CityJSON Loader

Données en entrées :

|  |
| --- |
|  |
| INPUT='INPUT' |
|  | DIVIDE\_BY\_OBJECT\_TYPE = 'DIVIDE\_BY\_OBJECT\_TYPE' |
|  | LOD\_AS = 'LOD\_AS' |
|  | LOAD\_SEMANTIC\_SURFACES = 'LOAD\_SEMANTIC\_SURFACES' |
|  | STYLE\_BY\_SEMANTIC\_SURFACES = 'STYLE\_BY\_SEMANTIC\_SURFACES' |
|  | SRID = 'SRID' |
|  | BBOX = 'BBOX' |
|  | OBJECT\_TYPE = 'OBJECT\_TYPE' |
|  |  |
|  | LODLOADINGTYPES = ['NONE', 'ATTRIBUTES', 'LAYERS'] |
|  | OBJECTTYPES = ['Building', 'Bridge', 'Road', 'TransportSquare', 'LandUse', 'Railway', 'TINRelief', 'WaterBody', 'PlantCover', 'SolitaryVegetationObject', 'CityFurniture', 'GenericCityObject', 'Tunnel'] |
|  |  |

Principe :

 Le logiciel analyse un fichier CityJSON et analyse sa structure arborescente pour identifier tous les objets de la ville. Ensuite, la géométrie et les attributs de chaque objet de la ville sont transformés en fonctionnalités QGIS. Enfin, ces fonctionnalités sont stockées dans des calques, qui sont ensuite ajoutés au projet en cours avec des styles de rendu 3D.

Le modèle de données fournit des définitions pour décrire les objets urbains (par exemple, les bâtiments et les infrastructures de transport) et les caractéristiques du paysage (par exemple, le terrain, la végétation et les plans d'eau). Dans CityJSON, chacune de ces fonctionnalités est un objet JSON avec le type respectif (Figure [1](https://onlinelibrary.wiley.com/doi/full/10.1111/tgis.12657#tgis12657-fig-0001) ), un certain nombre d'attributs et un ensemble de géométries. Les multiples géométries d'un objet urbain correspondent à différentes représentations 3D de la forme de l'objet selon différents niveaux de détail (LoD).

Un objet géométrique peut être un objet de l'un des types suivants : MultiPoint , MultiLineString , MultiSurface , CompositeSurface , Solid , MultiSolid ou CompositeSolid . Intuitivement, tous les types géométriques (à l'exception de MultiPoint et MultiLineString ) peuvent être considérés comme un ensemble de polygones décrivant la forme 3D d'un objet. Ce qui diffère entre les types de géométrie est la signification sémantique de la forme ; une géométrie Solide est considérée comme le volume délimité par les polygones susmentionnés, tandis qu'une MultiSurface ou une CompositeSurface décrit un objet de surface sans intérieur volumétrique composé de ces polygones. Chacun de ces polygones, qui dans la terminologie CityJSON est considéré comme une surface, peut également avoir une signification sémantique supplémentaire. Cela signifie que les surfaces individuelles peuvent avoir un type sémantique (par exemple, WallSurface , RoofSurface et FloorSurface ) avec des attributs qui représentent d'autres informations exclusives à cette surface.

Architecture du logiciel :

Chaque objet ville est composée de plusieurs surfaces = polygones 3D dans Qgis : le plugin doit créer des entités QGIS avec des géométries multi-polygones qui représentent les objets du modèle de ville. Cela fonctionne si la ville a un seul niveau de géométrie, c’est-à-dire,un seul LoD. Seulement une ville peut contenir plusieurs Lod.

Afin de résoudre ces problèmes, CityJSON Loader fournit diverses options pour différents mappages entre les objets de la ville et les fonctionnalités QGIS. Cela signifie que, selon le choix de l'utilisateur, une entité QGIS peut correspondre à : un objet ville, où toutes les géométries sont représentées par un multi-polygone; la LoD d'un objet ville, où un objet ville est divisé en plusieurs entités (une entité par géométrie LoD ;  ou une surface sémantique d'un objet de ville, où chaque surface d'un objet de ville forme sa propre caractéristique de polygone QGIS.

On divise les entités en couches :

Le logiciel est également conçu de manière à ce que les entités résultantes puissent être divisées en différentes couches en fonction de : leur type d'objet (par exemple, Bâtiment , Route et Terrain ) ou leur LoD (dès que les géométries LoD sont chargées en tant qu'individu caractéristiques). Ces options peuvent être combinées, de sorte que, par exemple, une couche soit créée pour chaque LoD de chaque type d'objet. Cela permet aux utilisateurs de visualiser et de traiter différents LoD séparément dans des ensembles de données multi-LoD. De plus, en divisant les fonctionnalités par type d'objet, il est plus facile de styliser et de gérer des objets de différents types.

Chargement du modèle : 5 étapes :

1. Tout d'abord, le logiciel parcourt les objets de la ville du modèle pour identifier tous les attributs contenus dans l'ensemble de données. Au cours de cette étape, une liste des attributs est construite pour être utilisée dans la création des couches QGIS.
2. Dans la deuxième étape, les sources de données et les couches QGIS sont créées. Cela se fait en énumérant les attributs identifiés dans la première étape. De plus, à cette étape, certains champs supplémentaires peuvent être créés : un pour la LoD de l'entité, dans les cas où l'utilisateur choisit de charger les géométries LoD en tant qu'entités individuelles ; et un pour les surfaces sémantiques, dans les cas où l'utilisateur souhaite charger la géométrie des surfaces d'un objet.
3. Au cours de la troisième étape, les fonctionnalités réelles de QGIS sont construites. Cela se fait en suivant les mappages décrits dans la section [3.1](https://onlinelibrary.wiley.com/doi/full/10.1111/tgis.12657#tgis12657-sec-0010) , ce qui signifie que les entités sont construites par objet de ville, par LoD ou par surface sémantique.
4. A la quatrième étape, les entités créées sont affectées aux couches créées lors de la deuxième étape. Cela se fait en fonction des attributs des entités et de la division souhaitée. Par exemple, si l'utilisateur choisit d'avoir différentes fonctionnalités LoD dans des couches individuelles, le champ LoD respectif est utilisé par le logiciel pour décider à quelle couche la fonctionnalité est attribuée.
5. Enfin, les couches sont ajoutées au projet QGIS actif. Une mise en forme de base est appliquée aux calques au cours de cette étape. Le logiciel active le rendu 3D pour chaque couche créée en appliquant le même style de couleur que la vue 2D. [4](https://onlinelibrary.wiley.com/doi/full/10.1111/tgis.12657#tgis12657-note-1004) Selon les préférences de l'utilisateur, un style basé sur des règles peut être appliqué en 3D afin que les entités soient colorées en fonction de leur type de surface sémantique (par exemple, rouge pour RoofSurface et blanc pour WallSurface ).