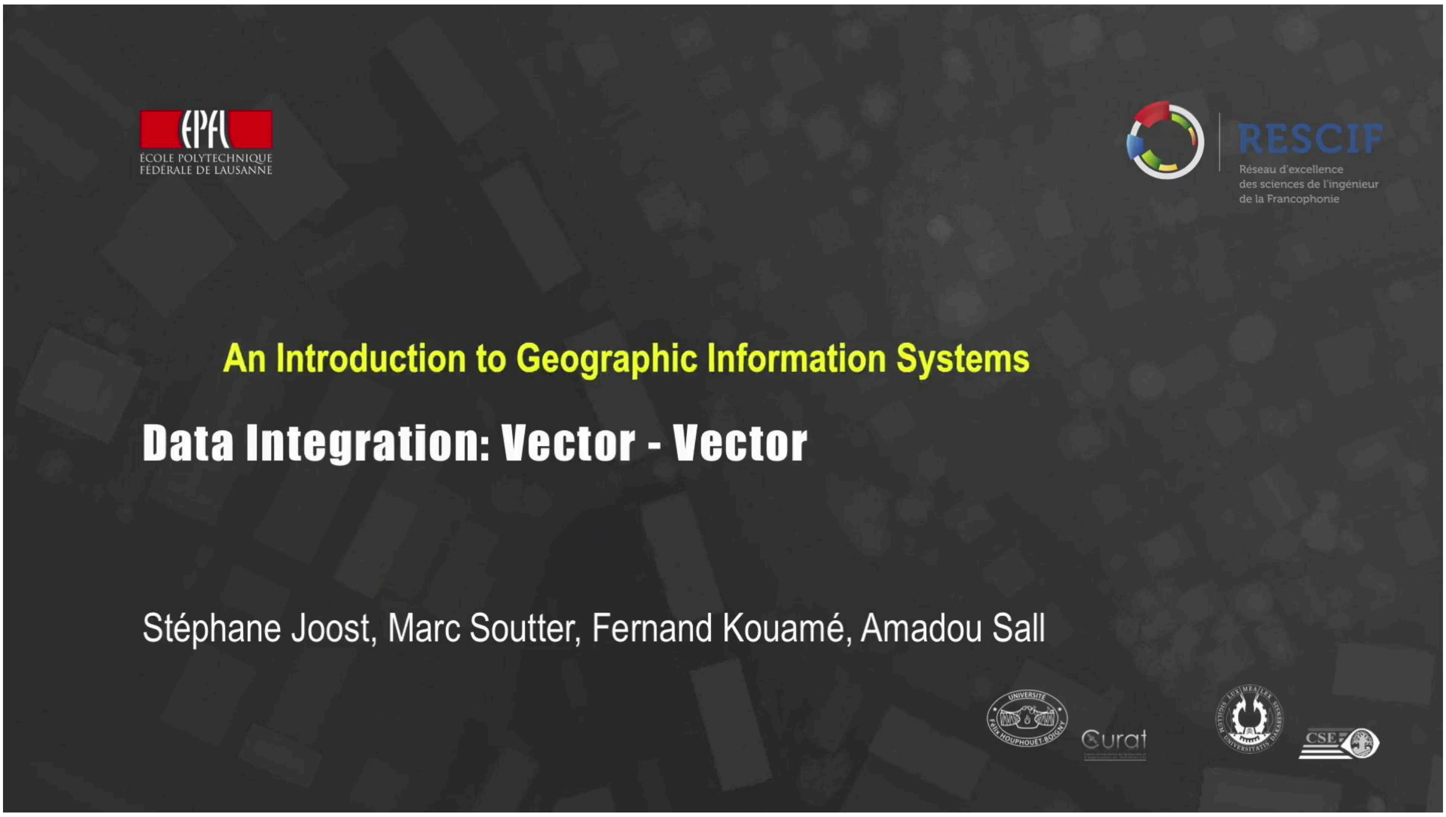
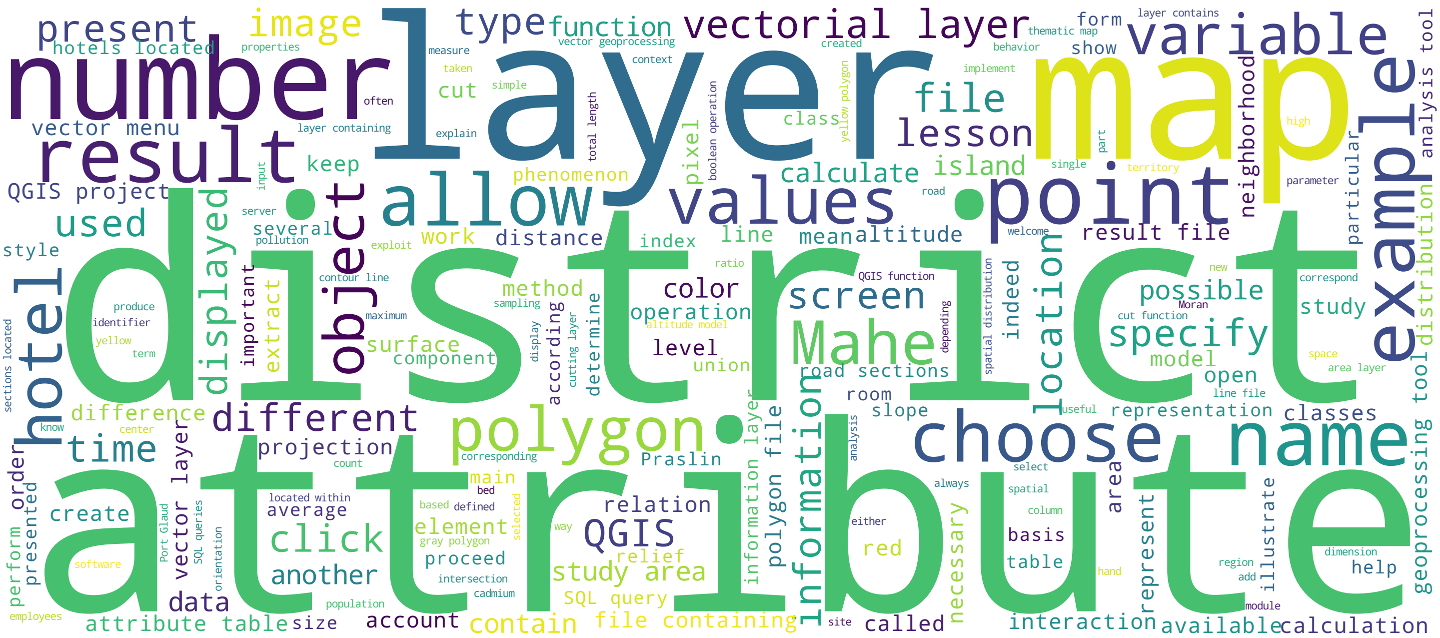
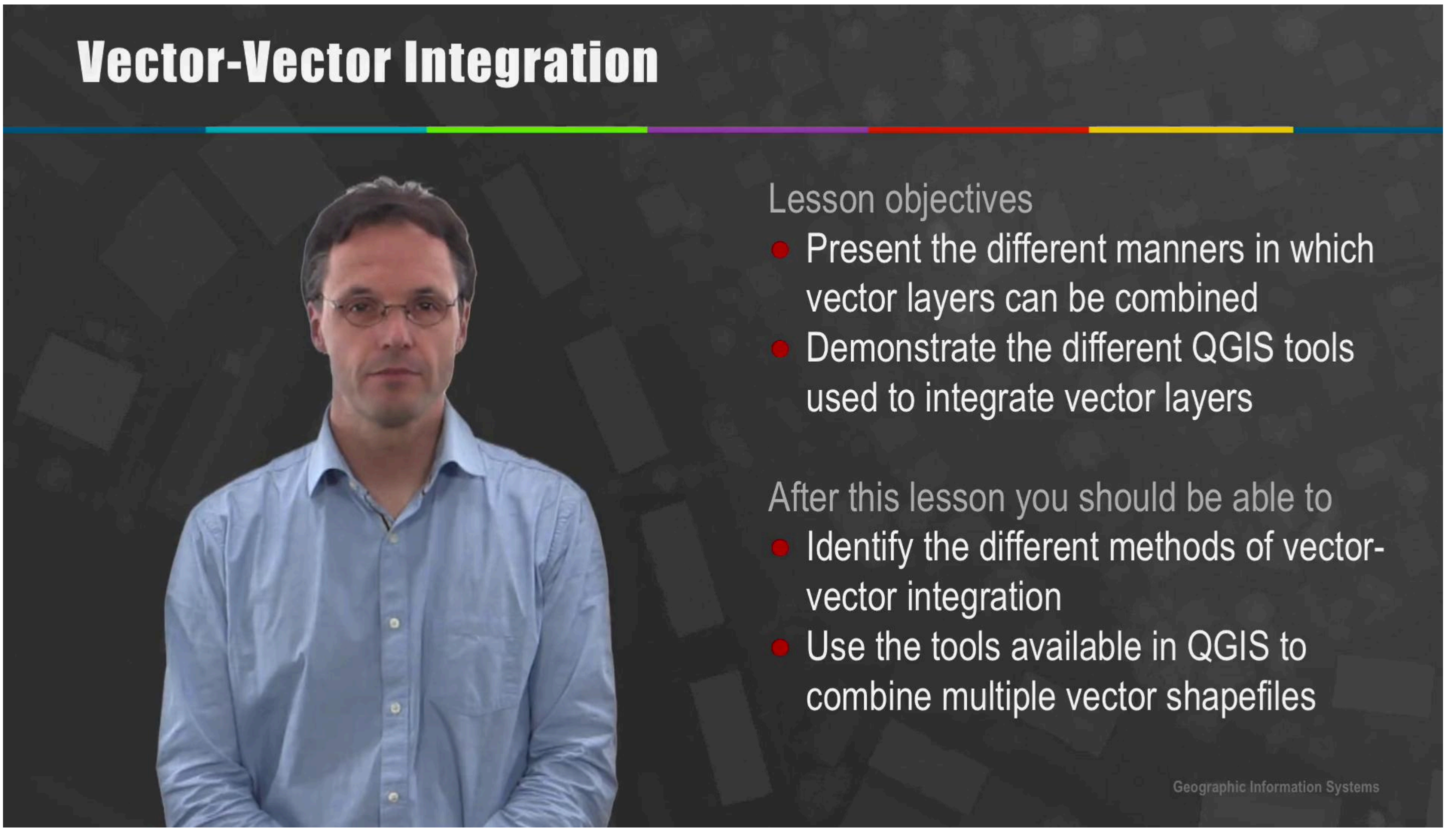




Search MOOC

Video

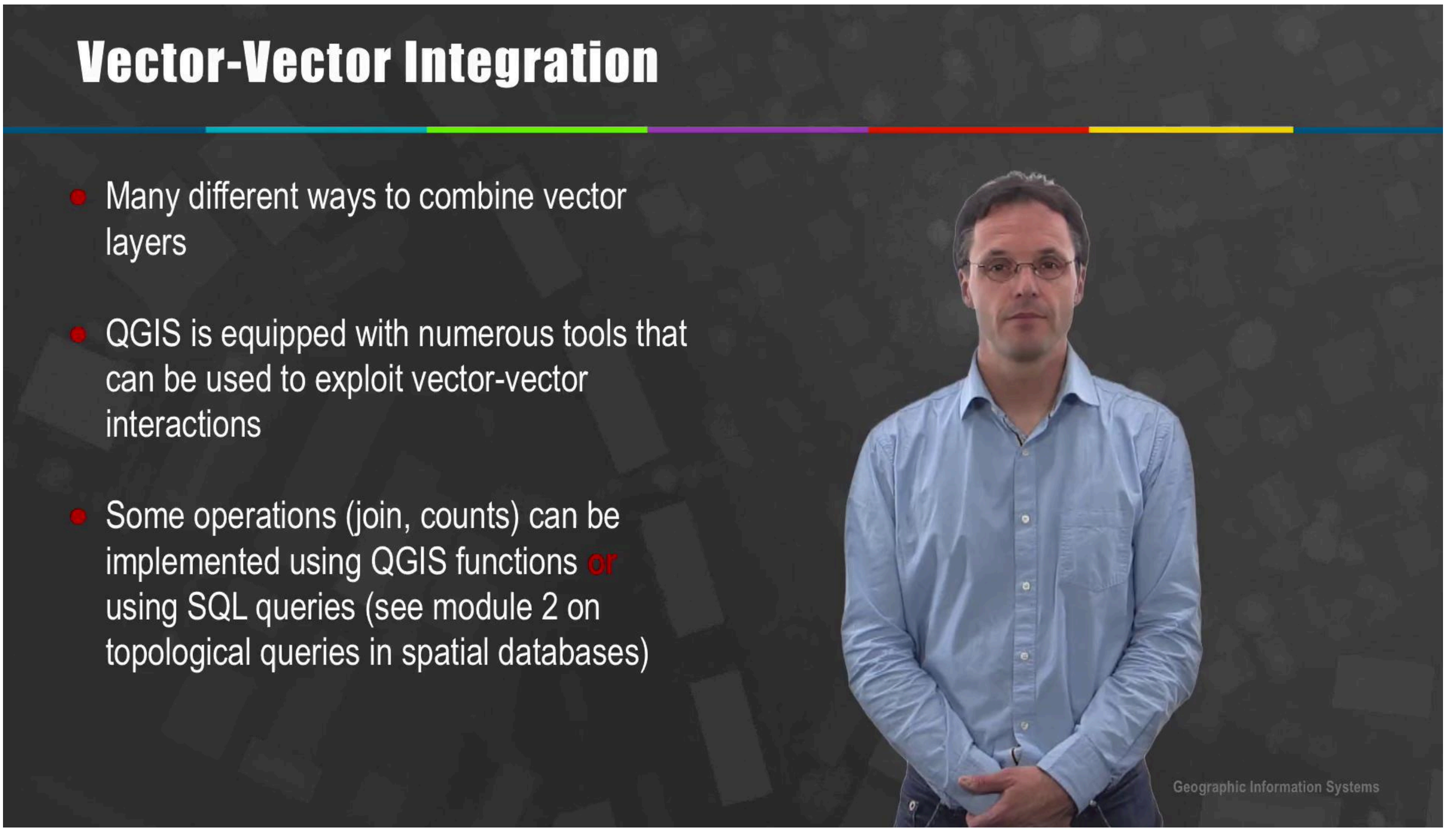


Hello and welcome to this third lesson dedicated to interactions between spatial information layers. We will focus this time on the operations that allow to connect several vectorial layers. Some of the vector-vector interactions can be implemented with the help of SQL queries which have already been dealt with in module 2 in the course on the topological spatial queries. This lesson is largely practical and it illustrates how the QGIS software can be used to make vectorial layers interact, typically, how to proceed to count the number of points stored in an information layer and located within a polygon located in another. The goals of this lesson are to explain to you what the different types of possible interactions are between several layers of vectorial geoinformation and to present what tools are available in QGIS that allow to perform the corresponding operations.

Notes

Summary

0m 31s

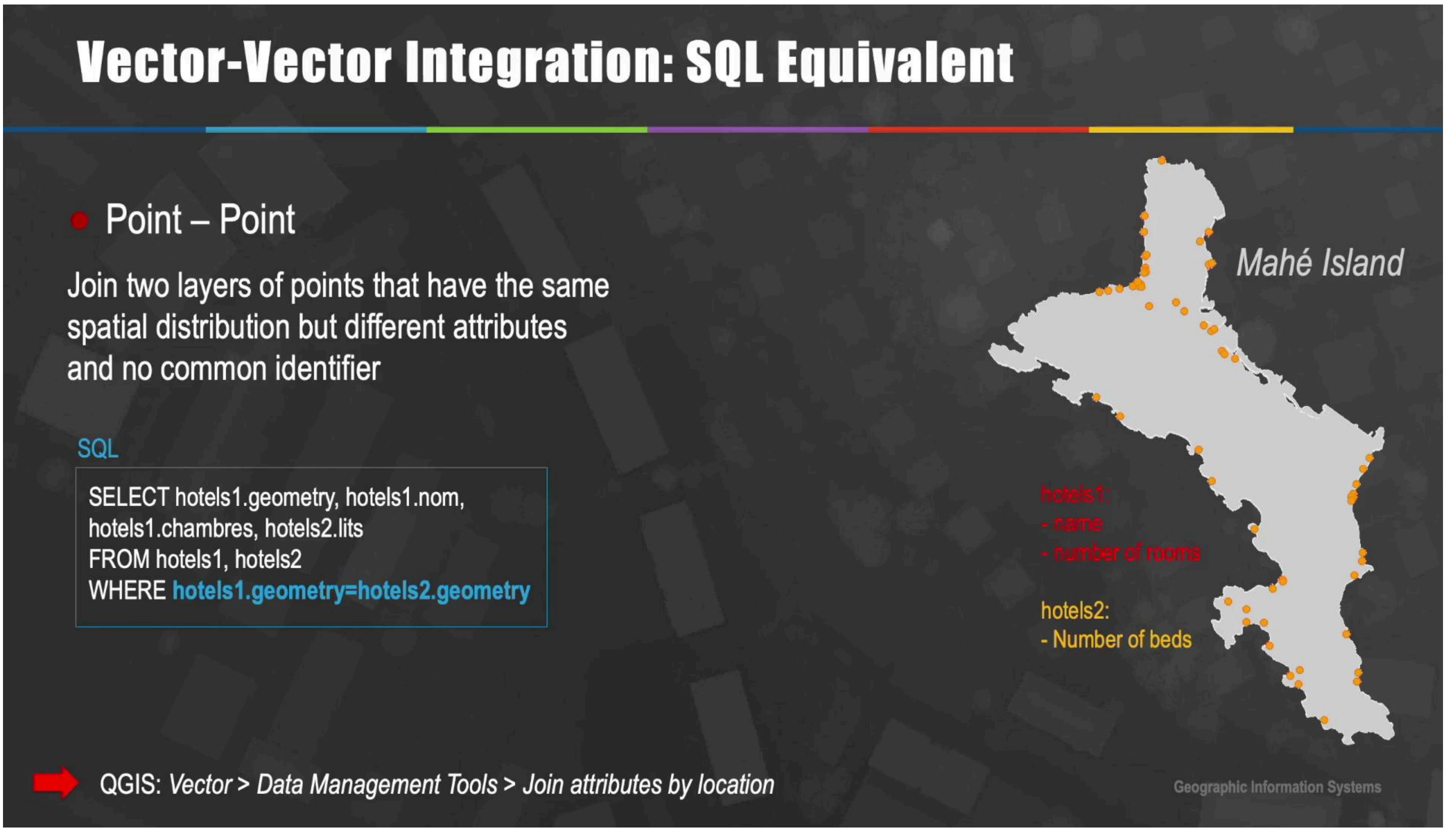


It is possible to exploit in different ways the interaction between several vectorial layers and the QGIS software contains a series of functions capable to implement them. It can be for example to cut a polygon located on a layer by means of a segment located on another. It should be noted that some operations like the junction of layers without common identifier or the enumeration of objects can be performed either through QGIS functions which are already existing, or by means of SQL queries that we presented in module 2 on topological spatial queries. We will now present you basic operations that it is essential to master to exploit the interaction between vectorial layers, by indicating the equivalent SQL queries when it is relevant.

Notes

Summary

1m 38s

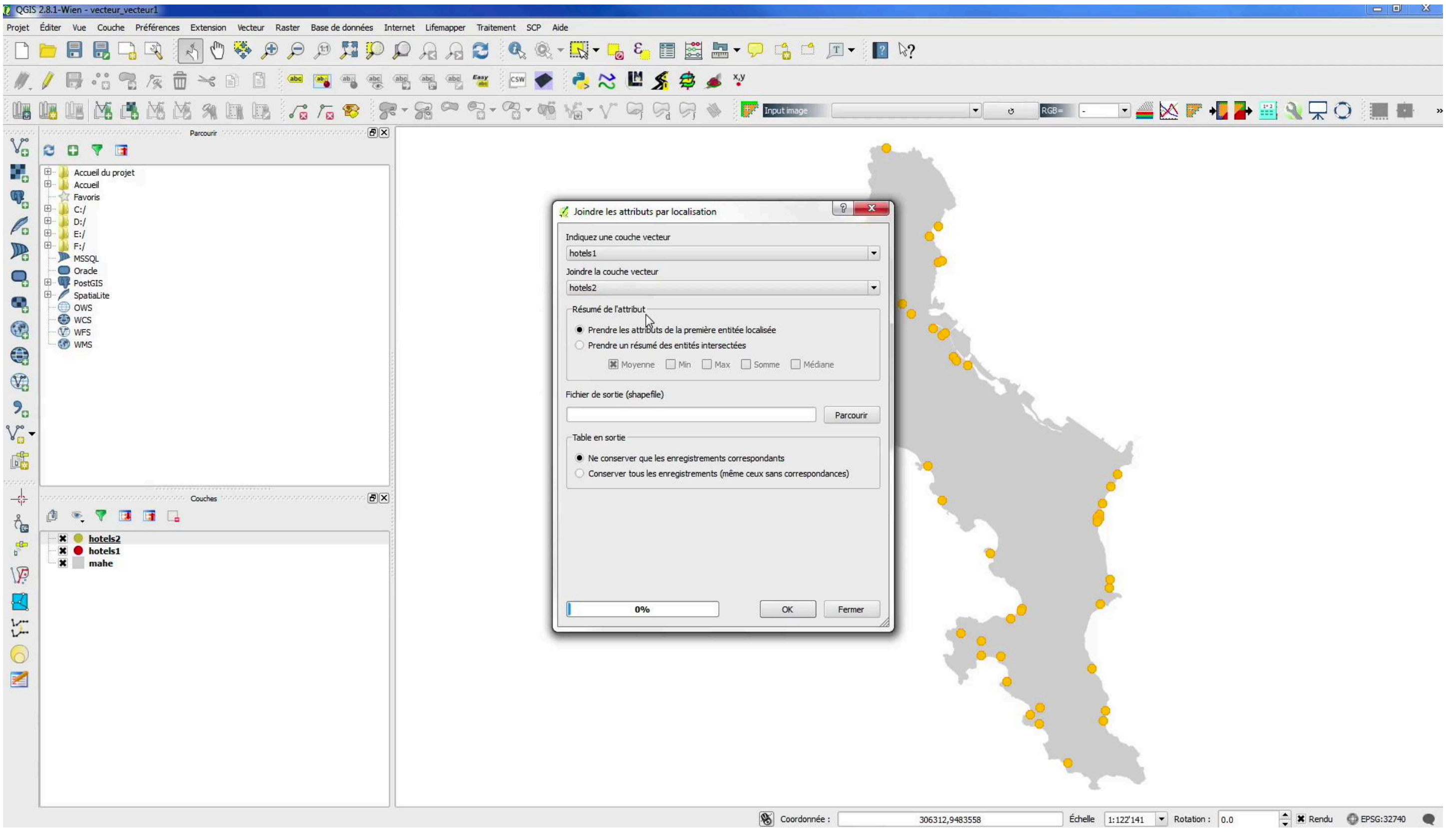


It can happen that a data set without a unique identifier and consisting of points representing here in our example of hotels, contains some of its attributes in a layer, here in red the name of the hotel and the number of rooms, and another part of its attributes in a second layer, here in orange the number of beds. To gather all the attributes in a single layer, it is necessary to proceed at the junction of the two files on the basis of spatial location. The corresponding SQL query, displayed here on the screen, uses the geometry of objects whose X and Y components must be strictly identical in both files so that the link can be made. The equivalent QGIS function is available in the "vector" menu under "data management tools", then "join the attributes by the location".

Notes

Summary

2m 31s

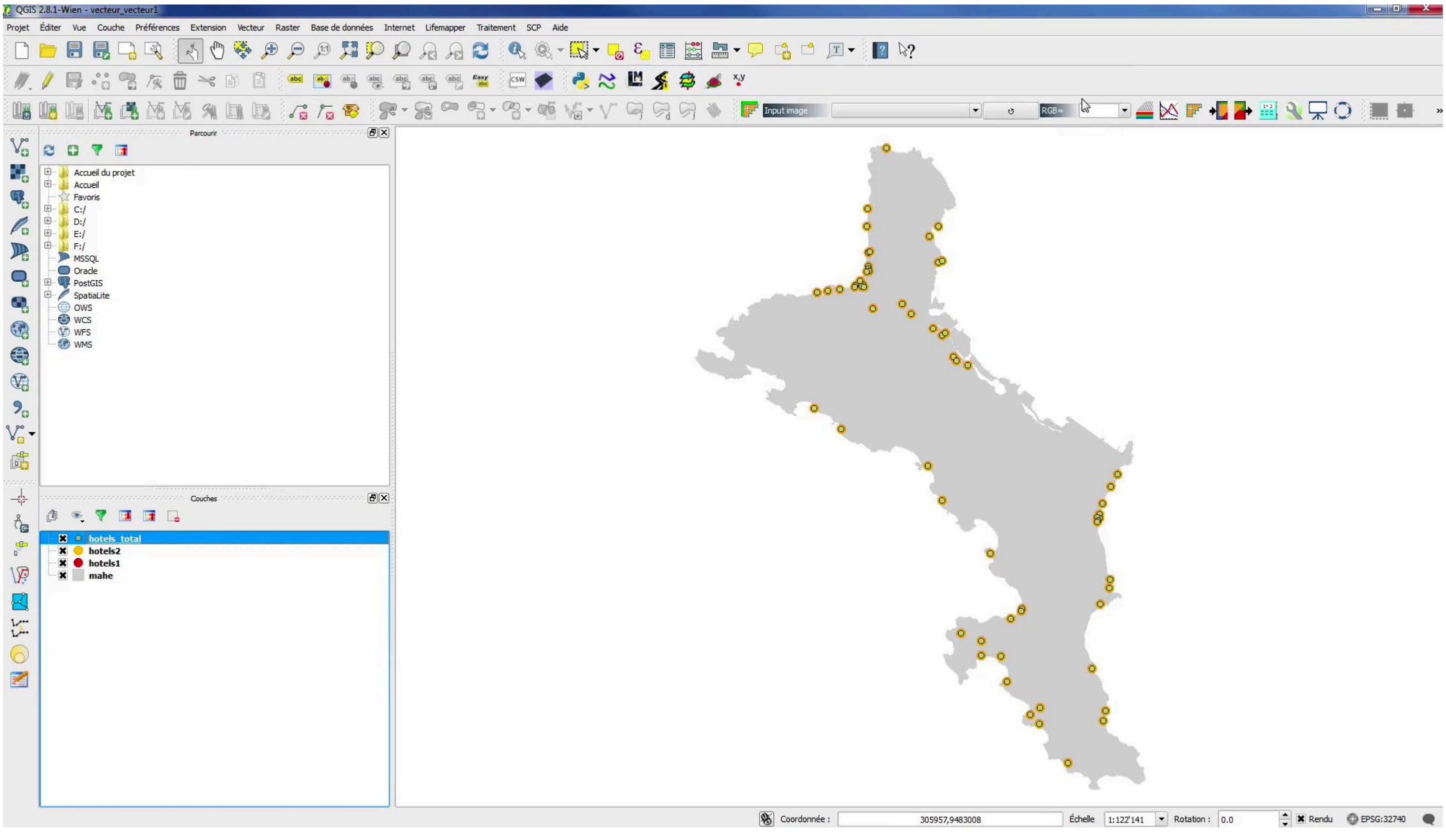


Let's see how to implement this function. In the QGIS project we have two vectorial layers of superimposed points containing information on the hotels of Mahe. In the attributes of the first layer, hotels1, we have the hotel name and the number of rooms. The second layer, hotels2, contains exactly the same hotels, but in the attributes, this time we only have the number of beds. We will join the attributes of these two layers by going under "vector" "data management tools", "join the attributes by the location". In the first menu "indicate a vector layer", we choose the vectorial layer hotels1, and in the menu "join the vector layer", we choose the layer hotels2. In the "summary of the attribute", we could choose to combine the available information on the two layers. If we have an attribute which has the same number in both tables, such as the number of employees, but that the values are not always the same between the two files, something that could happen for example if we have the employees in January in one file and in June in another file, we could choose here to take either the value of the first layer, or a summary of the values of the two layers, for example, the average, minimum or maximum number of employees.

Notes

Summary

3m 24s

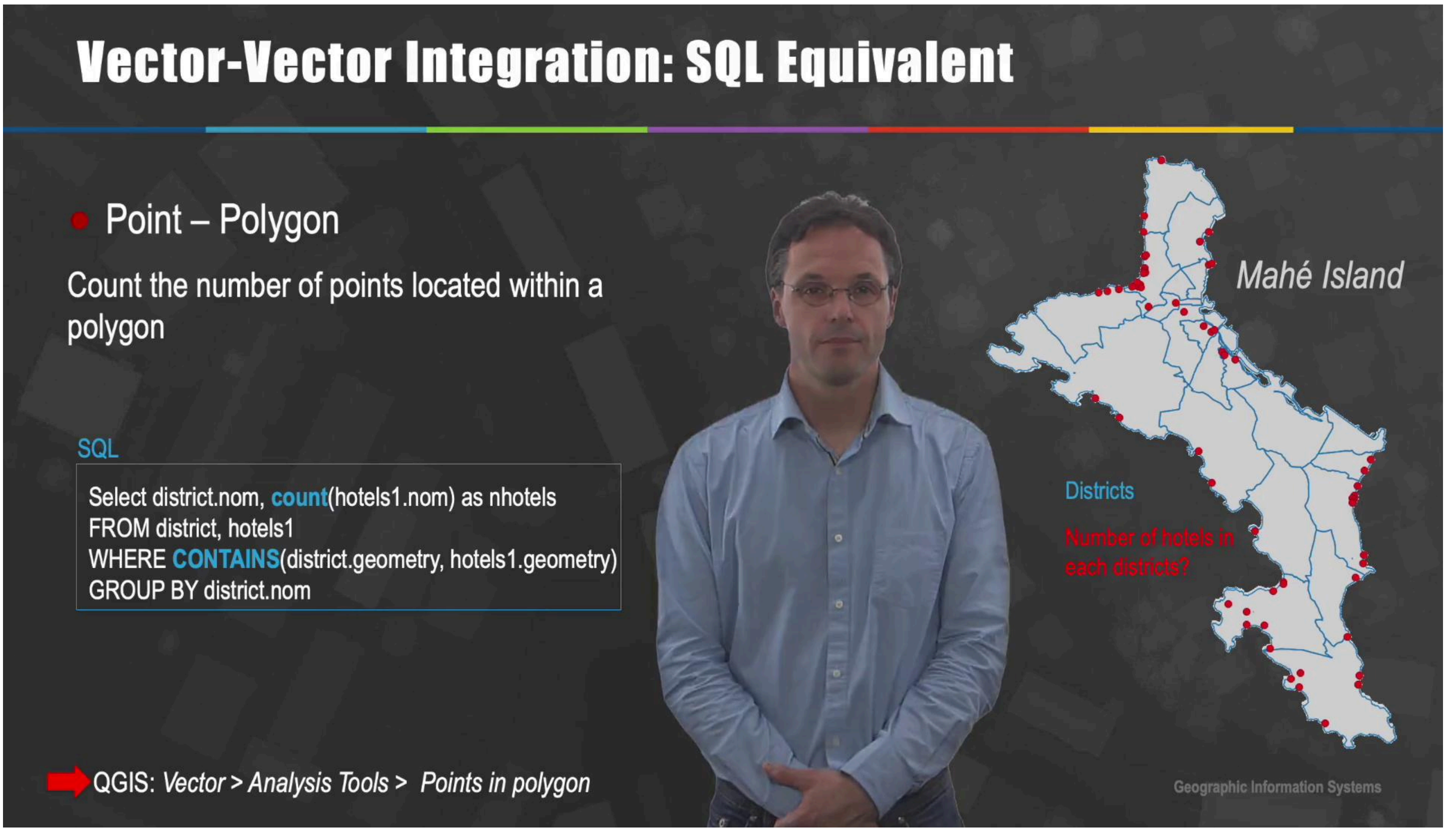


In our case, we don't have a common attribute and this part is therefore not useful. We then specify the name and location of the result file. Finally, in the last part of the window, you can choose to keep only the corresponding saved elements, that is to say the superimposed points or to keep also the elements saved without correspondance, for example in the case where some hotels would be present in only one of the two files. Here we choose to keep only the correspondences. And you can then click on OK. A new vector layer is created and this layer contains the points of all the hotels and in the attribute tables, we see that we recovered the attributes of the two initial layers, so the name, the number of rooms and number of beds.

Notes

Summary

4m 47s

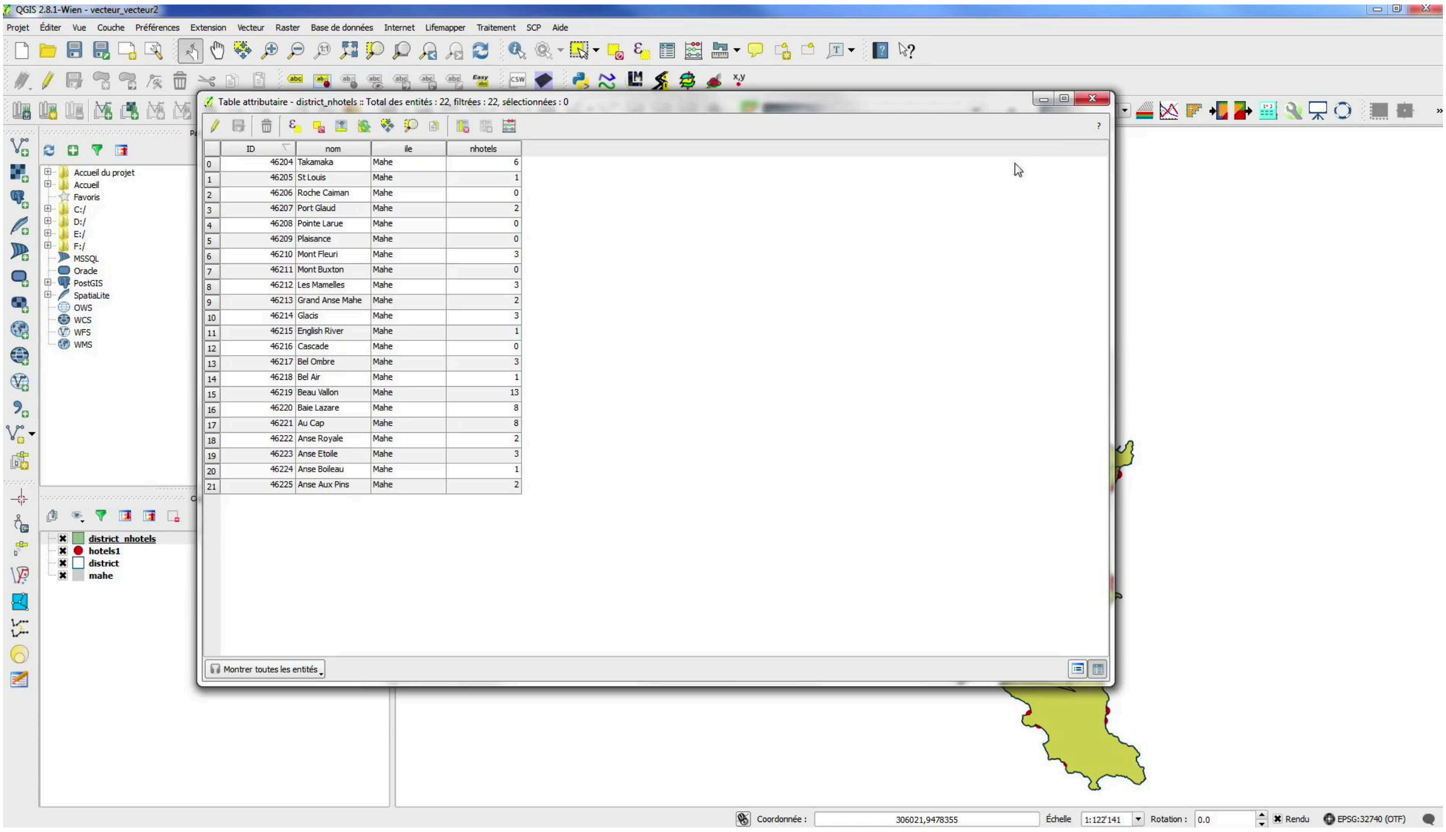


In this second situation, we will have to count the number of points stored in a first information layer and located within each polygon stored in another layer. In the proposed example, the territory of Mahe in the Seychelles is composed of a number of districts in blue within which we want to know the number of hotels represented here in red. The SQL query allowing to do this operation uses the "count", "contain", and "group" instructions. In QGIS, the function "points in a polygon" is accessible via the "vector" menu and then "analysis tools".

Notes

Summary

5m 43s

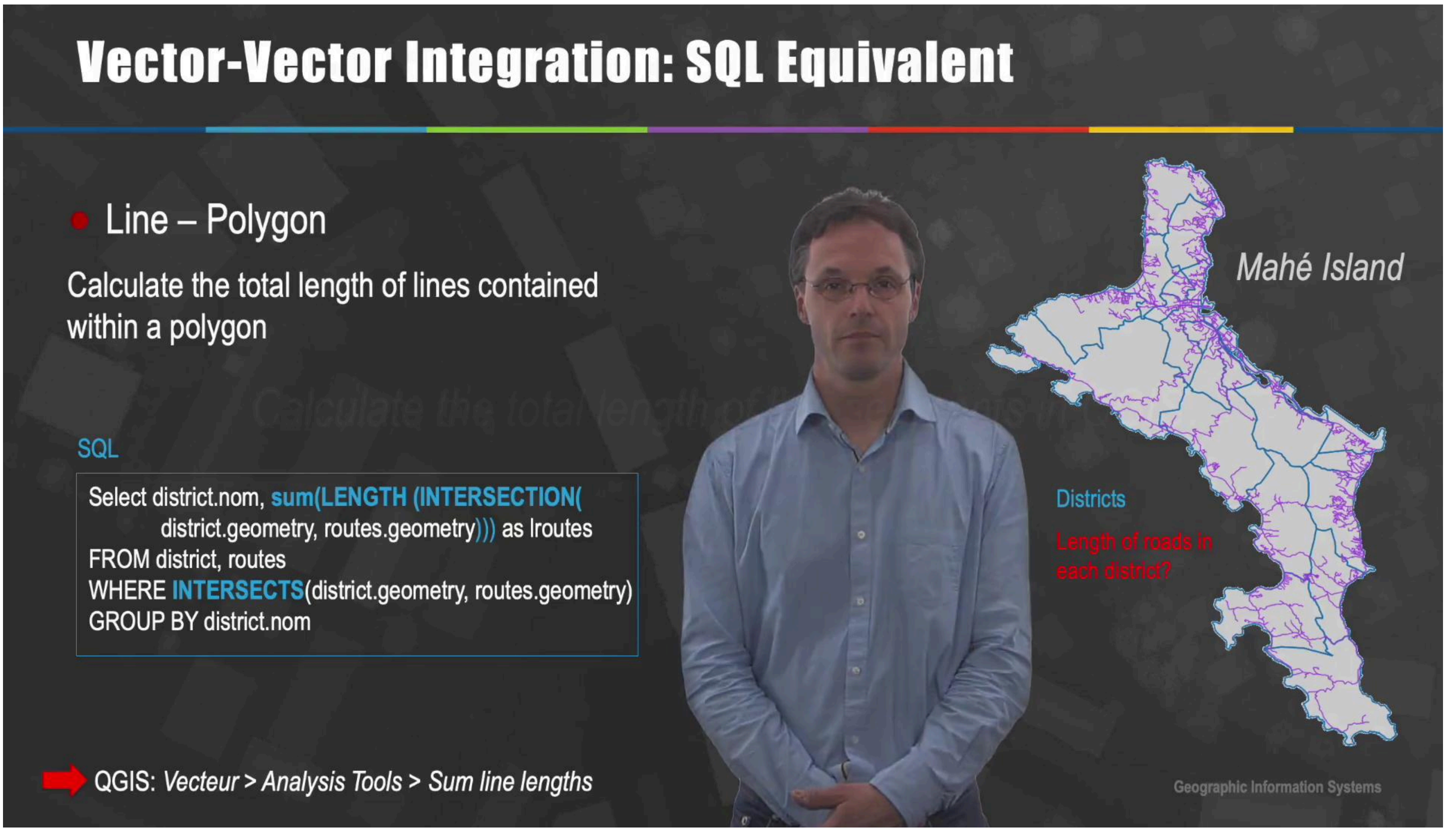


Here's how it works: in the QGIS project, we have a vectorial file of polygons containing the districts of Mahe, and a point file with the hotels. We will count the number of hotels located in each district using the "vector" tool, "analysis tools", "points in a polygon". In the "polygon vector layer as input" dropdown menu, we choose the district layer then under "points vector layer as input", we select hotels1. We can give a name to the column that will contain the number of hotels for each district. We will name it here: nhotels. Finally, we give the name and location of the result file. A warning appears to make us aware of a difference in projection because the file being created does not yet have an assigned projection system. So we can ignore this message here and click on OK. This is at the next step that we will specify the projection of the result layer. In our case, we work in the UTM 40 South zone system which is adapted for the Seychelles region. The result is displayed on the screen. It is a polygon file containing all the districts. If we open the attribute table, we see that there's a column nHotels indicating the number of hotels located in each district.

Notes

Summary

6m 22s

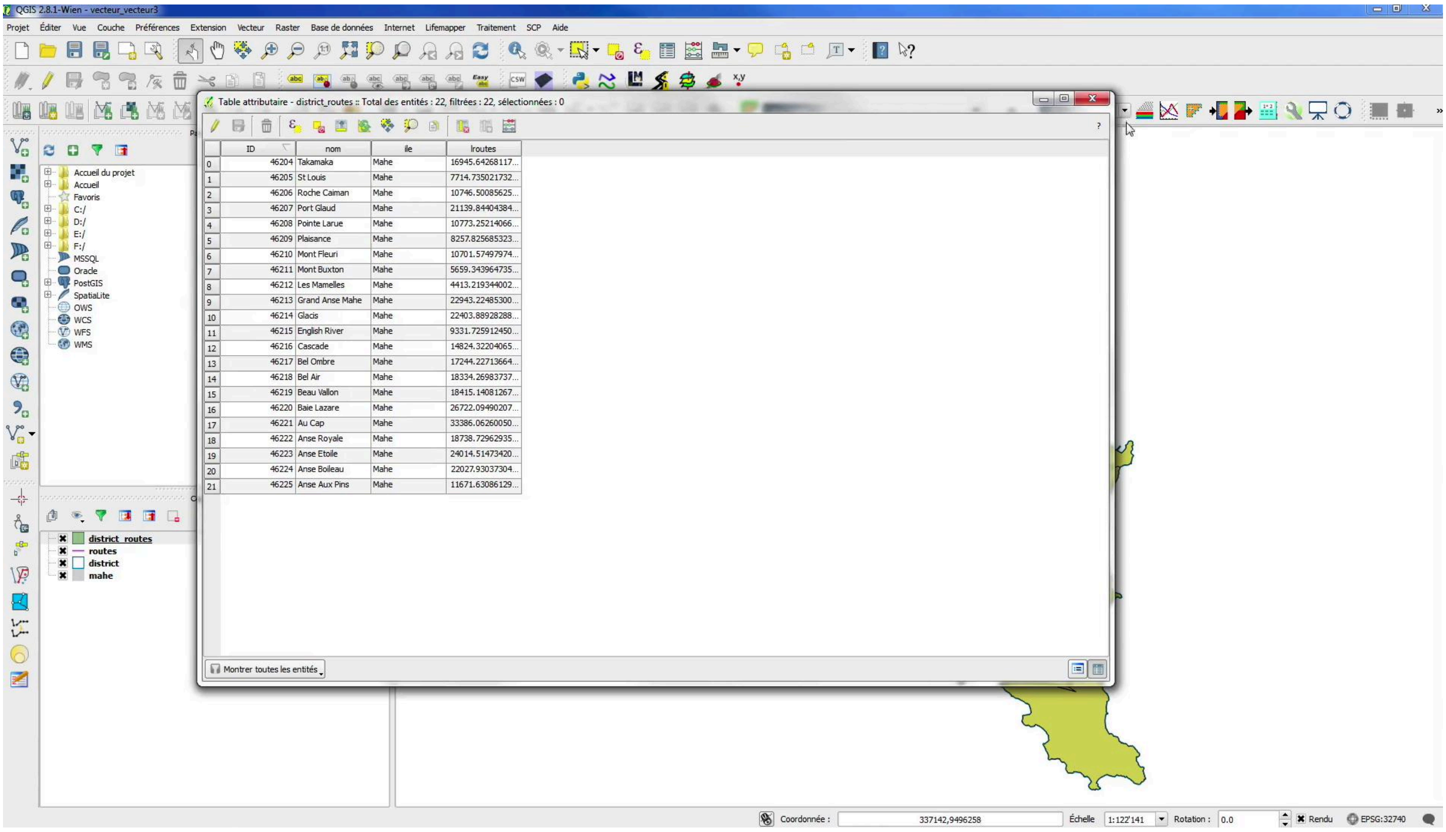


Now, still on the island of Mahe, we want to calculate the total length of the road network, here in purple, located within each district. The SQL query is more complex this time. It involves the "sum", "length" and "intersection" instructions, nested before grouping the distances by district. In QGIS it is the "total of the line lengths" function in the "vector" menu then "analysis tools", that allows to perform this operation.

Notes

Summary

7m 49s

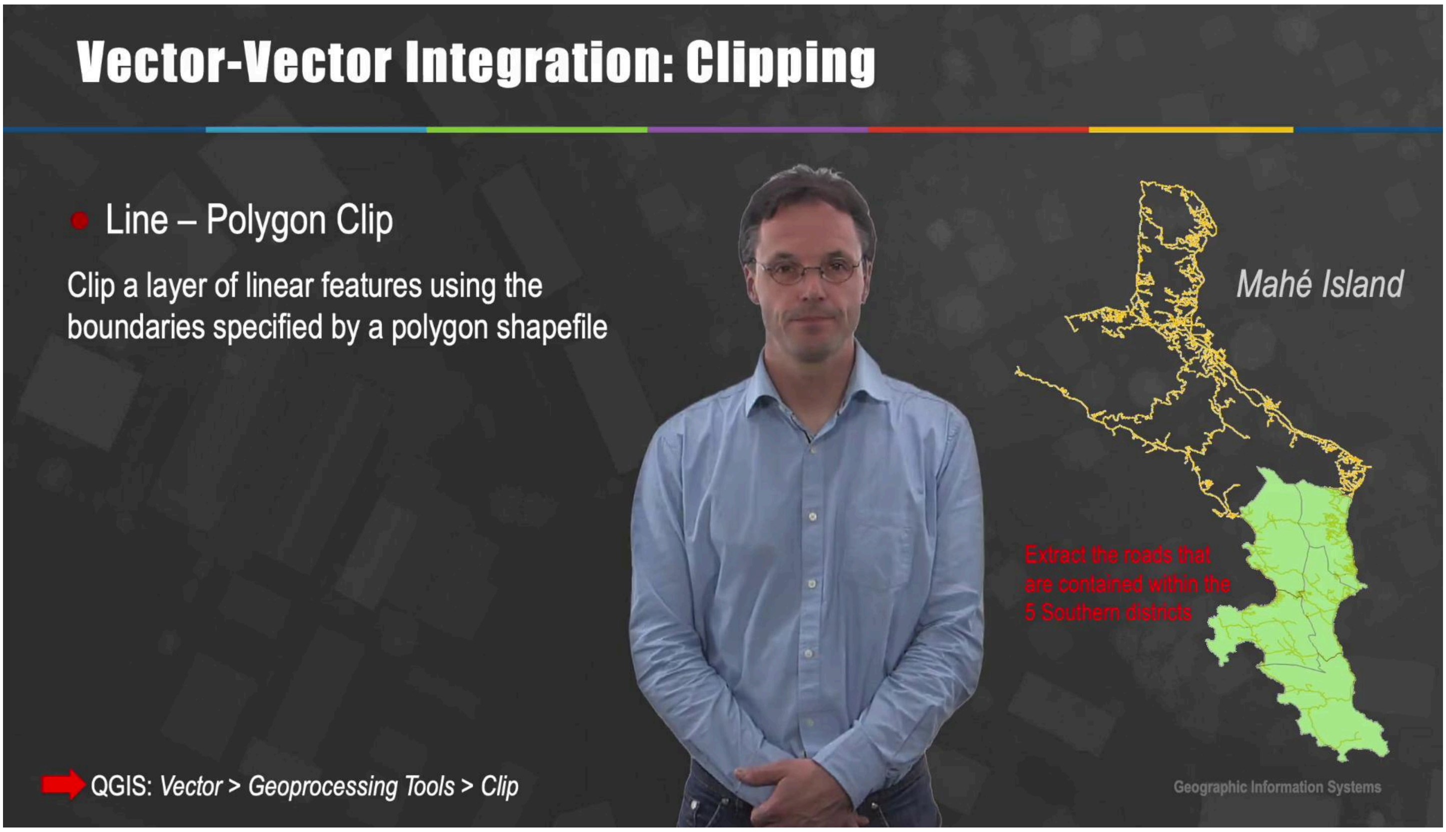


Let's see then how to proceed to calculate this sum of segments lengths by district. In the QGIS project we have the polygon file containing the districts of Mahe and a line file with the roads of the island. We will calculate the total length of road sections located in each district using the "vector" tool, "analysis tools", "total line lengths". Under "polygon vector layer as input", we choose the district layer and under "line vector layer as input", we choose the roads. We can then name the attribute that will contain the result. Here we will call it lroutes. Then we specify the name and location of the result file. We can then click on OK. The calculation will take some time because it is a relatively complex operation. The result is then displayed on the screen and we find the districts of Mahe and if we open the attribute table, we see that a column lroutes has been added, indicating the total length of road sections located in each district.

Notes

Summary

8m 18s

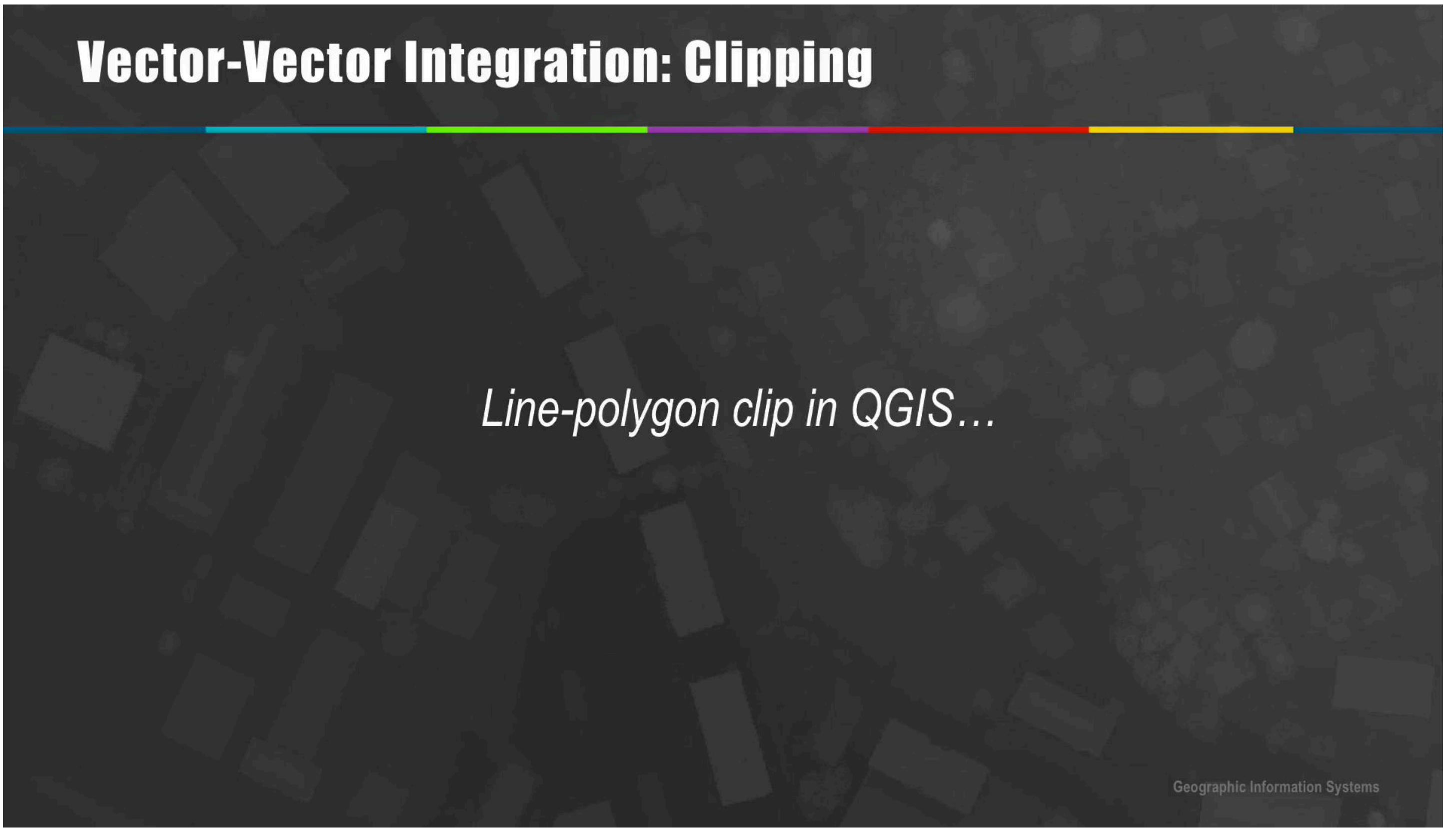


We will now present the different cutting operations of a vectorial layer by another. It is indeed possible to cut polygons with other polygons, or to cut lines with polygons. In the third case, we will speak more of extraction of points, segments or surfaces located inside of a polygon. The cutting as implemented in the "cut" function in QGIS, generates a result which contains only the attributes of the cut layer. The first example shown here is the case of lines cut by polygons. Still in Mahe whose road network is here displayed in yellow, we want to extract the roads located in the top five districts in the South of the island and represented in green.

Notes

Summary

9m 32s

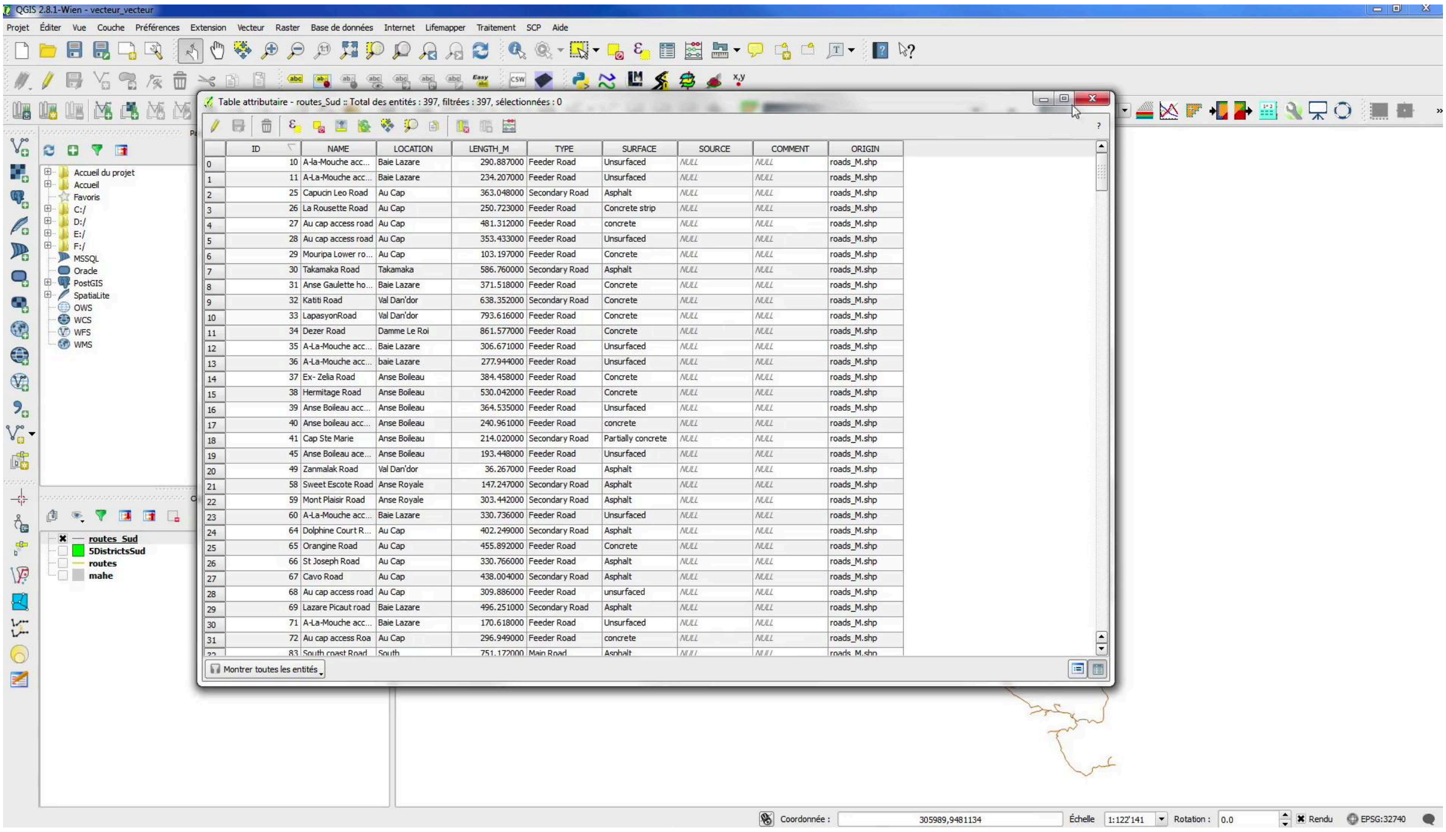


Here is how to do it in QGIS.

Notes

Summary

10m 18s

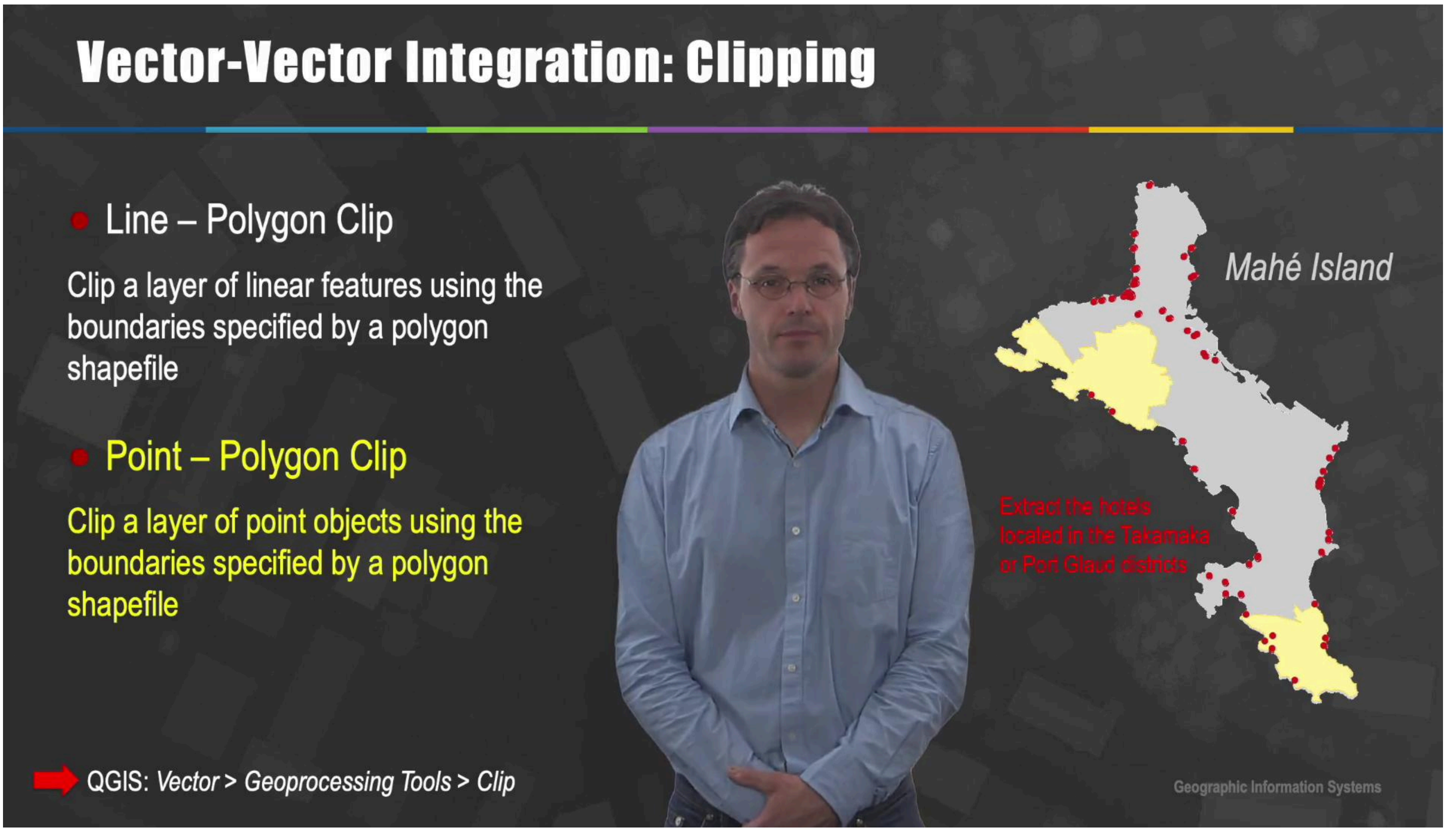


In the QGIS project we have the line layer containing the Mahe roads and a polygon file containing the most southerly five districts. We will cut the line file to retain only the road sections in the five districts. We use the cutting tool again under "vector" "geoprocessing tools" and "cut". As an input vectorial layer, we choose the roads and as a cutting layer, the polygon file containing the five districts. We must then specify the result file, then click on OK. The calculation will take some time and the result is displayed on the screen. It is indeed a line file containing only the road sections located in the five districts. In the attribute table, we see that all the attributes of the initial "road" file were recovered.

Notes

Summary

10m 21s

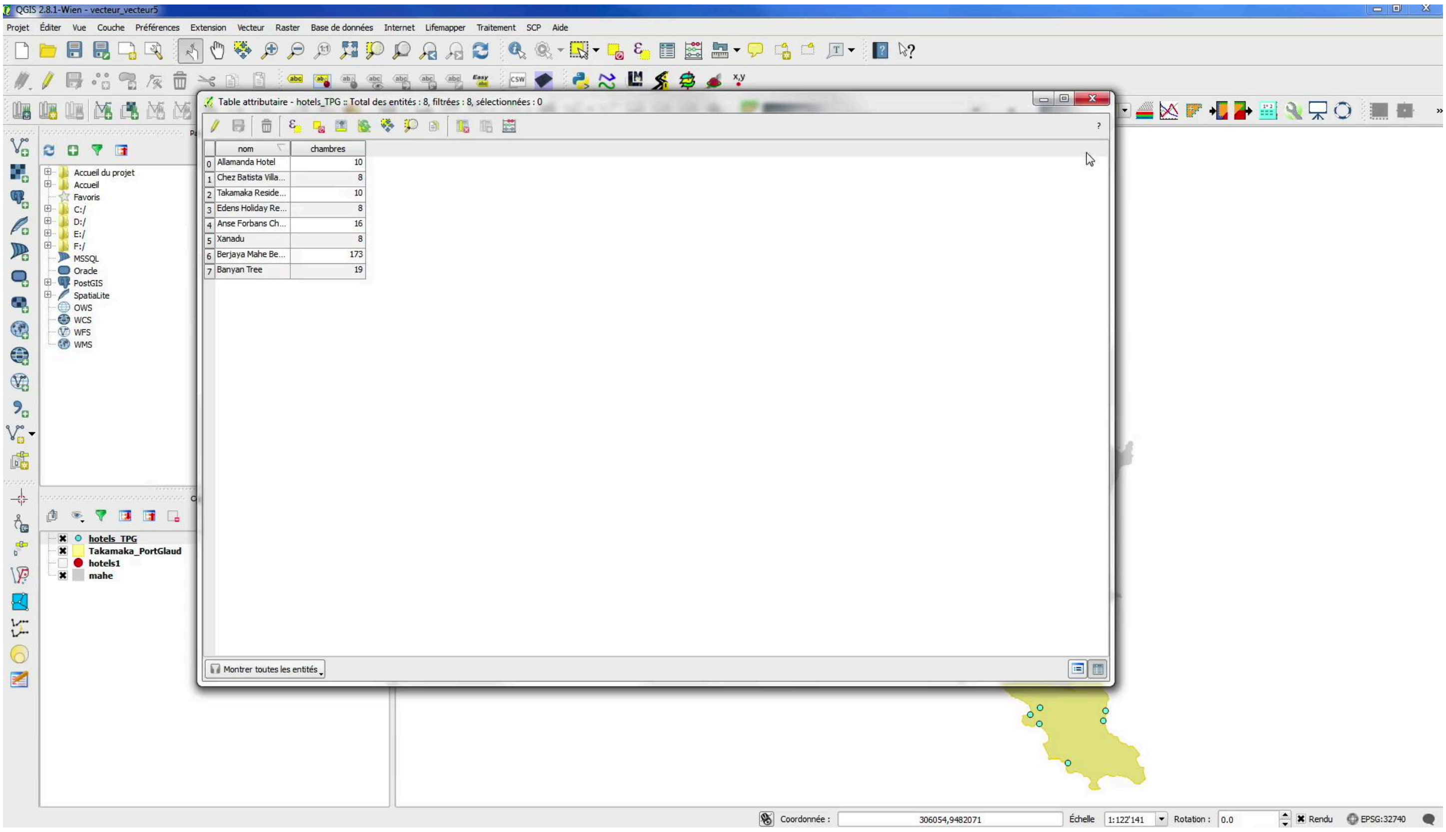


The case that we present now is common. It is the extraction of points located in polygons. On the island of Mahe, we want to extract the hotels, here represented in red, of the districts of Takamaka and Port Glaud represented here in yellow.

Notes

Summary

11m 19s

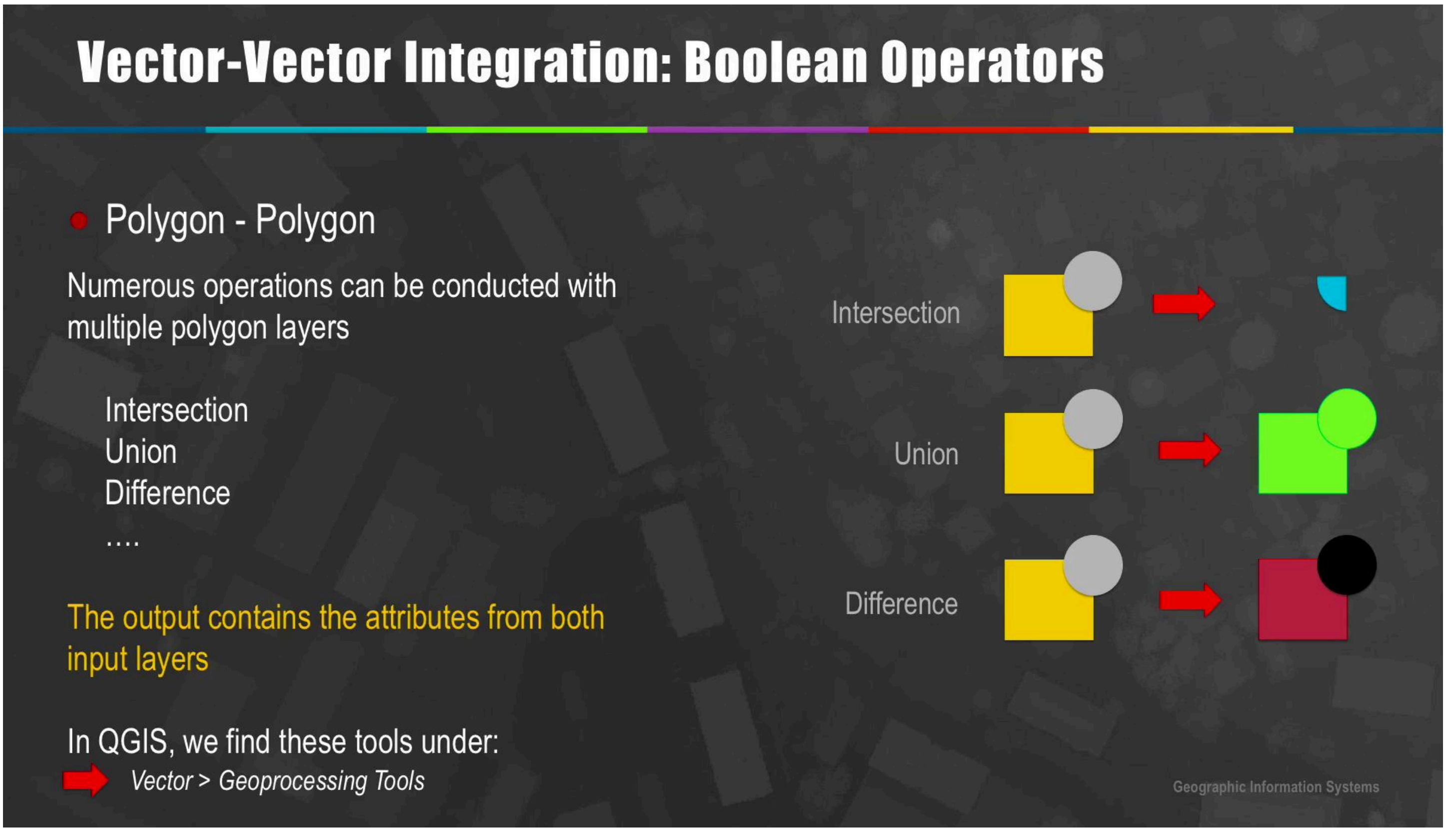


Let's see how it goes in QGIS. In the QGIS project we have the points file of the hotels with the name of the hotel and the number of rooms as attributes, as well as a polygon file containing two districts that of Takamaka in the South and Port Glaud further North. We will extract the hotels located in these two districts. For this we use the cutting tool which is found under "vector" "geoprocessing tools", "cut". As the vectorial layer we choose the layer to be cut, that is to say the point layer of the hotels. As the cutting layer, we choose the layer containing the polygons of the two districts. We then specify the name and location of the result file and we can click on OK. The result is displayed on the screen. This is a point file containing all the hotels located in the districts of Takamaka Port Glaud. In the attribute layer We find exactly the same attributes as for the initial layer of hotels.

Notes

Summary

11m 36s

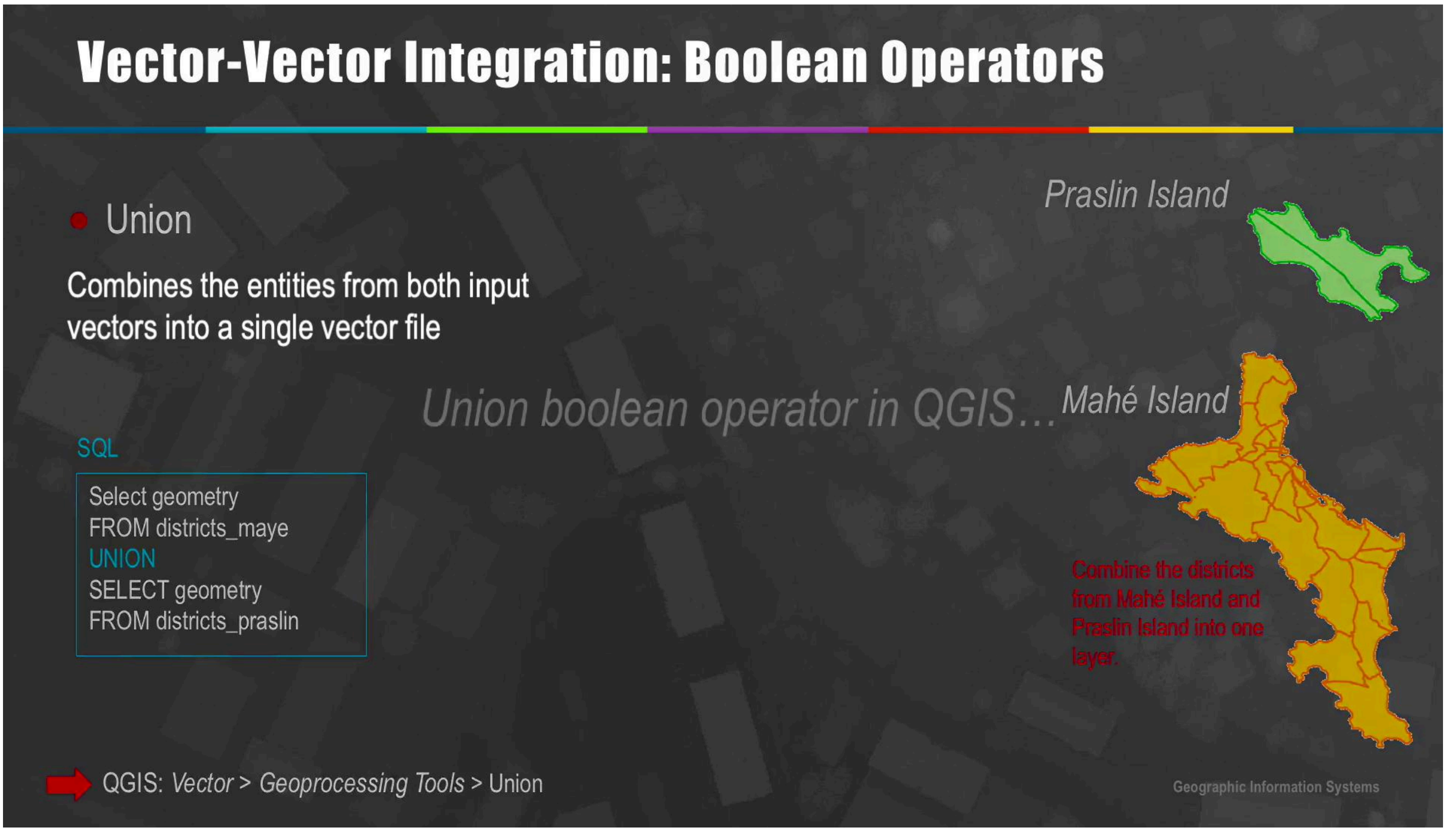


We will now look at another type of interaction between different vectorial layers. It is the boolean operations on the polygons. The boolean algebra or boolean calculation, is an algebraic approach of the logic expressed in terms of variables, operators and functions on the logic variables. This allows to use algebraic techniques to treat the expressions with two values of the calculation of proposals. These terms are very often used in the context of the multi-criteria analysis to assess the ability of certain territories to perform a specific function. In the case of two layers of superimposed polygons, we can be confronted to the following situations: the intersection between yellow polygon and the gray polygon produces the blue polygon. The union of the yellow polygon with the gray polygon gives green polygons. And the difference between the yellow polygon and the gray polygon gives the red polygon. The result of these boolean operations contains the attributes of the two layers and it is always in the "vector" menu under "geoprocessing tools", that we will find the QGIS functions capable of performing this type of processing.

Notes

Summary

12m 56s

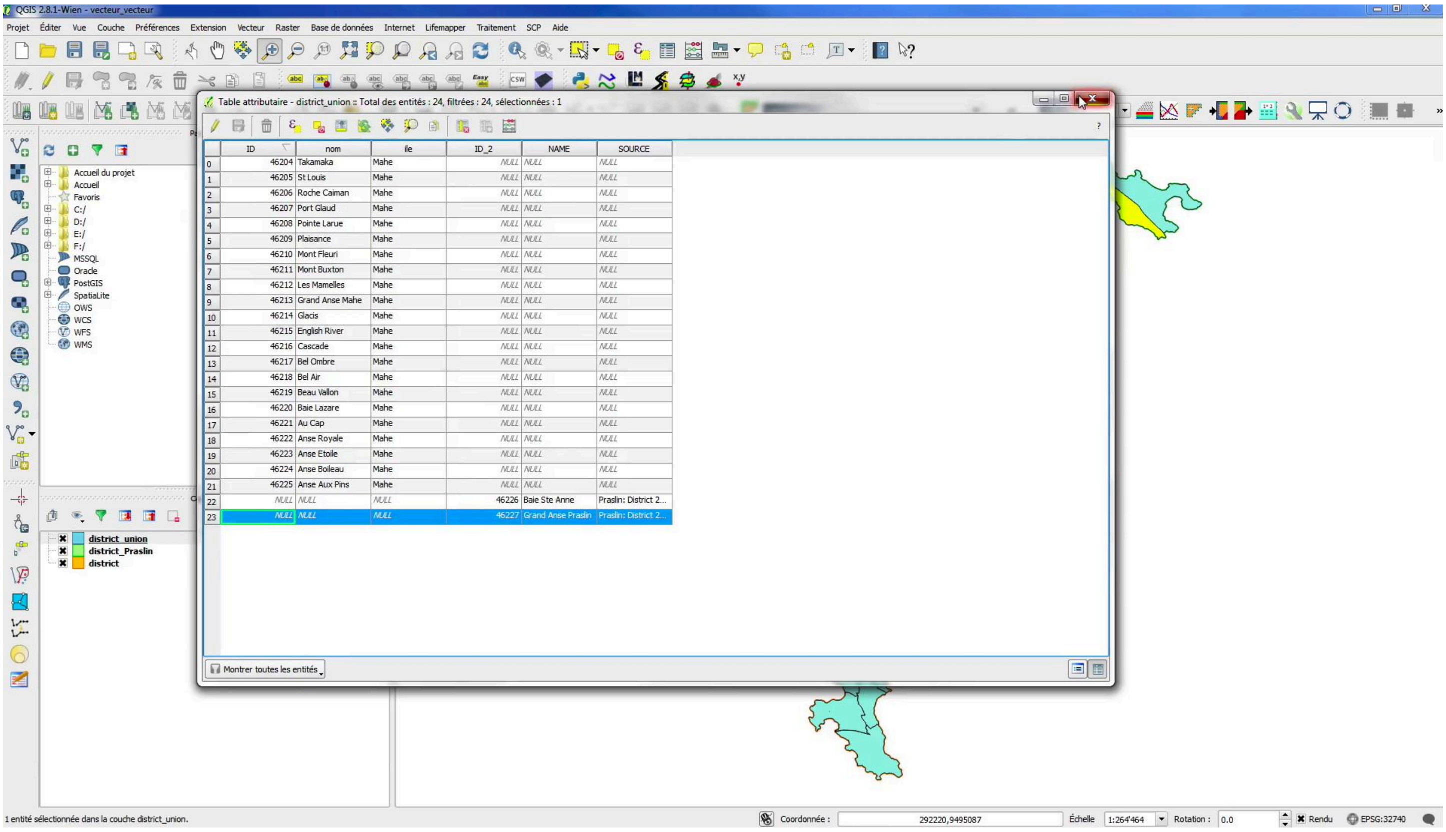


We show you an example to illustrate the "union" boolean operation. The aim here is to group in a single information layer the districts located on the island of Mahé and those located on the island of Praslin originally stored on two different files. This is to bring together the entities of both vectorial files in a single one. The corresponding SQL query calls for the "union" operator and in QGIS it is the "union" geoprocessing tool that should be used.

Notes

Summary

14m 08s

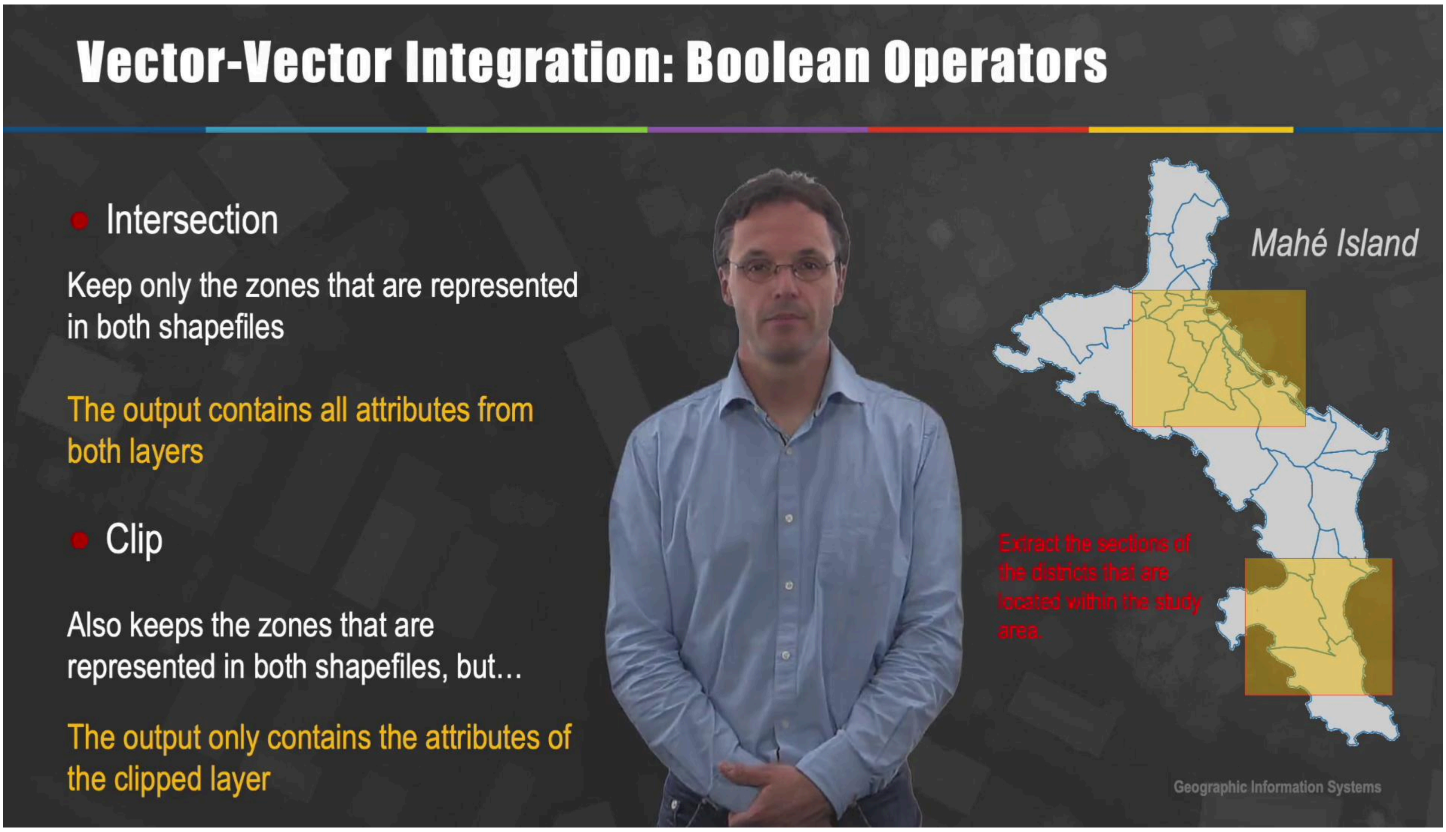


Here iss how it works: in the QGIS project, we have the districts of Mahe, with in the attributes table the identifier and the name of each district as well as the name of the island. We also have another polygon file containing the two districts of Praslin with for each district its name and a source attribute. We will merge these two files to obtain a single layer of polygons containing all the districts. For that, we use the "union" tool under "vector", "geoprocessing tools", "union". Under "input vectorial layer", we choose the "districts" layer that contains the districts of Mahe. Under "union layer", we select the districts of Praslin. We can then define the name and the directory where the result is saved and click on OK. We then have to specify the projection system of the result, in our case the UTM Zone 40 South system. And the result is displayed on the screen. It is indeed a polygon layer containing all the districts of Mahe and as well as those of Praslin. By opening the attribute table, we see that we find all the attributes of the districts of Mahe but also the attributes of the districts of Praslin. As these attributes are different, the Mahe districts have null values for the Praslin attributes and vice versa. The union therefore retrieves the attributes of both files.

Notes

Summary

14m 36s

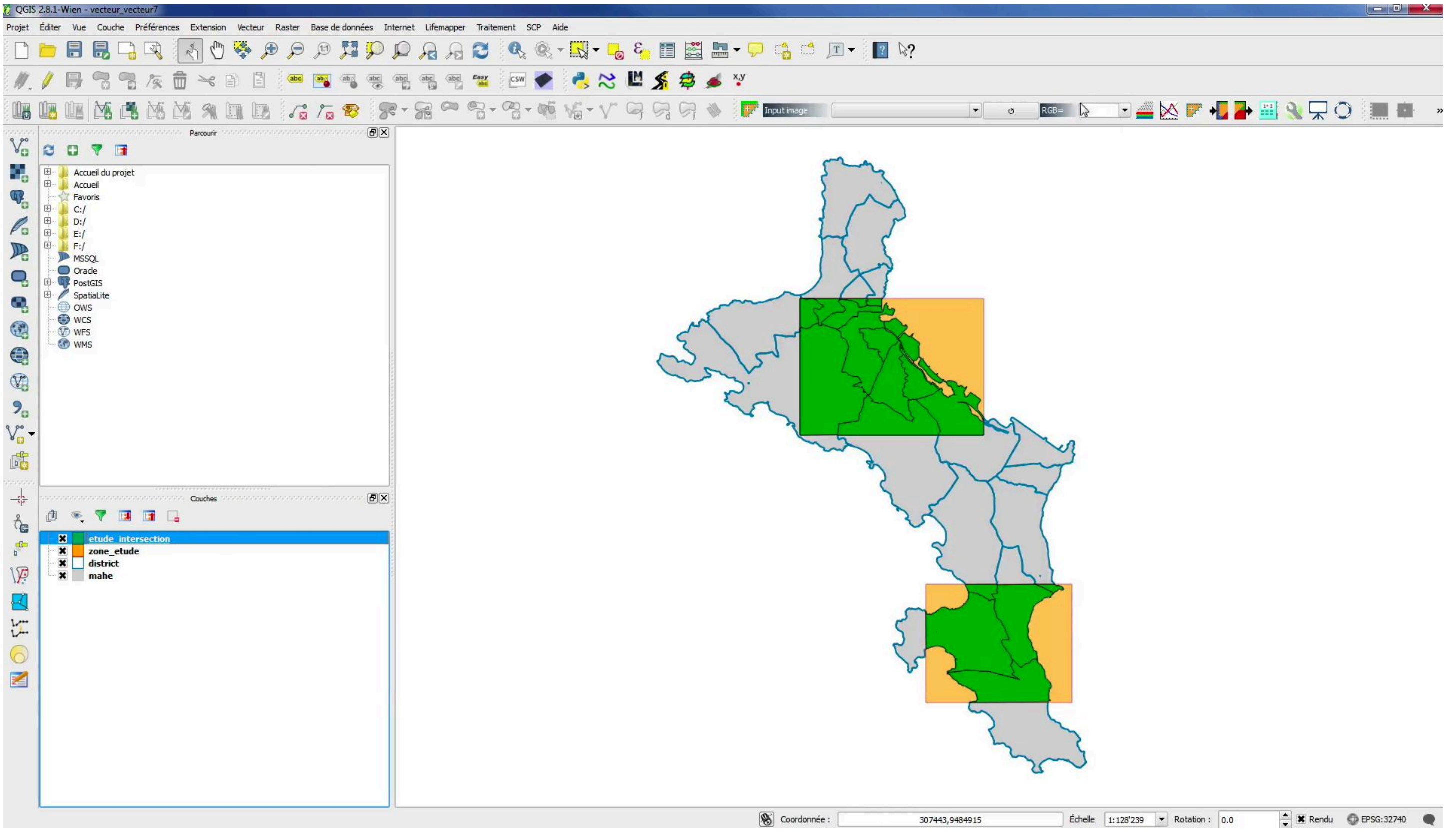


To illustrate the implementation of the "intersection" boolean operator we will extract the portions of the territory of the districts of Mahe located in the two study areas represented here by the two orange polygons. This means maintaining only the common areas between the two vector files in orange on the map. This operation provides a result which contains the attributes of the two layers. In the same configuration, we can also use the "cut" function which will also help to keep the common areas between the two vector files, but the result in this case contains only attributes of the cut layer.

Notes

Summary

16m 00s

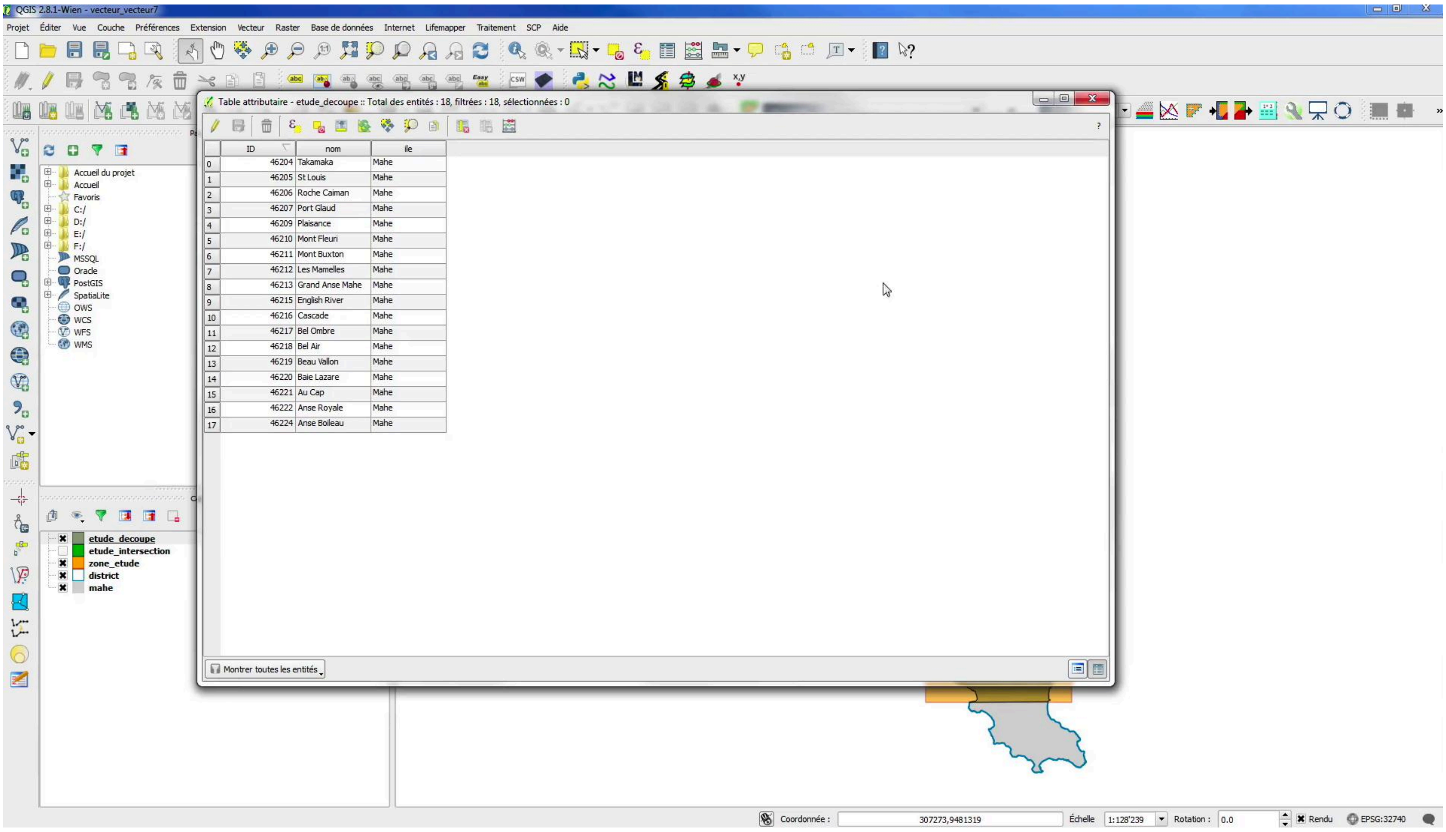


Here is how to proceed in QGIS: in the QGIS project we have the districts of Mahe which have as attributes an identifier, a name and the name of the island as well as a "study zone" layer which indicates the location of the two zones that we are interested in. This layer contains four attributes: an identifier, a name, a place and a year of study. We will first extract the portions of districts located in the two study areas using the "intersection" function in the "vector" menu "geoprocessing tools", "intersection". We choose the layer of districts as an input layer and the study area layer as the intersection layer. We specify the name and location of the result, then we can click on OK. Then we specify the projection system of the result, UTM Zone 40 South. The result is displayed on the screen and the created layer contains the districts portions located inside the two study areas. If we open the attribute table, we see that the result contains the attributes of the two original layers, both those of the district layer ID, name, island, but also those of the "study area" layer ID_ze, nom_ze, place, year. Like the union, the intersection keeps the attributes of the two combined layers.

Notes

Summary

16m 38s



The same polygons could have been obtained with the cutting function which was previously used, the difference is that the cutting function only keeps the attributes of the first layer. The cutting layer is used only to define the edges of the new polygons but its attributes are not taken into account. To convince you, we will apply the "cut" function to the same example. Under "vector" "geoprocessing tools", "cut", the layer of districts is selected as the input layer and the study area layer as a cutting layer. we save the result and click on OK. We have to specify the projection system then the layer appears on the screen. We get exactly the same polygons only with the intersection. However in the attribute table we do not have any attributes of the "study area" layer, only the attributes of the "districts" layer have been kept.

Notes

Summary

18m 04s