soit possible de réaliser des opérations similaires entièrement sur QGIS à l'aide de PostGis. L'idée ici est la même que la première solution : dans un premier temps, on convertit les polylignes des itinéraires de randonnées en points, puis de les projeter sur les polylignes de la BDTopo.

Solution retenue : Traitement automatique via PostGis

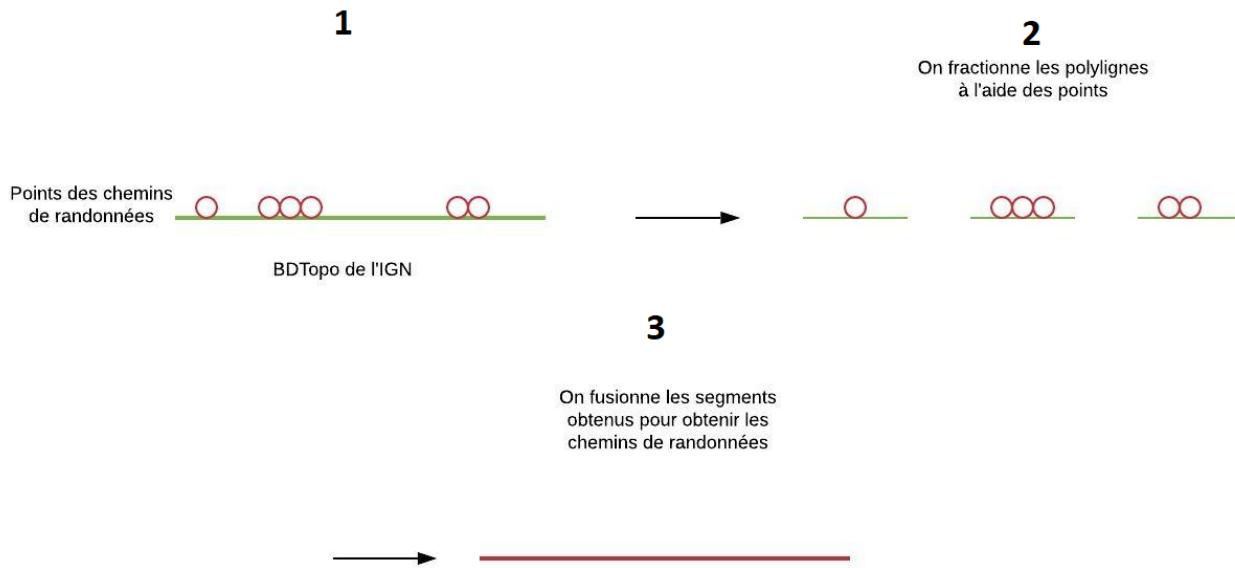
Figure 8 : Schéma représentant les différents géotraitements à réaliser



On commence donc par transformer les polylinéaires des itinéraires de randonnée en points sur QGIS à l'aide de l'outil « Extraire les sommets ». On migre ensuite les données vers Open Jump qui possède l'outil « Projeter des points sur des lignes » afin de positionner les points sur les polylinéaires.

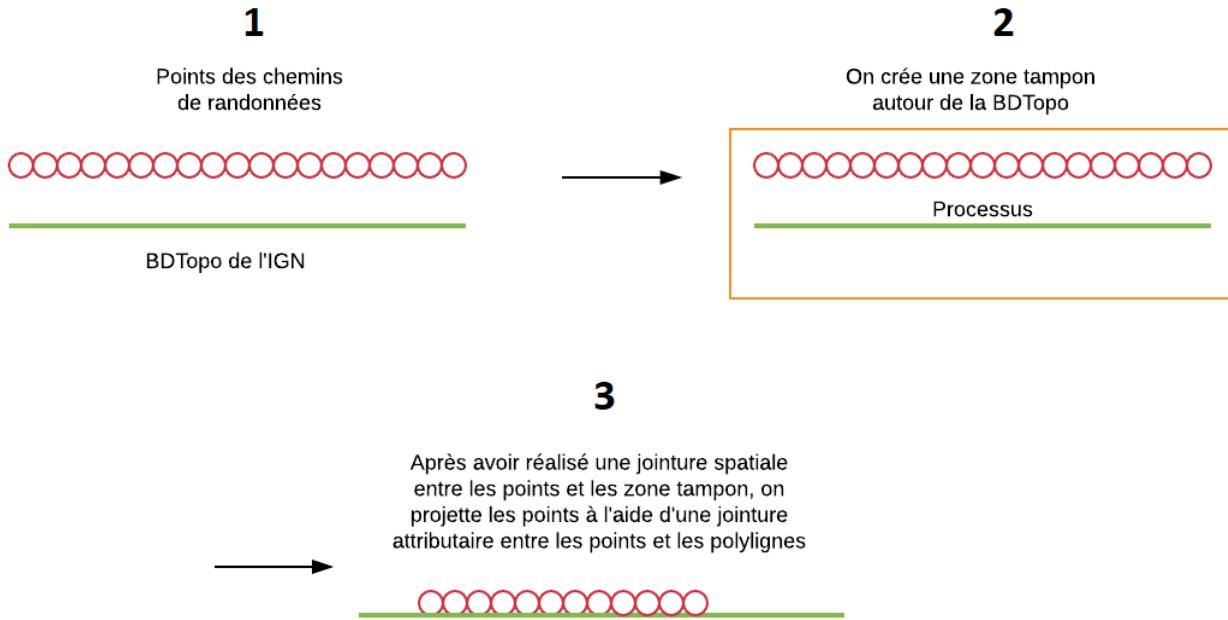
Nos points correspondant aux sommets des itinéraires de randonnées sont désormais positionnés sur les polylinéaires de la BDTopo. Cependant, à cette étape, un problème se pose : comment reconvertis les points en polylinéaires ? Il existe bien des outils sur Open Jump ou QGis pour convertir des points en polylinéaires, mais si on utilise cet outil, il risque d'y avoir à nouveau un décalage avec la BDTopo. La meilleure solution semble donc de fractionner les polylinéaires de la BDTopo à partir des points des itinéraires de randonnées, de sélectionner les segments des polylinéaires qui intersectent les points puis de fusionner ces segments entre eux.

Figure 9 : Schéma représentant la seconde étape



Il est possible de projeter des points sur les polylignes sur QGIS à l'aide d'une jointure attributaire. Cependant, pour faire cette connexion, il est donc nécessaire que les polylignes de la BDTopo et les points aient un champ en commun dans leurs tables attributaires, ce qui n'est pas le cas. Pour remédier à ce problème, nous avons pensé à créer une zone tampon autour des polylignes de la BDTopo puis de faire une jointure spatiale entre ces zones tampons et les points se trouvant à proximité des polylignes.

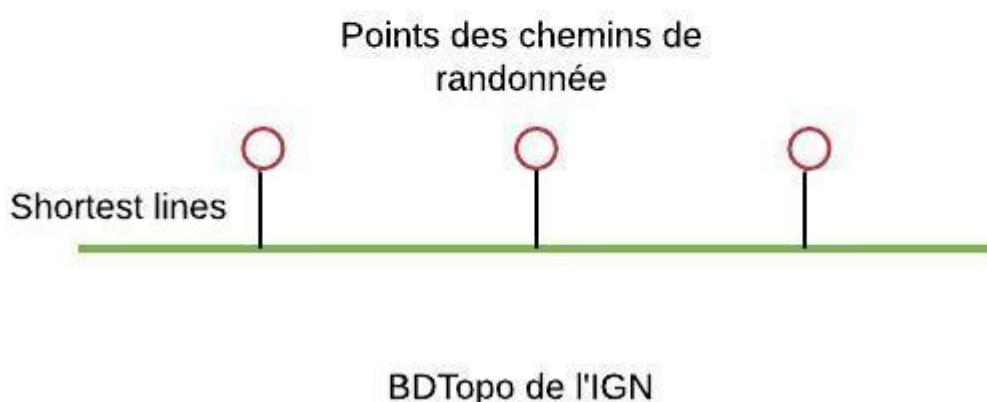
Figure 10 : Schéma représentant la jointure spatiale entre les points et les zones tampons



La fonction ST_ShortestLine de Postgis permet de créer, pour chaque objet d'une couche, une polyligne représentant la distance avec les objets d'une autre couche. L'idée ici est donc de créer des polylinéaires représentant, pour chaque point des itinéraires de randonnée, la distance avec la polyligne de la BDTopo la plus proche. Ensuite, on utilise les shortest lines pour découper la BDTopo en différents segments.

Figure 11 : Schéma représentant le découpage de polylignes via les shortest lines

On crée pour chaque point une ligne représentant la distance avec la polyligne de la BDTopo la plus proche



On découpe ensuite la BDTopo avec les lignes obtenues afin d'obtenir plusieurs segments différents



Une fois ces segments obtenus, on sélectionnera ceux qui correspondent aux itinéraires de randonnées et on fera un ST_Union afin d'obtenir les polylinéaires des itinéraires de randonnées avec la bonne géométrie.

L'ensemble du traitement à là aussi été réalisé sous forme de requête SQL sous PostGis. Voici le code détaillé avec les différentes étapes :

1^{ère} étape : création de la fonction ST_AsMultiPoint

On crée la fonction ST_AsMultiPoint qui nous permettra plus tard de convertir la couche de polylinéaires des itinéraires de randonnée en une couche de points.

```
CREATE OR REPLACE FUNCTION ST_AsMultiPoint(geometry) RETURNS geometry AS
'SELECT ST_Union((d).geom) FROM ST_DumpPoints($1) AS d;'
LANGUAGE sql IMMUTABLE STRICT COST 10;
```

2^{ème} étape : transformation de la couche des itinéraires de randonnée en une couche 2D

La couche du relevé GPS était initialement en 3D alors que celle de la BDTopo était en 2D. Cette requête convertit la couche des itinéraires de randonnée en une couche 2D grâce à la fonction ST_Force2D afin de pouvoir réaliser des géotraitements entre cette couche et celle de la BDTopo.

```
CREATE TABLE theo.chemins_entretenus_valdalliere_2D AS
SELECT id, maitre_oeu, km_aide_dp, date_maj, nom_epci, ST_Force2D(geo
m) as geom
FROM theo.chemins_entretenus_valdalliere;
```

3^{ème} étape : densification des nœuds des itinéraires de randonnée et conversion en multipoints

Une fois la couche convertie en 2D, on utilise la fonction ST_Segmentize pour la densifier puis la fonction ST_AsMultiPoint pour la convertir en une multitude de points.

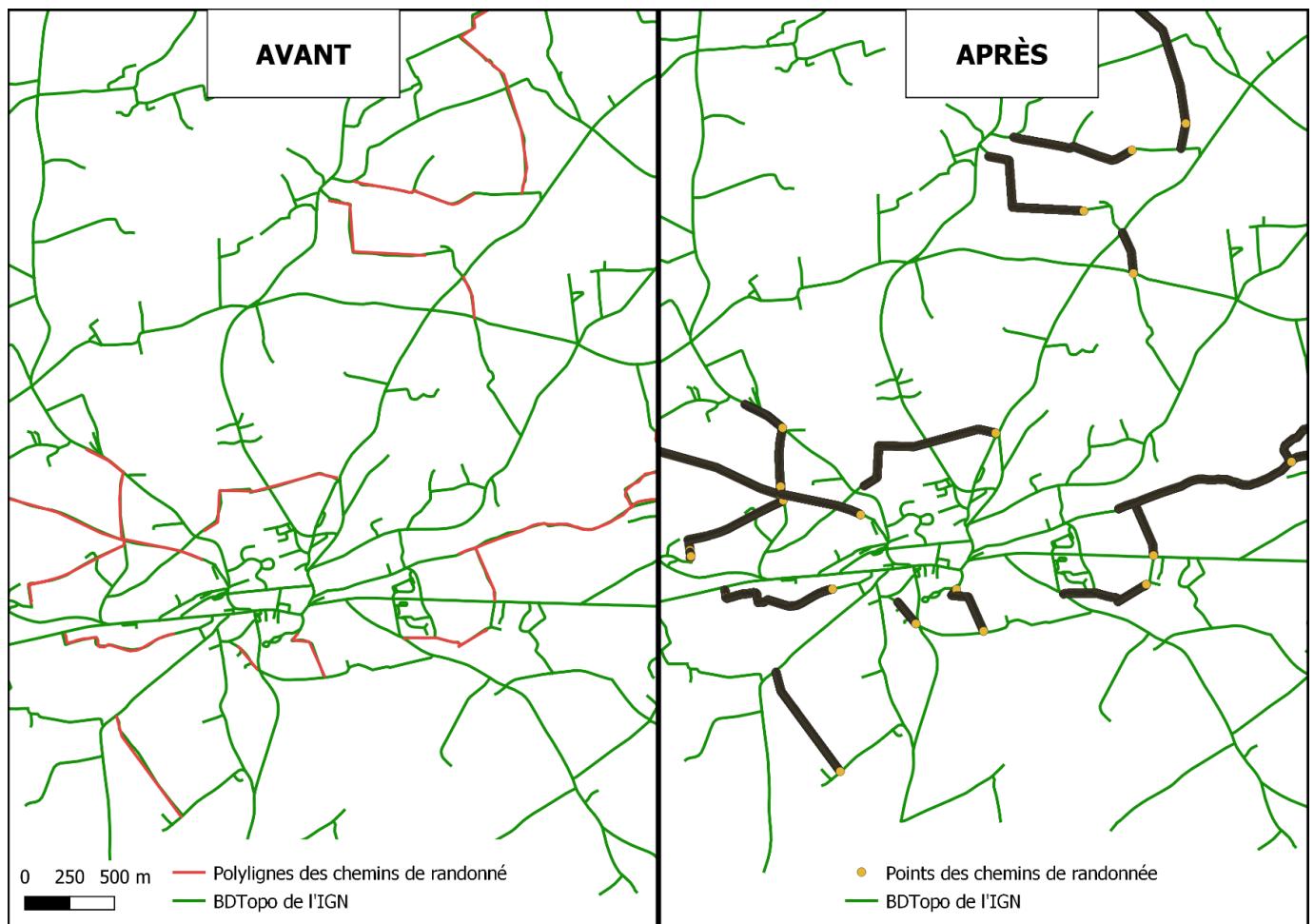
```
CREATE TABLE theo.chemins_entretenus_valdalliere_multipoints AS
```

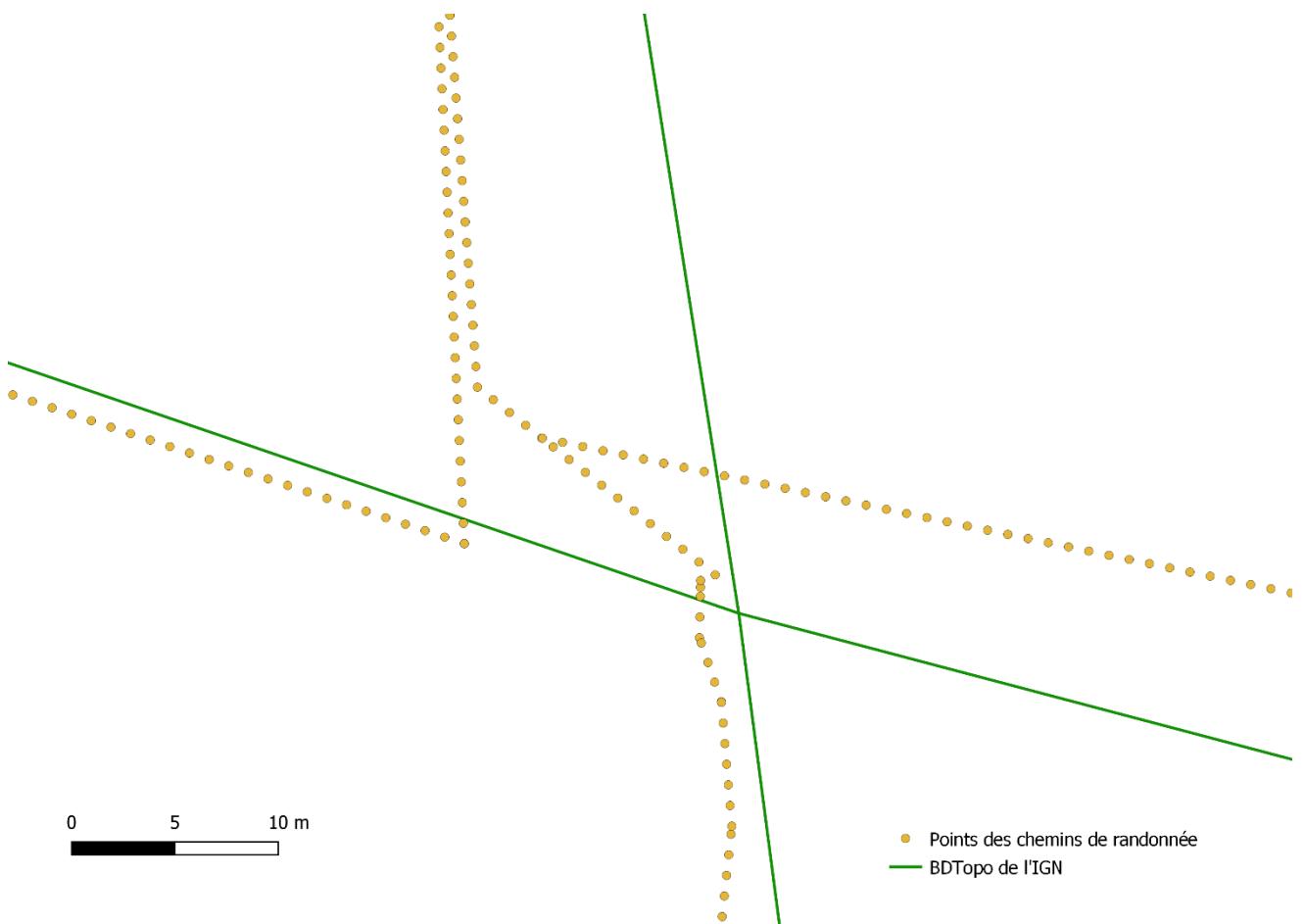
```

SELECT id, maitre_oeu, km_aide_dp, date_maj, nom_epci, ST_AsMultiPoint(st_
segmentize(geom,1))::geometry(MULTIPOINT,2154) as geom
FROM theo.chemins_entretenus_valdalliere_2d;

```

Figures 12 et 13 : Conversion des polylignes des itinéraires de randonnée en multipoints





4^{ème} étape : Conversion des multipoints en points

On dispose désormais d'une couche de points représentant les itinéraires de randonnées. Cependant, cette couche est une couche de multipoints, c'est-à-dire que l'ensemble des points ne constituent qu'un seul objet avec un seul identifiant. On convertit donc cette couche multipoints en une couche de points via la fonction ST_Dump pour pouvoir dissocier chaque point de la couche.

```
CREATE TABLE theo.chemins_entretenus_valdalliere_points AS
SELECT id, maitre_oeu, km_aide_dp, date_maj, nom_epci, (ST_Dump(a.geom)).geom
FROM theo.chemins_entretenus_valdalliere_multipoints a;
```

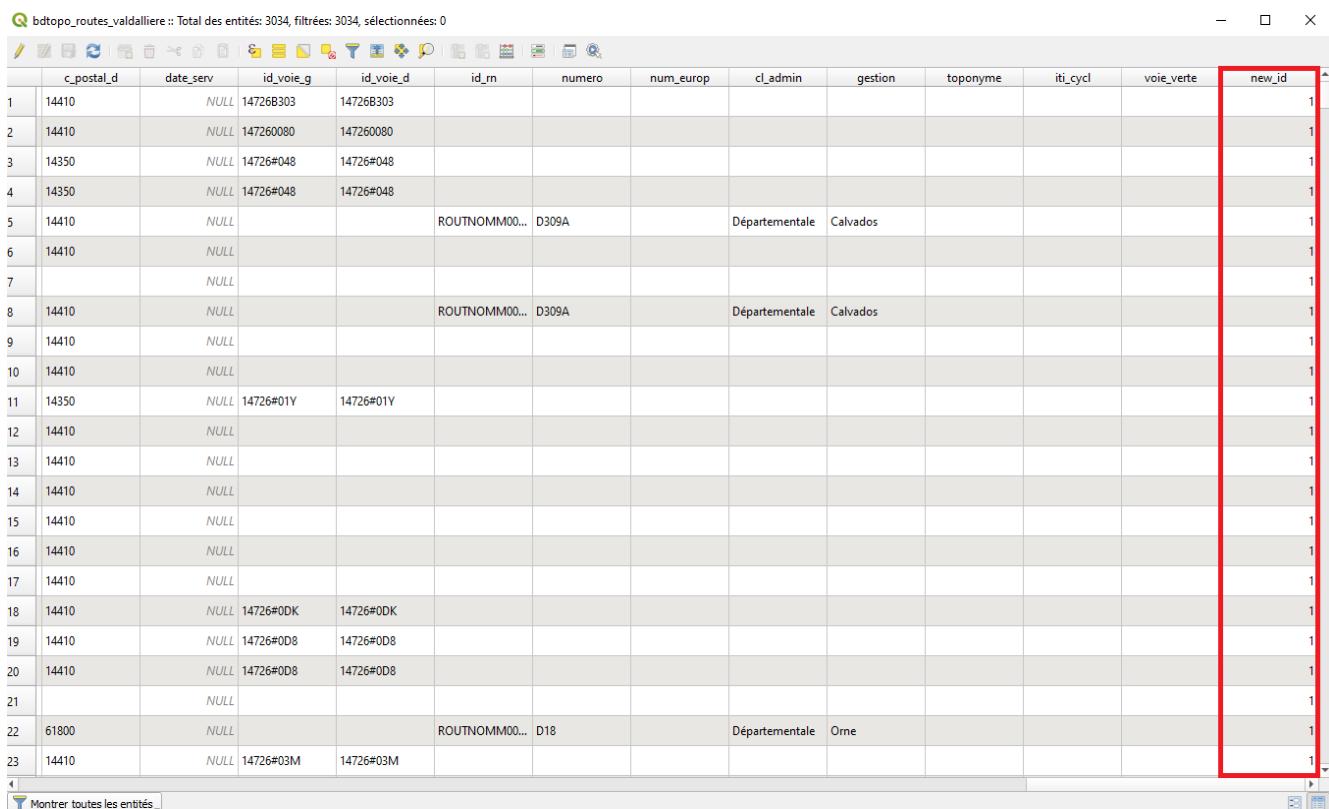
5^{ème} étape : Ajout d'un champ « new_id » à la table attributaire de la BDTopo

Par la suite, on aura besoin d'unifier l'ensemble des polylignes de la BDTopo sous un seul et même objet pour pouvoir créer une shortest line unique entre les points des itinéraires de randonnée et la

BDTopo. Pour cela, on crée un nouveau champ dans la table attributaire de cette couche à laquelle on assigne la valeur « 1 » pour l'ensemble des objets.

```
ALTER TABLE theo.bdtopo_routes_valdalliere ADD COLUMN new_id integer ;
UPDATE theo.bdtopo_routes_valdalliere SET new_id =1;
```

Figure 14 : Capture d'écran de la table attributaire de la BDTopo



The screenshot shows a database table named 'bdtopo_routes_valdalliere'. The table has 13 columns: c_postal_d, date_serv, id_voie_g, id_voie_d, id_rn, numero, num_europ, cl_admin, gestion, toponyme, iti_cycl, voie_verte, and new_id. The 'new_id' column is highlighted with a red border. The data consists of approximately 30 rows, each containing various route identifiers and administrative details. The 'new_id' column contains the value '1' for all rows.

	c_postal_d	date_serv	id_voie_g	id_voie_d	id_rn	numero	num_europ	cl_admin	gestion	toponyme	iti_cycl	voie_verte	new_id
1	14410	NULL	14726B303	14726B303									1
2	14410	NULL	147260080	147260080									1
3	14350	NULL	14726#048	14726#048									1
4	14350	NULL	14726#048	14726#048									1
5	14410	NULL			ROUTNOMM00...	D309A		Départementale	Calvados				1
6	14410	NULL											1
7		NULL											1
8	14410	NULL			ROUTNOMM00...	D309A		Départementale	Calvados				1
9	14410	NULL											1
10	14410	NULL											1
11	14350	NULL	14726#01Y	14726#01Y									1
12	14410	NULL											1
13	14410	NULL											1
14	14410	NULL											1
15	14410	NULL											1
16	14410	NULL											1
17	14410	NULL											1
18	14410	NULL	14726#0DK	14726#0DK									1
19	14410	NULL	14726#0D8	14726#0D8									1
20	14410	NULL	14726#0D8	14726#0D8									1
21		NULL											1
22	61800	NULL			ROUTNOMM00...	D18		Départementale	Orne				1
23	14410	NULL	14726#03M	14726#03M									1

6^{ème} étape : Unification de la BDTopo

Une fois ce nouveau champ créé, on utilise la fonction ST_Union afin d'unifier l'ensemble des polylignes de la BDTopo sous un seul et même objet.

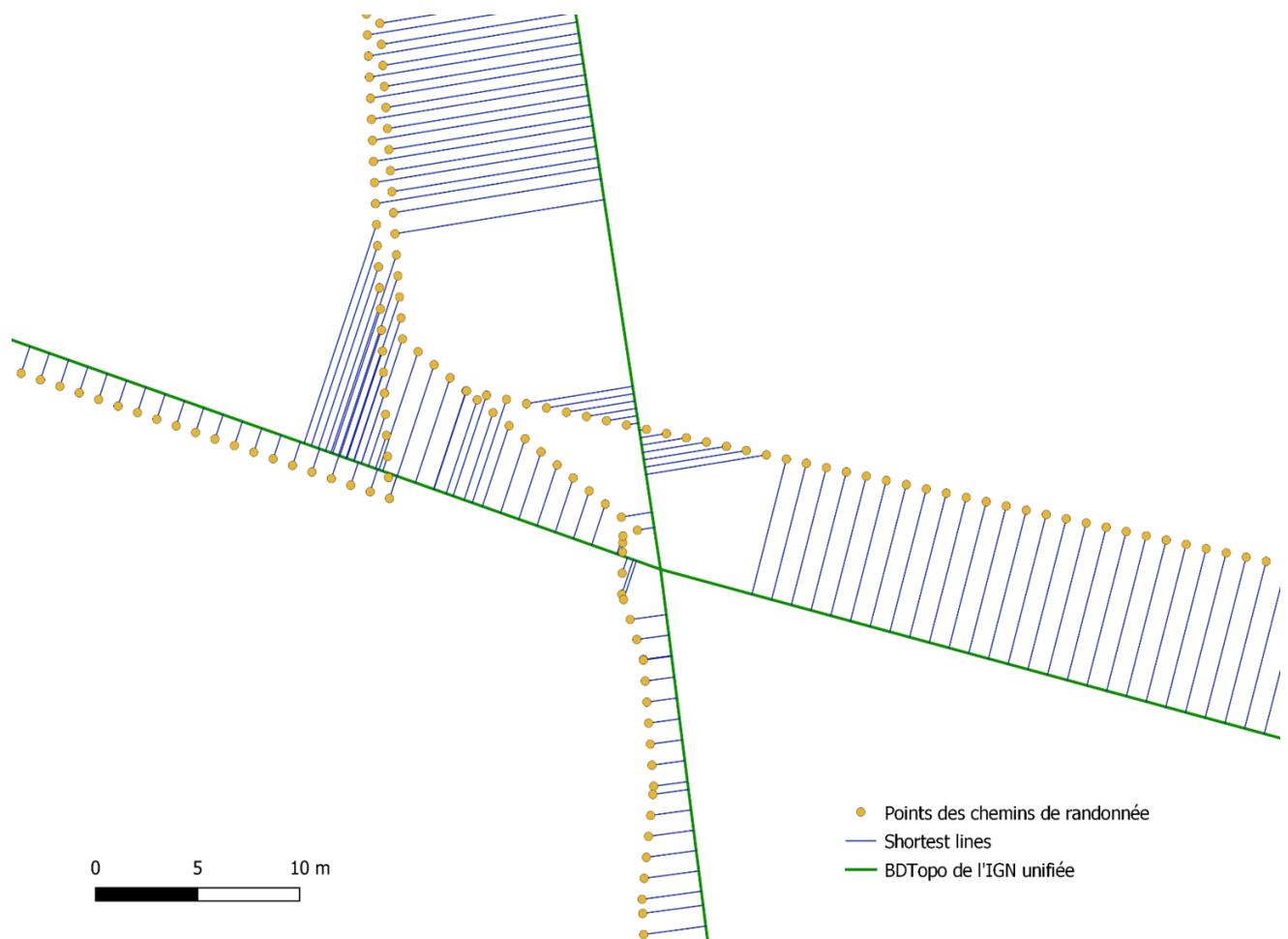
```
CREATE TABLE theo.bdtopo_routes_valdalliere_union AS
SELECT new_id, ST_Union(a.geom) as geom
FROM theo.bdtopo_routes_valdalliere a
GROUP BY new_id;
```

7^{ème} étape : Création des shortest lines entre les points des itinéraires de randonnée et la couche unifiée de la BDTopo

La fonction ST_ShortestLine va créer, pour chaque point de la couche des itinéraires de randonnées, une polyligne représentant la distance la plus proche avec les objets de la couche BDTopo. Il était donc important de n'avoir qu'un seul objet dans la couche de la BDTopo car sinon on aurait une multitude de polylignes pour chaque point.

```
CREATE TABLE theo.shortestlines AS
SELECT ST_ShortestLine(a.geom, b.geom) as geom
FROM theo.chemins_entretenus_valdalliere_points a, theo.bdtopo_routes_vald
alliere_union b;
```

Figure 15 : Création des shortest lines

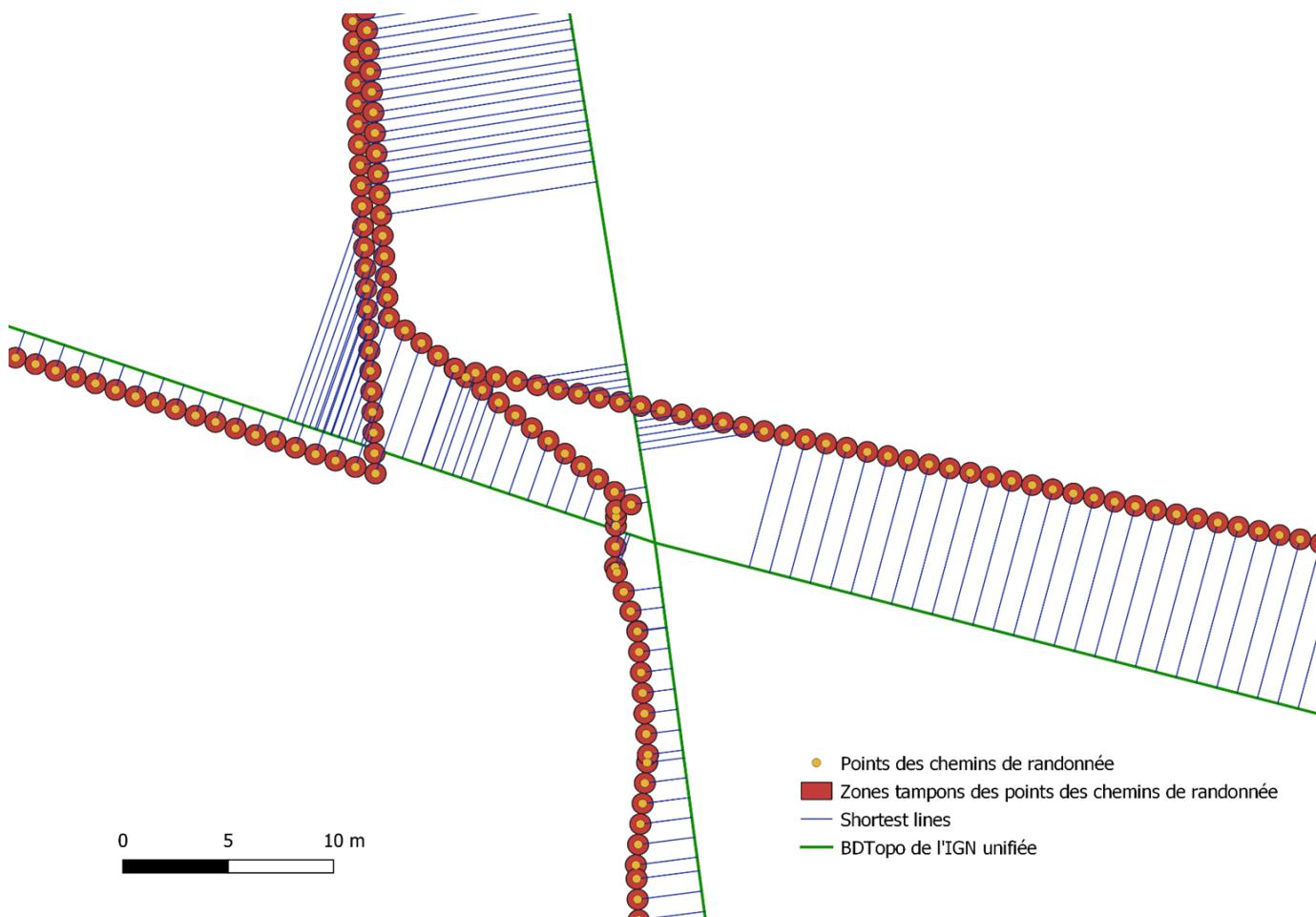


8^{ème} étape : Création de zones tampons autour des points

Pour l'instant, les shortest lines n'ont cependant pas de champs au sein de leur table attributaire. On souhaite donc transférer la table attributaire des points des itinéraires de randonnée vers les shortest lines. Pour cela, on crée, dans un premier temps, avec la fonction ST_Buffer, une zone tampon de 5 cm autour de chaque point qui vont intersecter les shortest lines et nous permettra ensuite de faire une jointure spatiale entre les deux couches.

```
CREATE TABLE theo.chemins_entretenus_valdalliere_points_buffer AS
SELECT a.id, a.maitre_oeu, a.km_aide_dp, a.date_maj, a.nom_epci, ST_Buffer
(a.geom , 0.5)  as geom
FROM theo.chemins_entretenus_valdalliere_points a;
```

Figure 16 : Crédit des zones tampons



9^{ème} étape : Jointure spatiale entre les shortest lines et les zones tampons des points des itinéraires de randonnée

Une fois les zones tampons créées, on fait la jointure spatiale entre la couche des shortest lines et celle des zones tampons à l'aide de la fonction ST_Intersects pour transférer la table de la deuxième couche vers la première.

```
CREATE TABLE theo.shortestlines_join AS
SELECT a.id, a.maitre_oeu, a.km_aide_dp, a.date_maj, a.nom_epci, b.geom as
geom
FROM theo.chemins_entretenus_valdalliere_points_buffer a, theo.shortestli
nes b
WHERE st_intersects(a.geom, b.geom) = true;
```

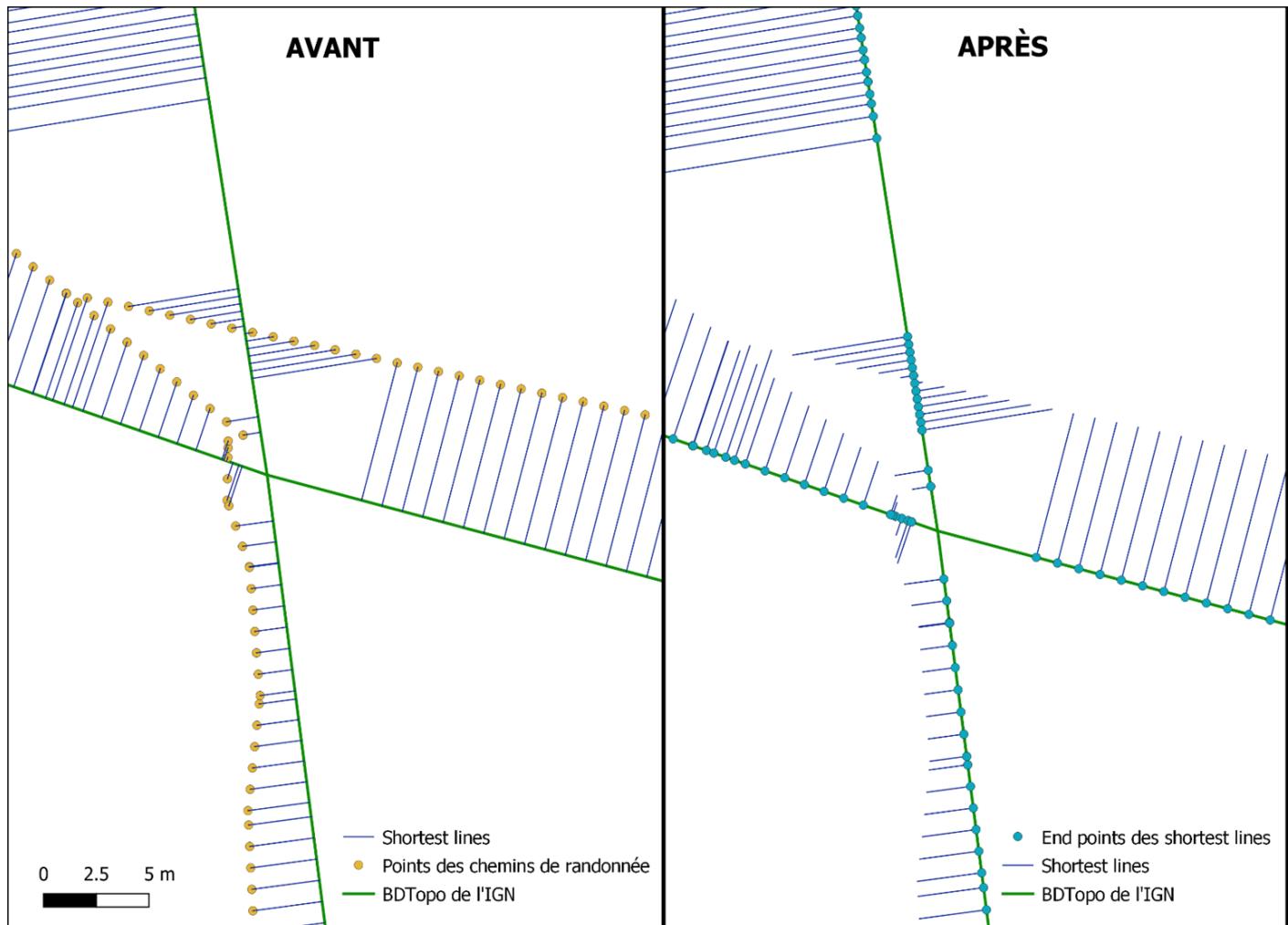
10^{ème} étape : Création de points à la fin des shortest lines

Maintenant que nous avons nos shortest lines, avant de découper la BDTopo, on souhaite ne garder que les polylignes de la BDTopo qui correspondent aux itinéraires de randonnée du relevé GPS. Pour cela, on va recréer les points des itinéraires de randonnée directement sur la BDTopo via la fonction ST_Endpoint qui crée des points à la fin des polylignes.

On utilise la fonction ST_EndPoint pour créer des points à la fin des shortest lines. On peut voir que ces points correspondent aux points des itinéraires de randonnées mais projetés sur la BDTopo de l'IGN. Ces points nous serviront ensuite pour sélectionner les polylignes de la BDTopo qui nous intéressent.

```
CREATE TABLE theo.shortestlines_endpoints AS
SELECT ST_EndPoint(a.geom) as geom
FROM theo.shortestlines_join a;
```

Figure 17 : End points des shortest lines



11^{ème} étape : Création d'une zone tampon autour de la BDTopo

On crée ensuite une zone tampon de 5 cm autour de la couche BDTopo de l'IGN, ce qui nous permettra par la suite de sélectionner les polylinéaires de la BDTopo qui correspondent aux itinéraires de randonnées du relevé GPS.

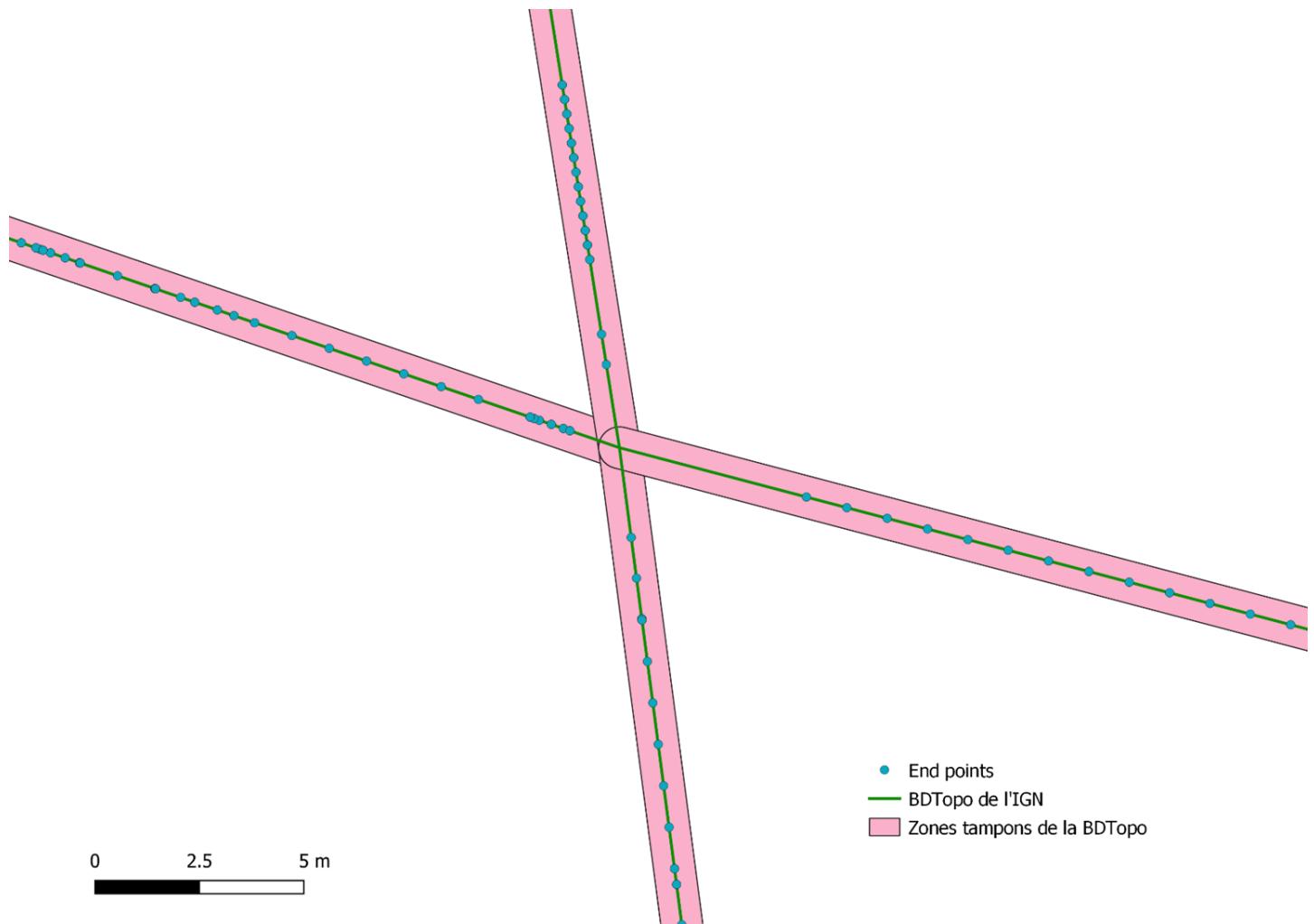
```
CREATE TABLE theo.bdtopo_routes_valdalliere_buffer AS
SELECT id, ST_Buffer(a.geom , 0.5) as geom
FROM theo.bdtopo_routes_valdalliere a;
```

12^{ème} étape : Compte le nombre d'end points à l'intérieur de chaque zone tampon

Une fois la zone tampon créée, on utilise la fonction « count » pour compter le nombre d'end points qui intersectent les zones tampons de la BDTopo. Ainsi, on sait pour chaque segment de la BDTopo si des end points se trouvent dessus et combien.

```
CREATE TABLE theo.bdtopo_routes_valdalliere_count AS
SELECT theo.bdtopo_routes_valdalliere_buffer.geom, count(theo.shortestlines
_endpoints.geom) AS totale
FROM theo.bdtopo_routes_valdalliere_buffer LEFT JOIN theo.shortestlines_end
points
ON st_contains(theo.bdtopo_routes_valdalliere_buffer.geom, theo.shortestli
nes_endpoints.geom)
GROUP BY theo.bdtopo_routes_valdalliere_buffer.geom;
```

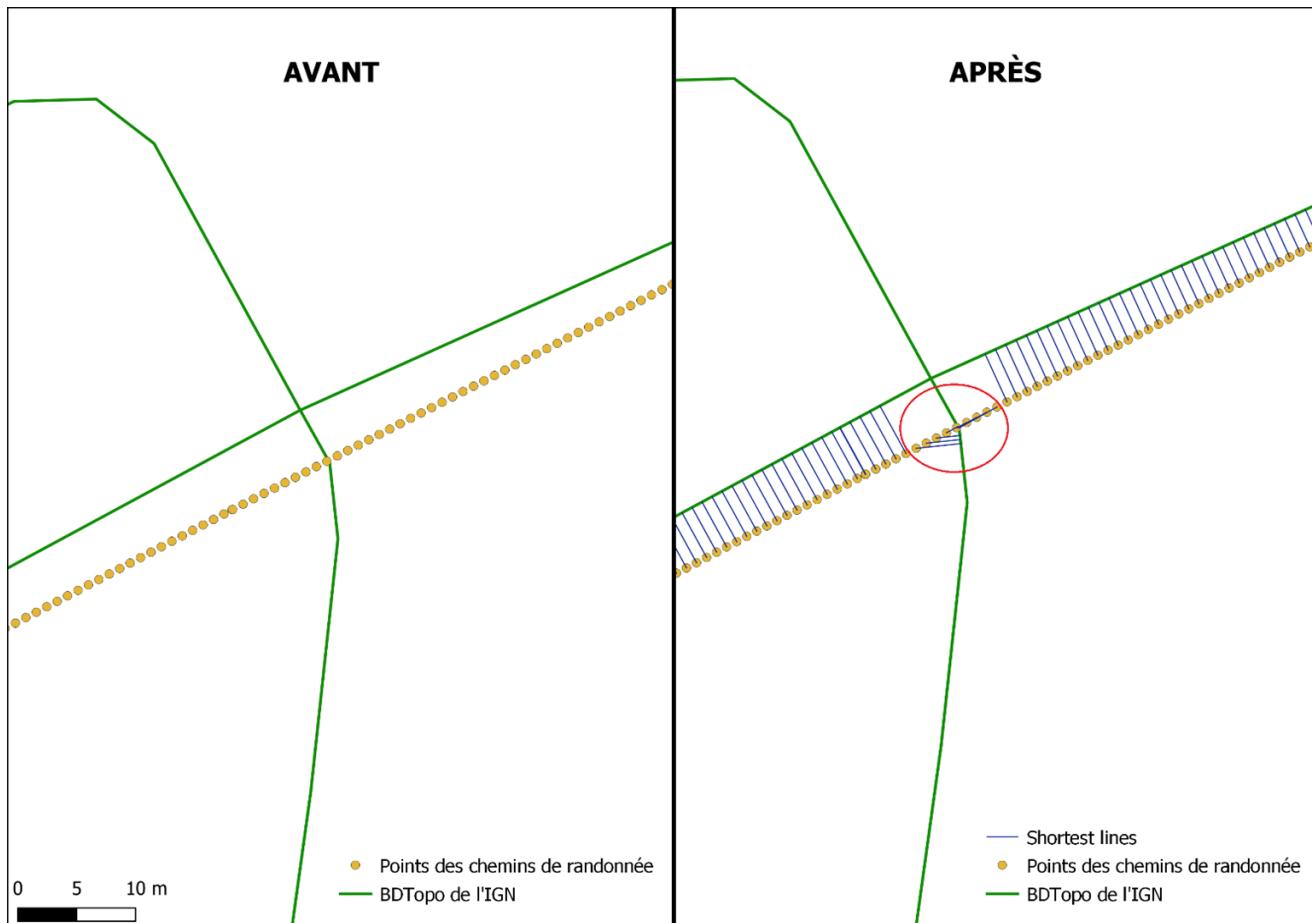
Figure 18 : Zones tampon de la BDTopo



13^{ème} étape : Création d'une nouvelle couche avec les zones tampons qui intersectent plus de 50 end points

L'un des problèmes rencontrés est que les shortest lines ne se sont pas toujours créées par rapport à la polyligne qui correspond au chemin de randonnée mais par rapport à d'autres polylignes plus proches de certains points.

Figure 19 : Exemple d'une erreur lors de la création des shortest lines



Pour corriger ce problème, on ne va sélectionner que les zones tampons de la BDTopo qui intersectent plus de 50 points, ce qui nous servira à éliminer plus tard les polylignes qui ne correspondent pas aux itinéraires de randonnée. Une fois que l'on sait le nombre d'end points qu'intersectent chaque zone tampon donc, on ne sélectionne que les zones tampons qui intersectent plus de 50 end points.

```
CREATE TABLE theo.bdtopo_routes_valdalliere_50points AS
SELECT a.geom
FROM theo.bdtopo_routes_valdalliere_count a
WHERE a.totale > 50;
```

14^{ème} étape : Découpe de la BDTopo avec les shortest lines

On découpe maintenant la BDTopo de l'IGN en différents segments à l'aide des shortest lines afin de pouvoir, par la suite, ne garder précisément que les segments qui correspondent aux itinéraires de randonnée.

15^{ème} étape : Sélection spatiale des segments de la BDTopo correspondant aux itinéraires de randonnée

A l'aide des zones tampons de plus de 50 end points, on fait une sélection spatiale afin de sélectionner les segments de la BDTopo découpée qui correspondent aux itinéraires de randonnée.

```
CREATE TABLE theo.chemins_entretenus_valdalliere_true AS
SELECT a.*
FROM theo.bdtopo_routes_valdalliere a, theo.bdtopo_routes_valdalliere_50po
ints b
WHERE ST_Within(a.geom, b.geom) = true;
```

16^{ème} étape : Unifie les itinéraires de randonnée

Désormais, il ne reste plus qu'à unifier les segments entre eux pour reconstituer les relevés GPS des itinéraires de randonnée avec la bonne géométrie.

```
CREATE TABLE theo.chemins_entretenus_valdalliere_final AS
SELECT id, maitre_oeu, km_aide_dp, date_maj, nom_epci, ST_Union(a.geom) as
geom
FROM theo.bdtopo_routes_valdalliere_true a
GROUP BY id, maitre_oeu, km_aide_dp, date_maj, nom_epci;
```

Figure 20 : Résultat final

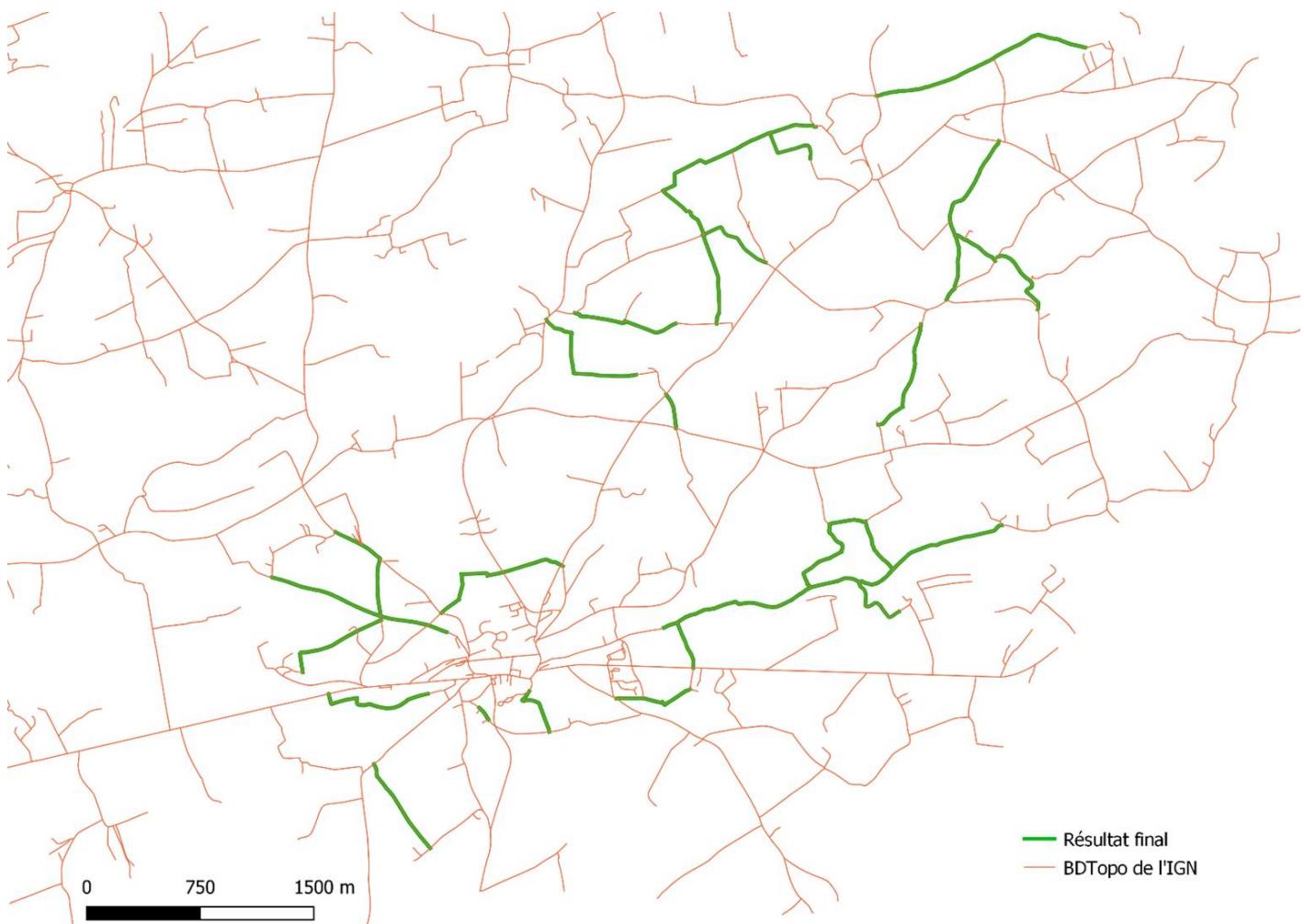


Figure 21 : Résultat final bis



Conclusion

L'utilisation de la fonction ST_ShortestLine nous donne un résultat satisfaisant avec un très faible pourcentage d'erreurs. La géométrie des itinéraires de randonnée est bien corrigée en dehors de quelques segments qui manquent parfois au résultat final.

Partie II : Mise en place de géotraitements avec PGRouting



Contexte : Intégrer la bibliothèque de fonctions PGRouting au SIG du Calvados

PGRouting est une extension de PostGIS proposant des fonctionnalités de routage géospatial et d'autres analyses de réseaux. Il permet ainsi de réaliser plusieurs géotraitements sur des polylignes : création de graphiques topologiques, calcul de l'itinéraire le plus court entre deux points ou tronçons de route, création d'isodistances...

Le Pôle SIG compte utiliser cette extension pour de futurs projets et souhaite auparavant réaliser des essais afin de mieux comprendre le fonctionnement des différentes fonctions de PGRouting.

Objectifs : Créer un itinéraire entre deux points et des isodistances depuis un point

Il y a ici deux objectifs :

1. Évaluer la création d'itinéraires : le Pôle SIG souhaite trouver un outil pour rechercher la plus courte distance entre deux emplacements du réseau de randonnées.
2. Crée des isodistances dans un graphe : Dans le cadre de la D.E.C.I (Défense Extérieure Contre les Incendies), le Pôle SIG cherche à représenter géographiquement les zones du réseau routier départemental se trouvant à moins de 200m des P.E.I (Points d'Eau Incendie). Pour cela, il faut pouvoir créer des isodistances autour de points.

Zone géographique test :

Pour faciliter les essais, ces derniers se concentreront sur une petite échelle : la commune de Bretteville-sur-Odon.

1. Crée un itinéraire entre deux points - Solution proposée : Utilisation de pg_dijkstra

L'algorithme Dijkstra a été conçu par l'informaticien Néerlandais Edsger Dijkstra en 1956. C'est un algorithme de recherche de graphe qui résout le problème de plus court chemin à source unique pour un graphe à coûts de chemin non négatifs. La fonction pg_dijkstra a ainsi pour but de calculer la

distance la plus courte entre deux sources (par exemple entre deux points ou deux polylignes) ou au sein d'un graphe.

Ainsi, on va, dans un premier temps, chercher à créer une polyligne représentant le chemin le plus court entre deux sources puis, dans un second temps, créer un isodistance autour de cette polyligne.

L'ensemble du traitement a été réalisé via un traitement automatique sur PostGIS :

1ère étape : Correction de la couche de BDTopo des routes

Le premier problème rencontré est que le champ « id » de la BDTopo des tronçons de routes est de type « *character varying* » alors que nous avons besoin d'un champ id de type « *integer* ». On crée donc un nouveau champ dans la table attributaire.

```
ALTER TABLE theo.bdtopo_troncon_route_bretteville ADD COLUMN new_id integer;
```

La création du nouvel id incrémenté se fait de manière manuel via la fonction « \$id » dans la calculatrice de champ de QGIS.

Figure 22 : Création d'un champ « new_id »

alias_d	c_postal_g	c_postal_d	date_serv	id_voie_g	id_voie_d	id_rn	numero	num_europ	cl_admin	gestion	toponyme	iti_cycl	voie_verte	new_id
				NULL										1
14760	14760			NULL										10
14760	14760			NULL	14101#004	14101#004								100
14760	14760			NULL	141010606	141010606								101
14760	14760			NULL	141010534	141010534								102
14760	14760			NULL	141010660	141010660								103
14760	14760			NULL	141010420	141010420								104
14760	14760			NULL	141010500	141010500								105
14760	14760			NULL	141010725	141010725								106
14760	14760			NULL	141010320	141010320								107
14760	14760			NULL										108
14760	14760			NULL	141010365	141010365								109
				NULL	14101#005	14101#005								110
14760	14760			NULL	141011053	141011053	ROUTNOMM00... D220		Départementale	Calvados				110
14760	14760			NULL	141010084	141010084								111
14760	14760			NULL	141010955	141010955								112
14760	14760			NULL	141010721	141010721								113
14760	14760			NULL	141010350	141010350								114
14760	14760			NULL	141011053	141011053	ROUTNOMM00... D220		Départementale	Calvados				115
14760	14760			NULL										116
14760	14760			NULL	141010196	141010196								117
14760	14760			NULL	141010533	141010533								118
14760	14760			NULL	141010533	141010533								119

2ème étape : Conversion de la couche BDTopo des routes en 2D

Autre problème avec la couche des routes : celle-ci est en 3D alors que le traitement nécessite d'avoir une couche 2D. On convertit donc la couche à l'aide de la fonction ST_Force2D.

```
CREATE TABLE theo.bdtopo_troncon_route_bretteville_2D AS
SELECT new_id AS id, nature, nom_1_g, nom_1_d, nom_2_g, nom_2_d, importance,
fictif, pos_sol, etat, date_creat, date_maj, date_app, date_conf, id_source,
prec_plani, prec_alti, nb_voies, largeur, it_vert, prive, sens, cyclable,
bus, urbain, vit_moy_vl, acces_vl, acces_ped, fermeture, nat_restr,
restr_h, restr_p, restr_ppe, restr_lar, restr_lon, restr_mat, bornedeb_g,
bornedeb_d, bornefin_d, bornefin_g, inseecom_g, inseecom_d, typ_adres, alias_g,
alias_d, c_postal_g, c_postal_d, date_serv, id_voie_g, id_voie_d, id_ln,
numero, num_europ, cl_admin, gestion, toponyme, iti_cycl, voie_verte,
ST_Force2D(geom) AS geom
FROM theo.bdtopo_troncon_route_bretteville;
```

3ème étape : Création des champs « source » et « target »

On rajoute les champs « source » et « target » dans la table attributaire de la BDTopo des routes qui nous serviront à créer un graphe topologique par la suite.

```
ALTER TABLE theo.bdtopo_troncon_route_bretteville_2D add column "source" integer;
ALTER TABLE theo.bdtopo_troncon_route_bretteville_2D add column "target" integer;
```

4ème étape : Création d'un graphe topologique de points représentant les nœuds du réseau de routes de la BDTopo

On utilise la fonction « pgr_createTopology » de PGRouting afin de créer un graphe topologique à partir de la couche 2D de la BDTopo de l'IGN.

```
select pgr_createTopology('theo.bdtopo_troncon_route_bretteville_2d', 0.0005, 'geom', 'id');
```

On voit ainsi que les champs « source » et « target » de la table attributaire de la couche 2D de la BDTopo sont désormais remplies.

Figure 23 : Capture d'écran de la table attributaire du graphe topologique

date_serv	id_voie_g	id_voie_d	id_m	numero	num_europ	cl_admin	gestion	toponyme	iti_cycl	voie_verte	source	target
NULL	141010196	141010196									76	73
NULL	141010955	141010955									214	215
NULL	141010725	141010725									66	327
NULL	141010084	141010084									192	213
NULL											246	247
NULL	141010721	141010721									78	216
NULL	141010320	141010320									207	208
NULL	141010725	141010725									198	59
NULL	141011053	141011053	ROUTNOMM00...	D220		Départementale	Calvados				211	212
NULL	141010365	141010365									209	210
NULL	141010660	141010660									202	203
NULL	141010500	141010500									206	67
NULL	141010420	141010420									204	205
NULL											348	347
NULL											477	348
NULL											589	590
NULL	14101#00V	14101#00V									583	584
NULL	14101#00T	14101#00T									581	1
NULL			ROUTNOMM00...	N814	E401	Nationale	DIR Nord Ouest	Autoroute des ...			587	390
NULL	141011053	141011053	ROUTNOMM00...	D220		Départementale	Calvados				495	582
NULL	14101#00U	14101#00U									578	579
NULL	14101#00S	14101#00S									576	577
NULL	141011053	141011053	ROUTNOMM00...	D220		Départementale	Calvados					

5ème : Création d'un champ de longueur dans la BDTopo

On crée un nouveau champ « length_m » dans la table attributaire de la BDTopo qui va calculer la distance de chaque tronçon.

```
alter table theo.bdtopo_troncon_route_bretteville_2d add column length_m integer;
update theo.bdtopo_troncon_route_bretteville_2d set length_m = st_length(st_transform(geom,2154));
```

Figure 24 : Crédation du champ « length_m »

gestion	toponyme	iti_cycl	voie_verte	source	target	length_m
				76	73	48
				214	215	34
				66	327	18
				192	213	99
				246	247	37
				78	216	71
				207	208	95
				198	59	62
Calvados				211	212	82
				209	210	164
				202	203	70
				206	67	11
				204	205	151
				348	347	60
				477	348	272
				589	590	305
				583	584	22
				581	1	93
DIR Nord Ouest	Autoroute des ...			587	390	34
Calvados				495	582	92
				578	579	14
				576	577	65
				582	70	61

6ème étape : Utilisation de « pgr_dijkstra »

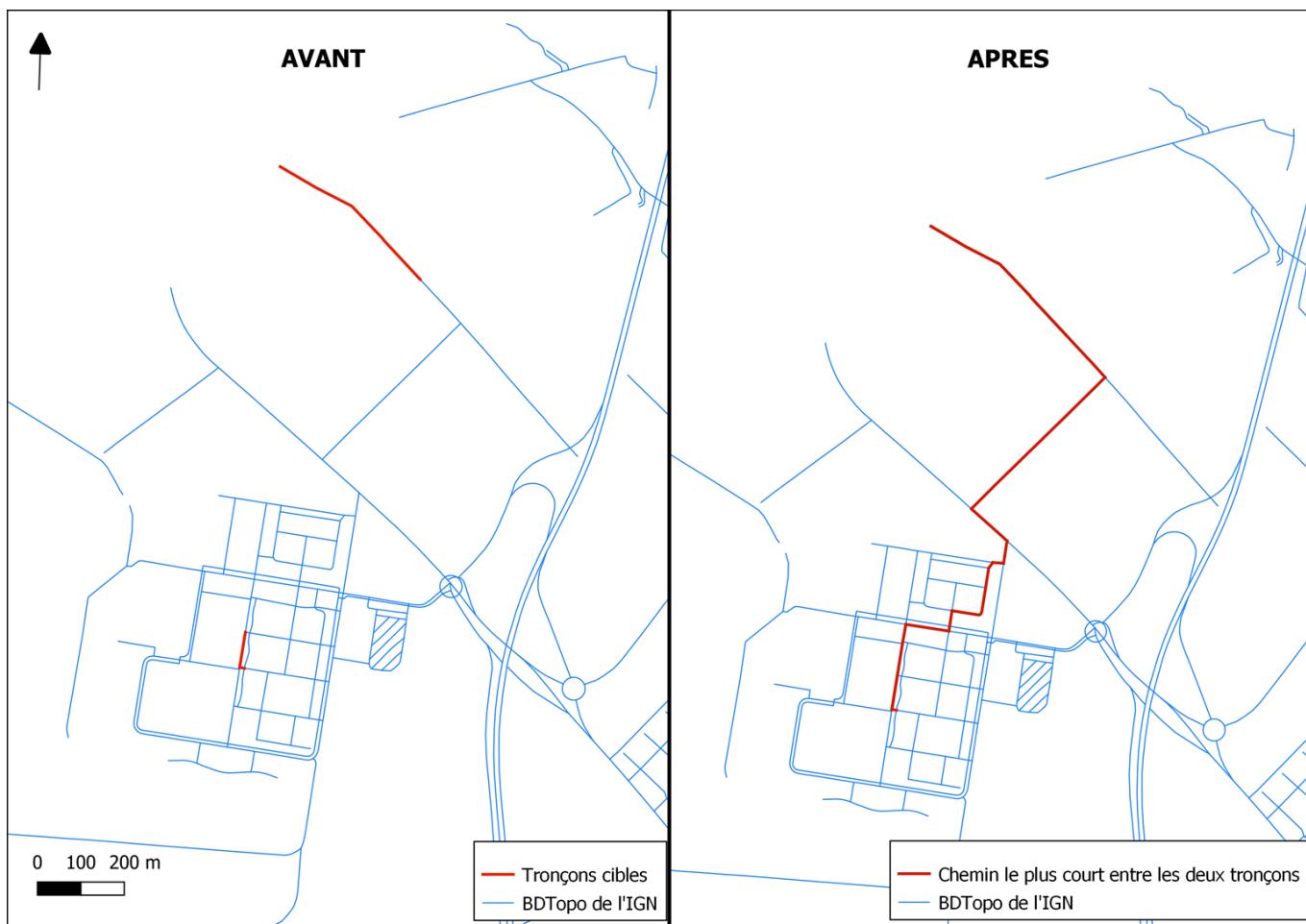
On utilise la fonction « pgr_dijkstra » de PGRouting afin de trouver le chemin le plus court entre deux tronçons. Ici on a choisi les tronçons ayant pour identifiant source « 373 » et « 673 » pour l'exemple.

```

SELECT seq, edge, b.geom AS "geom"
  FROM pgr_dijkstra(
    SELECT id as id, source, target,
           length_m as cost FROM theo.bdtopo_troncon_route_br
      etteville_2d',373, 673, false
) a INNER JOIN theo.bdtopo_troncon_route_bretteville_2d b ON (a.ed
ge = b.id) ORDER BY seq;

```

Figure 26 : Résultat final - Tracé du chemin le plus court entre deux tronçons



Conclusion

La fonction « pgr_dijkstra » permet effectivement de chercher le chemin le plus entre deux emplacements. Cette fonction pourra servir pour mettre en place une application en lien avec la base de donnée des itinéraires de randonnée.

2. Créer des isodistances - Solution proposée : Utilisation de « pgr_drivingdistance »

L'objectif ici est d'utiliser la BDTopo de l'IGN et les P.E.I de Bretteville-sur-Odon afin de créer des isodistances représentant la distance à partir d'un point. La principale différence entre une isodistance et une zone tampon est que l'isodistance se base sur un graphe. C'est à dire que ces isodistances donneront la distance exacte autour d'un point au sein de la BDTopo des routes de l'IGN.

Pour réaliser cette tâche, PGRouting dispose de la fonction « pgr_drivingdistance » permettant de sélectionner les tronçons se trouvant à une certaine distance d'un point.

L'ensemble du traitement a été réalisé via un traitement automatique sur PostGIS :

1ère étape : Conversion de la couche BDTopo des routes en 2D

Autre problème avec la couche des routes : celle-ci est en 3D alors que le traitement nécessite d'avoir une couche 2D. On convertit donc la couche à l'aide de la fonction ST_Force2D.

```
CREATE TABLE theo.bdtopo_troncon_route_bretteville_2D AS
SELECT new_id as id, nature, nom_1_g, nom_1_d, nom_2_g, nom_2_d, importance,
fictif, pos_sol, etat, date_creat, date_maj, date_app, date_conf, id_source,
prec_plani, prec_alti, nb_voies, largeur, it_vert, prive, sens, cyclable,
bus, urbain, vit_moy_vl, acces_vl, acces_ped, fermeture, nat_restr,
restr_h, restr_p, restr_ppe, restr_lar, restr_lon, restr_mat, bornedeb_g,
bornedeb_d, bornefin_d, bornefin_g, inseecom_g, inseecom_d, typ_adres, alias_g,
alias_d, c_postal_g, c_postal_d, date_serv, id_voie_g, id_voie_d, id_ln,
numero, num_europ, cl_admin, gestion, toponyme, iti_cycl, voie_verte,
ST_Force2D(geom) AS geom
FROM theo.bdtopo_troncon_route_bretteville;
```

2ème étape : Création d'un graphe de points représentant le début et la fin des polylignes de la BDTopo.

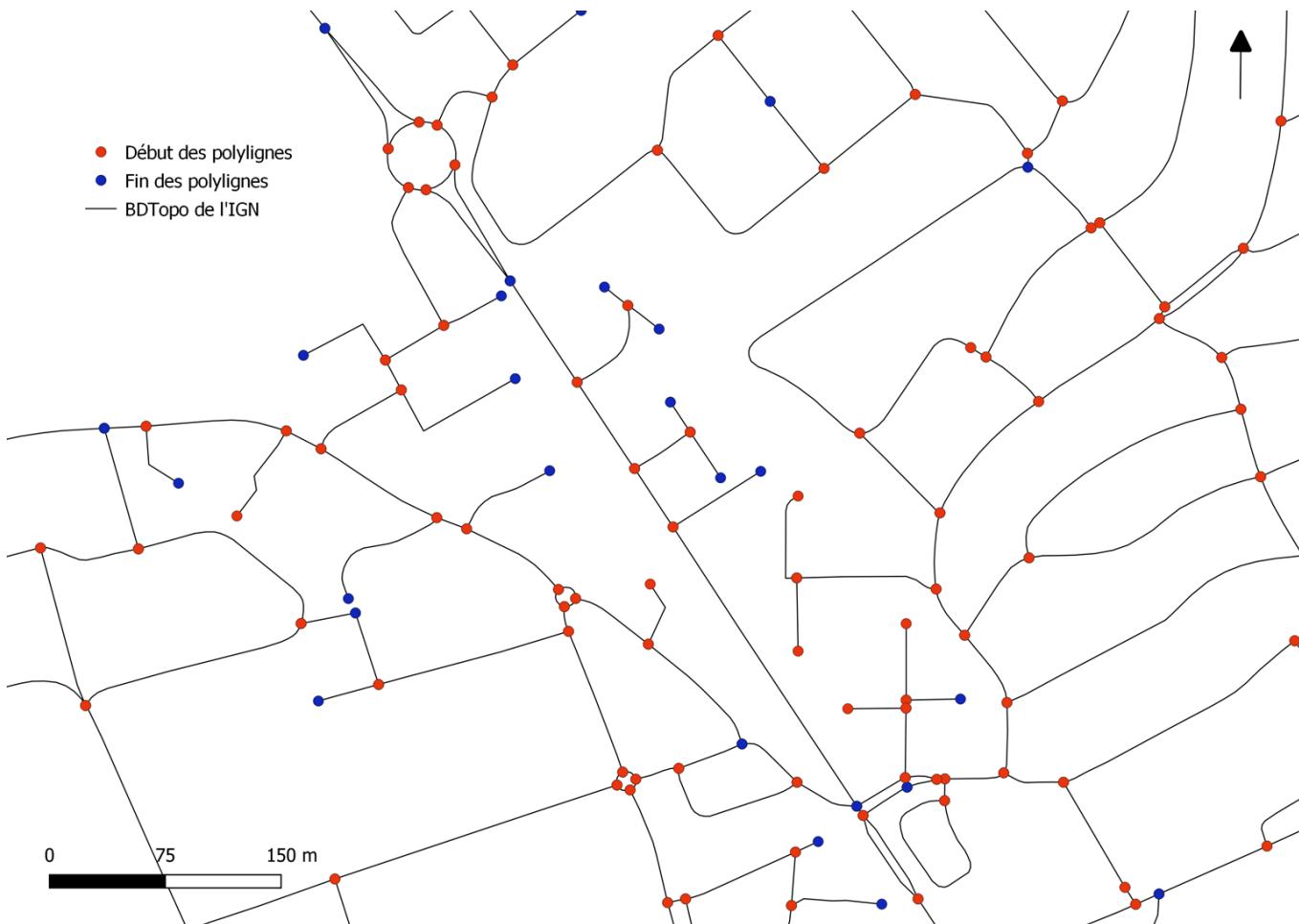
On utilise les fonctions ST_StartPoint et ST_EndPoint afin de créer une couche « bdtopo_troncon_route_bretteville_2d_ext » contenant deux graphes de points : l'un représentant le début des polylignes et l'autre la fin des polylignes de la BDTopo. On crée, en même temps, un nouveau graphe contenant les informations sur ces points.

```

CREATE OR REPLACE TABLE theo.bdtopo_troncon_route_bretteville_2d_ext AS
SELECT *, ST_StartPoint(geom), ST_EndPoint(geom)
FROM theo.bdtopo_troncon_route_bretteville_2d;

```

Figure 26 : Création de points au début et à la fin des polylignes de la BDTopo



3ème étape : Création d'un nouveau graphe de points réunissant les deux graphes

On crée un nouveau graphe de points « node » qui réunit les points de début et de fin des polylignes.

```
CREATE TABLE theo.node AS
```

```
SELECT row_number() OVER (ORDER by foo.p)::integer AS id,  
      foo.p AS geom  
FROM (  
    SELECT DISTINCT theo.bdtopo_troncon_route_bretteville_2d_ext.ST_StartPoint AS p FR  
    OM theo.bdtopo_troncon_route_bretteville_2d_ext  
    UNION  
    SELECT DISTINCT theo.bdtopo_troncon_route_bretteville_2d_ext.ST_EndPoint AS p FR  
    OM theo.bdtopo_troncon_route_bretteville_2d_ext  
) foo  
GROUP BY foo.p;
```

Figure 27 : On joint les nœuds de polylinéaires au sein d'une même couche



4ème étape : Création d'un graphe topologique contenant les informations des nœuds

On crée un graphe topologique, « network », à laquelle on joint la table attributaire de la couche des points des nœuds des polylignes (points du début et de la fin des polylignes).

```
CREATE TABLE theo.network AS
```

```
SELECT a.* , b.id as start_id, c.id as end_id
FROM theo.bdtopo_troncon_route_bretteville_2d_ext AS a
JOIN theo.node AS b ON a.ST_Startpoint = b.geom
JOIN theo.node AS c ON a.ST_EndPoint = c.geom;
```

Figure 28 : Table attributaire de la couche « network »

target	length_m	traveltime_min	st_startpoint	st_endpoint	start_id	end_id
62	21	0,0252	SRID=2154;POI...	SRID=2154;POI...	542	527
176	59	0,0708	SRID=2154;POI...	SRID=2154;POI...	519	528
289	49	0,0588	SRID=2154;POI...	SRID=2154;POI...	519	529
289	81	0,0972	SRID=2154;POI...	SRID=2154;POI...	517	529
461	247	0,2964	SRID=2154;POI...	SRID=2154;POI...	572	526
461	47	0,0564	SRID=2154;POI...	SRID=2154;POI...	524	526
62	123	0,1476	SRID=2154;POI...	SRID=2154;POI...	511	527
641	384	0,4608	SRID=2154;POI...	SRID=2154;POI...	463	535
641	460	0,552	SRID=2154;POI...	SRID=2154;POI...	559	535
300	47	0,0564	SRID=2154;POI...	SRID=2154;POI...	535	536
501	15	0,018	SRID=2154;POI...	SRID=2154;POI...	536	537
311	52	0,0624	SRID=2154;POI...	SRID=2154;POI...	528	532
311	105	0,126	SRID=2154;POI...	SRID=2154;POI...	554	532
441	326	0,3912	SRID=2154;POI...	SRID=2154;POI...	486	533
441	220	0,264	SRID=2154;POI...	SRID=2154;POI...	537	533
47	12	0,0144	SRID=2154;POI...	SRID=2154;POI...	73	74
47	104	0,1248	SRID=2154;POI...	SRID=2154;POI...	118	74
48	38	0,0456	SRID=2154;POI...	SRID=2154;POI...	74	75
48	71	0,0852	SRID=2154;POI...	SRID=2154;POI...	68	75
12	34	0,0408	SRID=2154;POI...	SRID=2154;POI...	67	71
16	104	0,1248	SRID=2154;POI...	SRID=2154;POI...	47	73
16	46	0,0552	SRID=2154;POI...	SRID=2154;POI...	59	73
470	20	0,0240				

5ème étape : Calcul du temps de parcours des tronçons de route

On ajoute d'abord deux champs à la table attributaire de la couche « network » : « len_km » et « time_driving ».

```
ALTER TABLE theo.network add column len_km double precision
```

```
ALTER TABLE theo.network add column time_driving double precision
```

Dans le champ « len_km », on calcule la distance de chaque polyligne du graphe topologique.

```
UPDATE theo.network set len_km = (ST_Length(geom))/1000;
```

Dans le champ « time_driving », on calcule le temps qu'il faut pour parcourir chaque polyligne en km/h.

```
UPDATE theo.network set time_driving = ((60/5)::double precision)*len_km;
```

Figure 29 : Crédation des champs « len_km » et « time_driving »

length_m	traveltime_min	st_startpoint	st_endpoint	start_id	end_id	len_km	time_driving
21	0,0252	SRID=2154;POI...	SRID=2154;POI...	542	527	0,021328640791...	0,255943689493...
59	0,0708	SRID=2154;POI...	SRID=2154;POI...	519	528	0,059263901322...	0,711166815867...
49	0,0588	SRID=2154;POI...	SRID=2154;POI...	519	529	0,048575514900...	0,582906178804...
81	0,0972	SRID=2154;POI...	SRID=2154;POI...	517	529	0,081049176402...	0,972590116831...
247	0,2964	SRID=2154;POI...	SRID=2154;POI...	572	526	0,247452595578...	2,96943114694522
47	0,0564	SRID=2154;POI...	SRID=2154;POI...	524	526	0,046508320997...	0,55809985197041
123	0,1476	SRID=2154;POI...	SRID=2154;POI...	511	527	0,123207603897...	1,47849124676872
384	0,4608	SRID=2154;POI...	SRID=2154;POI...	463	535	0,383771856176...	4,60526227411347
460	0,552	SRID=2154;POI...	SRID=2154;POI...	559	535	0,459632374693...	5,51558849632372
47	0,0564	SRID=2154;POI...	SRID=2154;POI...	535	536	0,047062871482...	0,564754457788...
15	0,018	SRID=2154;POI...	SRID=2154;POI...	536	537	0,014525328290...	0,17430393948112
52	0,0624	SRID=2154;POI...	SRID=2154;POI...	528	532	0,051666281510...	0,619995378124...
105	0,126	SRID=2154;POI...	SRID=2154;POI...	554	532	0,104863590962...	1,25836309155306
326	0,3912	SRID=2154;POI...	SRID=2154;POI...	486	533	0,325594435730...	3,90713322876653
220	0,264	SRID=2154;POI...	SRID=2154;POI...	537	533	0,219712636016...	2,63655163219334
12	0,0144	SRID=2154;POI...	SRID=2154;POI...	73	74	0,011738824472...	0,140865893667...
104	0,1248	SRID=2154;POI...	SRID=2154;POI...	118	74	0,103933302186...	1,24719962623216
38	0,0456	SRID=2154;POI...	SRID=2154;POI...	74	75	0,037649435586...	0,451793227043...
71	0,0852	SRID=2154;POI...	SRID=2154;POI...	68	75	0,070614260874...	0,847371130488...
34	0,0408	SRID=2154;POI...	SRID=2154;POI...	67	71	0,034182597911...	0,410191174940...
104	0,1248	SRID=2154;POI...	SRID=2154;POI...	47	73	0,103954316889...	1,24745180267561
46	0,0552	SRID=2154;POI...	SRID=2154;POI...	59	73	0,046476875971...	0,557722511653...
26	0,0248	SRID=2154;POI...	SRID=2154;POI...	545	546		

6ème étape : Crédation du champ « nearest_node »

Cette fois ci c'est dans la couche des points P.E.I que l'on ajoute un champ « nearest_node » qui servira à calculer quels nœuds des polylignes de la BDTopo sont les plus proches de chaque P.E.I.

```
ALTER TABLE theo.sdis_points_eau_incendie_bretteville
ADD Column nearest_node integer
```

7ème étape : Calcul de la distance entre les P.E.I et les nœuds des polylignes les plus proches

Dans un nouveau champ « temp », on calcule la distance entre chaque P.E.I et le nœud des polylignes qui lui est le plus proche.

```
CREATE TABLE theo.temp AS
SELECT
    DISTINCT ON (a.id) a.id, b.id AS start_id,
    ST_Distance(a.geom, b.geom) AS dist
FROM theo.sdis_points_eau_incendie_bretteville a, theo.node b
ORDER BY a.id, ST_Distance(a.geom, b.geom), b.id;
```

8ème étape : Mise à jour de la couche des P.E.I

On remplit le champ « nearest_node » de la couche des P.E.I avec la table « temp ».

```
UPDATE theo.sdis_points_eau_incendie_bretteville
SET nearest_node =
(SELECT start_id
FROM theo.temp
WHERE temp.id = theo.sdis_points_eau_incendie_bretteville.id);
```

Figure 30 : Table attributaire de la couche des P.E.I

comm_cod	etat_cod	debit_1_ba	capacite	date_pesee	date_ctrl	nearest_node
101	DI	65	NULL	2017-11-15	2016-03-09	226
101	DI	94	NULL	2017-11-15	2016-03-09	272
101	DI	49	NULL	2017-11-15	2016-03-09	324
101	DI	155	NULL	2017-11-15	2016-03-10	423
101	DI	84	NULL	2017-11-15	2016-03-07	408
101	DI	92	NULL	2017-11-15	2016-03-07	434
101	DI	89	NULL	2017-11-15	2016-03-09	297
101	DI	66	NULL	2017-11-15	2016-03-09	243
101	DI	103	NULL	2017-11-15	2016-03-09	217
101	IN	17	NULL	2017-11-27	2016-03-07	525
101	DI	44	NULL	2017-11-15	2016-03-07	593
101	DI	120	NULL	NULL	2016-03-10	431
101	DI	111	NULL	2017-11-15	2016-03-09	208
101	DI	84	NULL	2017-11-15	2016-03-09	347
101	DI	82	NULL	2017-11-15	2016-03-09	358
101	DI	77	NULL	2017-11-15	2016-03-10	224
101	DI	72	NULL	2017-11-15	2016-03-07	484
101	DI	83	NULL	2017-11-15	2016-03-07	518
101	DI	77	NULL	2017-11-15	2016-03-07	461
101	DI	67	NULL	2017-11-15	2016-03-07	552
101	IN	28	NULL	2017-11-15	2016-03-07	577
101	DI	72	NULL	2017-11-15	2016-03-07	618
101	DI	62	NULL	2017-11-15	2016-03-10	574

8ème étape : Sélection des itinéraires se trouvant à moins de 500m d'un nœud sélectionné

On utilise la fonction « pgr_drivingDistance » de PGRouting afin de sélectionner, à partir des points des débuts de polylignes, les itinéraires se trouvant à moins de 500m d'un point sélectionné dans le graphe (ici le nœud sélectionné est celui portant l'identifiant 674).

```

CREATE TABLE theo.ISO_500 AS
SELECT seq, cost, agg_cost, node, edge, l.geom FROM pgr_drivingDistance(
    'SELECT id as id, start_id as source, end_id as target, len_km as
cost FROM theo.network',
    674,0.5, false) AS foo JOIN theo.network l ON l.id = foo.edge
;

```

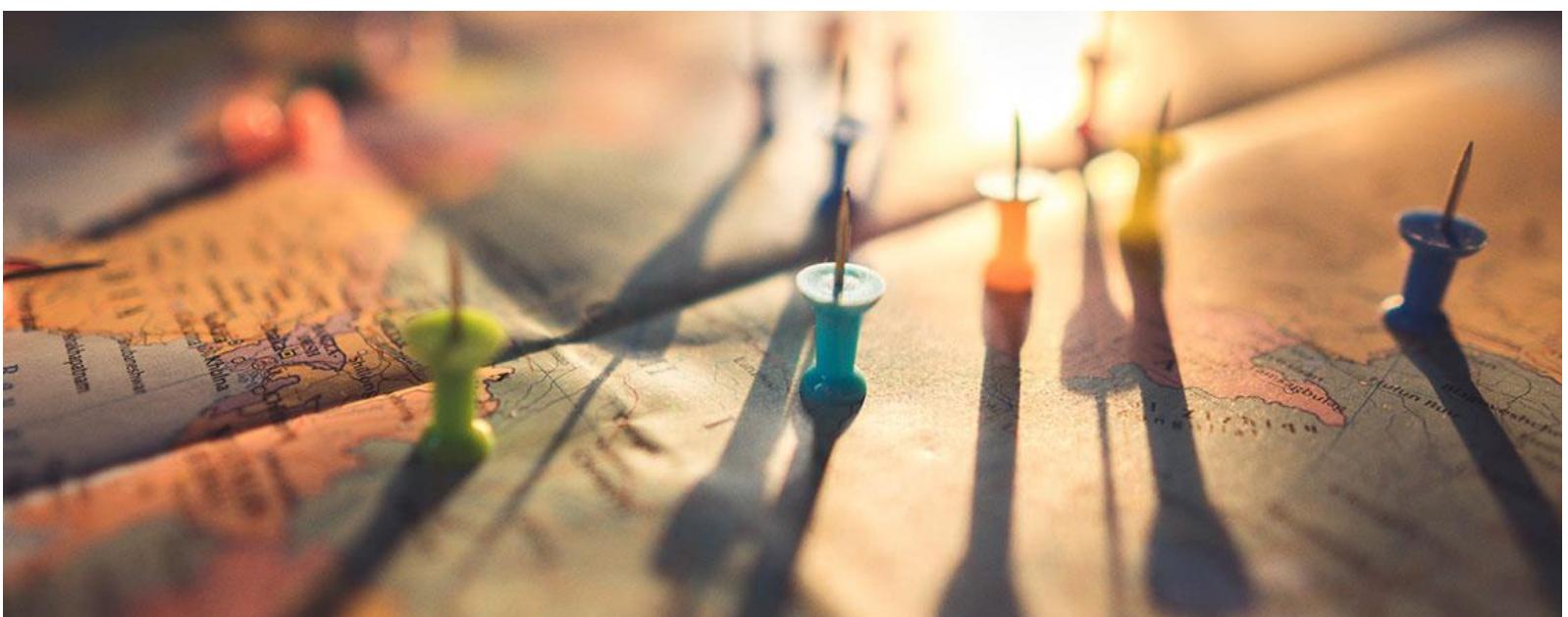
Figure 31 : Sélection des polylignes se trouvant à moins de 500m d'un nœud



Conclusion

L'utilisation de la fonction « pgr_driving » distance de PGRouting a permis de sélectionner les itinéraires permettant d'accéder à un emplacement à partir d'une certaine distance mais n'a cependant pas permis de créer des isodistances.

Partie III : Publier des données en open data sur data.calvados.fr



Contexte : Production d'un catalogue de données en ligne

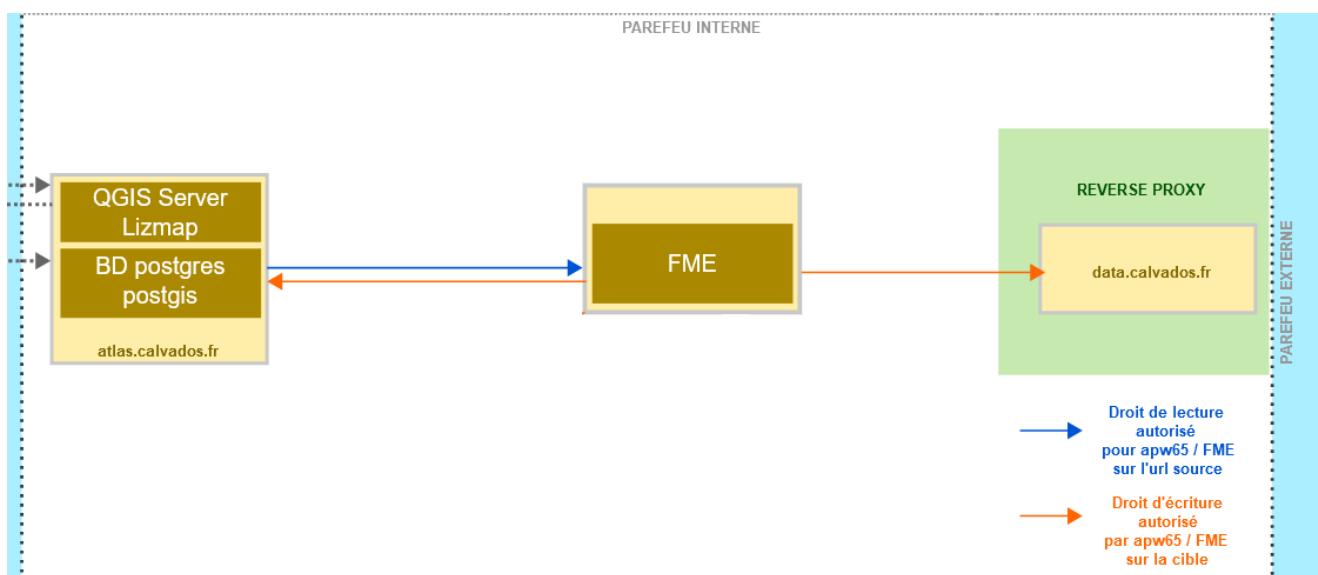
Le Département dispose d'une base de données SIG qui alimente ces différents outils : Mapéo Calvados, SIR, Calvados Atlas. En complément de ces outils, le Département du Calvados a souhaité mettre en ligne un catalogue de données géographiques dont certaines données seraient en libre téléchargement en open data : data.calvados.fr.

Objectif : Automatiser la publication des données géographiques

Automatiser l'extraction des données la base de données SIG pour permettre leur publication sur data.calvados.fr et data.gouv.fr. En outre, on souhaite que ce portail puisse être mis à jour quotidiennement. Il faut donc écrire un script que l'on lancera chaque soir.

Solution proposée : Publication automatique avec FME

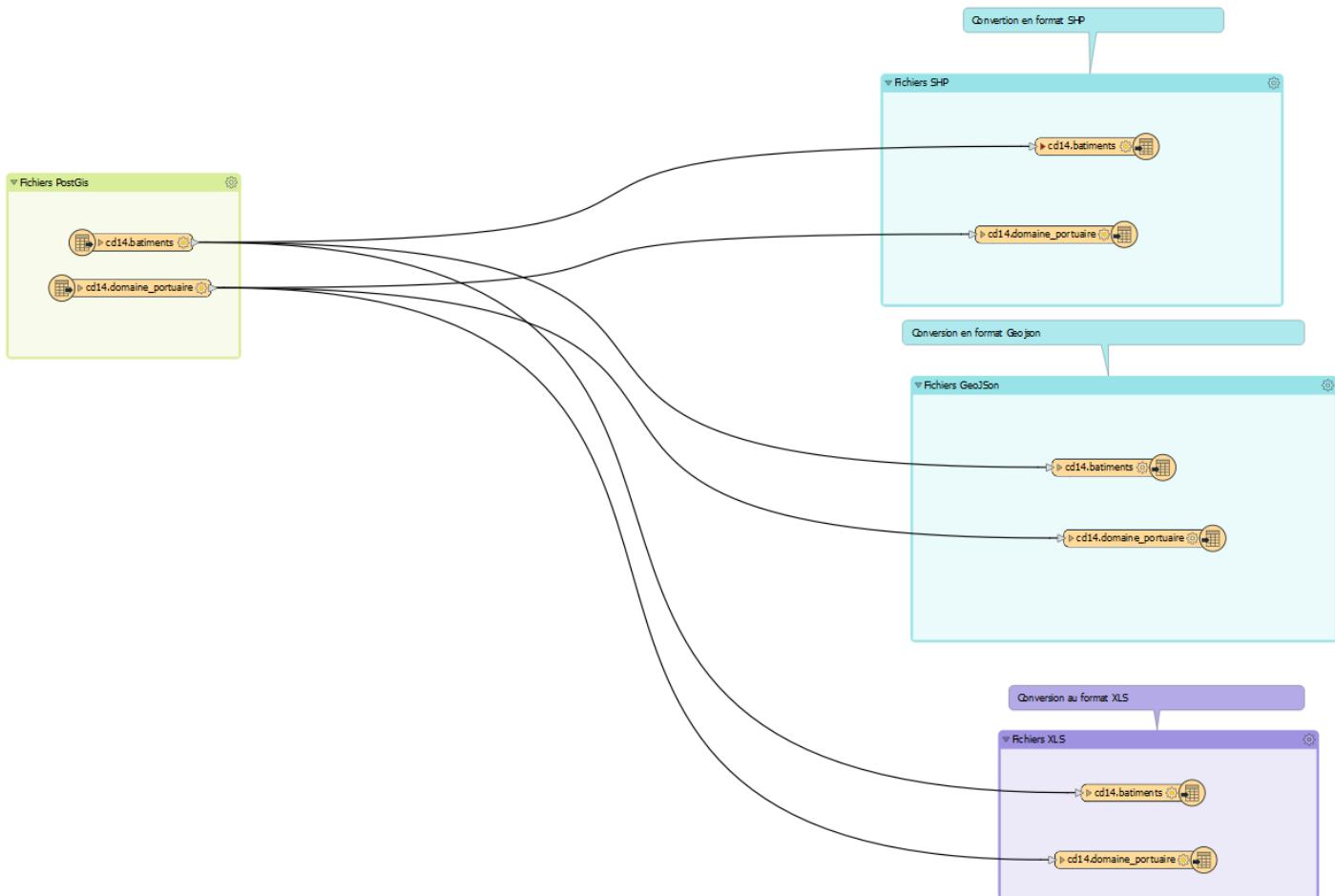
Figure 32 : Architecture des serveurs



Les données sont stockées dans une base de données PostgreSQL/GIS au sein LAN CD14. L'objectif est de les exporter sur un serveur hors du LAN CD14 afin que les données soient accessibles pour le grand public. Pour cela, on utilisera le logiciel FME, un outil de traitements de données conçu à l'origine pour manipuler des informations géographiques mais qui a évolué par la suite afin de traiter des données de toute nature : bases de données, bureautique, tableurs, CRM, images, XML, plans CAO, cloud, big data, nuages de points...

FME permet d'extraire, de transformer et de charger des données géographiques via des scripts semblables à des modelbuilders et de réaliser des géotraitements à l'aide de transformeurs. Le logiciel nous permettra de convertir les données PostGis en données GeoJson, Shapefile et XLMS. Une fois converties, ces données seront envoyées sur le serveur FTP à l'aide du transformeur « FTPCaller ».

Figure 33 : Exemple d'un workbench FME réalisé

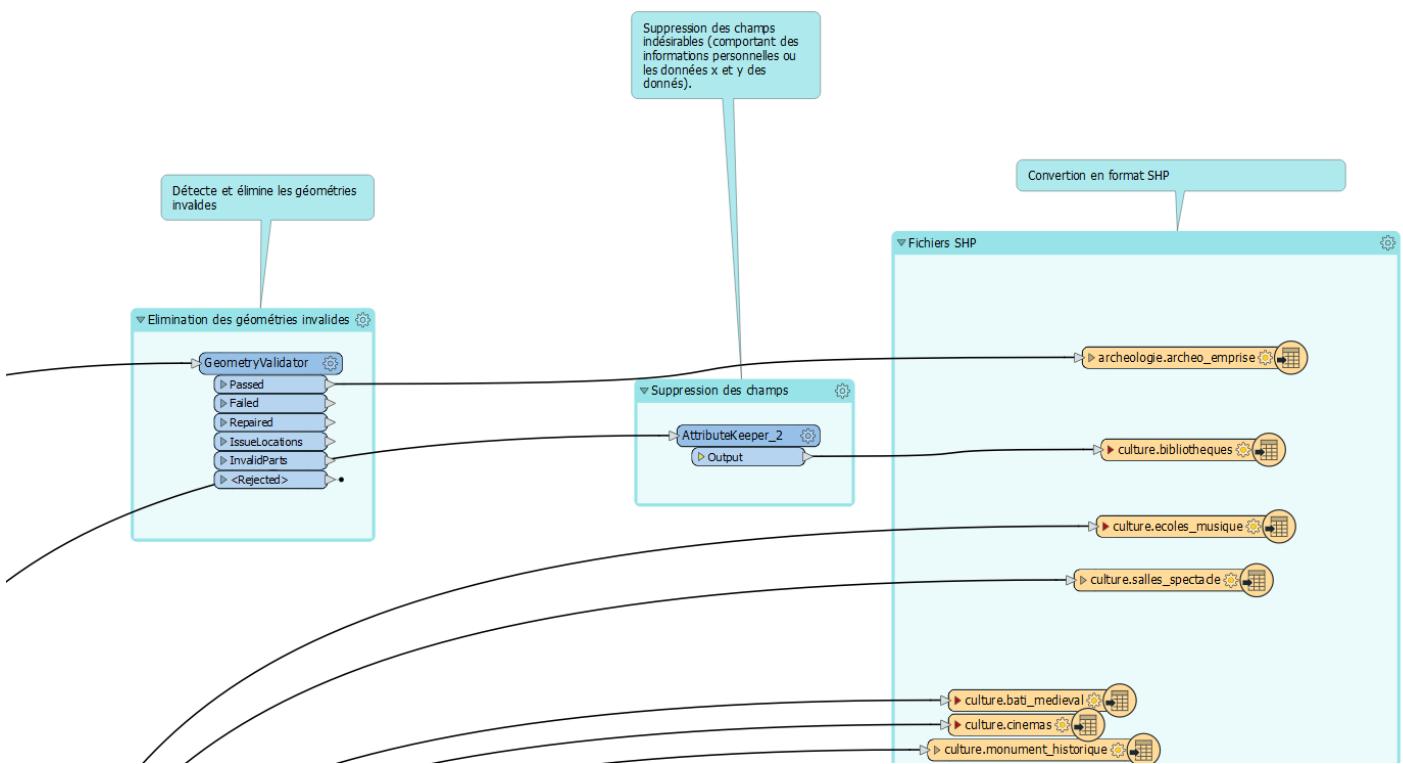


La première tâche à réaliser était de convertir les données PostGis en format shapefile. Cependant, trois problèmes se posaient :

Certains jeux de données comportaient des erreurs géométriques, comme la couche de l'emprise archéologique sur le territoire, ce qui rendait la conversion des couches en shapefile impossible. Il a donc fallu au préalable éliminer les objets géométriquement invalides via le transformeur « GeometryValidator ».

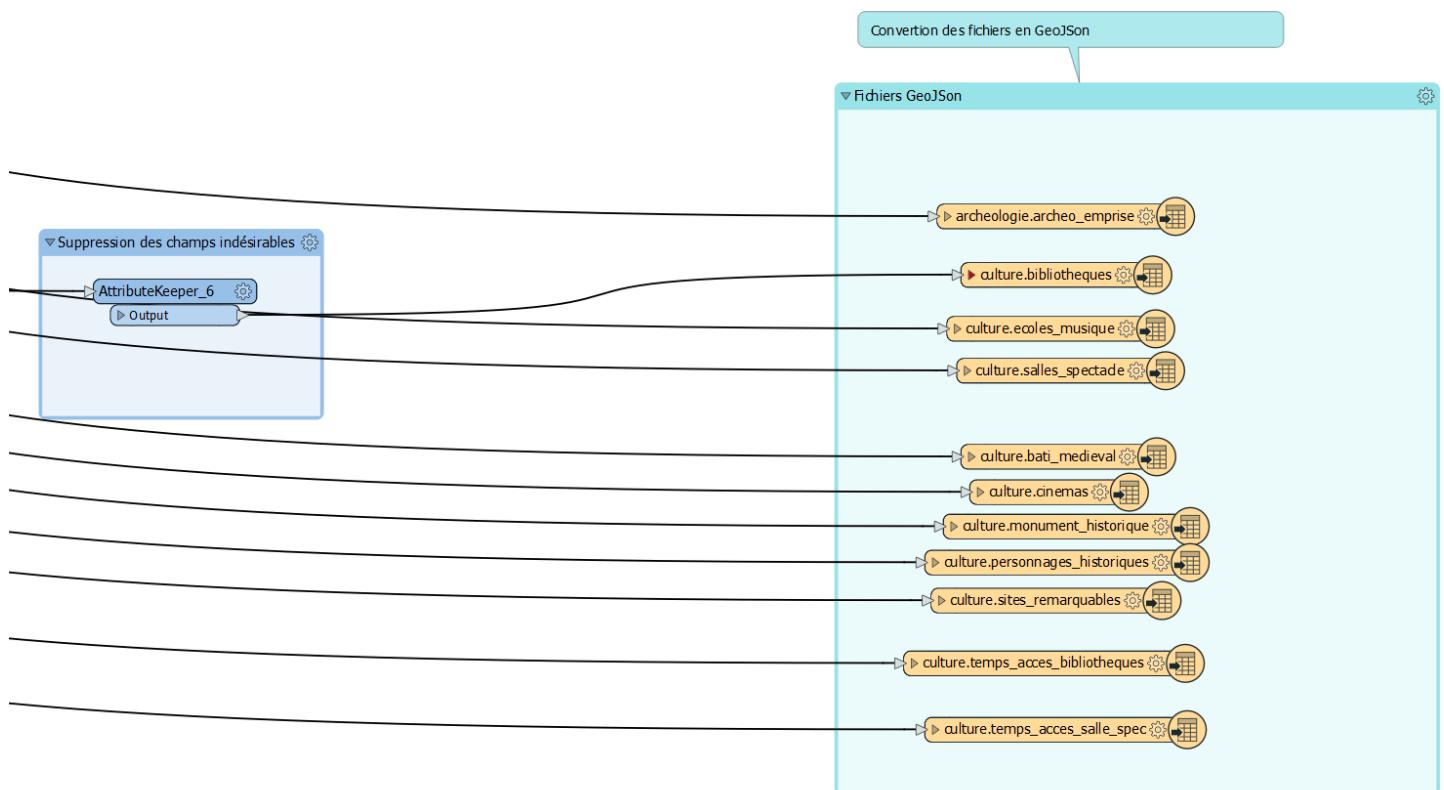
Enfin, certaines données comportaient des données attributaires inutiles (comme les coordonnées x et y des objets) ou auxquelles le grand public n'est pas censé avoir accès (le nom de certains agents du département), comme l'oblige le RGPD (Règlement Général sur la Protection des Données) du règlement européen. Le transformeur « AttributeKeeper » a servi à supprimer ces informations des couches finales.

Figure 34 : Conversion des données Postgis au format Shapefile



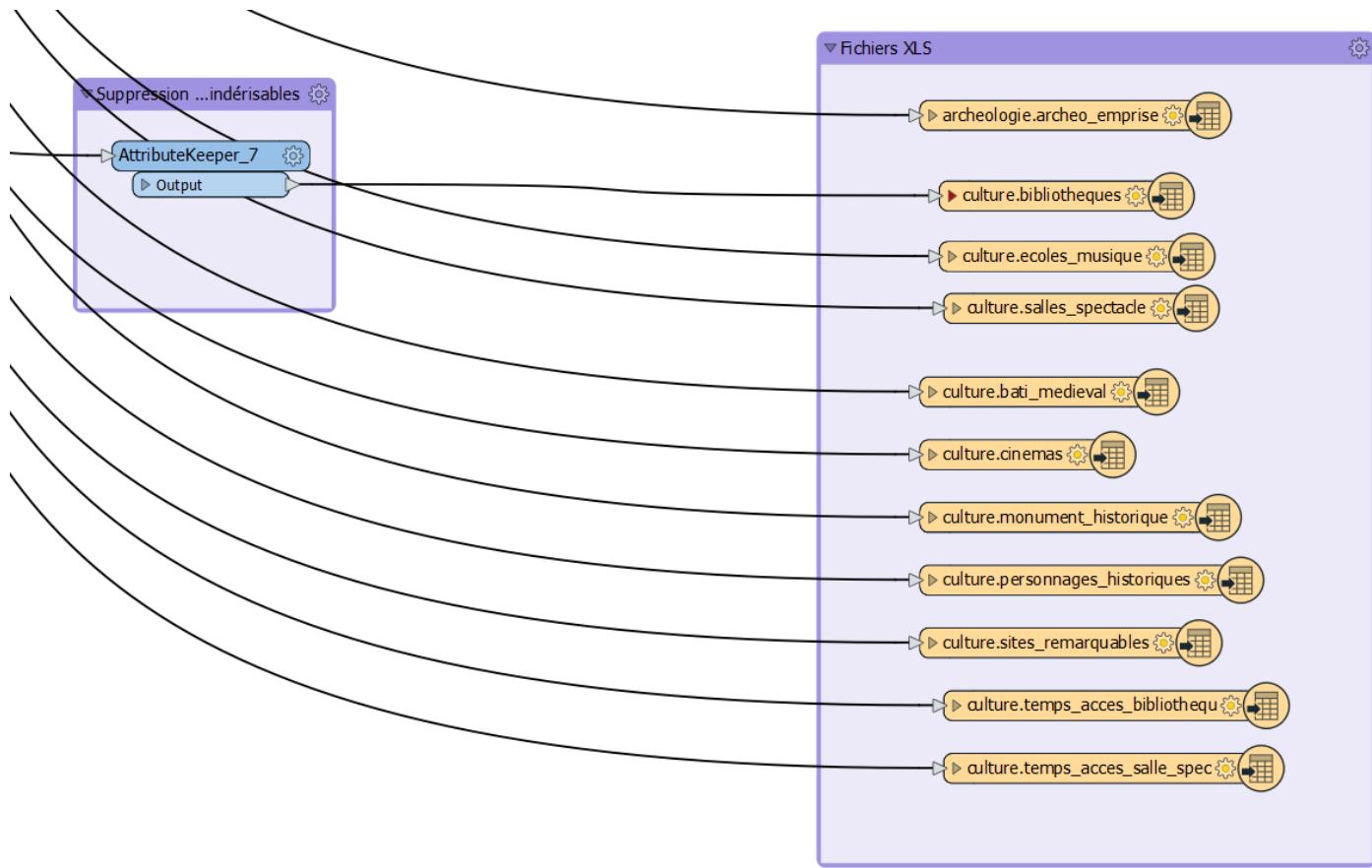
La seconde étape a été de convertir les données au format GeoJSON. Ici, la seule étape secondaire a été de supprimer les données attributaires indésirables.

Figure 35 : Conversion des données PostGis au format GeoJson



Troisième étape : conversion des données au format XLS. Ici, la conversion suit les mêmes étapes que pour la conversion au format GeoJson.

Figure 36 : Conversion des données au format XLS

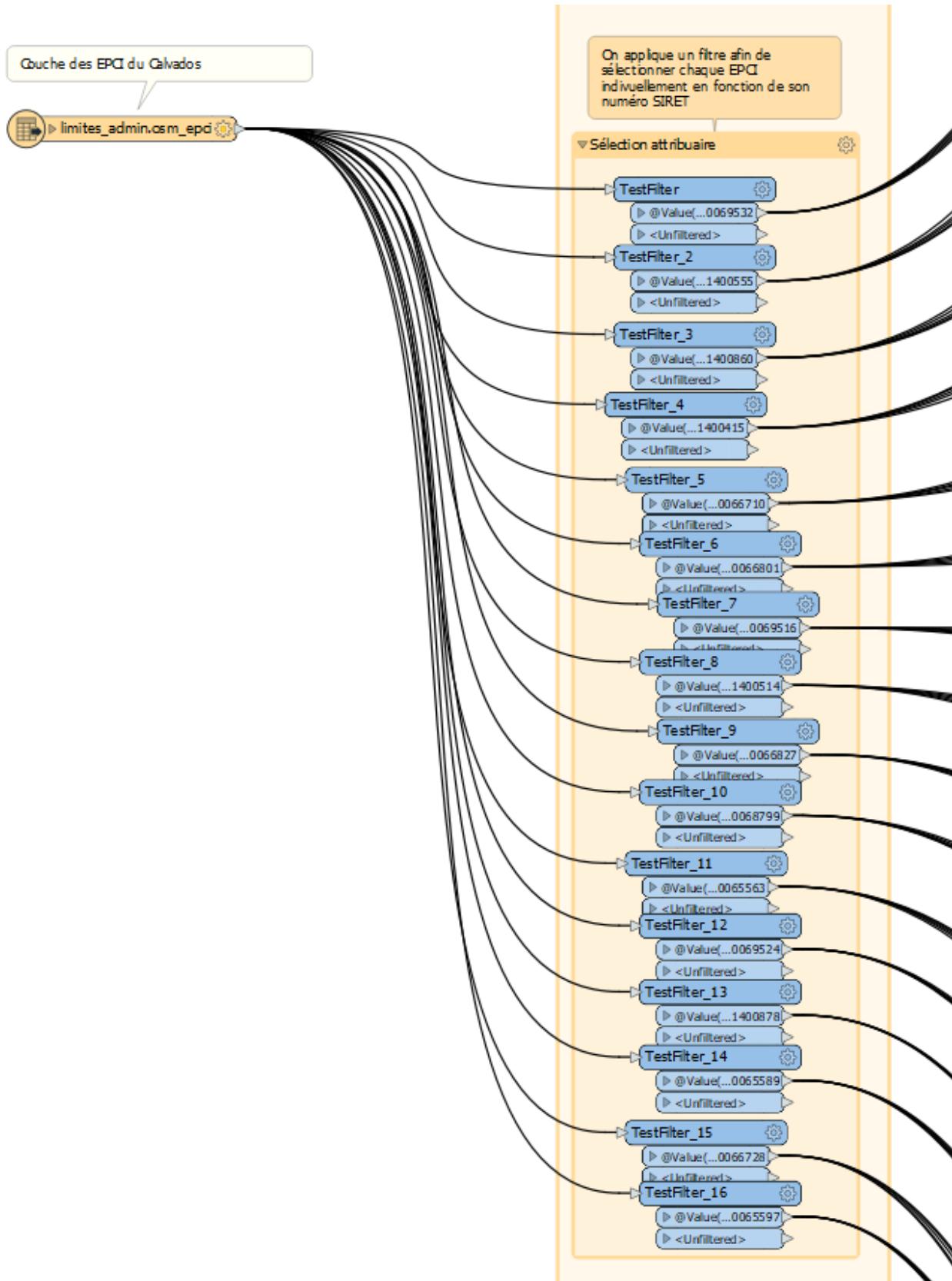


Une fois les conversions effectuées, on a tenté de transférer les données sur le serveur apl430 via le transformeur « FTP Caller ».

Il était aussi nécessaire d'importer les données cadastrales en ligne au format dxf, geojson, shapefile et svg. Cependant, l'importation de ces données nécessitant un traitement plus important, celui-ci a été réalisé dans un autre workbench.

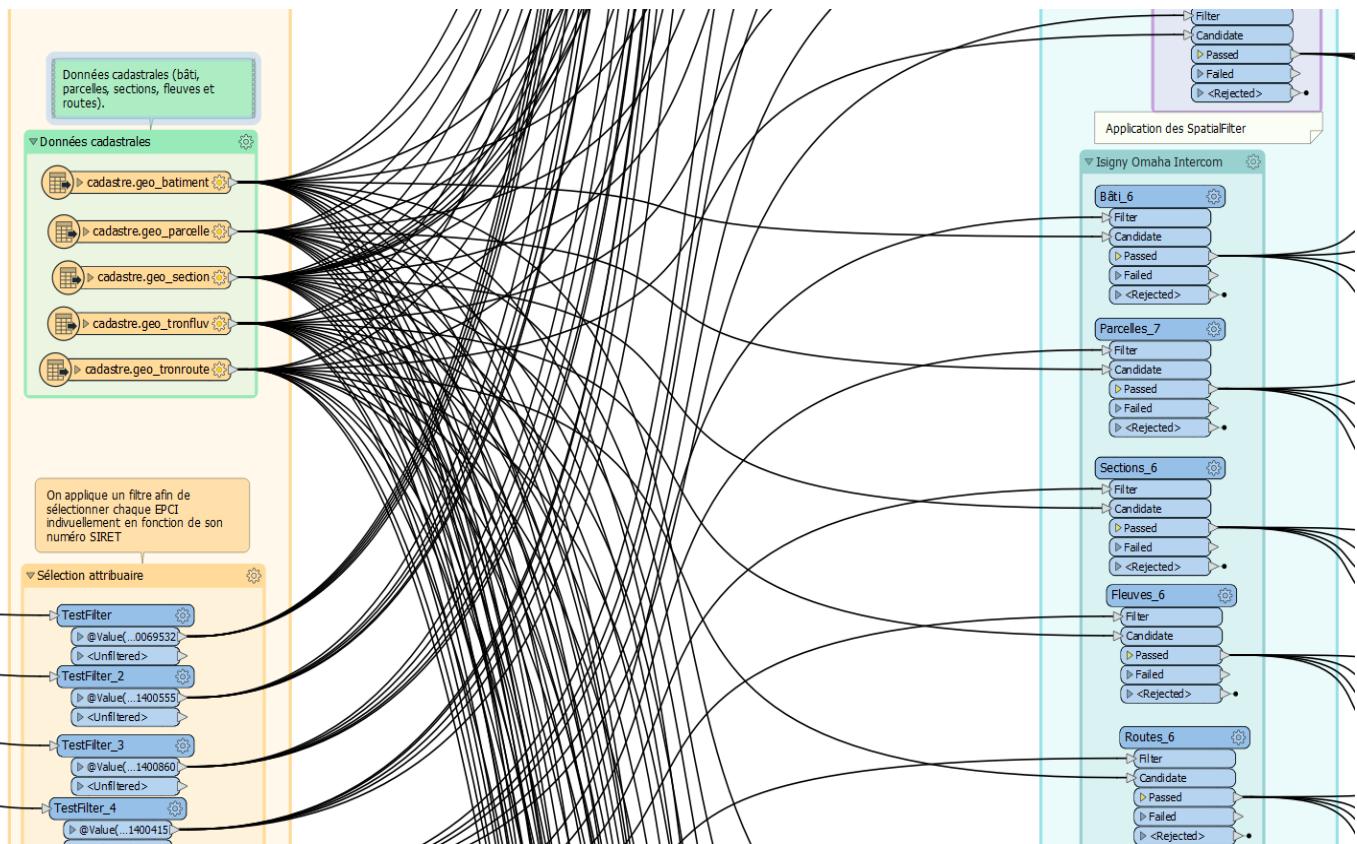
L'objectif ici était de pouvoir proposer à l'utilisateur d'avoir accès aux données cadastrales par epci. Il a donc aussi fallu dans un premier temps importer ces EPCI. Pour cela, on a importé la couche OSM des EPCI issu de PostGis à laquelle on a appliqué différents transformateurs « TestFilter », qui permettent de faire l'équivalent des requêtes attributaires de Qgis, afin de ne sélectionner pouvoir sélectionner chaque EPCI indépendamment les unes des autres.

Figure 37 : Sélection des EPCI via le transformeur « TestFilter »



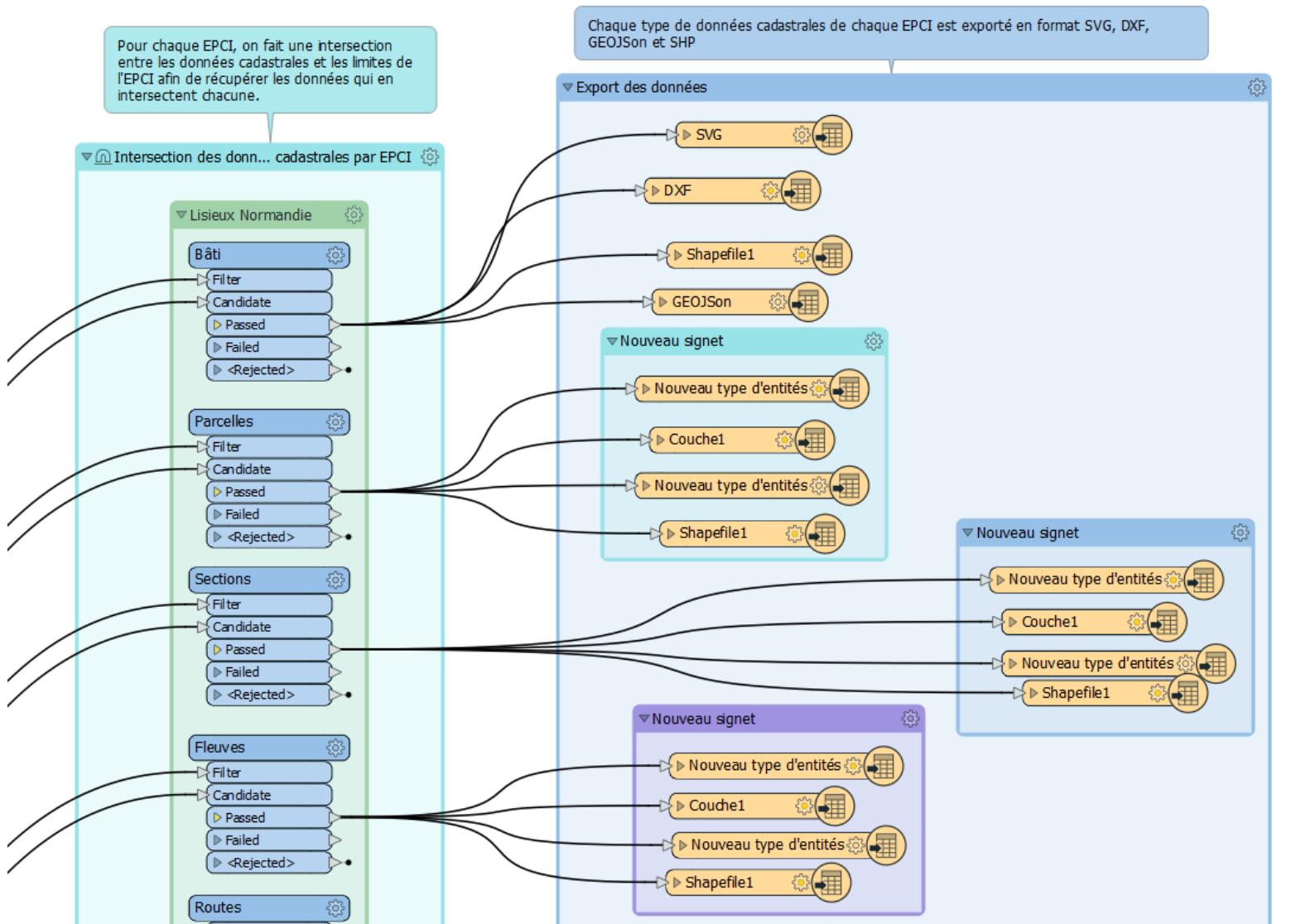
Ensuite, on importe les données cadastrales du bâti, des parcelles, des sections, des tronçons de fleuves et des tronçons de route. Il faut ensuite découper ces données cadastrales avec les couches des EPCI. Pour cela, on applique, pour chaque EPCI, cinq transformateurs « SpatialFilter », qui permettent de faire l'équivalent des sélections spatiales de Qgis.

Figure 38 : Sélection spatiale des données cadastrales via les EPCI



Une fois cette sélection spatiale effectuée, on va exporter chaque donnée cadastrale en quatre formats, déjà cités précédemment : DXF, Geojson, Shapefile et SVG.

Figure 39 : Export des données cadastrales



Une fois toutes les données exportées, on les met en ligne via le logiciel FileZilla, un client FTP qui nous permet de migrer les données directement vers Data.calvados.fr. A l'origine, l'exportation des données devait elle aussi être automatisée mais le serveur cible est en SFTP et FME ne permet l'exportation que vers des serveurs FTP. Une fois ce problème réglé, l'ensemble de l'opération sera automatisé.

Figure 40 : Capture d'écran du logiciel FileZilla

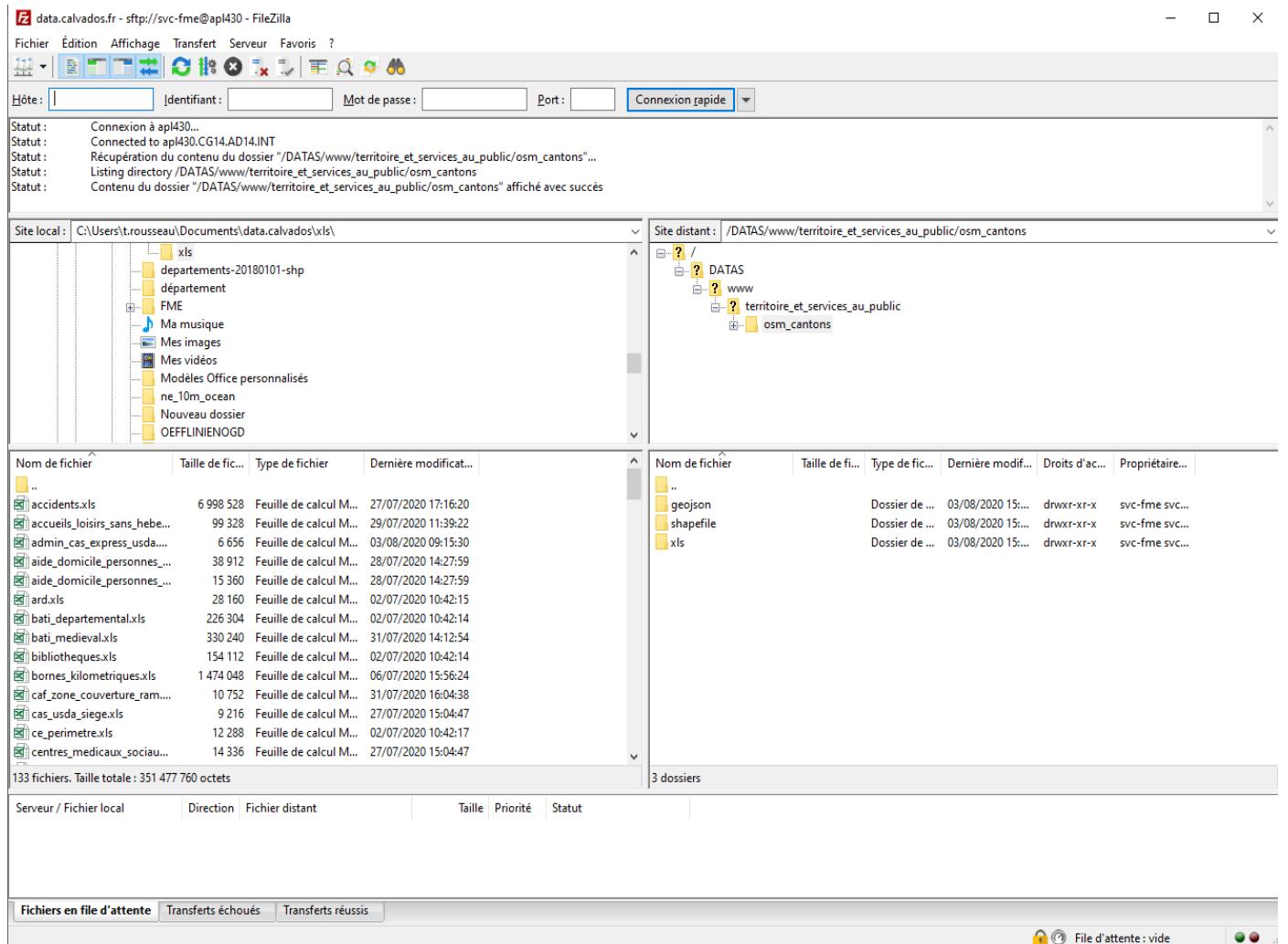


Figure 41 : Capture d'écran de data.calvados.fr

Index of /

<u>Name</u>	<u>Last modified</u>	<u>Size</u>	<u>Description</u>
 logo/	2020-07-22 18:47	-	
 bati departemental/	2020-07-31 10:58	-	
 cadastre/	2020-07-31 10:51	-	
 culture/	2020-07-31 14:16	-	
 enfance et jeunesse/	2020-07-31 16:48	-	
 environnement/	2020-08-03 11:01	-	
 orthos/	2020-07-20 10:36	-	
 routes/	2020-07-27 17:24	-	
 sante et autonomie/	2020-07-31 16:53	-	
 solidarite/	2020-07-31 16:53	-	
 sport et tourisme/	2020-07-28 15:11	-	
 territoire et servic..>	2020-08-03 15:19	-	

Enfin, on met aussi les données en téléchargement sur Isogéo, une plateforme en ligne permettant de cataloguer facilement des données. Isogéo permet de mettre les données de data.calvados.fr à disposition des différents services du département. En effet, pour chaque couche créée, une fiche de métadonnées lui est dédiée sur Isogéo, résumant les informations sur celle-ci (que contient-elle ? Comment a-t-elle été collectée?) et des liens de téléchargement sous les différents formats sont aussi proposés.

Isogeo a en effet développé OpenCatalog, un outil permettant d'oublier des données dans un catalogue en ligne de manière très simple. On peut ainsi choisir un nombre d'utilisateurs limité ayant accès aux données, une barre de recherche est mise à disposition ainsi que des filtres dynamiques permettant d'accéder rapidement aux données. Les informations essentielles sont mises en avant et l'utilisateur peut rapidement visualiser et/ou télécharger la donnée décrite.

Figure 42 : Capture d'écran d'OpenCatalog d'Isogéo

Conclusion

L'utilisation de FME et de FileZilla a permis de remplir l'ensemble des objectifs fixés : on a pu transformer les données PostGis en différents formats (DXF, GeoJSON, Shapefile, ...) via FME puis les envoyer sur data.calvados.fr via FileZilla.

Les données qui étaient hébergées sur le serveur Lizmap du Département sont donc désormais disponibles en open data sur le site data.calvados.fr.

CONCLUSION GÉNÉRALE

Avec ce stage, le Pôle SIG avait plusieurs objectifs dont certains qui ont dû être abandonnés en raison de l'épidémie de Covid 19 qui a entraîné une réduction de la durée du stage. En effet, le stage devait, à l'origine, se concentrer aussi sur des projets comme l'établissement d'un cahier des charges pour la mise en place d'une base de données sur les ENS (Espaces Naturels Sensibles) ou encore sur la mise en place d'une base de données sur les ERP (Établissements Recevant du Public). Finalement, il a été choisi de se concentrer sur trois projets : l'automatisation du traitement des itinéraires de randonnée, la création d'isodistances et enfin la publication de données en open data sur data.calvados.fr.

Pour remplir les objectifs fixés, plusieurs outils ont été mis à ma disposition : QGIS a été le principal outil SIG utilisé. Pour le traitement automatique des données, PostGis a été d'une grande aide. Le logiciel m'a permis des fonctions SQL puissantes et ainsi facilement apparier les relevés GPS des itinéraires de randonnée sur la BDTopo de l'IGN mais aussi corriger les erreurs géométriques. Le logiciel FME a également été d'une grande aide, il m'a permis de mettre à jour et transformer les données postgres/gis en données plus facilement exploitable (geojson, xls, etc.) et téléchargeables sur data.calvados.fr.

Au cours du stage, j'ai aussi pu être familiarisé avec nombreux outils SIG : QGIS, PostGis, pgRouting, l'extension de PostGis consacré à la gestion des données géographiques routières, ou encore l'ETL FME et l'outil de catalogage Isogéo. Au cours de ce stage, eu l'occasion d'approfondir grandement les notions de SQL et d'utilisation de FME vues durant ma formation en licence professionnelle.

J'ai notamment automatiser des géotraitements permettant d'optimiser grandement le temps de traitement des données géographiques.

Ce stage m'a donc d'abord permis d'appréhender plusieurs étapes du travail d'un chef de projet junior, notamment la création de donnée, la vérification de celles-ci puis leur partage en ligne. Cette expérience sera fort utile pour ma carrière professionnelle future.

WEBOGRAPHIE

DÉPARTEMENT DU CALVADOS :

Calvados Atlas : <https://atlas.calvados.fr/>

Site du Département du Calvados : <https://www.calvados.fr/accueil.html>

data.calvados.fr : <https://data.calvados.fr/>

FME :

Veremes.com : <https://www.veremes.com/produits/fme/fme-desktop>

ISOGEO :

Isogeo.com : <https://www.isogeo.com/>

OpenCatalog : <https://www.isogeo.com/nos-produits/OpenCatalog>

PGRouting :

Manuel de PGRouting : <https://docs.pgrouting.org/2.0/fr/doc/index.html>

Rapport de stage, Jeux Olympiques Paris 2024 : Couverture des équipements sportifs :

https://www.u-orme.fr/images/pdf/Rapport_JO_Chailloux_ELBakkouchi_Alary.pdf

Using pgr_distance for isodistances : <https://gis.stackexchange.com/questions/275005/using-pgr-drivingdistance-for-isodistances>

POSTGIS :

Introduction à PostGIS : <http://www.postgis.fr/chrome/site/docs/workshop-foss4g/doc/index.html>

Split lines with points, the postgis way : <https://mygisnotes.wordpress.com/2017/01/01/split-lines-with-points-the-postgis-way/>

TABLE DES FIGURES

Figure 1 : Organigramme de la hiérarchie des services au sein du Département	5
Figure 2 : Schéma du fonctionnement du Pôle SIG	6
Figure 3 : Capture d'écran du portail Mapéo Calvados	7
Figure 4 : Capture d'écran de SIR.....	8
Figure 5 : Capture d'écran de Calvados Atlas.....	9
Figure 6 : Carte des relevés GPS à Valdallière.....	12
Figure 7 : Exemple d'une erreur de tracé obtenue avec le relevé GPS	13
Figure 8 : Schéma représentant les différents géotraitements à réaliser	15
Figure 9 : Schéma représentant la seconde étape	16
Figure 10 : Schéma représentant la jointure spatiale entre les points et les zones tampons	17
Figure 11 : Schéma représentant le découpage de polylignes via les shortest lines	18
Figures 12 et 13 : Conversion des polylignes des itinéraires de randonnée en multipoints.....	20
Figure 14 : Capture d'écran de la table attributaire de la BDTopo	22
Figure 15 : Création des shortest lines	23
Figure 16 : Création des zones tampons	24
Figure 17 : End points des shortest lines.....	26
Figure 18 : Zones tampon de la BDTopo	28
Figure 19 : Exemple d'une erreur lors de la création des shortest lines	29
Figure 20 : Résultat final	31
Figure 21 : Résultat final bis	32
Figure 22 : Création d'un champ « new_id ».....	35
Figure 23 : Capture d'écran de la table attributaire du graphe topologique.....	37
Figure 24 : Création du champ « length_m ».....	38
Figure 26 : Création de points au début et à la fin des polylignes de la BDTopo	41
Figure 27 : On joint les noeuds de polylignes au sein d'une même couche	42
Figure 28 : Table attributaire de la couche « network ».....	43
Figure 29 : Création des champs « len_km » et « time_driving »	45
Figure 30 : Table attributaire de la couche des P.E.I	47
Figure 31 : Sélection des polylignes se trouvant à moins de 500m d'un noeud	48
Figure 32 : Architecture des serveurs	50
Figure 33 : Exemple d'un workbench FME réalisé	51
Figure 34 : Conversion des données Postgis au format Shapefile	52
Figure 35 : Conversion des données PostGis au format GeoJSon	53
Figure 36 : Conversion des données au format XLS	54
Figure 37 : Sélection des EPCI via le transformeur « TestFilter ».....	55
Figure 38 : Sélection spatiale des données cadastrales via les EPCI	56
Figure 39 : Export des données cadastrales	57
Figure 40 : Capture d'écran du logiciel FileZilla.....	58
Figure 41 : Capture d'écran de data.calvados.fr	59
Figure 42 : Capture d'écran d'OpenCatalog d'Isogéo	60

ACRONYMES ET DÉFINITIONS

-Appariement : Action d'assortir par pair. Ici, l'idée est d'associer chaque itinéraire de randonnée à la polyligne de la BDTopo de l'IGN qui lui correspond.

-DXF : Format de fichier créée par la société Autodesk servant à échanger des fichiers DAO ou CAO entre systèmes CAO n'utilisant pas le même format de fichier natif (comme les fichiers PostGIS).

-ENS (Espaces naturels sensibles) : Espaces dont le caractère naturel est jugé menacé et qui vont donc être classé dans un périmètre « sensible » au sein duquel l'urbanisation est interdite. Les ENS sont un outil départemental de protection de l'environnement via une acquisition foncière ou la signature d'une convention avec le propriétaire du terrain.

Le Pôle SIG a pour projet de créer une base de données regroupant les informations géographiques sur les ENS du Calvados.

-ERP (Etablissements recevant du public) : Bâtiments dans lesquels des personnes extérieures (le personnel n'étant pas compté) sont admises et qui sont donc soumis à des exigences réglementaires applicables.

Le Pôle SIG a pour projet de créer une base de données regroupant les informations géographiques sur les ERP du Calvados.

-GeoJSON : Format d'encodage de données géographiques utilisant la norme JSON (Javascript Object Nation).

-LAN (Local Area Network) : Réseau informatique au sein duquel les terminaux (ordinateurs ou autres) sont en liaison au niveau local, c'est à dire sans utiliser d'accès à Internet.

-Open data : Données numériques laissées en libre accès aux usagers.

-P.E.I (Points d'eau incendie) : Ensemble des points d'eau (comme les bouches à incendies) pouvant servir à lutter contre les incendies dans le cadre d'intervention de sapeurs-pompiers.

-RGPD (Règlement général sur la protection des données) : Règlement de l'Union Européenne lié à la protection des personnelles des citoyens de l'Union.

-Serveur FTP (File Transfer Protocol) : Serveur permettant de transférer des fichiers par Internet ou via un réseau local (un LAN).

-Serveur SFTP : Serveur semblable au serveur FTP au sein duquel les identifiants d'authentification sont chiffrés.