Intégration et visualisation de données 3D Lidar et tuiles 3D Vectorielles

Laboratoire 3

[**Étape 1 Importation et nettoyage des nuages de points**](#_ec8u3bf6aztu) **2**

[**Étape 2 Importation des limites terrestres et découpage du nuage de points**](#_fvbp9sj6bfxr) **5**

[**Étape 3 Simplification du nuage de points**](#_7ijt6xek4lo) **7**

[**Étape 4 Ajout de rasters géoréférencés**](#_23l8rlxg5p8f) **8**

[**Étape 7 Ajout des empruntes et details de bâtiments**](#_mtbwt3oe7v) **12**

[**Étape 8 Jointure des propriétés du nuage de points dans les polygones**](#_vj3u30n7iutj) **14**

[**Étape 9 Visualisation du résultat dans MaplibreGL**](#_hctb93vskptd) **18**

# Étape préliminaire

Télécharger les données

[Lab3 data.zip](https://drive.google.com/file/d/1GNEDPSGwSdACGDY3BhyQMRrtKXjtsQb2/view?usp=drive_link)

# Étape 1 Importation et nettoyage des nuages de points

* Ajouter les 6 nuages de points suivante avec un ***LAS READER***

<http://depot.ville.montreal.qc.ca/geomatique/lidar_aerien/2015/301-5041_2015.laz>

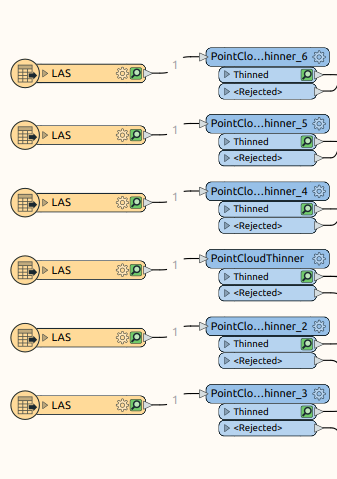
<http://depot.ville.montreal.qc.ca/geomatique/lidar_aerien/2015/301-5040_2015.laz>

<http://depot.ville.montreal.qc.ca/geomatique/lidar_aerien/2015/301-5039_2015.laz>

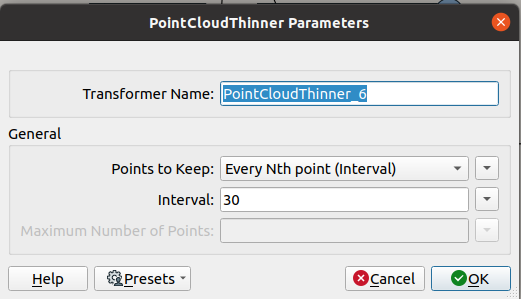
<http://depot.ville.montreal.qc.ca/geomatique/lidar_aerien/2015/300-5041_2015.laz>

<http://depot.ville.montreal.qc.ca/geomatique/lidar_aerien/2015/300-5040_2015.laz>

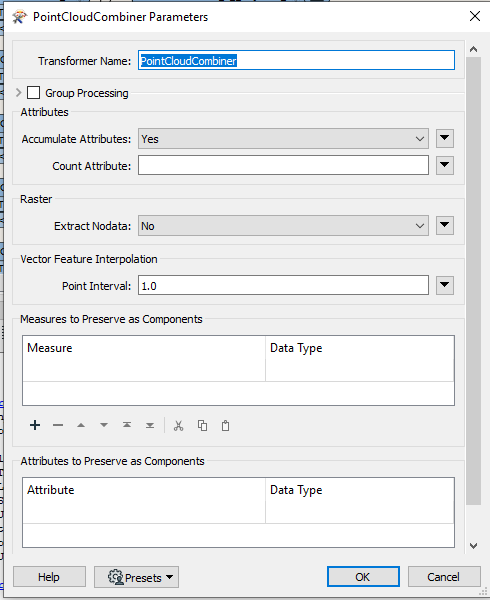
<http://depot.ville.montreal.qc.ca/geomatique/lidar_aerien/2015/300-5039_2015.laz>



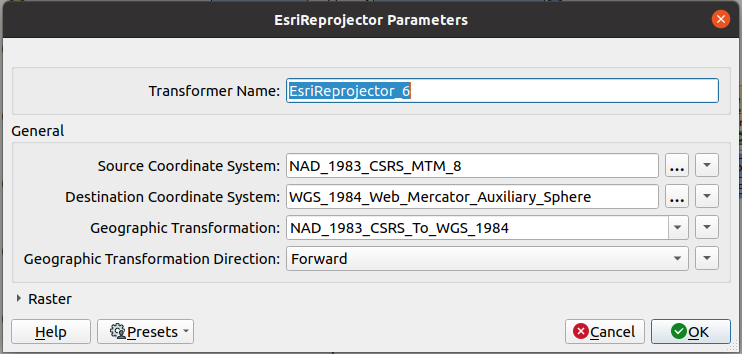
* Ajoutez sur chacun des reader un ***Thinner*** avec un filtre de 30



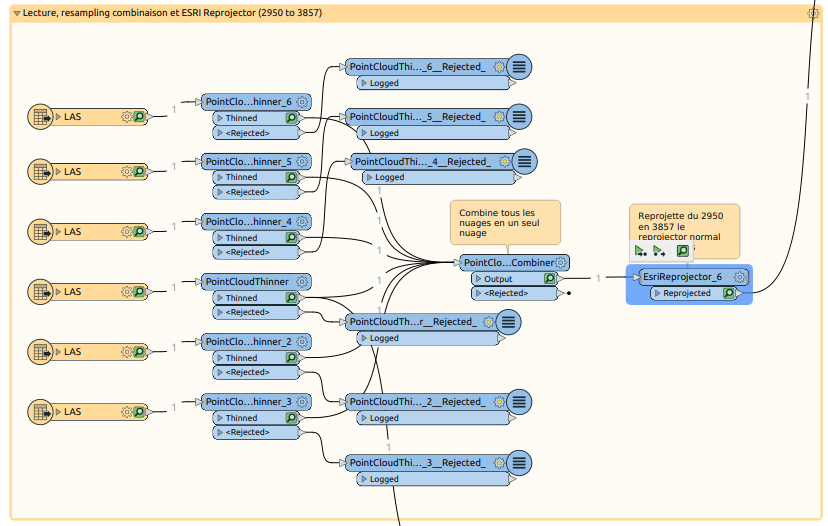
* Ensuite ajoutez y un ***pointcloudcombiner*** pour combiner les 6 nuages de points en 1 seul

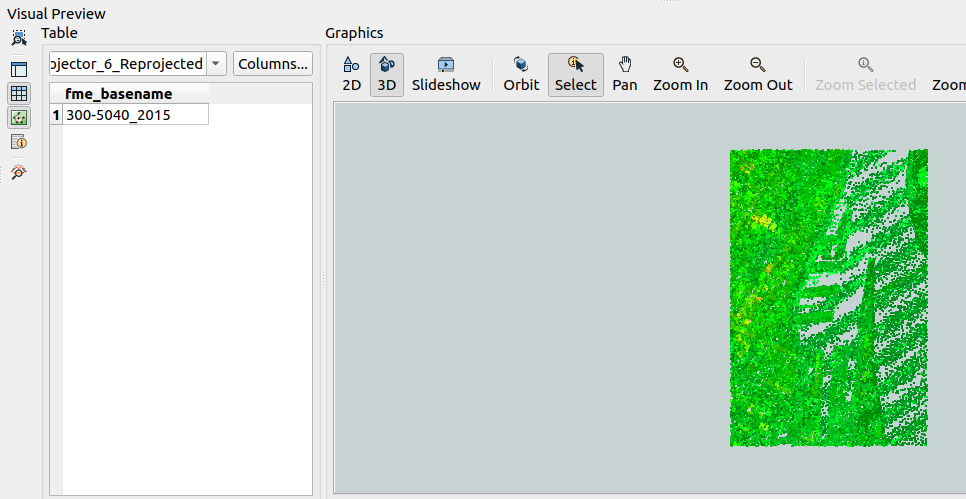


* Puis un ***EsriReprojector (2950 to 3857)***



* La première étape devrait ressembler à ca



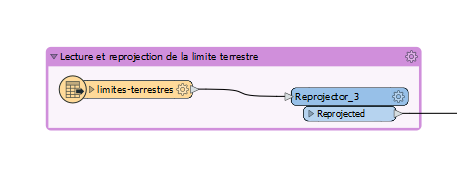


# Étape 2 Importation des limites terrestres et découpage du nuage de points

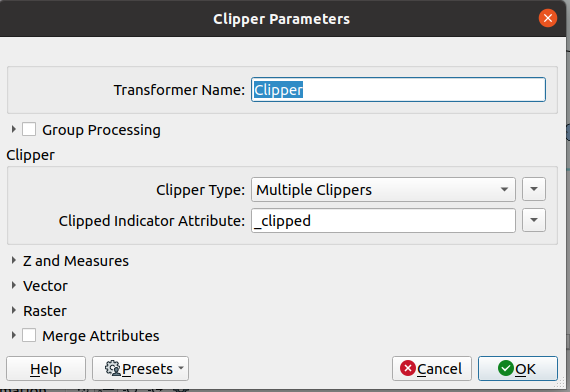
* Ajoutez un ***GeoJSONReader (URL)*** puis un ***Reprojector 4326 vers 3857***

Datasource:

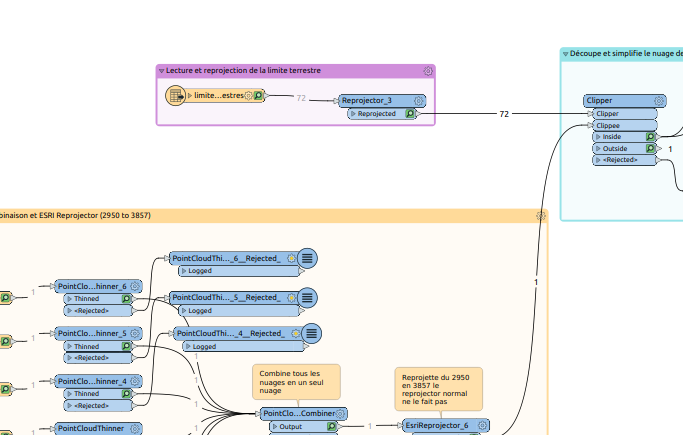
<https://data.montreal.ca/dataset/b628f1da-9dc3-4bb1-9875-1470f891afb1/resource/92cb062a-11be-4222-9ea5-867e7e64c5ff/download/limites-terrestres.geojson>



Ajoutez un ***Clipper*** pour découper le nuage de puis avec la limites terrestres pour enlever les points superflus qui dépasse de notre zone de travail



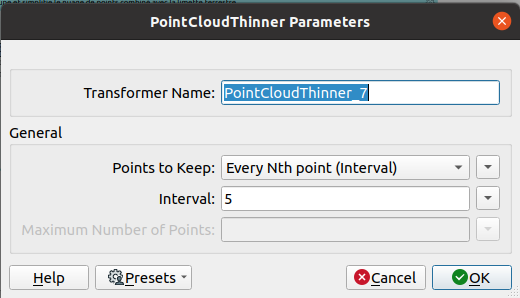
Cela devrait ressembler à ca

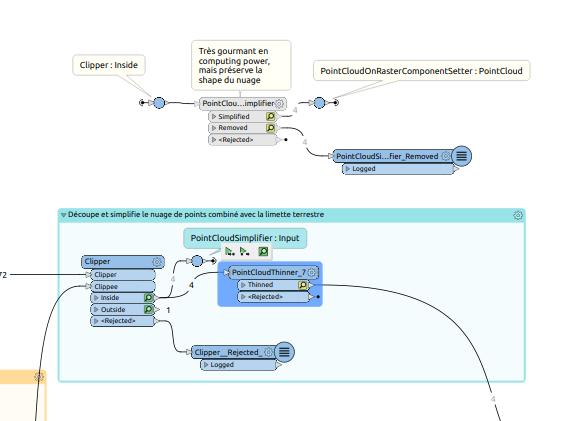


Connectez le nuage de points combiné comme ***Clippee*** et les limites terrestres comme ***Clipper***

# Étape 3 Simplification du nuage de points

Ensuite refaite un point ***thinner*** interval de 5 pour encore simplifier le nuage de points, normalement nous devrions effectuer un ***PointCloudSimplifier*** qui rend le nuage plus léger mais conserve sa forme, néanmoins ce processus est très gourmand en processing power, nous allons donc utiliser un thinner pour les biens de l’exercice

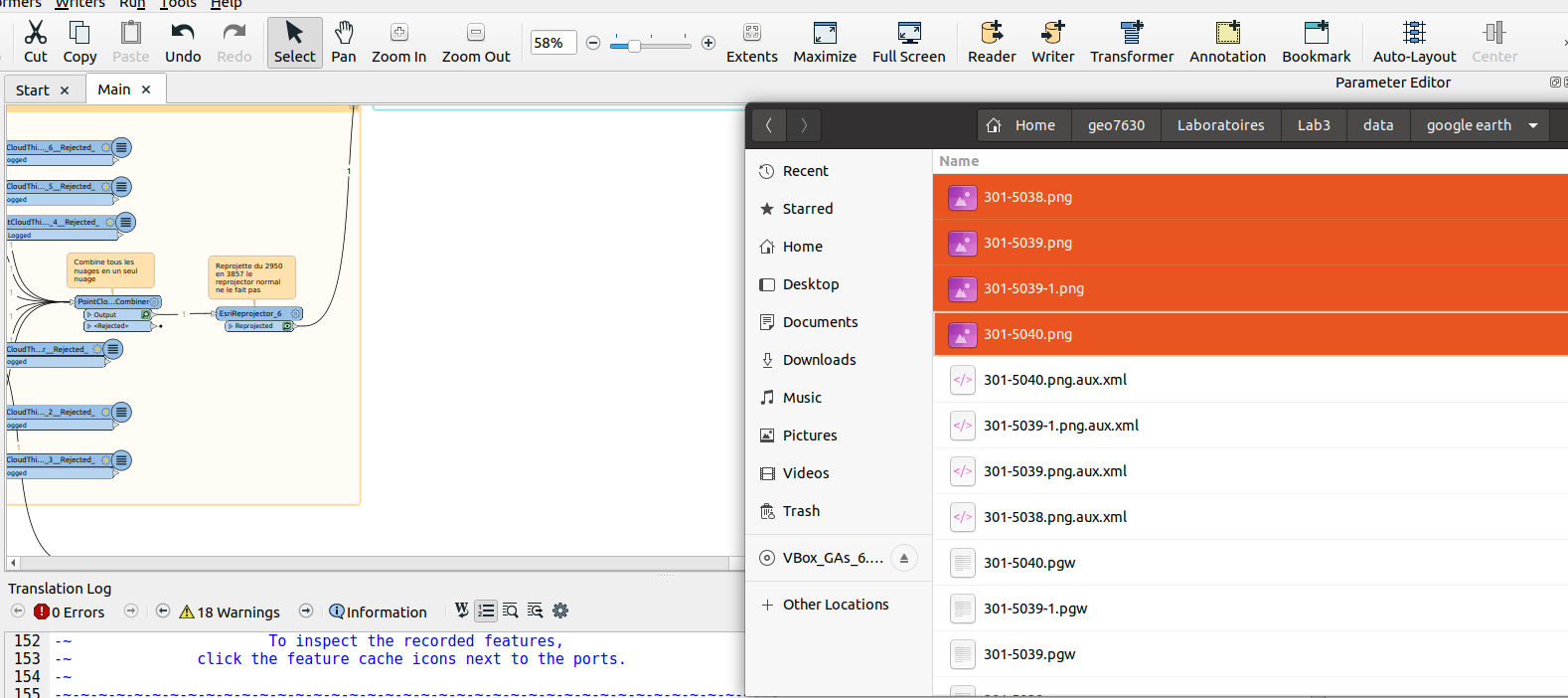




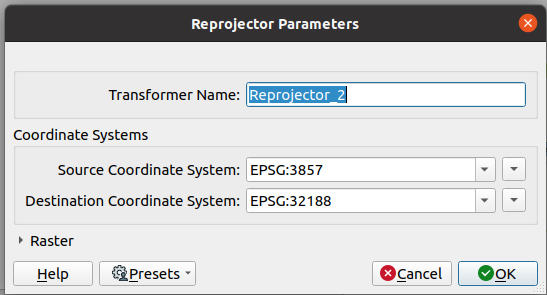
# Étape 4 Ajout de rasters géoréférencés

Cet ajout nous permettra d’ajouter la couleur du raster à notre nuage de points

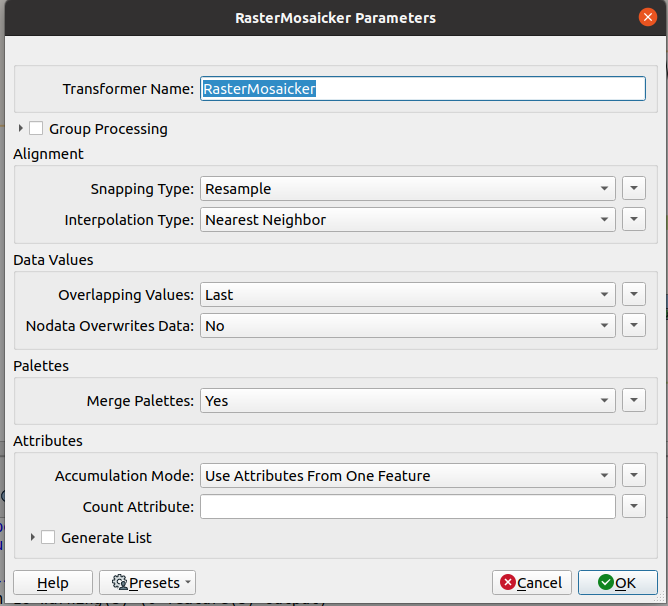
* Ajouter les 4 rasters d’un seul coup en sélectionnant les 4 et en faisant un drag and drop



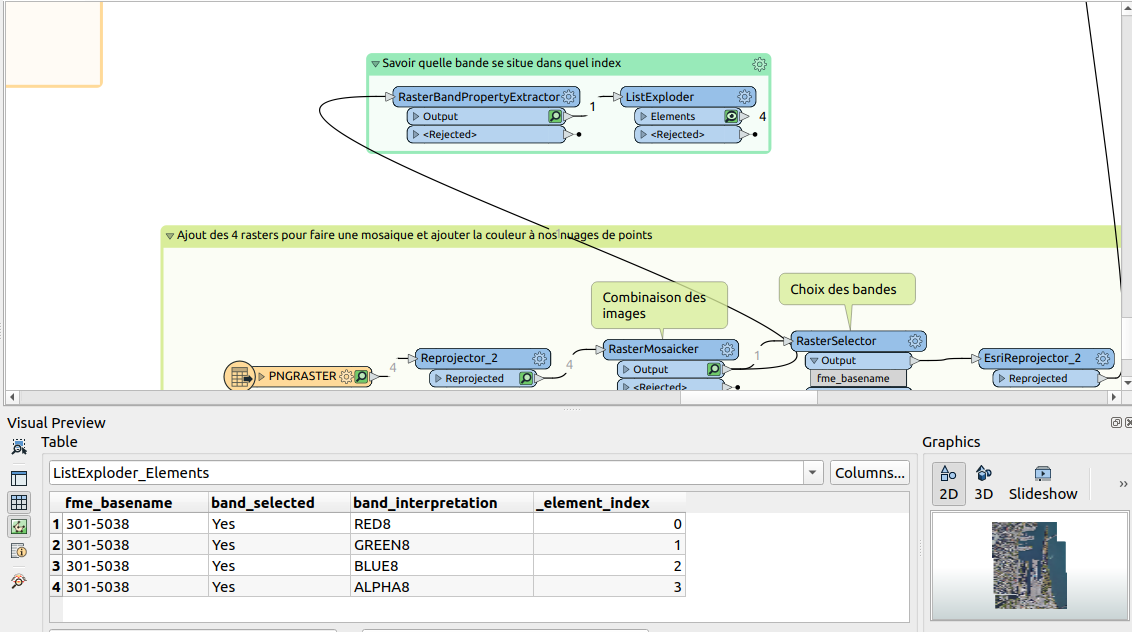
* Ajoutez un ***reprojector*** 3857 vers 32188 (équivalent au 2950 mais au standard de reprojection FME)



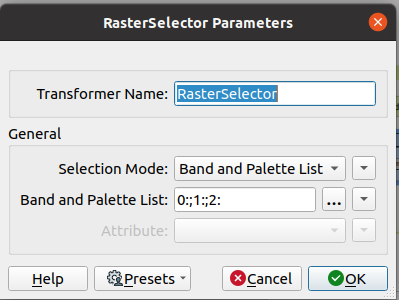
* Ajouter un ***rasterMosaicker***



* Puis ajouter un ***RasterSelector*** pour choisir les 3 bandes RGB et supprimer la bande alpha
* Pour connaître sur quelle bande se situe quelle couleur vous pouvez ajouter un ***rasterPropertyExtractor*** et collez y un ***listExploder*** sur la propriété BAND



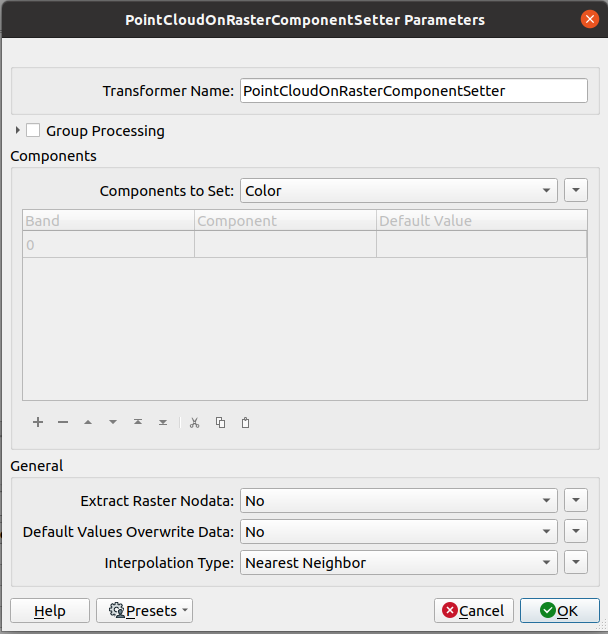
* Maintenant paramétrer le ***RasterSelector*** pour ne pas sélectionner la bande alpha



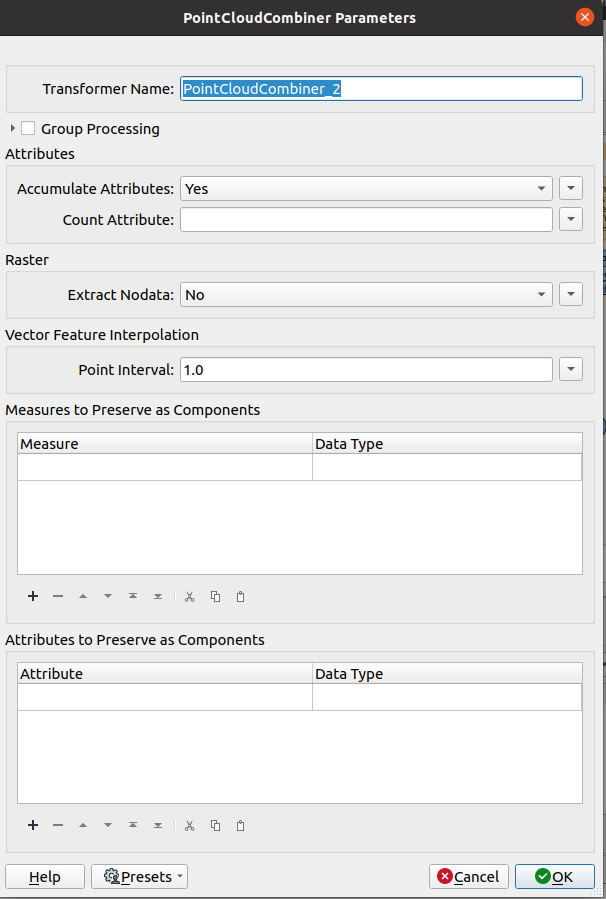
* Ajouter un ***EsriReprojector (2950 to 3857)*** avec les mêmes paramètres que celui de la première étape

Étape 6 Jointure raster et nuage de points

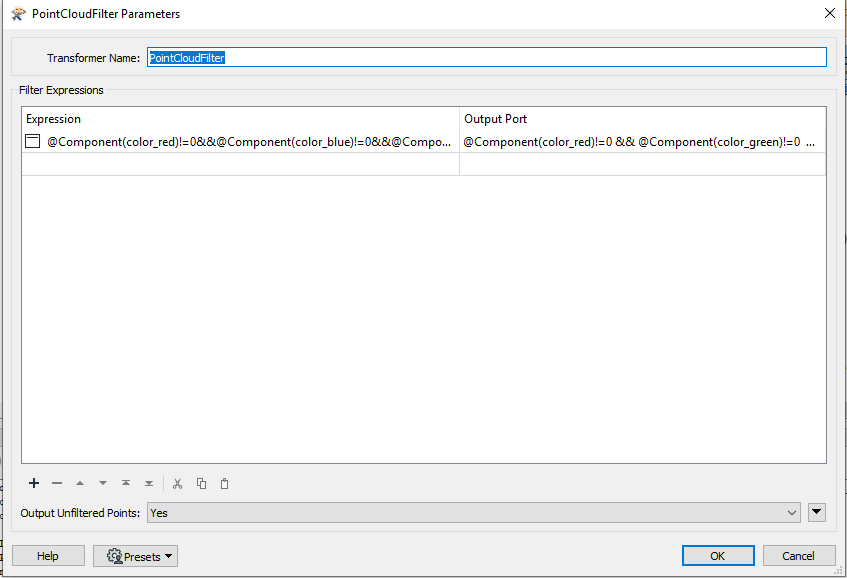
* Ajoutez la couleur dans votre nuage de point en ajoutant les valeur du raster avec un ***PointCloudOnRasterComponentSetter***



* Ensuite on va combiner le résultat en 1 seul nuage avec un ***PointCloudCombiner***



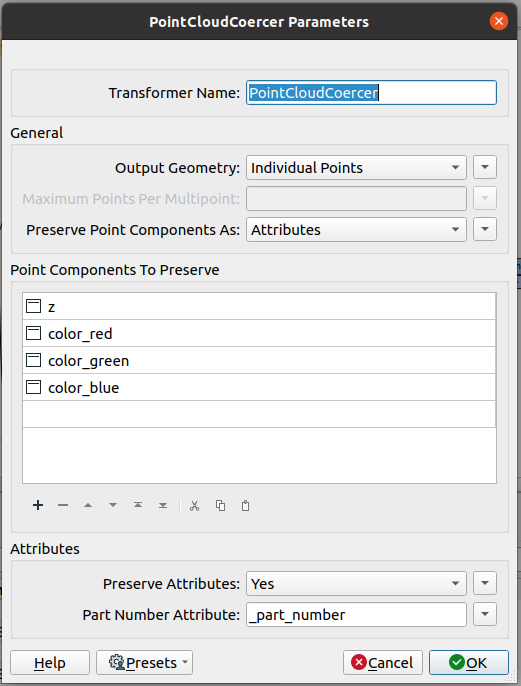
* Ensuite on va filtrer les valeurs du nuage de points dont le raster n’a pas donné de valeur avec un ***pointCloudFilter***



Pour filtrer on va choisir les valeurs RGB qui sont toutes à 0 avec l’expression suivante :

*@Component(color\_red)!=0&&@Component(color\_blue)!=0&&@Component(color\_green)*

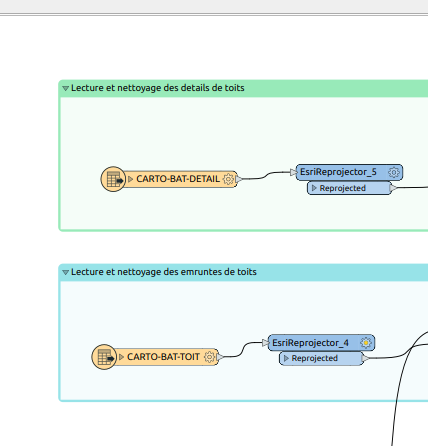
* Puis pour terminer nous allons transformer le nuage de points en couche de vecteurs ponctuels simple avec un ***PointCloudCoercer***, en s’assurant de garder les composantes nécessaires pour la suite

**

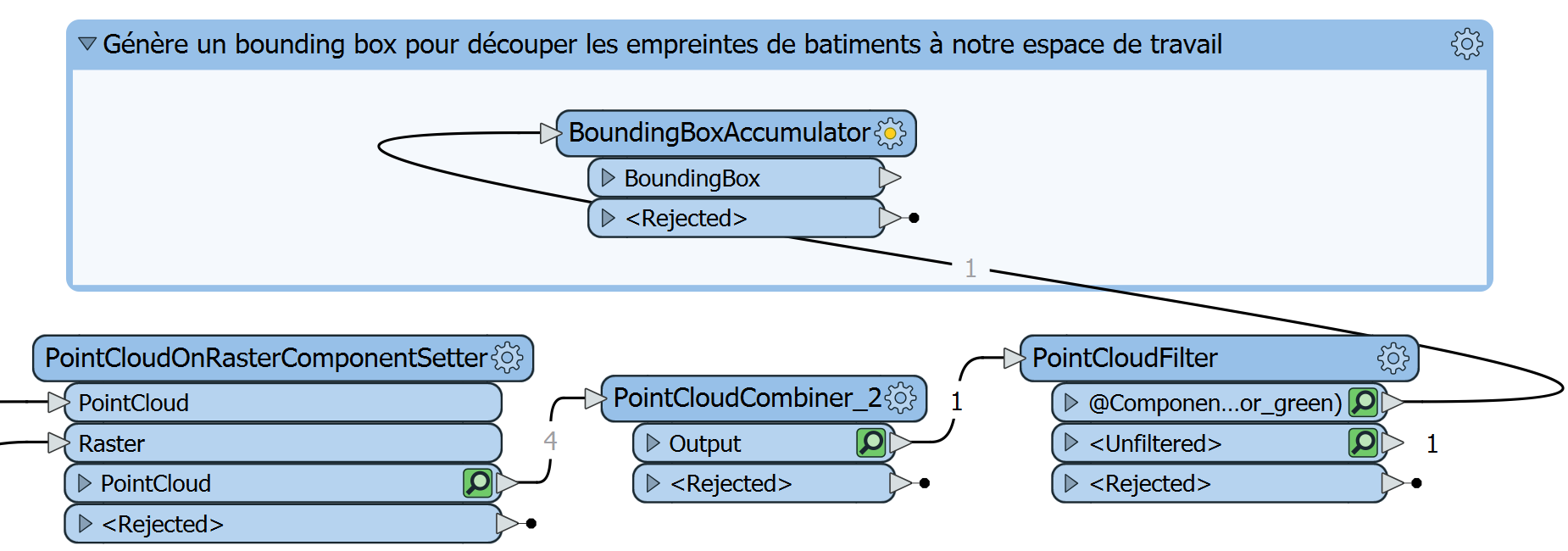
# Étape 7 Ajout des empruntes et details de bâtiments

Maintenant que le nuage de point est nettoyé et préparé, nous allons assigner le Z et la couleur aux polygones de bâtiments

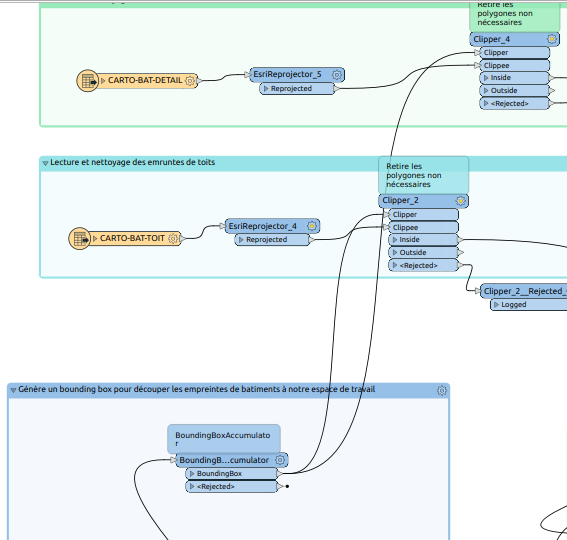
* Ajoutez 2 ***sources shapefiles*** pour les empreintes de toits (polygones) et les détails des toits (lignes)
* Reprojetez 2950 en 3857 avec un ***EsriReprojector***



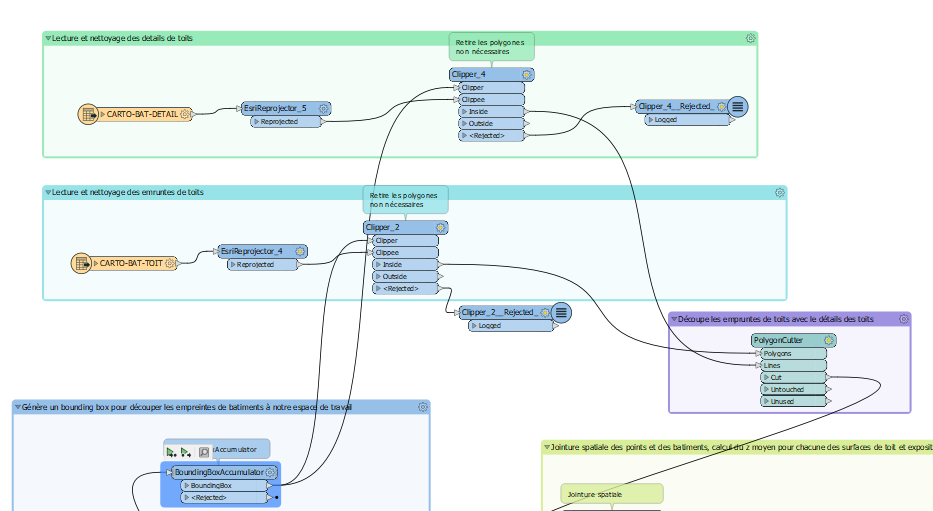
* Calculez le bounding box du nuage de points avec un ***BoundingBoxAccumulator***



* Ensuite on découpe avec un Clipper les polygones et les lignes

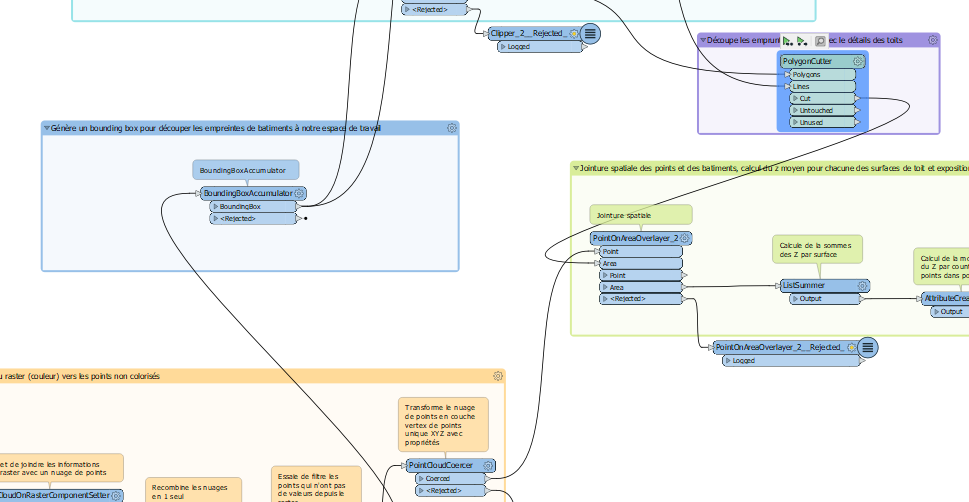


* Et on découpe les empreintes de toits avec les détails avec un ***PolygonCutter*** issue du ***FMEHub***

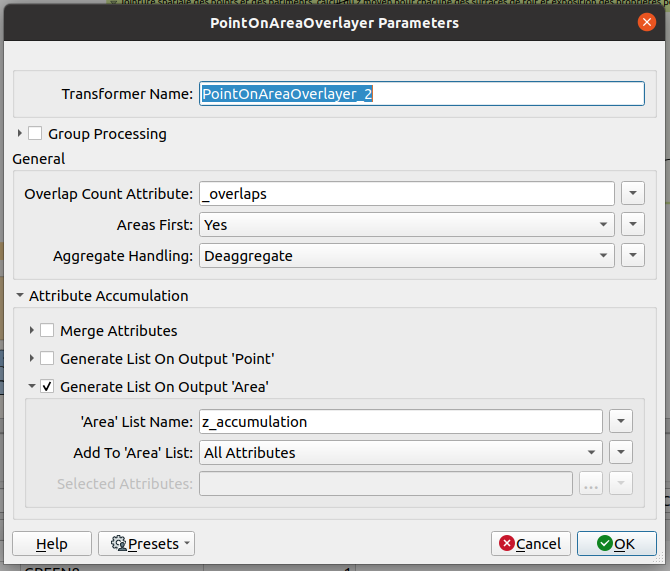


# Étape 8 Jointure des propriétés du nuage de points dans les polygones

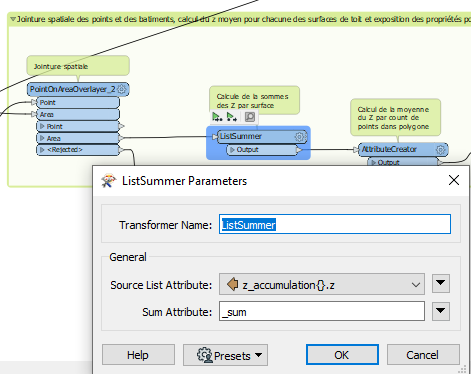
* Puis on vient joindre les polygones détaillés avec les points du nuages de points pour y injecter les valeurs de Z et de couleur du batiments avec un ***PointOnAreaOverlayer***



! Attention ! ici on va accumuler les informations de Z et de couleurs pour ensuite calculer les moyennes qui nous permettront d’attribuer à chaque “bâtiments” une hauteur moyenne et une couleur

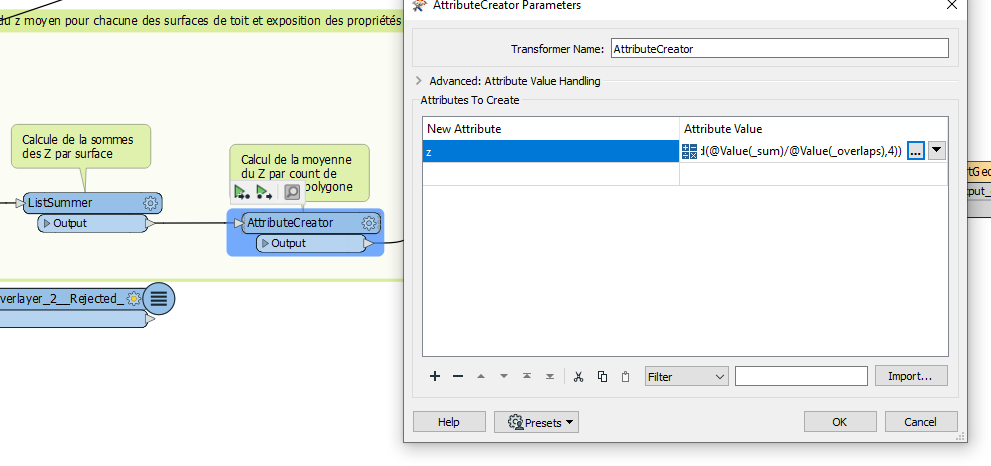


* On ajoute un ***ListSummer***

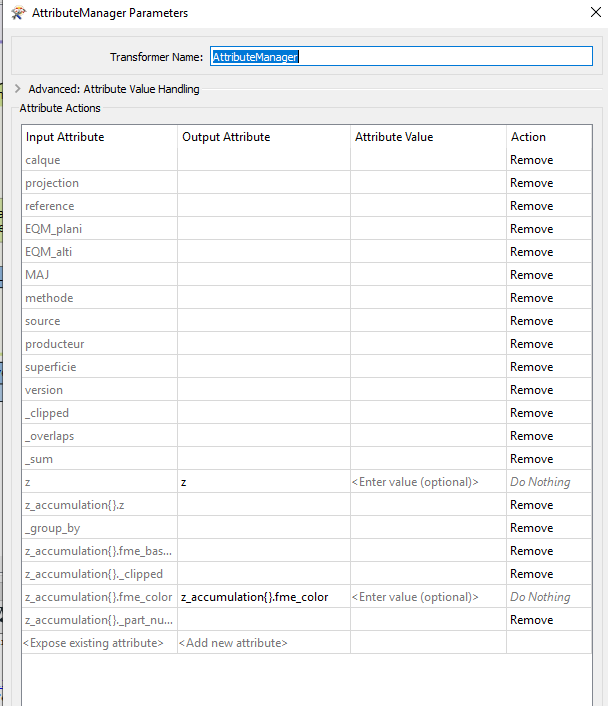


* Ensuite on va créer un attribut pour calculer la moyenne ***AttributeCreator***

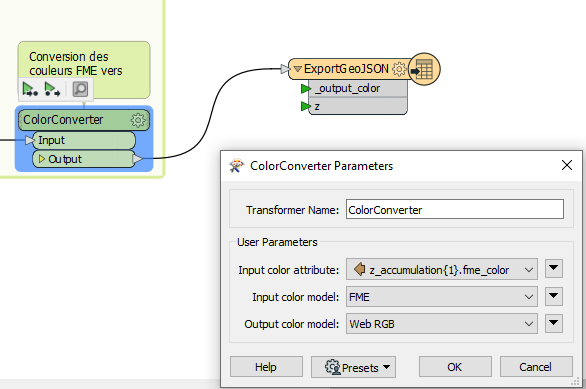
*@Evaluate(@round(@Value(\_sum)/@Value(\_overlaps),4))*



* Ensuite on fait un peu de ménage pour garder seulement les 2 propriétés qui nous intéressent avec un ***AttributeManager***



* Ensuite on transforme le FME color en RGB ou WebRGB avec un ***ColorConverter*** (FME Hub)

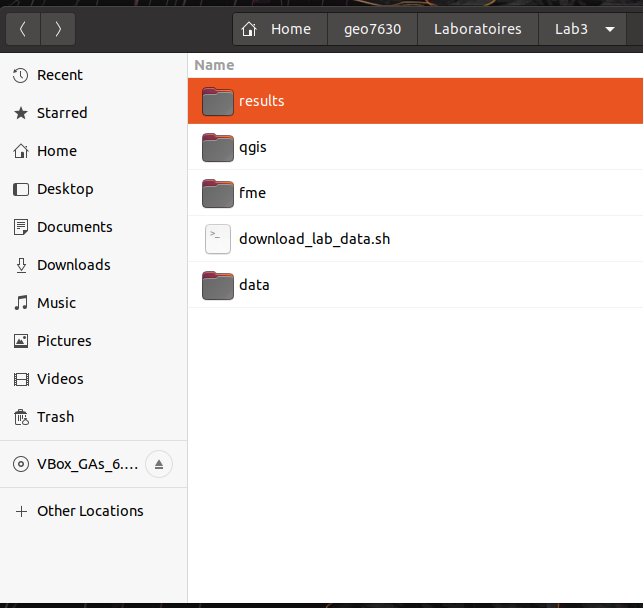


* Puis on exporte notre résultat avec ***GeoJSONWriter*** dans le répertoire :
  + *…/Lab3/results/bati3d.json*

# 

# Étape 9 Visualisation du résultat dans MaplibreGL

* Ouvrez un navigateur de fichiers et allez dans le dossier Lab3



* On ouvre le fichier MaplibreGL.html en double cliquant dessus dans le répertoire /Lab3/results/MaplibreGL.html
* Cliquez sur le bouton importer en haut à gauche et choisissez le bati3d.json, cela vous permettra de naviguer et visualiser les bâtiments à votre guise.

