

LearnOSM

Relations

Reviewed 2015-09-20

In the [Beginner's Guide](#) we explained that there are three types of objects that can be drawn in OpenStreetMap - points (nodes), lines (ways), and polygons (closed ways). Lines contain numerous points, and the line itself carries the attributes that define what it represents. Polygons are the same as lines, except that the line must finish where it begins in order to form a shape.

In fact, we lied to you, because there is one other type of object in OpenStreetMap, called relations. In the same way that a line consists of other points, a relation contains a group of other objects, be they points, lines, or polygons. If you are looking to advance your editing skills, then understanding and knowing how to properly edit relations is important.

For example, imagine that you want to map a building that has courtyards in the center. You would need to draw a polygon around the outside of the building, and you would want other polygons around the courtyards to indicate that they are not part of the building. This is an example of a relation. The relation would contain several polygons - and the attributes of the building would be attached to the relation, not the polygons.



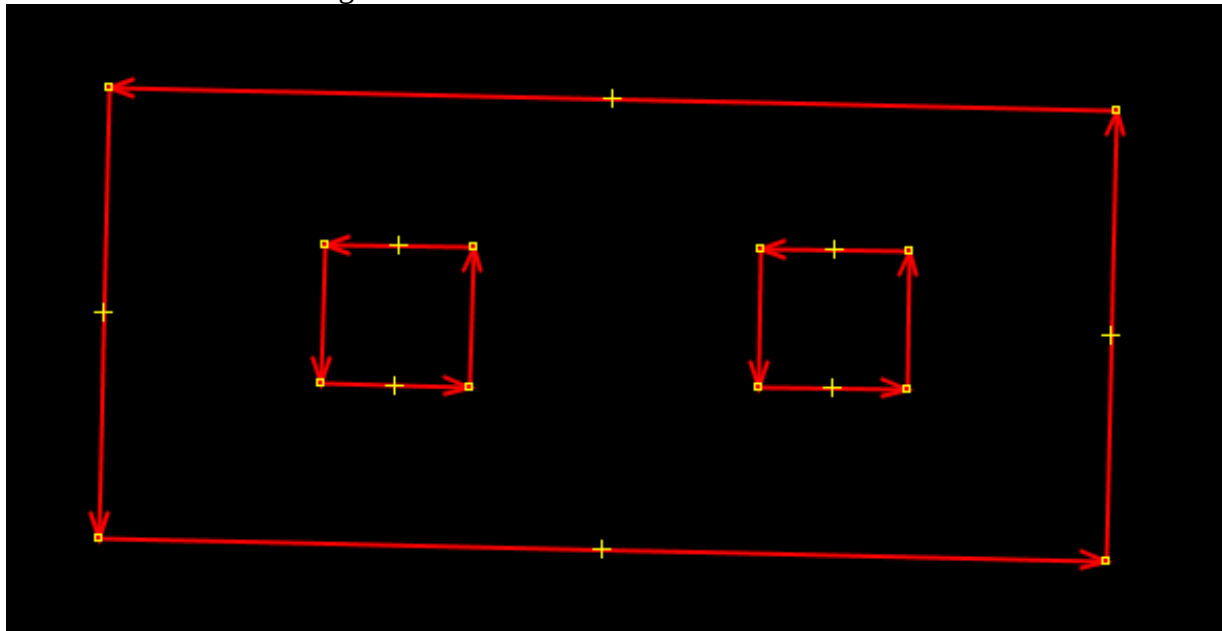
Relations are used to represent anything that requires a collection of objects to define. Other examples are bus routes (a collections of lines), long and complex objects (rivers or roads), or multiple polygons that are all part of one location (like buildings in a university).

There are mainly four types of relations you will encounter in OSM: Multipolygons, Routes, Boundaries, and Restrictions (such as, no left turns). In this section we will cover Multipolygons and Routes.

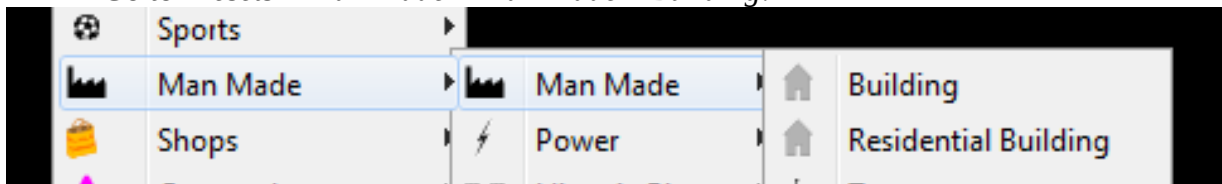
Creating Multipolygon Relations

Let's see how to create a multipolygon relation like the one shown above.

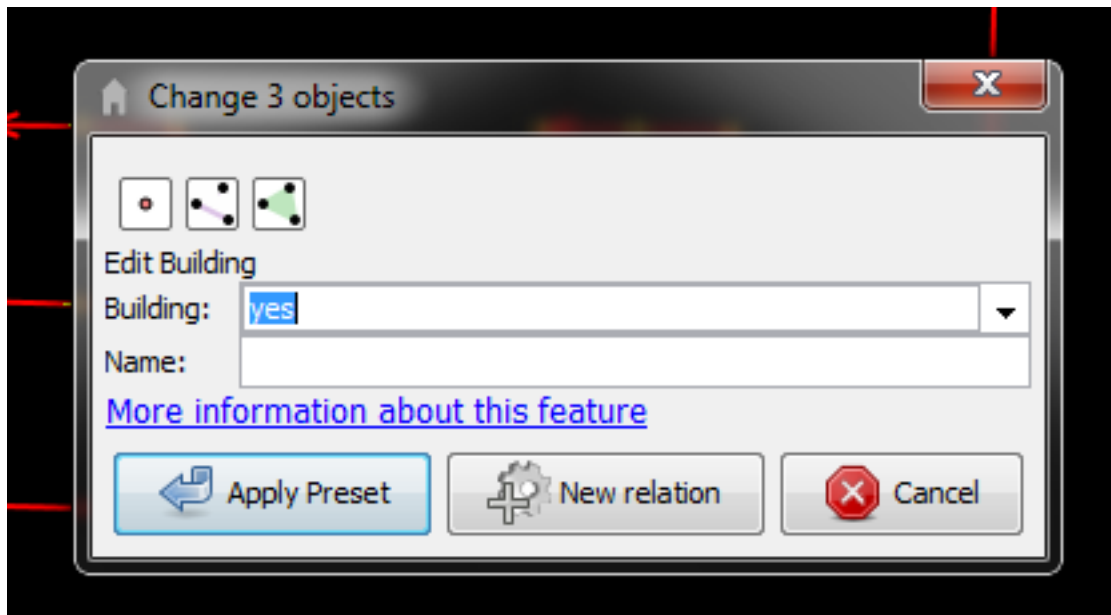
- First, draw your shapes. In this case we will draw three polygons, an outer rectangle, and two smaller rectangles.



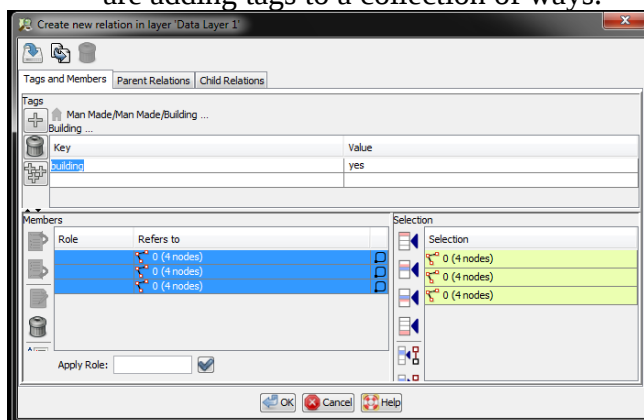
- Select all of the polygons. Remember you can select multiple objects by holding SHIFT and clicking on each.
- Go to *Presets->Man Made->Man Made->Building*.



- Click on "New Relation."



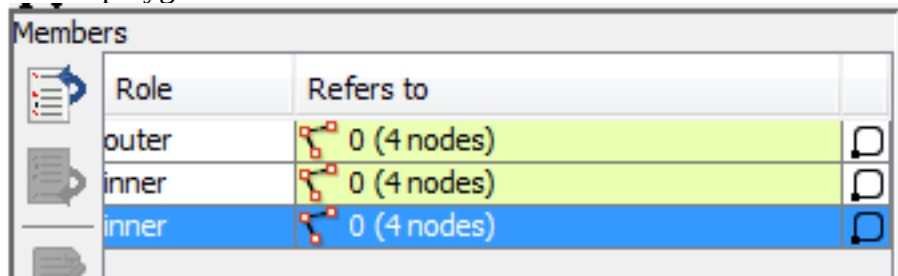
- You should now see the relations window. This is a little bit complex because now you are adding tags to a collection of ways.



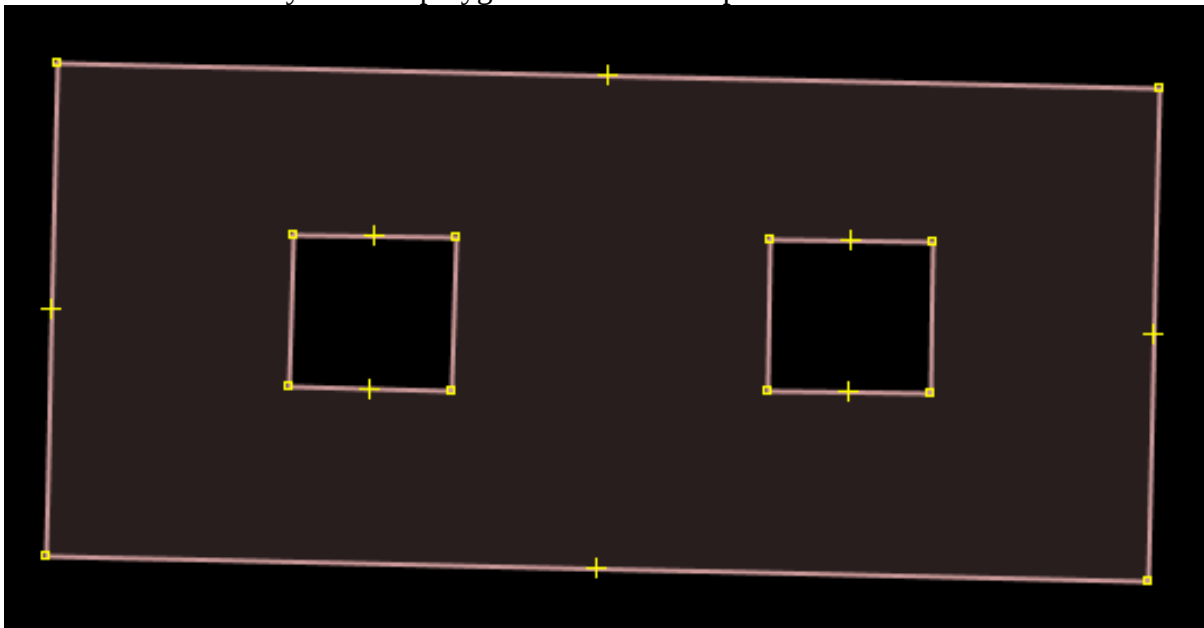
- Observe that at the top of the panel are the tags for the relation. These tags work the same way as tags always work.
- At the bottom is a list of the members of the relation. This relation has three members - that is, the three ways that are part of our relation.
- We need to do a couple of things to finish defining our multipolygon. First, notice that because we used the building preset we already have one tag defined for us, *building=yes*. We need to add one more tag that defines the **type** of the relation. We must add a tag that says *type=multipolygon*.
- Click in the tag box and add this tag.

Key	Value
building	yes
type	multipolygon

- Next we need to define what are called **roles**. Each member of a relation has a role, which indicates what that member's purpose is. In this case, the role of the outside polygon must be defined as **outer** and the role of the two inner polygons must be defined as **inner**. These are the roles that are available for members of a multipolygon.
- In the lower left panel select each member in the list. You can see the member that is selected will be highlighted in the map window. Enter **outer** and **inner** for the correct polygons.



- Click OK and your multipolygon relation is complete!



When you create a multipolygon like this it will be rendered on the map like so:



Multipolygons can be used for any complex object that requires inner and outer polygons, like a building or a river with patches of land inside it. Detailed multipolygon information can be found

on the [OSM Wiki](#).

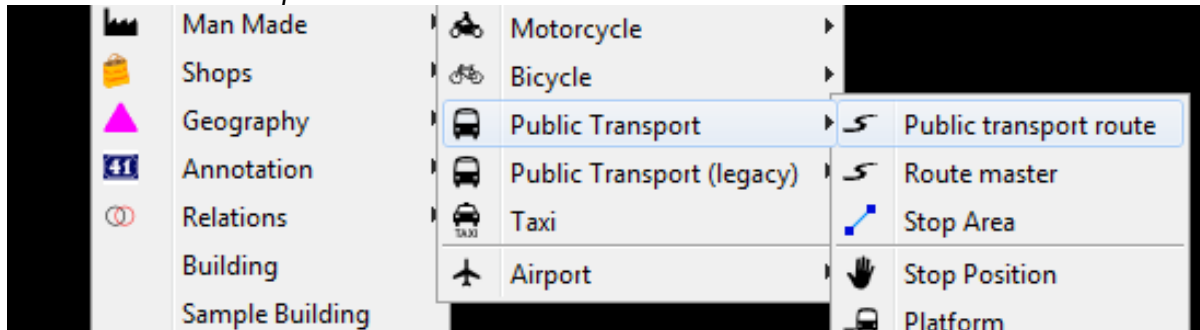
Route Relations

Relations are also very useful for representing long routes. For example, a bus or bicycle route follows various road segments. To define such a route, we can create a relation which contains all of the road segments that are part of the route. Additional features, such as bus stops can also be members of a route relation.

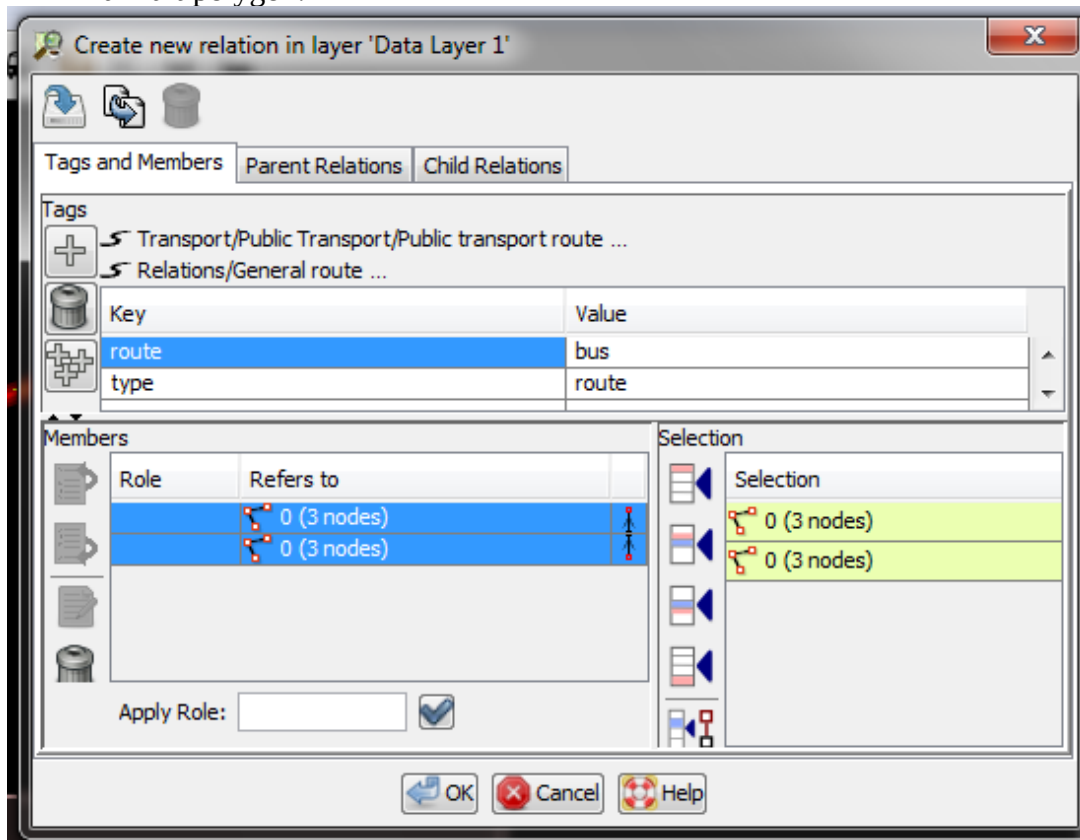


Let's see how to create a relation for a bus route in JOSM:

- First, select all of the ways which the route runs along. You may need to split some streets into separate segments if only part of them belong to the relation. You can do this using the “Split Way” tool.
- When all the segments are selected, go to *Presets->Transport->Public Transport->Public transport route*.



- Next to Route type, select “bus.”
- Click “New Relation.” You will see the relation window pop up, just as when you create a multipolygon.




- Notice that there are already tags which define the relation as a route. Instead of *type=multipolygon*, we have *type=route*. We also have a tag defining it as a bus route, as opposed to another type of route.
- You may now think, what should the **role** of the members be? Well, in the case of a route, we don't need to define the role of the members. By leaving the role blank the software

knows that they are simply part of the route. We could also define the role of each segment as **route**, but it is not necessary.

- Click OK and your route relation will be complete!

If you want a list of all the relations on the map, you can open the relations panel by

clicking on this button on the left: . This will open a panel where you can select, edit, and create new relations.

Summary

Relations can be difficult to understand and do not need to be used often, but they are necessary to know about. Every so often you may realize that you need a relation to map something correctly, and you can use this knowledge, and find more information on the OSM Wiki, to help you map relations correctly.

Was this chapter helpful? [Let us know and help us improve the guides!](#)

- learnosm@hotosm.org
- [@learnOSM](#)
- [Hosted on Github](#)



Official [HOT OSM](#) learning materials



Humanitarian
OpenStreetMap
Team