Evaluating Methods for Downloading OpenStreetMap Data

J. C. Ehrig-Page
University of Nebraska Omaha
jehrigpage@unomaha.edu

INTRODUCTION

OFTEN, ONE OF THE greatest limitations in creating maps or performing spatial analysis is acquiring data. OpenStreetMap (OSM) is useful resource which offers a wide variety of easily accessible datasets under an Open Database License (ODbL), including those that can be hard to find for some areas, such as parks, building footprints, and street centerlines. There are several ways to access and download OSM data, but many of these methods can be difficult to use due to their limited functionality or complexity. Additionally, they may require users to process the data, or purge unnecessary portions. In this piece, I explore multiple methods for downloading OSM data and evaluate them alongside the QuickOSM plugin for QGIS.

OPEN SOURCE DATA AND OSM

OSM, SUPPORTED BY THE OpenStreetMap Foundation (OSMF), is a community-driven collection of global geospatial data portals that is maintained by a diverse range of enthusiasts and professional geospatial users (openstreet-map.org/about). One of the main advantages of OSM data is that it is open source, meaning it is free for anyone to use, access, modify, and share, as long as they preserve the origin attribution and openness. This enables its contributors to build large, complex datasets that would be impossible to create by small groups or with limited funds.

While open source data has many advantages, it is not without its limitations. The data quality is inconsistent when compared to proprietary data. Some geographic locations might have more accurate and detailed information due to more people adding and updating data, while other areas may have little to no data at all. Additionally, the lack of standards or methods for data entry, along with contributions from users with limited skills, sometimes makes open source data less accurate when compared to proprietary data. For example, Figure 1 shows some of the limitations of OSM data: each marker represents unresolved notes or known errors in the data. Nonetheless, open source data can be a viable source of spatial information that can be beneficial even with its limitations, especially in instances where it is more up-to-date than or has information that is not present in other sources.

Accessing, editing, and downloading OSM data can be achieved in several ways through different Application Programming Interfaces (APIs). Editing data is handled by the main OSM API, through which users can add and modify data. In contrast, the Overpass API is a read-only option, and is optimized for consuming selected portions of the data. Clients are able to send a query to the Overpass API with parameters such as location, object type, or other desired properties, and receive a response
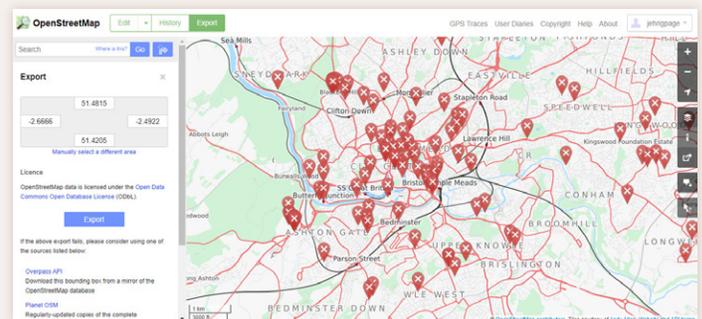
Figure 1. *OpenStreetMap web portal. Red X markers represent unresolved notes and errors, demonstrating the potential issues with OSM data.*

containing information for the features that satisfy the query. Though OSM's APIs allow interaction with the data, using them successfully can require extensive knowledge in programming and coding, making them difficult to use for many users.

Alternatively, users may use the OSM website to export data in the OSM file format, which can be opened in some GIS software, such as QGIS. While its use does not require programming skills, the tool is basic and limited. First, there are no options to filter the data by layer type (e.g., roads, natural features, etc.) and the location selection tool is limited. This means that users will have to remove unwanted data after it has been downloaded. Furthermore, the tool is limited to downloading datasets that include fewer than 50,000 nodes. The ability to download more nodes is essential where a user needs to acquire significant amounts of data for large regions or highly dense urban areas where nodes would commonly exceed this limit.

The OSM web portal suggests using one of many third-party options to access OSM data, such as Planet

Sub Region	Quick Links		
	.osm.pbf	.shp.zip	.osm.bz2
Africa	[.osm.pbf] (3.9 GB)	✕	[.osm.bz2]
Antarctica	[.osm.pbf] (29.0 MB)	[.shp.zip]	[.osm.bz2]
Asia	[.osm.pbf] (8.3 GB)	✕	[.osm.bz2]
Australia and Oceania	[.osm.pbf] (790 MB)	✕	[.osm.bz2]
Central America	[.osm.pbf] (383 MB)	✕	[.osm.bz2]
Europe	[.osm.pbf] (21.7 GB)	✕	[.osm.bz2]
North America	[.osm.pbf] (9.4 GB)	✕	[.osm.bz2]
South America	[.osm.pbf] (2.1 GB)	✕	[.osm.bz2]

Figure 2. *The Geofabrik portal's list of OSM data by "Sub Region," showing file sizes and data formats.*

OSM or Geofabrik. Due to their large sizes, the datasets will typically be grouped into regions. For example, Geofabrik offers datasets broken down by continents, countries, and sub-national regions (see Figure 2). The whole of Europe is currently around 21.7 GB, while Great Britain is about 1 GB of data. Geofabrik does not offer data at the city level, thus country or sub-national data has to be downloaded, with users then needing to extract the portion in which they are interested or manually delete the excess. These data portals are viable options in obtaining data that users may need, but they do not provide extensive tools to filter data and select the relevant portions.

QGIS

QGIS IS A WIDELY USED program that supports OSM data and, like OSM, it is free and open source (qgis.org/en/site/about). The software is a volunteer-driven project maintained by a community of dedicated volunteers and organizations who develop its code and maintain documentation. The software allows users to create, edit, visualize, and analyze data from several vector, raster, and database file formats—including OSM files—and can be used on a variety of platforms including Windows, Mac, Linux, and BSD.

QGIS can also be enhanced with a series of plugins that perform a variety of tasks that are not part of the core QGIS software. These plugins have been developed to streamline workflows or perform a specific type of analysis. The QGIS plugin manager (Figure 3) accesses a collection of ready-to-use plugins that can be directly installed. As of November 2019, there are several QGIS plugins that utilize OSM data, but only two (QuickOSM and OSMDownloader) are designed to download the data. Other plugins, such as Pelicas Geocoding and ORS Tools, may query OSM data to pull additional geospatial information or provide geocoding services.

**Figure 3.** *QGIS Plugin Manager showing OSM plugins.*

OSMDOWNLOADER

OSMDownloader is a plugin developed by Luiz Andrade (Github username: lcoandrade) that uses the Overpass API to download OpenStreetMap data. It features a simple user interface that is easier to use than the Overpass API itself. To use the plugin, the user draws a rectangular area on the QGIS map window using the selection tool. The plugin then queries the Overpass API and downloads all spatial data within the designated rectangle. In Figure

4, for example, I've specified an extent covering the area around Bristol, in the United Kingdom. The plugin provides an easier interface than either the OSM web portal or Geofabrik, and the custom extents keep file sizes manageable; the data for the Bristol area added up to only about 513 MB.

Although the OSMDownloader plugin provides access to OSM data, it combines all the data into a few layers that are separated only by geometry type (points, lines, and polygons), as shown in Figure 5. Datasets that might normally be separated into discrete layers, such as streets and rivers, are grouped together. Simliar to Geofabrik and Planet OSM, this requires additional data processing to separate out the feature types that are needed and removing those that are not. This additional process can be very time-consuming.

QUICKOSM

QuickOSM is a plugin that, like OSMDownloader, provides access to OSM data using the Overpass API. Developed by Étienne Trimaille (Github username: Gustry), QuickOSM provides a GUI to build Overpass API queries. Additionally, there is a tool for loading OSM or PBF files. PBF files, like OSM files, store spatial data and the format was created to support future extensibility and flexibility.

The QuickOSM tool has several options for adding data to QGIS. The **Quick query** (Figure 6) is a fast and easy tool to search for different datasets, allowing them to be filtered by the extent of whatever data are currently loaded into QGIS, or by querying based on location information

Figure 4. OSMDownloader user interface, showing the extent of the selected area.
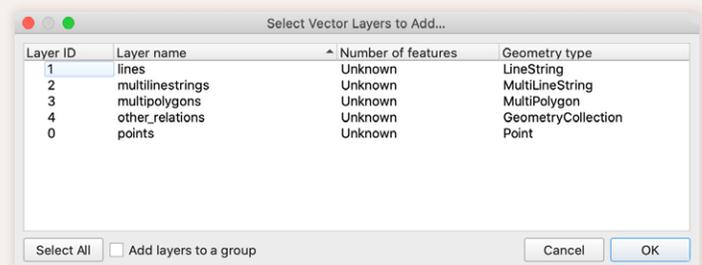
Figure 5. The file obtained through OSMDownloader contains layers differentiated by geometry type.

found in those data. The second option is to use the **Query** tool to build your own custom query scripts. When using the **Quick query** option, such scripts are created automatically for the **Query** tool in QuickOSM. The third option QuickOSM provides is the ability to open data in the OSM file format. While QGIS already supports the OSM file format, the QuickOSM plugin allows the user to filter the file by geometry type.
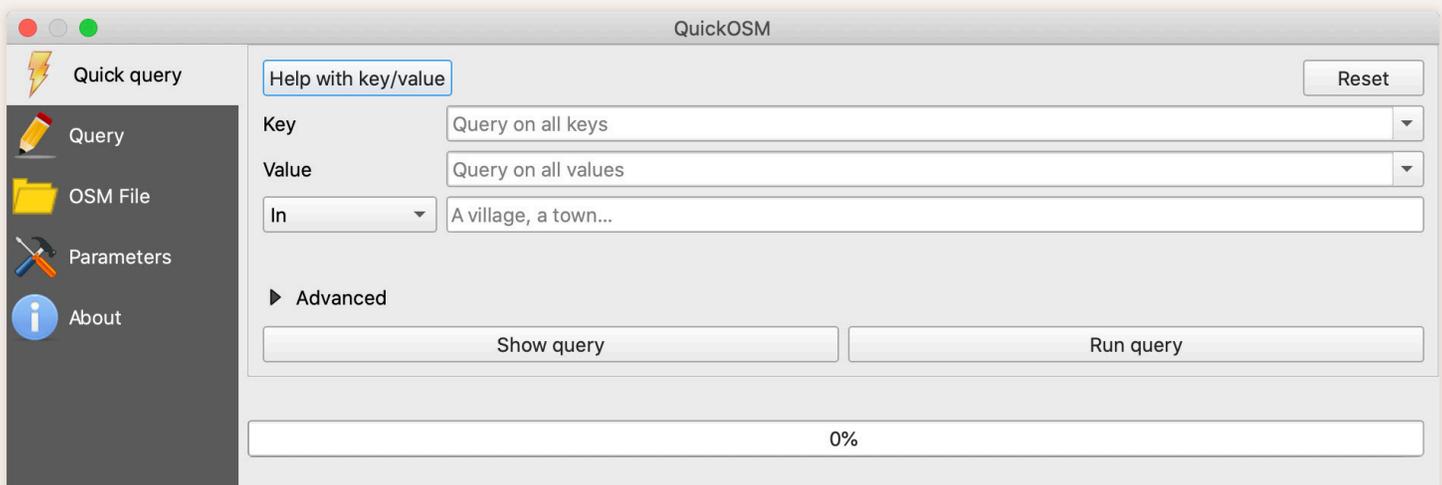
Figure 6. QuickOSM **Quick query** user interface, showing options to filter data.

Figure 7. *A map in QGIS, showing datasets before the addition of OSM data.*

QUICKOSM WALKTHROUGH

THE QUICKOSM PLUGIN USER INTERFACE allows users to query data and filter results using multiple inputs. This helps them quickly access only the data and information needed for a specific project. In order to further evaluate the QuickOSM tool, let's walk through a sample project to show the benefits of using the tool. In this project example, we begin by loading data showing parks and green spaces in the city of Bristol into QGIS, obtained through Bristol's open data portal (**opendata.bristol.gov.uk**). The resulting map (Figure 7) is somewhat sparse; road and building data would provide additional context to the map. Using the QuickOSM plugin, we will add this data.

QUICKOSM - QUICK QUERY

The plugin offers three main fields to filter data: **Key, Value,** and option to choose a spatial location. Under the
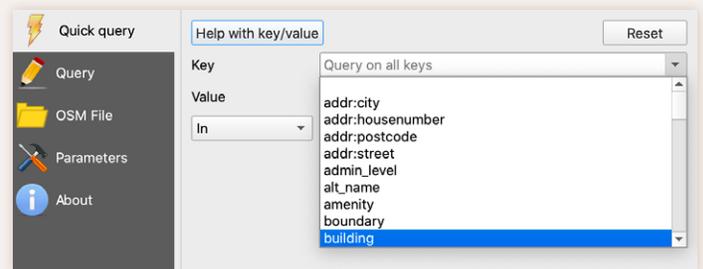
Figure 8. *Selecting building data from OSM.*

Key dropdown menu (Figure 8), there are options for the type of feature to select. We will choose **building** from the list. Just below **Key,** the **Value** field (Figure 9) allows the user to filter by specific types of attributes. This option is only available on certain feature types after the **Key** is selected. In this case, we can select **Query on all values** to capture all building types.
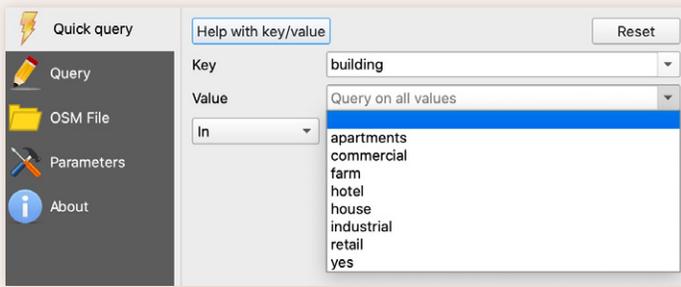
Figure 9. The **Value** dropdown allows users to specify building type.

Once we have described what sort of data we want, there are several methods for selecting data in a specific area. For this example, only buildings within Bristol are needed, and downloading a larger area would be unnecessary. The options for selecting a target area, using the dropdown menu below **Value**, are shown in Figure 10.

By clicking **Advanced**, a series of options appear (Figure 11) that allow greater customization of what type of data is returned, as well as letting the user specify how long to wait if the server cannot process the request, and a permanent file location in which to save the data.

QUICKOSM - QUERY

Once a query is created through the **Quick query** window, we can switch to the **Query** window (Figure 12) to view the script that will be sent to the Overpass API. Advanced users can edit the script, fine-tuning the output.
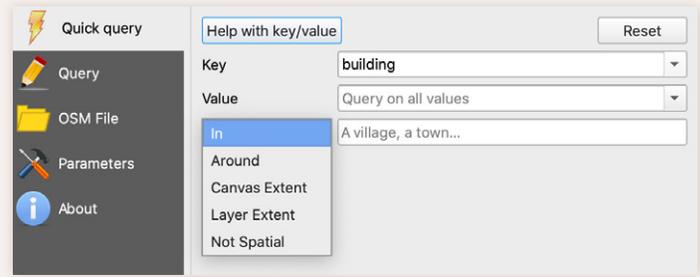
In: Select data within a specified location. This example will select buildings within Bristol.
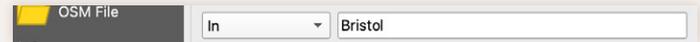
Around: Select data within a specified buffer distance of a specified location. This example will select buildings in Bristol and within 1000m of Bristol
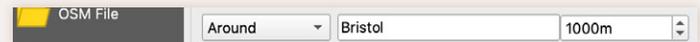
Canvas Extent: Select data within the current map canvas (map view).This example will select only buildings visible in the map view.
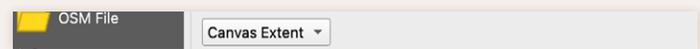
Layer Extent: Select data within the extent of a currently loaded dataset. This example will select buildings within our public spaces and not outside of them.

Not Spatial: Select data without any boundary limit. This example will select all buildings in OSM, worldwide.
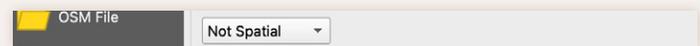
Figure 10. Options for selecting by location.
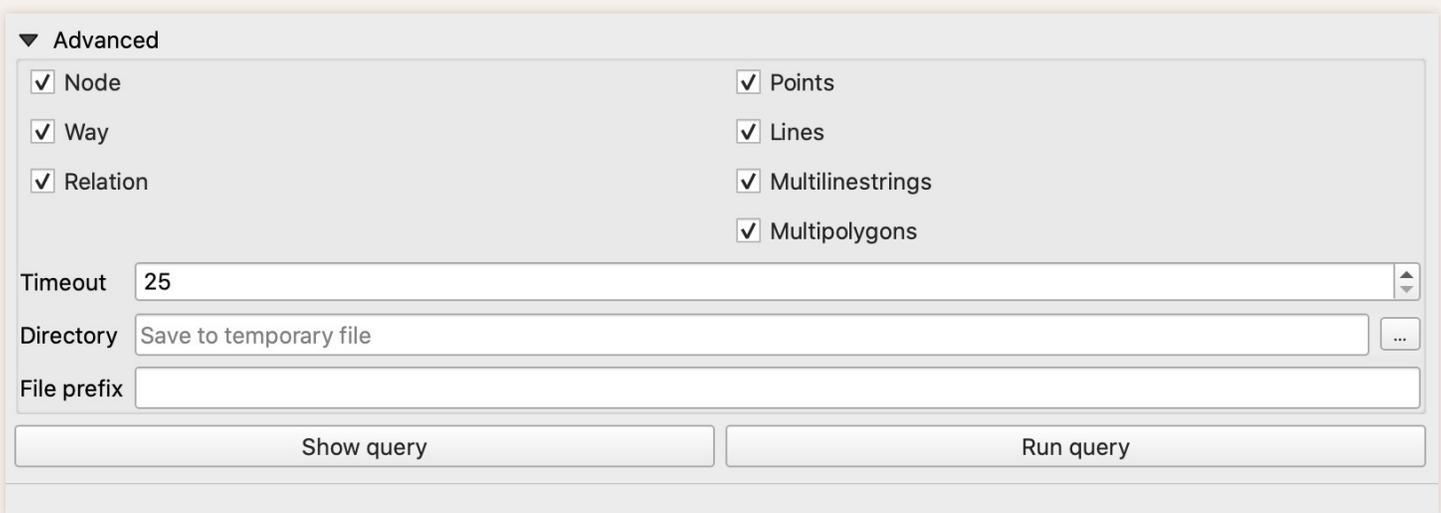
Figure 11. Advanced settings let us select data types, specify timeout settings, and choose where to save data.

Combined with the **Quick query** tool, users can quickly use the **Query** tool to create customized queries.

The advanced options of the **Query** window (Figure 13) are similar to **Quick query**, allowing for additional conditions on what data is returned. Using them, we can control the type of data returned and specify where it is saved, and we can additionally limit the geographical extent of the data.

EXECUTING QUICKOSM QUICK QUERY AND QUERY TOOLS

Running the query will execute the request to an Overpass database. If a file location is designated in the **Directory**, the file will be saved to the disk. Otherwise, if we leave it blank, the file will be stored in a temporary file location
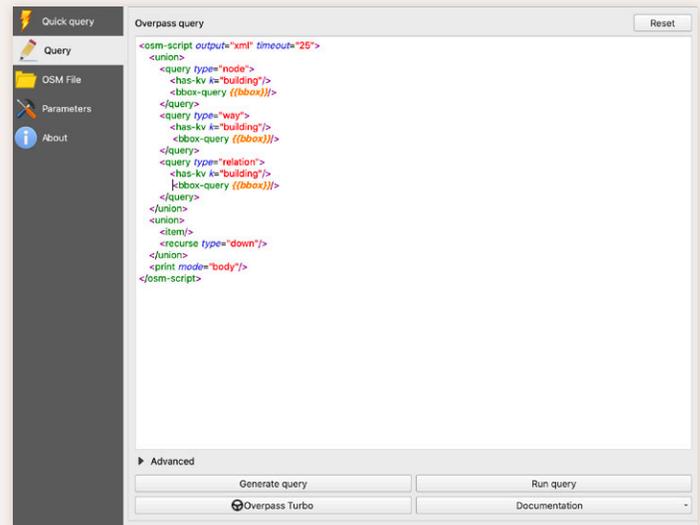
Figure 12. The Query text editor allows the user to further customize the query.
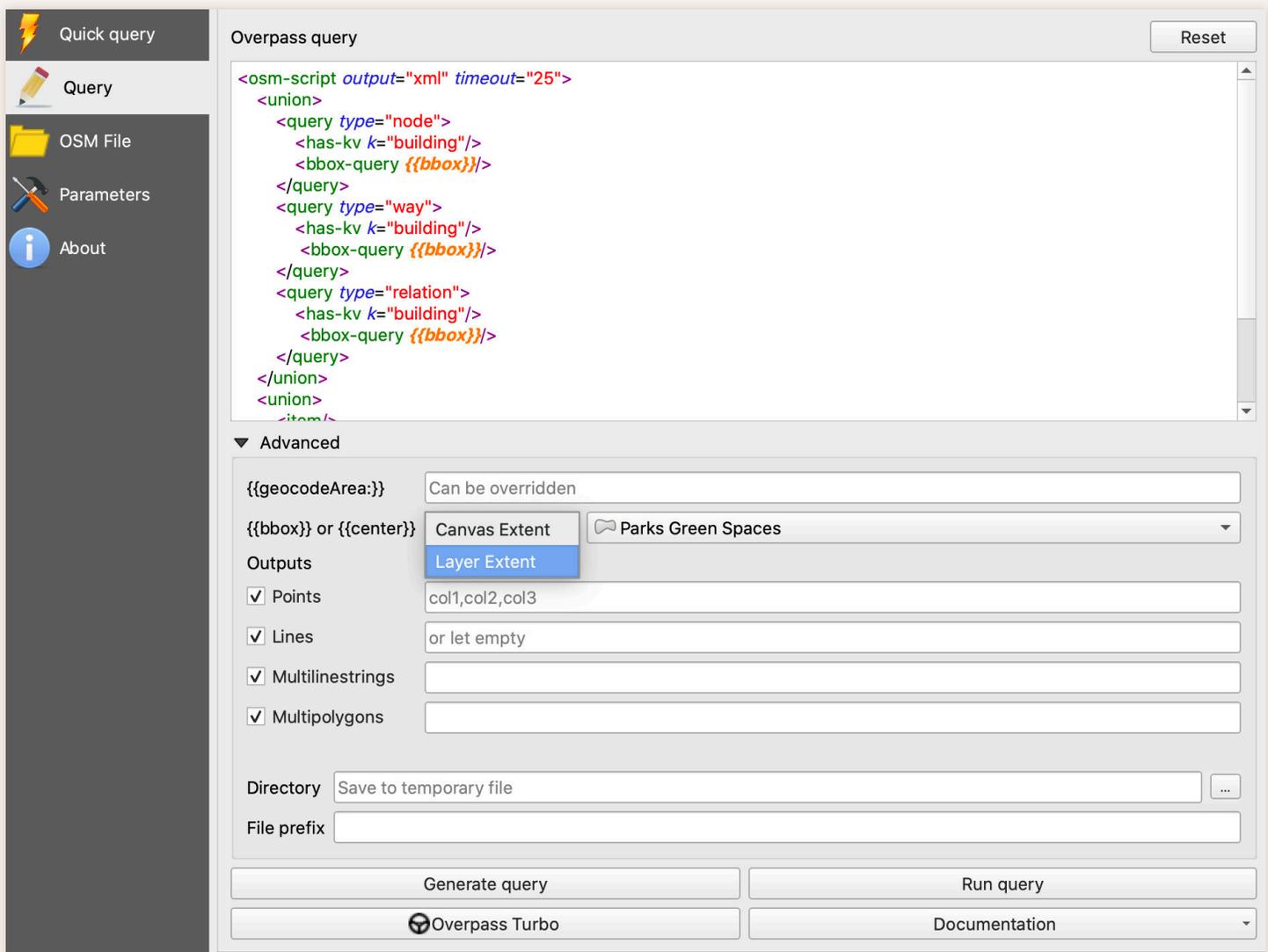
Figure 13. Advanced options to select data extent and type, as well as save-as options.

and be removed once QGIS is closed. Once the process is complete, we can close the plugin tool to view the new layers in the map and the **Layers** panel. Figure 14 shows that the building layers have been added as temporary files. These new OSM layers deliver additional spatial context and augment the parks and green spaces layers.

QUICKOSM - OSM FILE

The QuickOSM plugin has an option to add OSM data in OSM format (Figure 15). As noted earlier, both PlanetOSM and Geofabrik allow users to download OSM data in this format. However, QGIS already supports the OSM format, so this part of the plugin does not offer anything new outside of the core application capabilities. Like the other QuickOSM tools, there are options to import specific data types and to open the data as a temporary file.

SAVE OSM DATA

There are two methods to save OSM data accessed through the QuickOSM plugin. The first one is one mentioned previously: the plugin allows the user to designate a location in which to save the data as it is downloaded. However, QGIS also allows a user to work with data files in a temporary state. This method stores data to
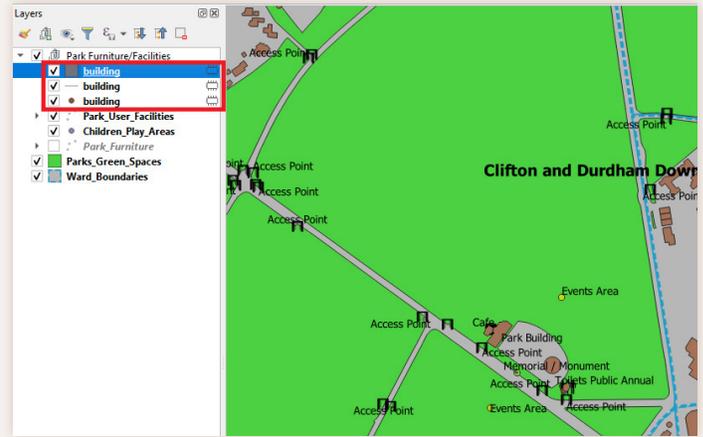
Figure 14. *QuickOSM temporary layers. The red box highlights the temporary layers added.*
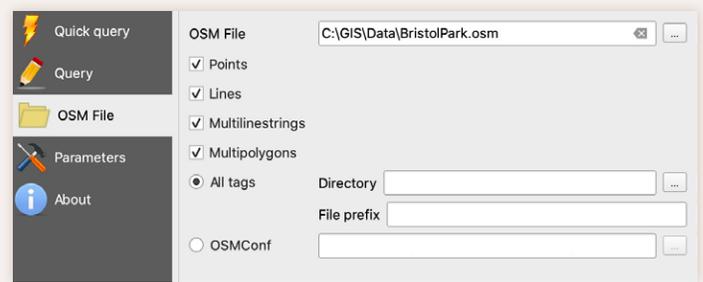

Figure 15. *OSM file import options.*
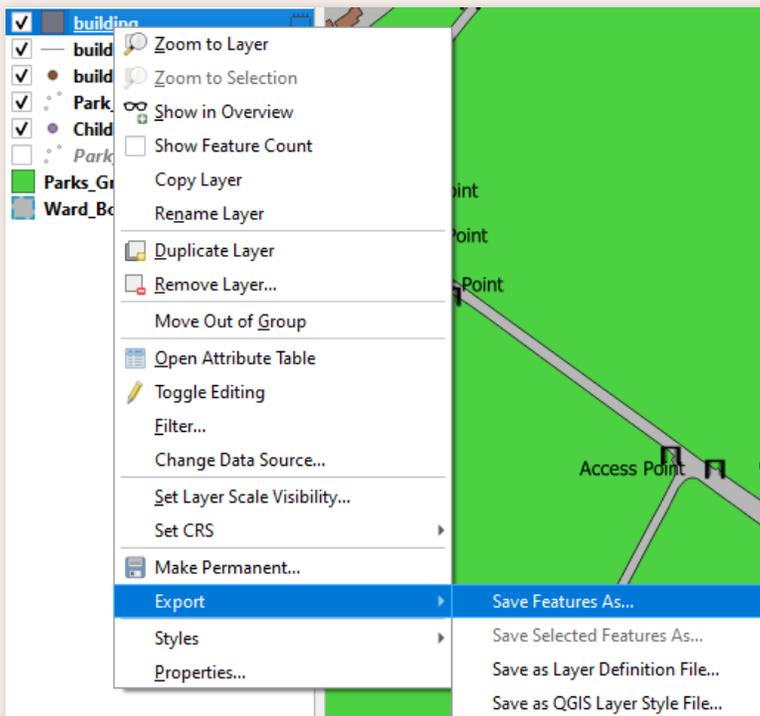
Figure 16. *Saving temporary layers from the QGIS table of contents.*
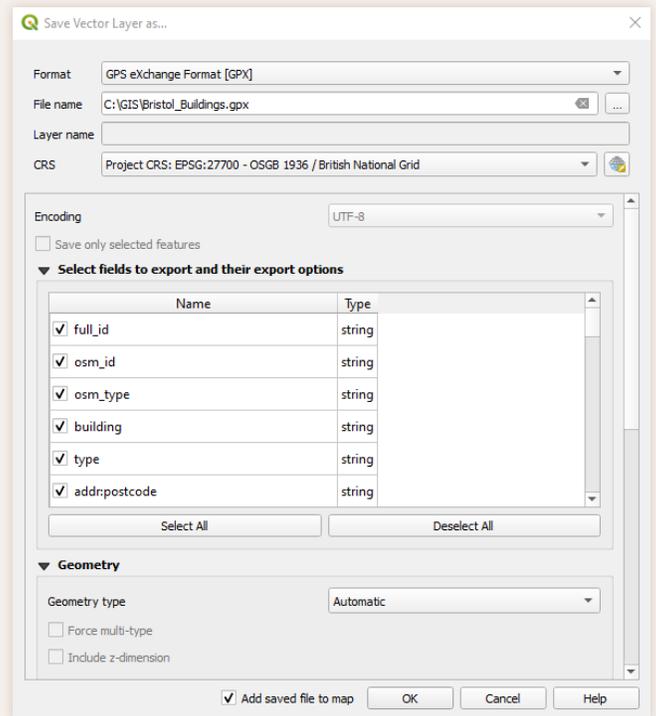
a temporary folder on the local computer, which requires disk space. However, if the downloaded data is not needed, it can be discarded without having to remembering to remove the associated file. QGIS will display an icon in the **Layers** panel next to any layer that is stored temporarily, as can be seen in Figure 14. The second option for saving the OSM data is to simply save the temporary layers, by right-clicking the layer, selecting **Export**, then **Save Features As…**, and then selecting a location in which to save the data and specifying a file format (Figure 16).

FINAL MAP

In our example, the data we obtained from OpenStreetMap can give additional context to existing data we obtained from Bristol's open data portal. In Figure 7, there is data on public spaces, but there is no information in the map regarding the spaces in between the public spaces. Figure 17 shows additional context using OSM data: buildings, roads, bus stops, and crossing locations have all been added, offering information about accessibility, and the setting of the public spaces.

SUMMARY

Although QuickOSM and OSMDownloader both provide fast and easy tools to access and download OpenStreetMap data, QuickOSM is the more versatile of two, allowing a user to select specific data types and datasets. The interface in QuickOSM is easy to use compared to OSMDownloader and has options for selecting specific datasets and locations. Additionally, the interface in

QuickOSM does not limit the user to a specific menu of options when selecting a dataset, instead allowing experienced users to customize the data query. Using QGIS and the QuickOSM plugin greatly enhances the accessibility of OpenStreetMap data. They can save considerable time by eliminating much of the post-processing needed when downloading data using other methods.
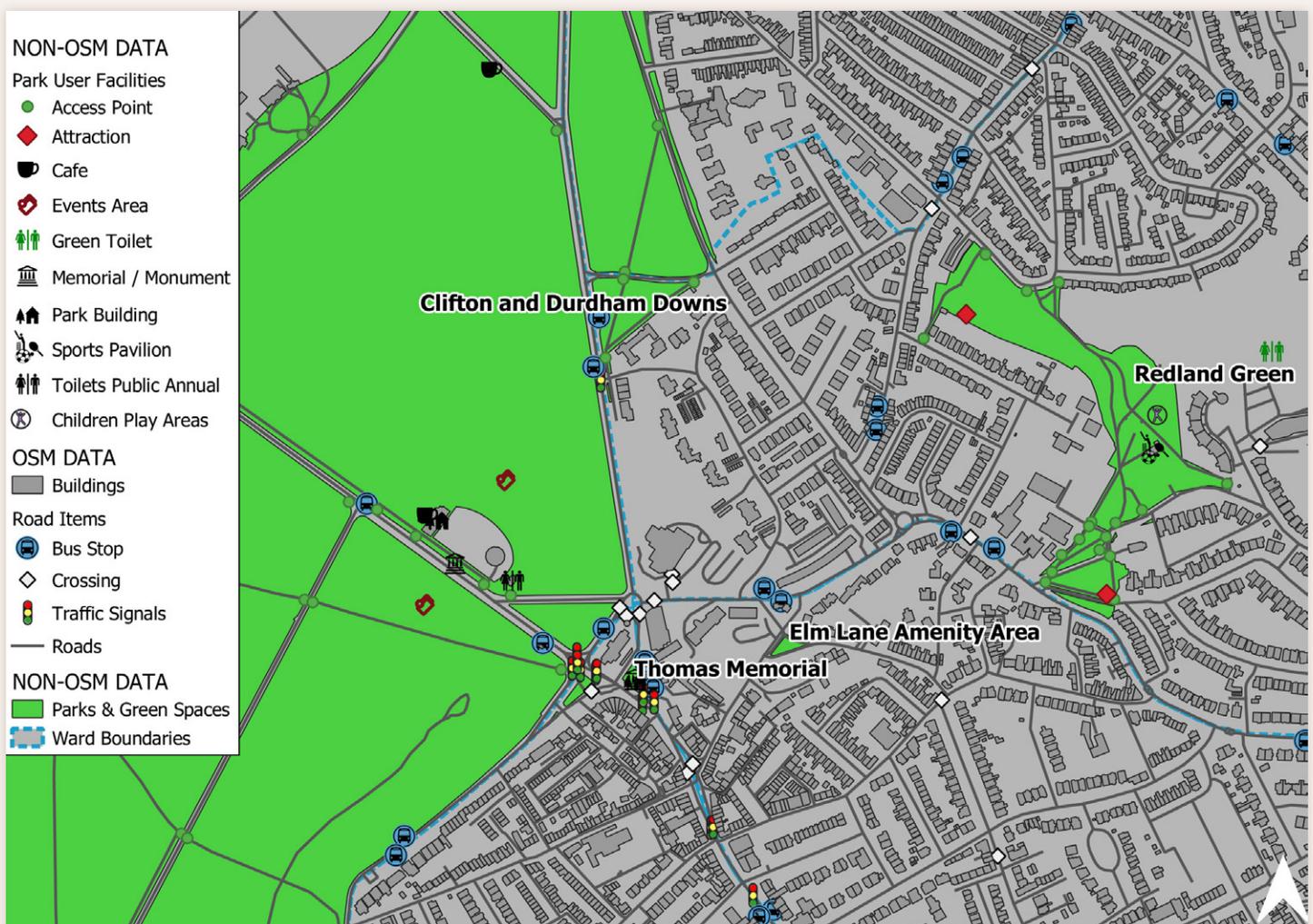
Figure 17. *Map after the addition of OpenStreetMap data.*