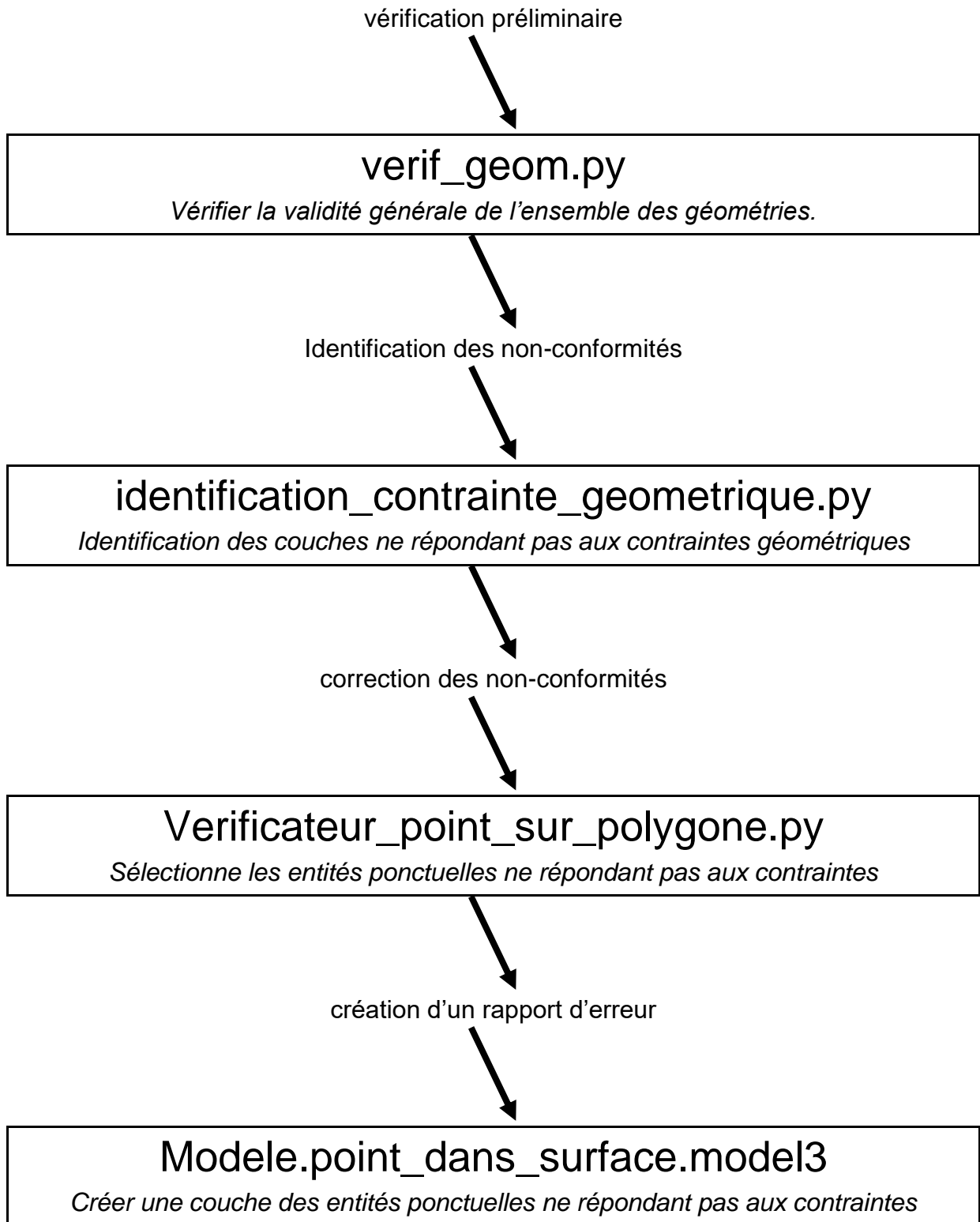


DOCUMENTATION

WORKFLOW	1
Vérificateur de géométrie	2
1) Comment se servir script	2
2) Quel logiciel utiliser	2
3) Fonctionnalité	2
4) Comment l'exécuter	2
5) Type de résultat attendu :	3
Identification des contraintes géométriques	3
Partie 1 : Mise en place des couches et vérifications de celles-ci	4
Partie 2 : Validation des relations spatiales	4
Partie 3 : Contrôle des superpositions et de la connectivité spatiale des entités	4
Vérificateur de point sur des polygones	7
C'est quoi cet outil ?	5
Qu'est-ce qui est compatible avec ?	5
Comment l'utiliser ?	5
Modèle de point dans une surface	7

WORKFLOW

Ordre d'exécution des scripts :

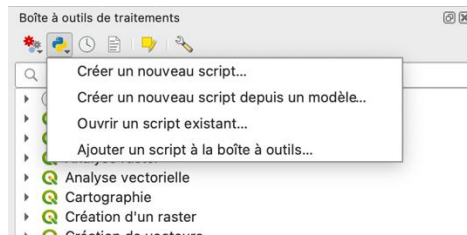


Vérificateur de géométrie

— *verif_geom.py* —

1) Comment se servir script

- Dans l'onglet de boîte à outil de traitement, ouvrir le script enregistré



2) Quel logiciel utiliser

- Utilisation du logiciel QGIS
- Version 3.24 de QGIS ou toute version ultérieure

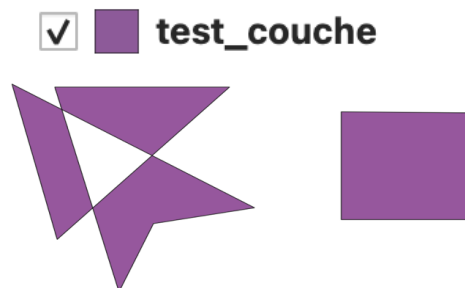
3) Fonctionnalité

- Le script va renvoyer pour une couche donnée, les polygones ayant une géométrie invalide et les polygones ayant une géométrie valide, sous forme de couche temporaire.
- Il va aussi renvoyer, dans la console, le nombre de polygones erronés.

4) Comment l'exécuter

- Choisir une couche de polygone en entrée à tester
- Possibilité de spécifier le nom des couches en sortie ou de les enregistrer

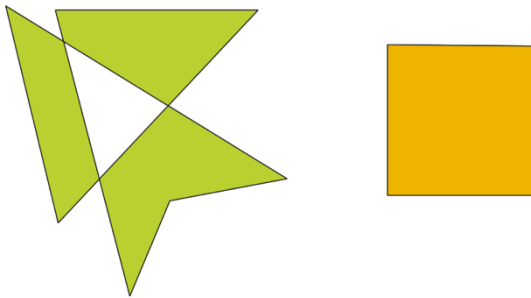
Une couche de départ contenant des polygones :



5) Type de résultat attendu :

On utilise le modèle ou le script :

☐ ☒ *polygone_non_valide*
☐ ☒ *polygone_valide*



Nombre d'erreurs affichées dans la console :

```
7 >>> .exec(compile(Path('/var/folders/f3/t1r0xw214x19wlth06lrh9x80000gn/T/tmpf8jut736').read_text(),  
8      . 'Sans titre 0', 'exec'))  
8 Nombre de polygones avec erreur : 1  
9
```


Identification des contraintes géométriques

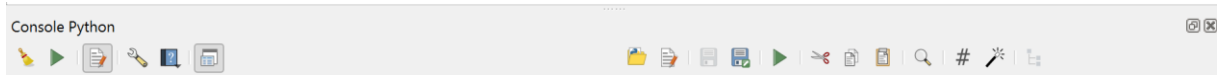
— *identification_contrainte_geometrique.py* —

Attention, ce script peut ne pas marcher sur les ordinateurs de l'Université en raison de la bibliothèque geopandas.

Avant d'utiliser ce script, il faut l'ouvrir dans Visual Studio Code est de modifier la ligne 6 de ce script en mettant votre chemin d'accès.

```
6 GPKG_PATH = "chemin/vers/le/fichier/IUGA_plan_complet.gpkg"
```

Ce script s'utilise directement sur QGIS avec la console python (extensions  console python). Puis d'ouvrir le fichier dans cette console.



Ce script, en quatre parties, sert à contrôler les erreurs possibles, il suit les étapes suivantes :

Partie 1 : Mise en place des couches et vérifications de celles-ci

- Chargement des couches à partir du fichier Geopackages « IUGA_plan_complet.zip »
- Vérification des erreurs géométriques, c'est-à-dire les géométries non valides, les auto-intersections ainsi que les doublons.

Partie 2 : Validation des relations spatiales

- La couche « infrastructure » doit croiser la voirie ou les espaces verts
- Les points d'accès doivent être attaché aux bâtiments
- Les arbres doivent être situés dans un espace vert et avoir une distance d'au moins 1m
- Le mobilier urbain ne doit pas être placé sur des routes ou des bâtiments
- Les aires de jeux doivent être situés à l'intérieur des espaces verts

Partie 3 : Contrôle des superpositions et de la connectivité spatiale des entités

- Détecte les superpositions et les chevauchements de polygone d'une même couche
- Vérifie les passages piétons c'est-à-dire s'il est contenu dans au moins une entité de voirie et si chaque passage piéton a au moins deux côtés en contact avec une voirie.

En dernier lieu, le script regroupe et affiche les erreurs trouvées.

```
Auto-intersections·trouvées·dans·voirie
Auto-intersections·trouvées·dans·passage_pieton
Auto-intersections·trouvées·dans·espace_vert
Auto-intersections·trouvées·dans·batiment
Auto-intersections·trouvées·dans·aire_de_jeux
Certains·infrastructures·ne·sont·pas·connectées·à·une·voirie·ou·un·espace·vert
Certains·points·d'accès·ne·sont·pas·liés·à·un·bâtiment.
Certains·arbres·ne·sont·pas·situés·dans·un·espace·vert.
Certains·arbres·sont·trop·proches·(<1m).
Certains·mobilier·urbains·sont·placés·dans·une·voirie·ou·un·bâtiment
Certains·aires·de·jeux·ne·sont·pas·situées·dans·un·espace·vert
Superpositions·trouvées·dans·voirie:·[(18,·19),·(19,·21)]
Auto-intersections voirie:·52·erreurs·trouvées.
```

Vérificateur de point sur des polygones

— verificateur_point_sur_polygone.py —

C'est quoi cet outil ?

Ce script vérifie la qualité spatiale d'une couche de points dans QGIS en identifiant deux types d'erreurs : les points situés en dehors de tout polygone dans une couche de polygones donnée,

et les points trop proches les uns des autres (moins d'un mètre de distance). Il sélectionne tous ces points problématiques afin que l'utilisateur puisse les examiner ou les corriger. Cela permet de s'assurer que tous les points sont correctement localisés dans les zones définies et bien espacés, ce qui est utile pour corriger des erreurs de numérisation ou valider des données dans des flux de travail en géomatique.

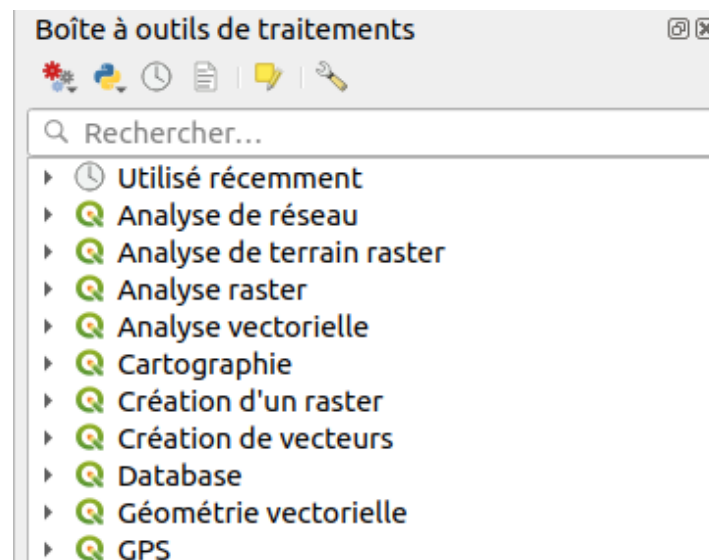
Qu'est-ce qui est compatible avec ?

le script est compatible avec QGIS uniquement et toutes ses versions.

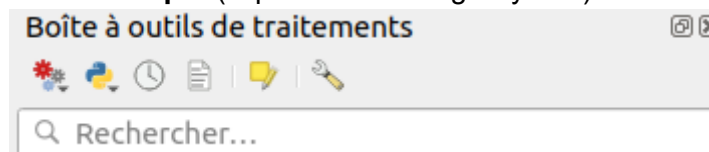
Comment l'utiliser ?

Étape 1 :

- Lancez **QGIS**
- Chargez vos couches (**polygone** et **points**)
- Ouvrez la **Boîte à outils de traitement**



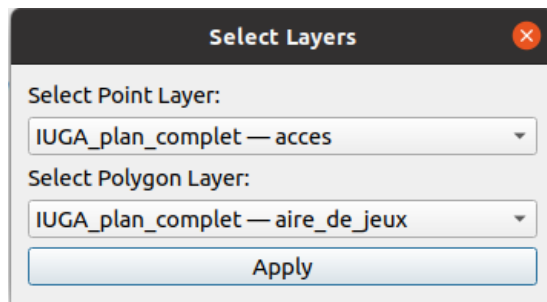
- Sélectionnez l'outil **Scripts** (l'option avec le logo Python)



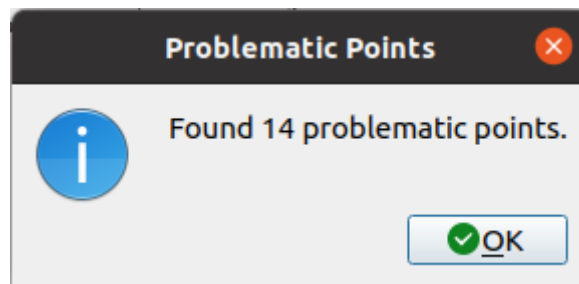
- Choisissez comment ouvrir le script : soit **en tant que nouveau script** (vous copiez-collez le contenu), soit en sélectionnant un **script existant** depuis un fichier
- Lancez le script en cliquant sur le **bouton vert Exécuter** ►

Étape 2 :

- Lors de l'exécution, une boîte de dialogue s'ouvre où vous devez sélectionner vos deux couches (**polygone** et **points**)



- Sélectionnez vos deux couches, puis appuyez sur « **Apply** »
- Une **nouvelle boîte de dialogue** s'affiche et vous informe du **nombre de points qui ne se trouvent pas sur la couche polygone**

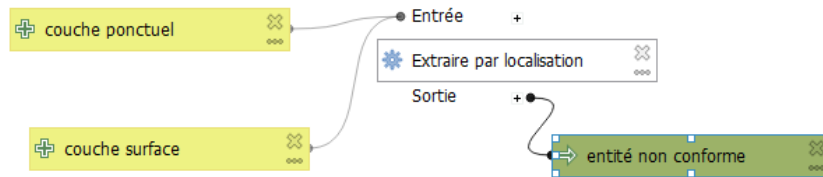


- En cliquant sur **OK**, une **nouvelle boîte de traitement** s'ouvre : vous avez alors le choix entre **lancer le traitement** (et sélectionner les points concernés) ou **annuler la sélection**

Modèle de point dans une surface

— Modele.point_dans_surface.model3 —

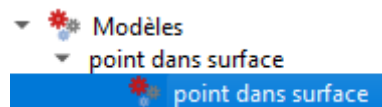
Modèle de vérification de géométrie pour Qgis 3.34



Modele.point_dans_surface.model3 est un modèle qui permet d'ajouter un traitement dans la boîte à outil Qgis.

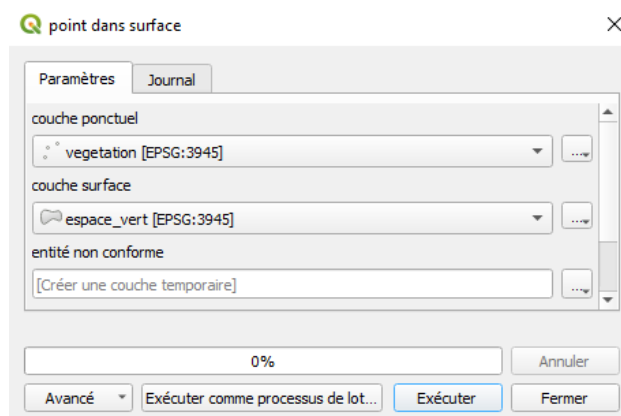
Après avoir lancé Qgis, il faut aller dans le menu « Modèles » de la boîte à outils de traitement (), et sélectionner « ajouter un modèle à la boîte à outils ». Il faut ensuite sélectionner le fichier Modele.point_dans_surface.model3 dans le répertoire où il a été rangé. L'outil « point dans surface » est maintenant implémenté.

Pour utiliser l'outil « point dans surface », on peut y accéder depuis la boîte à outils de traitement dans l'onglet « Modèle » :



L'outil "point dans surface" permet de tester pour une couche de points donnée, si leur localisation intersecte et une couche de polygone en particulier. Le traitement renvoie une couche de résultat « entité non conforme » correspondant aux entités ponctuel n'intersectant pas la couche de polygone voulue.

Pour utiliser l'outil point « dans surface » il faut que les couches à étudier soient préalablement chargées dans Qgis. Il faut ensuite indiquer la couche de point dont on veut tester la géométrie dans le champ « couche ponctuel », puis indiquer la couche de polygone qui va nous servir à tester la précédente dans le champ « couche surface ». Le champ optionnel « entité non conforme » permet de paramétrer la couche résultat si besoin.



Dans le cadre des spécifications techniques et qualitatives pour un inventaire urbain de l'avenue Marie Reynoard et du site de l'Institut d'Urbanisme et de Géographie Alpine, ce modèle permet de tester le respect des contraintes concernant la végétation et les

infrastructures réseau qui sont des couches ponctuelles qui doivent impérativement intercepter une autre couche de données.