



3DCityDB Tools

for

QGIS

Installation and user guide

Version 0.8.4

Last update: 28 August 2023

License

This presentation is licensed under the [Creative Commons License CC BY-NC-SA 4.0](https://creativecommons.org/licenses/by-nc-sa/4.0/). According to CC BY-NC-SA 4.0 permission is granted to share this document, i.e. copy and redistribute the material in any medium or format, and to adapt it, i.e. remix, transform, and build upon the material under the following conditions:



- **Attribution:** You must give appropriate credit, provide a link to the license, and indicate if changes were made. You may do so in any reasonable manner, but not in any way that suggests the licensor endorses you or your use.
- **NonCommercial:** You may not use the material for commercial purposes.
- **ShareAlike:** If you remix, transform, or build upon the material, you must distribute your contributions under the same license as the original.
- **No additional restrictions:** You may not apply legal terms or technological measures that legally restrict others from doing anything the license permits.

Motivation

Plugin
overview

Software
installation

Software
use

Advanced
options

Software
uninstall

Current
limitations

QGIS
Package

Resources

Motivation

Plugin
overview

Software
installation

Software
use

Advanced
options

Software
uninstall

Current
limitations

QGIS
Package

Resources

Motivation

IDEA:

Motivation

Plugin overview
Software installation
Software use
Advanced options
Software uninstall
Current limitations
QGIS Package
Resources
Changelog

- **CityGML 3D City Database:** Why not letting users benefit from *directly* working with the 3DCityDB?
 - No need to work with files
 - Editing of features attributes could become way easier
 - SQL/relational model are rather well-known in and outside the GIS user community
 - Last but not least.... "3D city models belong best in a database" 😊



Motivation

BUT:

Motivation

Plugin overview

Software installation

Software use

Advanced options

Software uninstall

Current limitations

QGIS Package

Resources

Changelog

- 3DCityDB structure is rather complex
 - Lots of nested tables, intricate structure
 - Data management is difficult, although some functions are provided (e.g. delete functions)
 - There can be multiple citydb schemas in the same database instance (aka "scenarios")
- CityGML does not follow the Simple Feature for SQL model (SFS)
 - Nested features
 - One feature can have multiple representations (multiple LoDs, multiple geometry types)
- The existing **Importer/Exporter** offers some functionalities, but its *raison d'être* is basically different (...as the name says!)

Motivation

SO:

Motivation

Plugin overview

Software installation

Software use

Advanced options

Software uninstall

Current limitations

QGIS Package

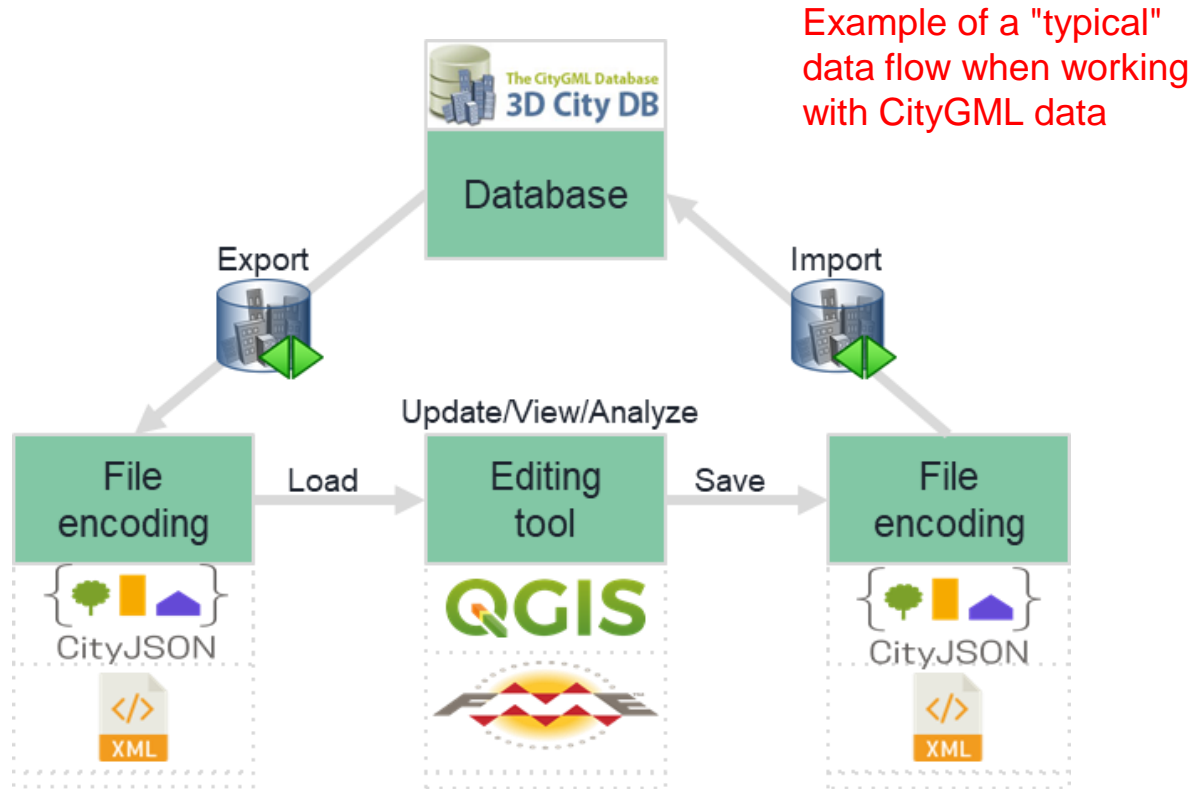
Resources

Changelog

- Why not using **QGIS**?
 - Well-known and established open-source software
 - Rather mature, version 3.28 LTR released in autumn 2022, well documented
 - Native support for PostgreSQL/PostGIS, and for Oracle Spatial
 - Has strong 2D and some (less mature) 3D visualisation functionalities
 - Can be extended with Python-based plugins

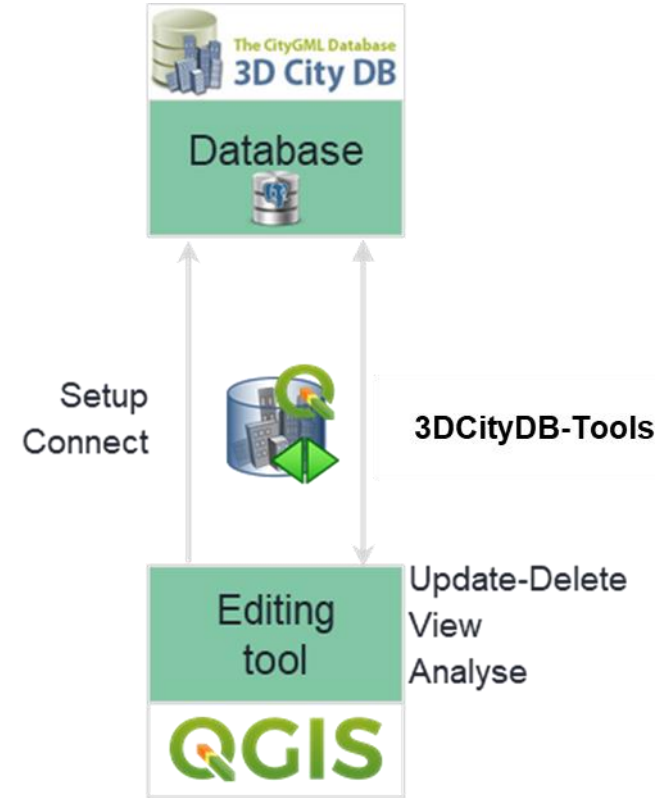
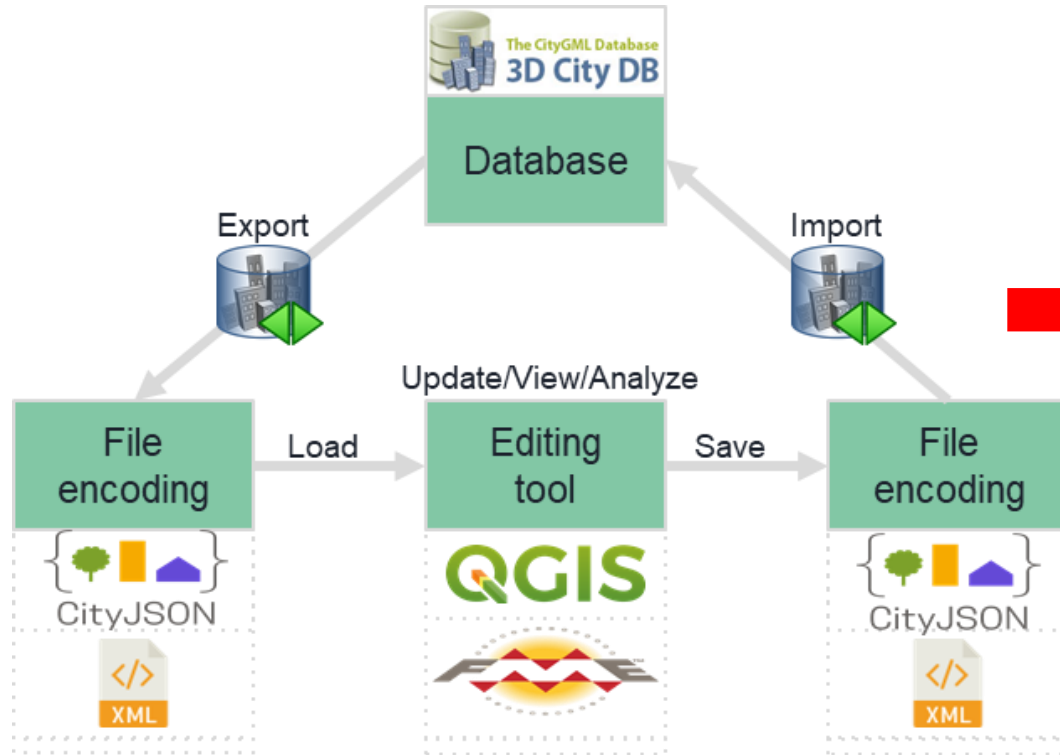


Motivation



Motivation

Vision / goal of the plugin



Motivation

Plugin
overview

Software
installation

Software
use

Advanced
options

Software
uninstall

Current
limitations

QGIS
Package

Resources

Plugin overview

Main functionalities

- Create "**SFS-like layers**" to hide 3DCityDB complexity when interacting with data
 - Deal efficiently with multi-LoD / different geometries / implicit representations
 - Up to ≈600 possible combinations in CityGML!
 - Merge all standard attributes of a CityObject into a single "table"
- **Deal with** the possibly *huge size of city models* stored in a database
- Support for **multiple citydb schemas** in the same 3DCityDB instance
- Support for **multiple users** with **different privileges** (read-only, read-write)
- **Editing of attributes**: possible (depending on user privileges)
- **Deletion of features**: possible (depending on user privileges)
- Editing of geometries: NOT possible

Plugin overview

Motivation

Plugin overview

Software installation

Software use

Advanced options

Software uninstall

Current limitations

QGIS Package

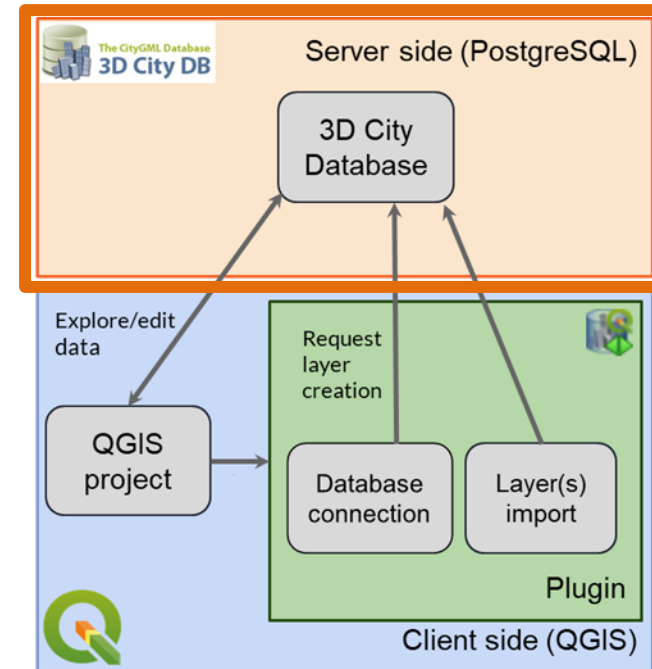
Resources

Changelog

Server-side

PostgreSQL "QGIS Package"

- Creates and manages layers as views (for attributes) linked to materialized views (for geometry) following the SFS model
- Manages
 - users and privileges
 - multiple citydb schemas
- Adds default users with ro & rw privileges



Plugin overview

Motivation

Plugin overview

Software installation

Software use

Advanced options

Software uninstall

Current limitations

QGIS Package

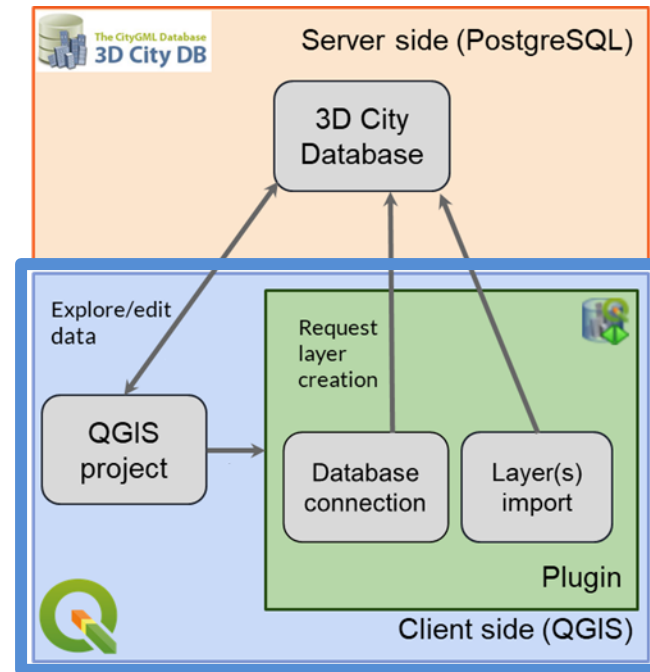
Resources

Changelog

Client-side

QGIS plugin “3DCityDB-Tools”

- Manages database connections + installation of the **QGIS Package**
- Allows for GUI-based
 - layer creation and management
 - management of multiple citydb schemas
 - editing of feature attributes
- GUI includes
 - support for children tables (e.g. generic attributes)
 - CityGML enumerations
 - Codelists
- Creates a hierarchical Table of Contents



IMPORTANT NOTICE

The following slides assume that you are already familiar with the **3DCityDB Suite**

In particular you should:

- Have an already installed 3DCityDB database instance
- Be able to use the 3DCityDB Importer/Exporter
- Be able to import CityGML data into the 3DCityDB
- (Optionally) be able to create additional citydb schemas

Otherwise:

- Refer to the slides in "**3DCityDB_Suite_QuickInstall.pdf**" (also in the same folder of this file) and/or
- Follow the tutorial: <https://github.com/3dcitydb/tutorials>

Last but not least...

- You may profit from a basic knowledge of the main CityGML concepts ☺
- Otherwise, here a crash course for free: <http://www.urbangeobigdata.it/?p=195>

Installation

Software requirements

- CityGML 3D City Database **v. 4.x** for PostgreSQL
 - <https://github.com/3dcitydb/3dcitydb-suite/releases>
 - <https://3dcitydb-docs.readthedocs.io/en/latest/>
 - **BEWARE:** 3DCityDB v. 3.x and older are NOT supported!
- PostgreSQL **v. 10 or higher**, PostGIS **v. 2.0 or higher**
 - <https://www.postgresql.org/download/>
 - **BEWARE:** NOT compatible with older versions
- QGIS **v. 3.22 LTR or v. 3.28 LTR**
 - <https://qgis.org/en/site/forusers/download.html>
 - **BEWARE:** Not tested/not supported with other versions
- PgAdmin (suggested, not required)
 - <https://www.pgadmin.org/download/>

Installation

Compatibility matrix

QGIS version	3DCityDB-Tools works?	Comments
QGIS 3.30, 3.32	✓	Should work, but not supported
QGIS 3.28 LTR	✓	Works
QGIS 3.26, 3.24	✓	Should work, but not supported
QGIS 3.22 LTR	✓	Works. Reference version used for development
QGIS 3.20	✗	May work, but not supported
QGIS 3.18	✗	"Import selected layers" button always disabled
QGIS 3.16 LTR	✗	User's GUI won't load. Issues with (outdated?) method QgsExtentGroupBox.setMapCanvas()

Motivation
 Plugin overview
Software installation
 Software use
 Advanced options
 Software uninstall
 Current limitations
 QGIS Package
 Resources
 Changelog

Testing machines

Machines used for development and testing:

OS	Processor(s)	HD	RAM	PostgreSQL	PostGIS
Windows 10 21H2 64bit	Core i7-8650U 1.7 GHz	SSD 2 TB	32 GB	14, 64bit	3.2
Ubuntu 20.04.3 LTS 64bit	Intel i7-7500U (4) 3.500GHz	SSD 250 GB	8 GB	12, 64bit	3.1
Mac OS (11.6.2 64bit)	Core i9-9980HK	HDD 1TB	32 GB	14, 64bit	3.1
Ubuntu 18.4 LTS 64bit	Virtual Machine	HDD 2 TB	16 GB	10, 64bit	3.0
Windows 10 22H2 64bit	Core i7-8565U 1.8 GHz	SSD 250GB + HDD 1 TB	16 GB	15, 64 bit	3.1
Ubuntu 22.04 LTS 64bit	Virtual Machine	100 GB	8 GB	14, 64 bit	3.2

Motivation

Plugin
overview

Software
installation

Software
use

Advanced
options

Software
uninstall

Current
limitations

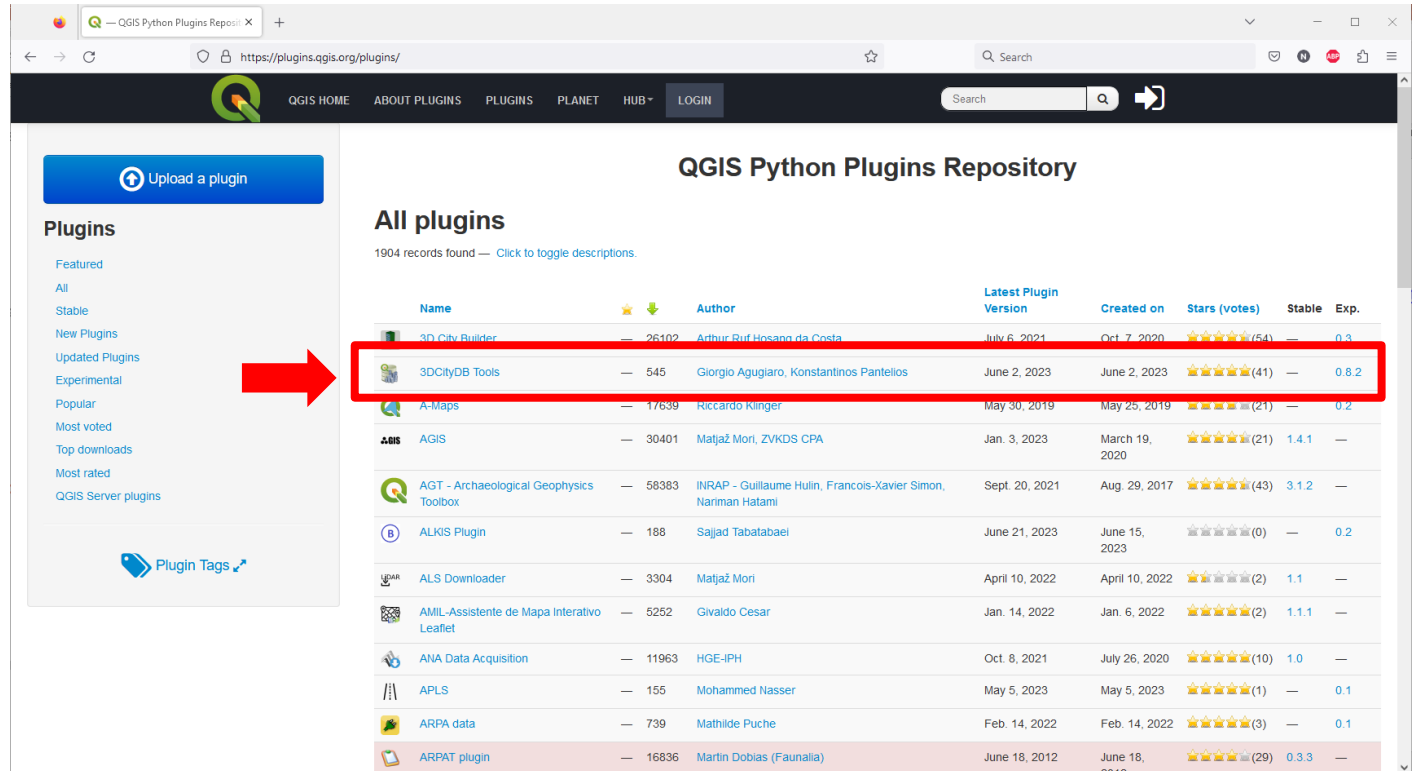
QGIS
Package

Resources

Installation via QGIS Plugins repository

Since version 0.8.2, the plugin is available also via the **QGIS Plugins repository**. This is the preferred (and easiest!) way to install it! Link: <https://plugins.qgis.org/plugins/>

Motivation
 Plugin overview
Software installation
 Software use
 Advanced options
 Software uninstall
 Current limitations
 QGIS Package
 Resources
 Changelog



QGIS Python Plugins Repository

All plugins
 1904 records found — [Click to toggle descriptions.](#)

Name	Stars	Author	Latest Plugin Version	Created on	Stars (votes)	Stable	Exp.
3D City Builder	26102	Arthur Bur Hossano da Costa	July 6, 2021	Oct 7, 2020	★★★★★ (54)	0.3	
3D City Tools	545	Giorgio Aguiaro, Konstantinos Pantelios	June 2, 2023	June 2, 2023	★★★★★ (41)	0.8.2	
A-Maps	17639	Riccardo Klinger	May 30, 2019	May 25, 2019	★★★★★ (21)	0.2	
AGIS	30401	Matjaž Mori, ZVKDS CPA	Jan. 3, 2023	March 19, 2020	★★★★★ (21)	1.4.1	
AGT - Archaeological Geophysics Toolbox	58383	INRAP - Guillaume Hulin, Francois-Xavier Simon, Nariman Hatami	Sept. 20, 2021	Aug. 29, 2017	★★★★★ (43)	3.1.2	
ALKIS Plugin	188	Sajjad Tabatabaei	June 21, 2023	June 15, 2023	★★★★★ (0)	0.2	
ALS Downloader	3304	Matjaž Mori	April 10, 2022	April 10, 2022	★★★★★ (2)	1.1	
AMIL-Assistente de Mapa Interativo Leaflet	5252	Givaldo Cesar	Jan. 14, 2022	Jan. 6, 2022	★★★★★ (2)	1.1.1	
ANA Data Acquisition	11963	HGE-IPH	Oct. 8, 2021	July 26, 2020	★★★★★ (10)	1.0	
APLS	155	Mohammed Nasser	May 5, 2023	May 5, 2023	★★★★★ (1)	0.1	
ARPA data	739	Mathilde Puche	Feb. 14, 2022	Feb. 14, 2022	★★★★★ (3)	0.1	
ARPAT plugin	16836	Martin Dobias (Faunalia)	June 18, 2012	June 18, 2012	★★★★★ (29)	0.3.3	

Installation via QGIS Plugins repository

Simply select it, and download it. If QGIS is already installed, it will be loaded automatically to the right folder. Done! 😊

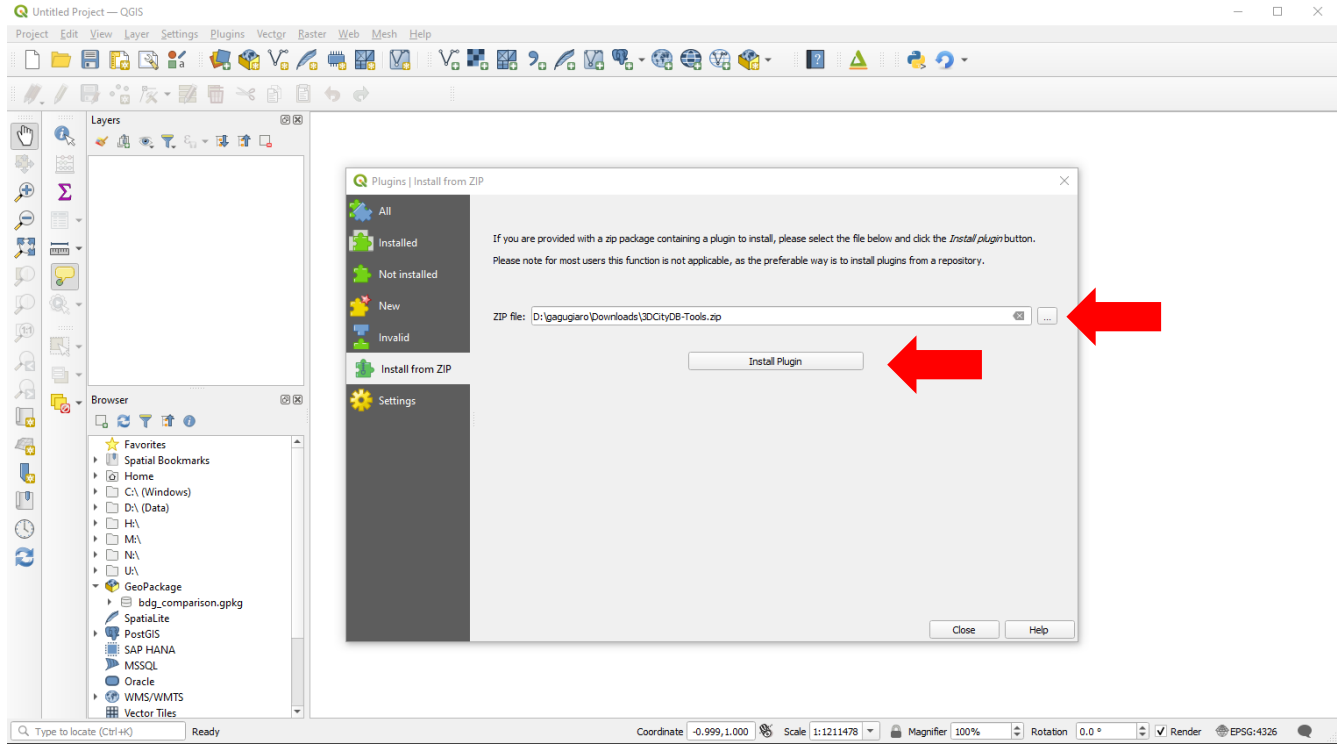
Motivation
 Plugin overview
Software installation
 Software use
 Advanced options
 Software uninstall
 Current limitations
 QGIS Package
 Resources
 Changelog



The screenshot shows the QGIS Python Plugins Repository website. The browser address bar displays <https://plugins.qgis.org/plugins/citydb-tools/>. The website has a dark header with navigation links: QGIS HOME, ABOUT PLUGINS, PLUGINS, PLANET, HUB, and LOGIN. A search bar is located on the right. The main content area is titled "QGIS Python Plugins Repository" and features a "3DCityDB Tools" plugin. On the left, there is a sidebar with a list of plugins categorized by "Featured", "All", "Stable", "New Plugins", "Updated Plugins", "Experimental", "Popular", "Most voted", "Top downloads", "Most rated", and "QGIS Server plugins". A red arrow points from the "Upload a plugin" button to the "Download latest" button. The plugin details for "3DCityDB Tools" include a star rating of 4.5 (41 votes), a description: "Tools to visualize and manipulate CityGML data stored in the 3D City Database", and tabs for "About", "Details", and "Versions". The "Details" tab is selected, showing a description of the plugin's functionality and its components: "The 'Layer Loader': it loads data as 'classical' layers into QGIS. All CityGML LoDs are supported, as well as multiple citydb schemas and multiple database users. Feature attributes can be edited, changes are stored back directly into the database;", "A 'Bulk Deleter': it allows to cleanup the database, or to delete selected feature types;", and "QGIS Package Administration": it allows to install the server-side part of the plug-in, and to manage database user privileges. Further details can be found in the PDF files contained in the 'user_guide' subfolder of the plugin installation directory. On the GitHub repository, some test datasets can be downloaded.

Manual installation

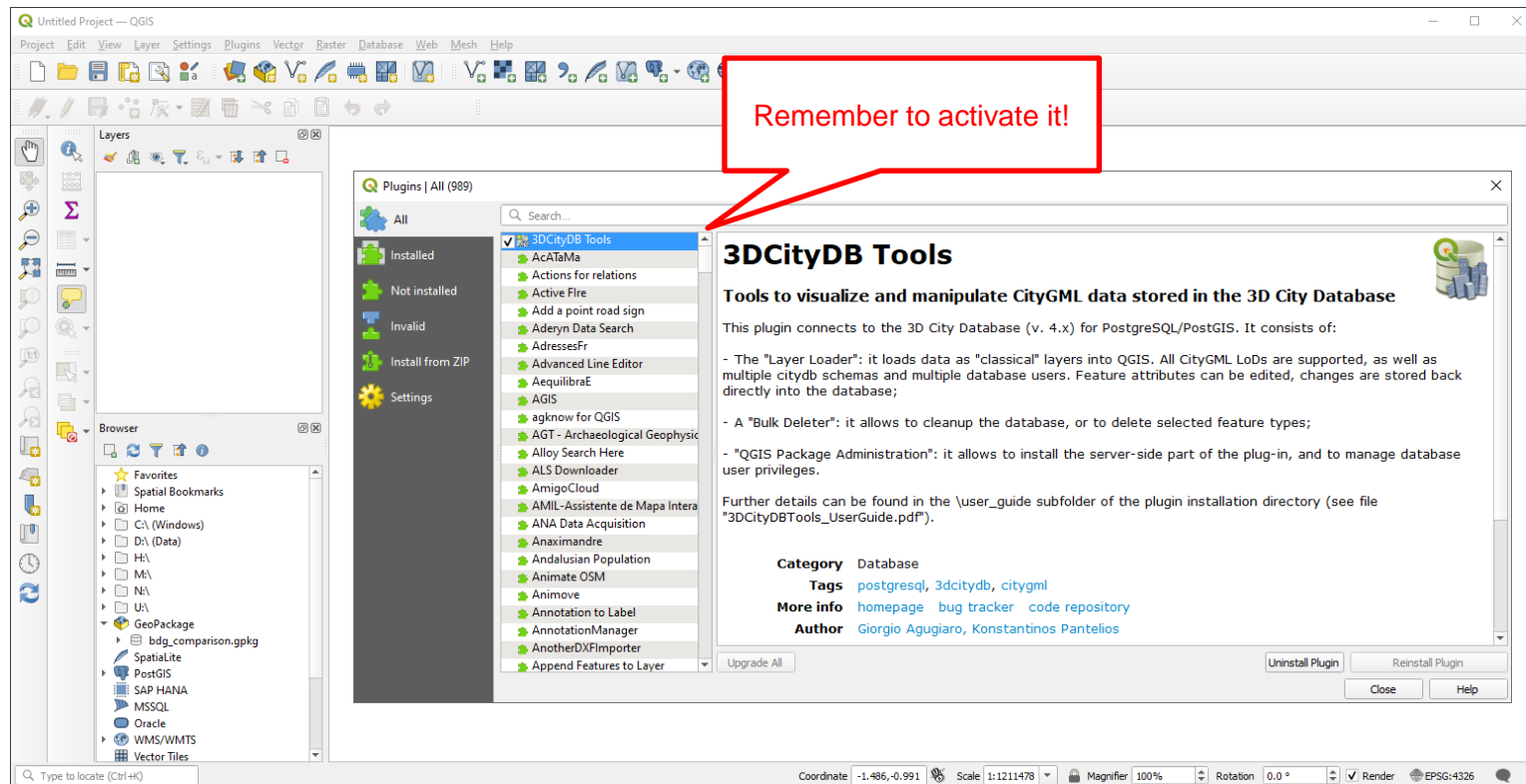
- The plugin is available also as **zip file** from the **GitHub repository**
- In QGIS, open the Plugins\Manage and install plugins window, and choose "Install from ZIP". Select the zip file and click the "Install Plugin" button



Manual installation

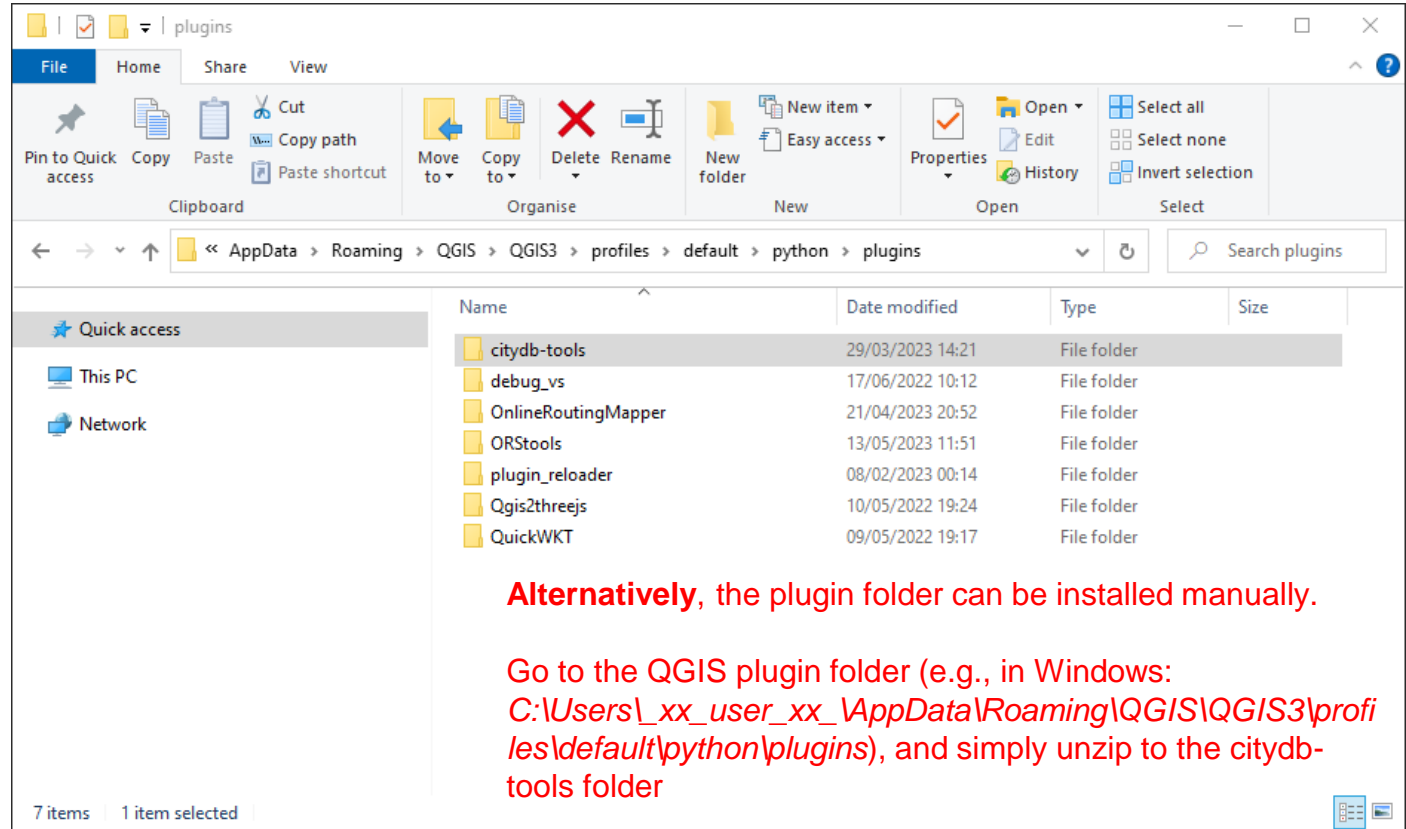
- Upon installation, you must activate the plugin

Motivation
 Plugin overview
Software installation
 Software use
 Advanced options
 Software uninstall
 Current limitations
 QGIS Package
 Resources
 Changelog



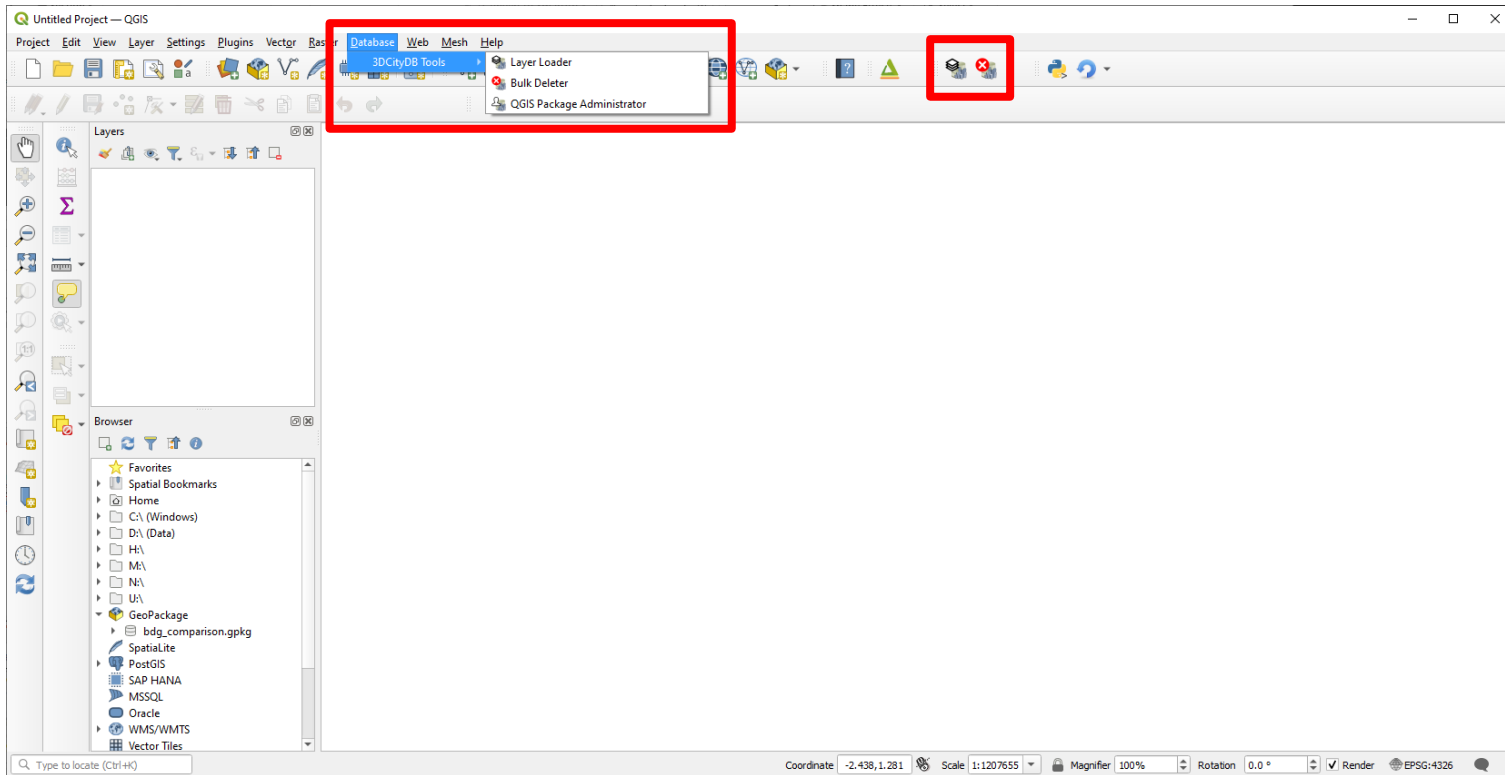
Manual installation

Motivation
 Plugin overview
Software installation
 Software use
 Advanced options
 Software uninstall
 Current limitations
 QGIS Package
 Resources
 Changelog



Front-end installation

The **Database** menu will now contain a new entry, and the icons will be visible in the database icon bar (if the icon bar is activated)



Motivation
 Plugin overview
Software installation
 Software use
 Advanced options
 Software uninstall
 Current limitations
 QGIS Package
 Resources
 Changelog

Back-end installation

The back-end installation consists in installing the **QGIS Package** into a 3DCityDB instance. It can be carried out using the "QGIS Package Administrator" GUI of the front-end. The **database administrator** is responsible for setting up in advance the server-side for *any* database user.

In general, **4 steps** are necessary:

- a) Installation of the QGIS Package (i.e. the "qgis_pkg" schema)
- b) Selection of the database users (e.g. "giorgio")
- c) Creation of a user schema for each selected user (e.g. "qgis_giorgio")
- d) Definition of the database privileges for each user and for each citydb schema (i.e. "read-only", "read & write", "none")

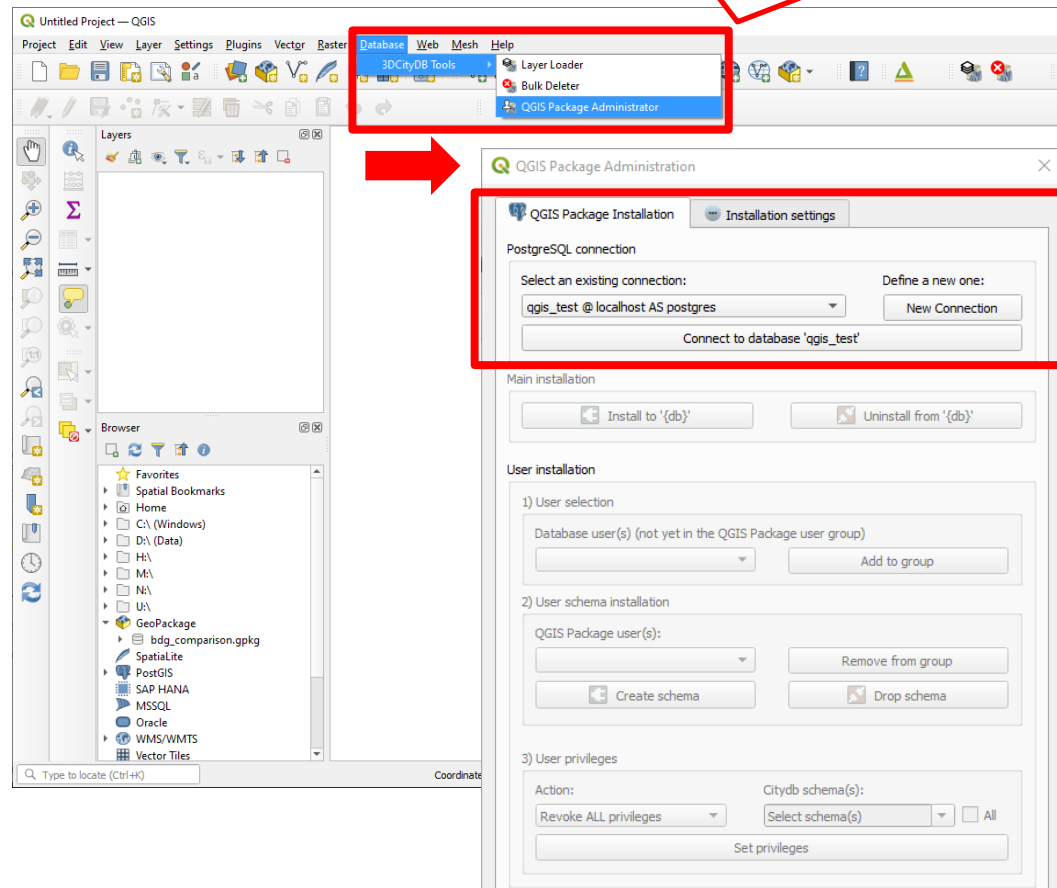
It is possible to perform:

- A "**simplified installation**", which carries out all 4 operation at once and automatically installs 2 default users
- A "**normal installation**", where the administrator has complete control over each step

Back-end installation

As **database administrator**
(e.g. "postgres"):

1) Create a new connection
or use an existing one to
the desired 3DCityDB
instance (here: "qgis_test")



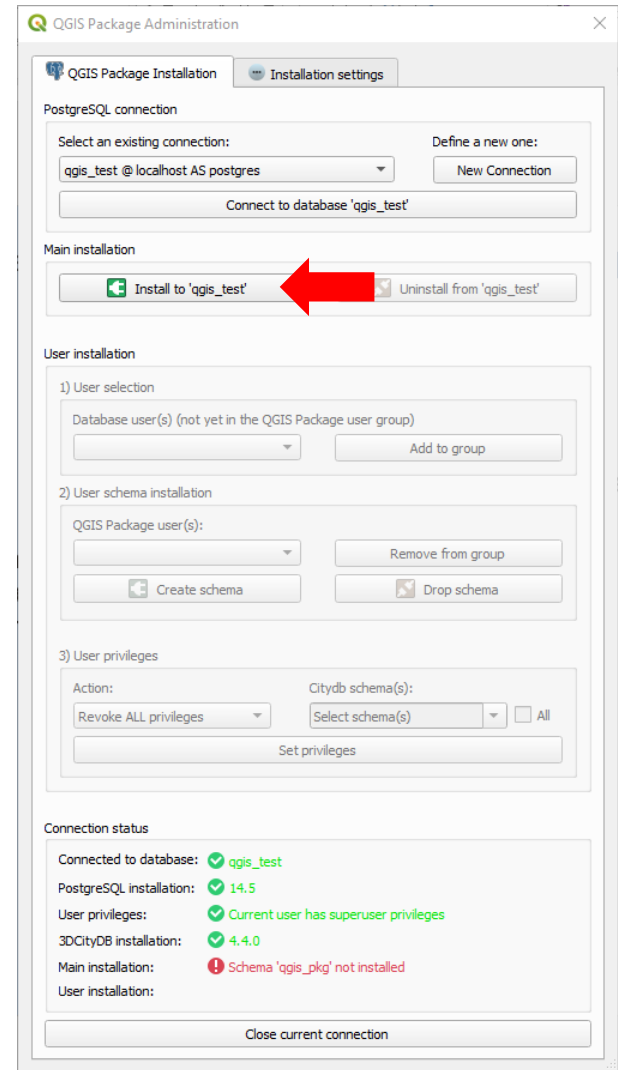
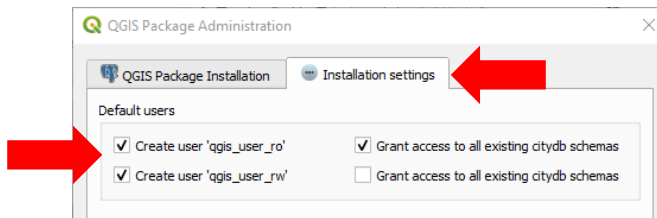
Back-end installation

2.1) "Simplified" installation

The QGIS Package and up to two default users are installed at once:

- User "**qgis_user_ro**" with read-only privileges
- User "**qgis_user_rw**" with read & write privileges
- Both users have access to all citydb schemas in the database at the moment of the installation
- **Note bene:** Their privileges and access rules can be changed at *any* time after installation. See later the "normal"-installation slides

Before clicking the Install button, go to the "Installation settings" tab and check the desired options



Back-end installation

2.1) "Simplified" installation

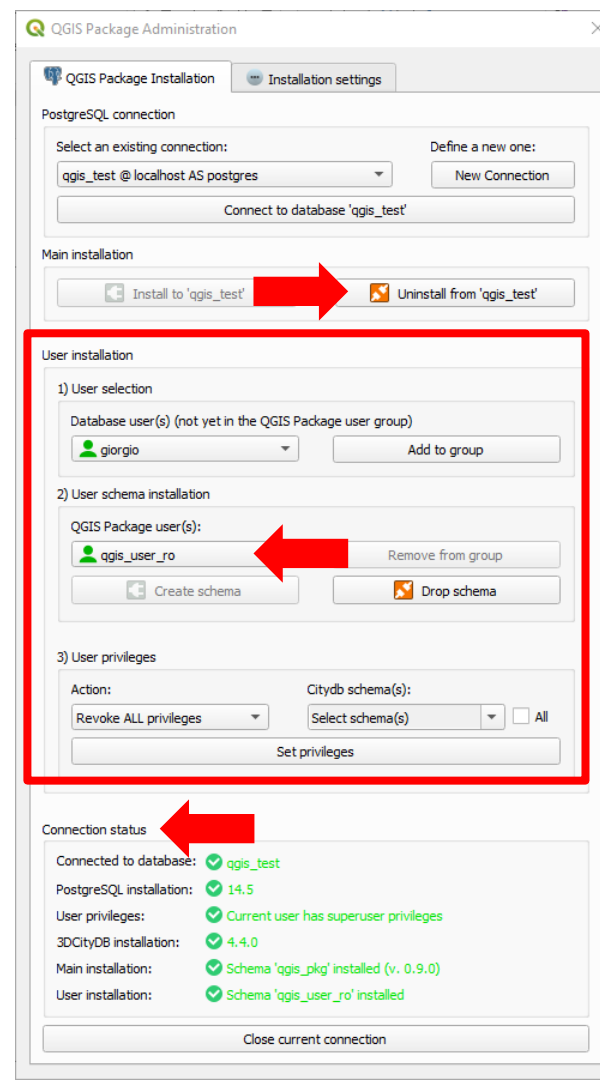
Upon successful installation:

- The **Uninstall button** is activated (in case you want to uninstall the QGIS Package)
- The **User Installation box** is activated
- You are notified in the **Connection status**

Done! 😊 Close the "QGIS Package Administration" GUI. You can now start using the plugin (e.g. the "Layer Loader" or the "Bulk Deleter") using the credentials of one of the default users

Please observe that:

- The default user(s) are automatically added to the **QGIS Package user(s)**
- You can edit the privileges in the **User privileges box**, or leave them as they are



Back-end installation

2.1) "Simplified" installation

In the 3DCityDB, the **qgis_pkg** schema is added and, depending on the chosen options, the **qgis_user_ro** and/or the **qgis_usr_rw** schemas are created, too

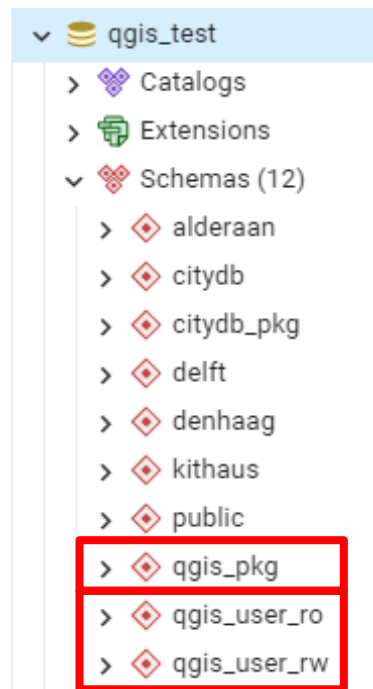
The credentials for the default users are:

User "qgis_user_ro":

- user name: **qgis_user_ro**
- password: **qgis_user_ro**

User "qgis_user_rw":

- user name: **qgis_user_rw**
- password: **qgis_user_rw**



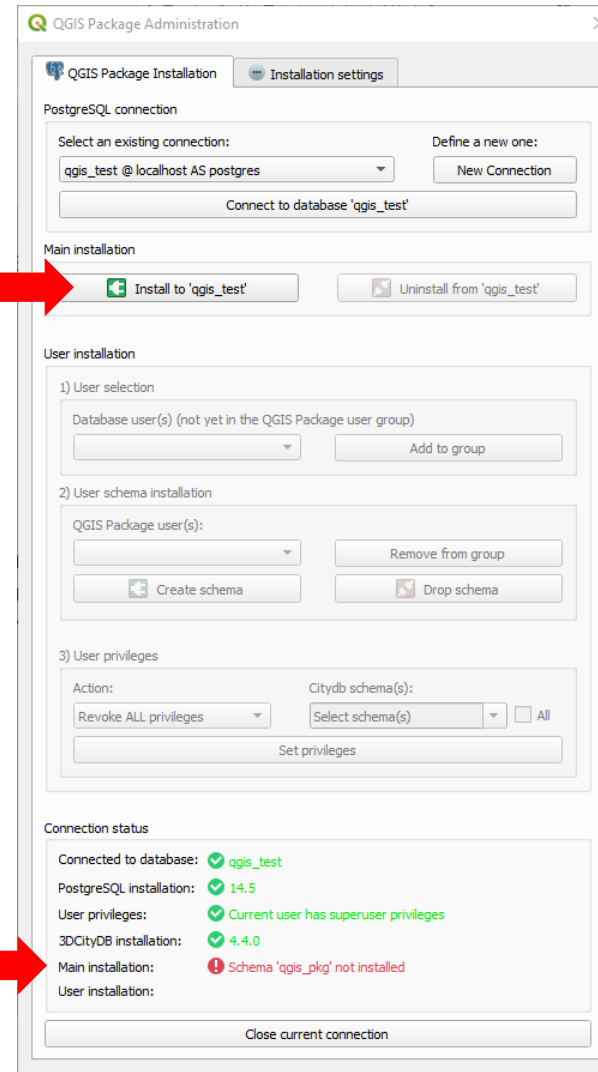
Back-end installation

2.2) "Normal" installation

Using the "normal" installation, the administrator has complete control over each one of the 4 required installation steps

Step a) To install the GIS Package, click the **Install button** (here: install to database "qgis_test")

The **Connection status box** in the lower part of the dialog will keep you informed.

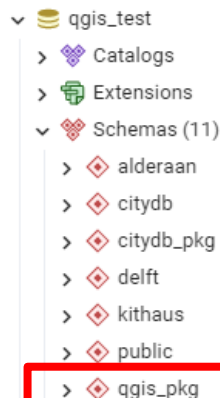


Back-end installation

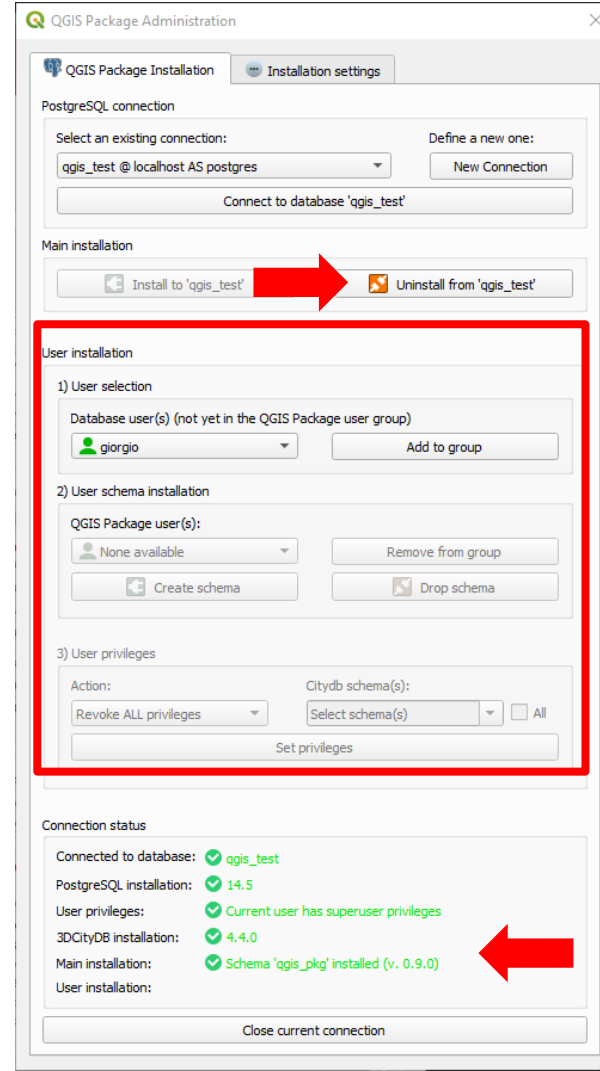
2.2) "Normal" installation

Upon successful installation:

- The **Uninstall button** is activated (in case you want to immediately uninstall)
- The **User Installation box** is activated
- You are notified in the **Connection status box**



The "**qgis_pkg**" schema is created in the the selected current database



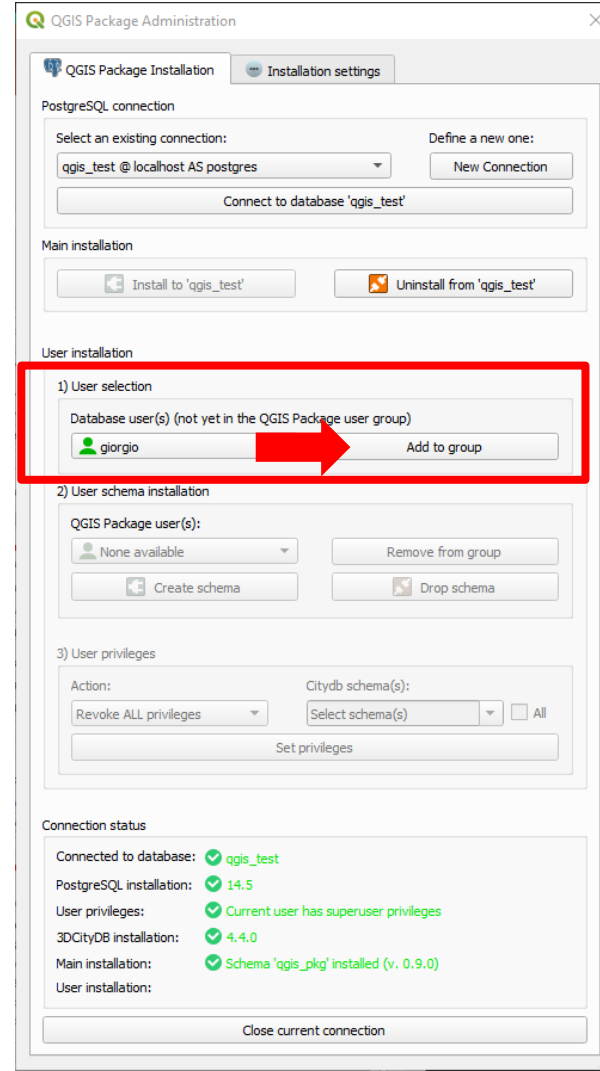
Back-end installation

2.2) "Normal" installation

Step b) Choose from all database users the one(s) to add to the QGIS Package user group. Click the **Add to group** button



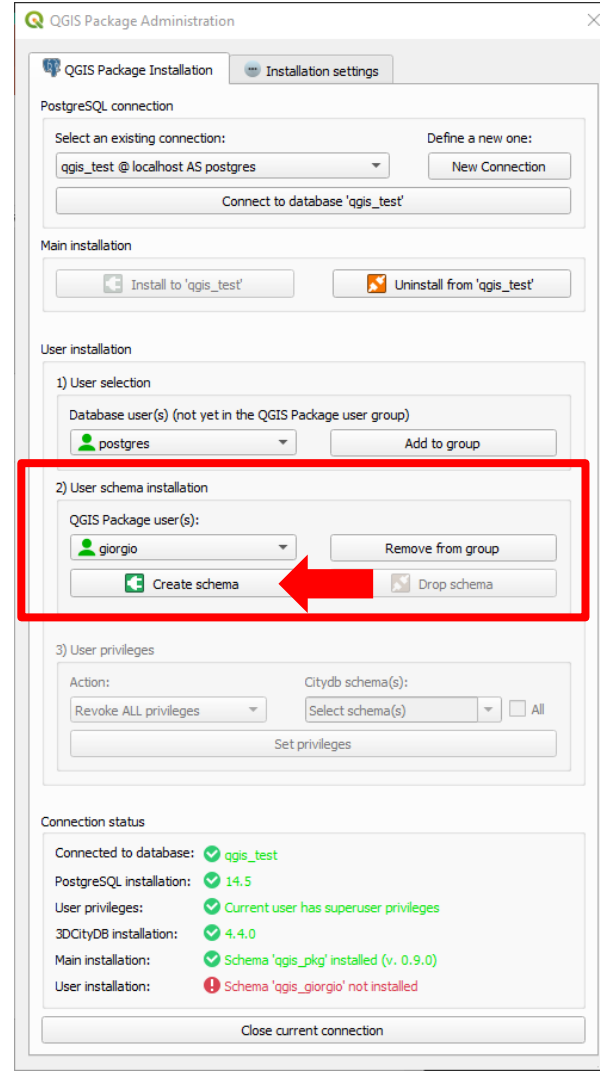
For each 3DCityDB there group named "**qgis_pkg_usrgroup_**" + **database name** is created. It contains those users that will be allowed to interact with the database from the front-end. Example: for database "qgis_test" there is a group called "qgis_pkg_usrgroup_qgis_test".



Back-end installation

2.2) "Normal" installation

Step c) Create the user schema for the selected user(s) belonging to the group

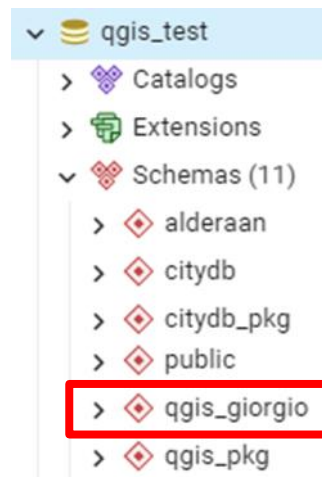


Back-end installation

2.2) "Normal" installation

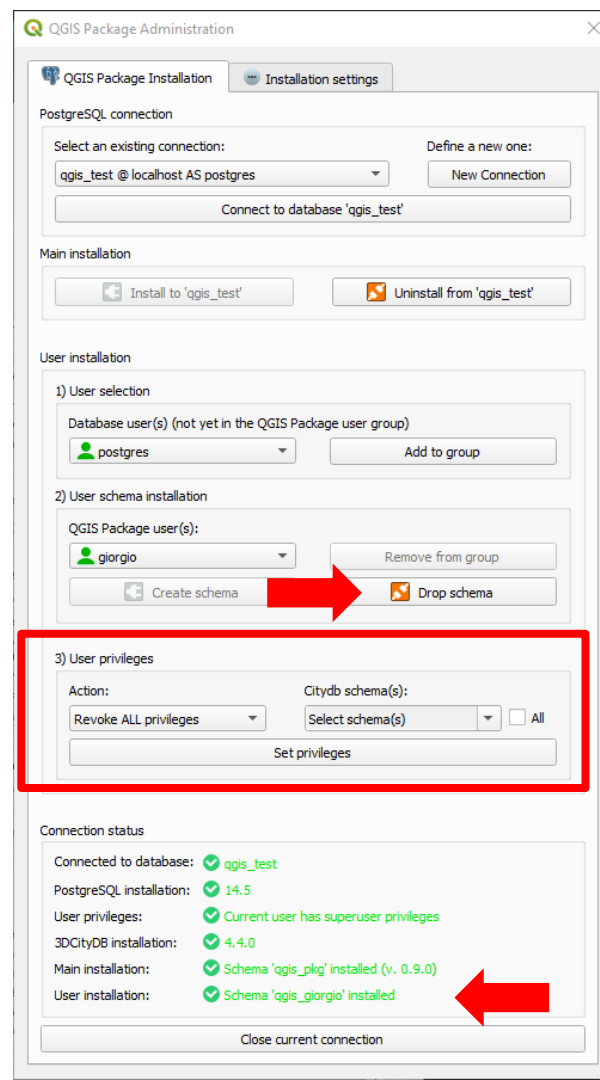
Upon successful creation of the user schema:

- The **Drop schema** button is activated (in case you want to drop the schema you just created)
- the **User privileges box** is activated
- You are notified in the **Connection status box**



A schema named "**qgis_** + **user name**" is created.

Example: for user "giorgio", schema "qgis_giorgio" will be created.



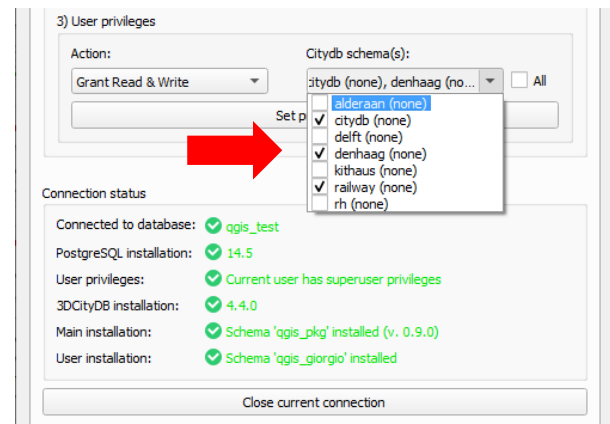
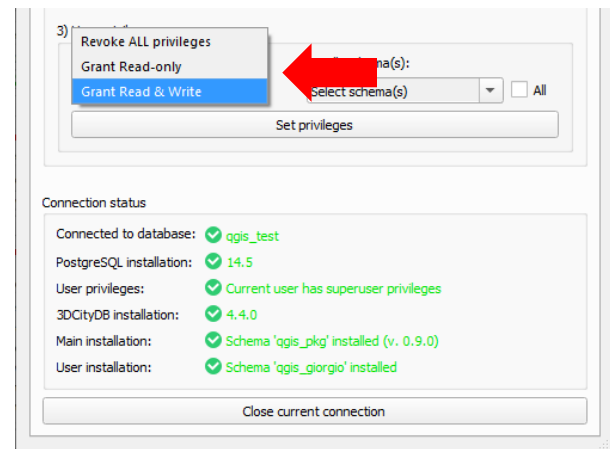
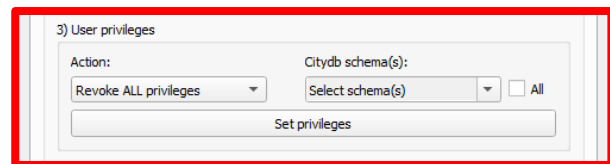
Back-end installation

2.2) "Normal" installation

Step d) For the selected user, set the database privileges ("read-only", "read & write", "none") for each of the existing citydb schemas

You can assign different privileges to different citydb schemas – or revoke them.

Click the **Set privileges** button to apply the settings. The privileges status in the drop down menu will be updated accordingly.



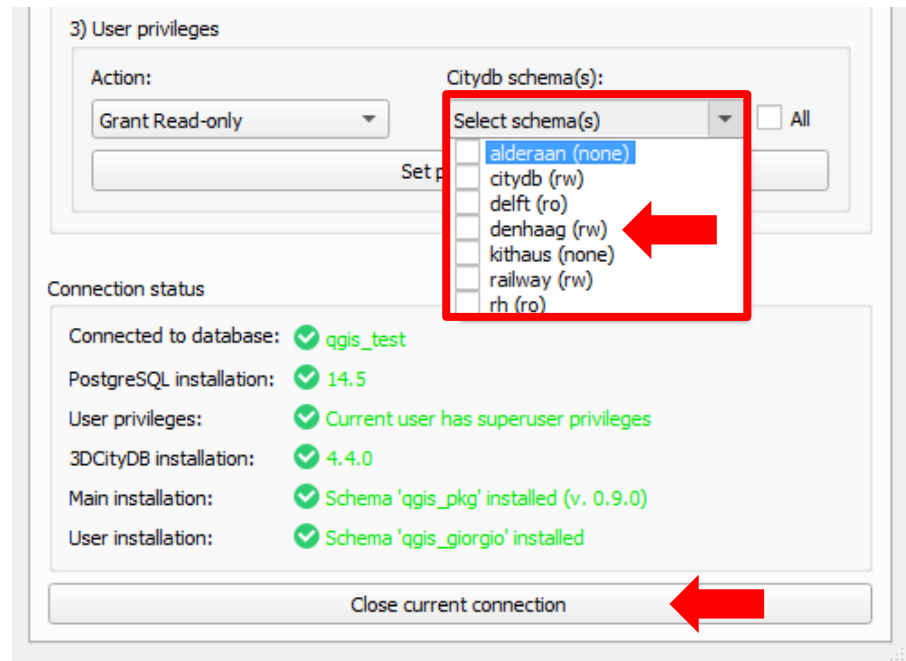
Back-end installation

2.2) "Normal" installation

Every time new privileges are set, the status in the drop down menu is updated with "ro" (read-only), "rw" (read & write) or "none".

Once you are done, you can click the **Close the current connection** button

You can now use the "Layer Loader" or the "Bulk Deleter"



Motivation

Plugin
overview

Software
installation

Software
use

Advanced
options

Software
uninstall

Current
limitations

QGIS
Package

Resources

Layer Loader

Open the **Layer Loader** from the menu or by clicking on the corresponding icon



Motivation

Plugin overview

Software installation

Software use

- Layer Loader
- Bulk Deleter

Advanced options

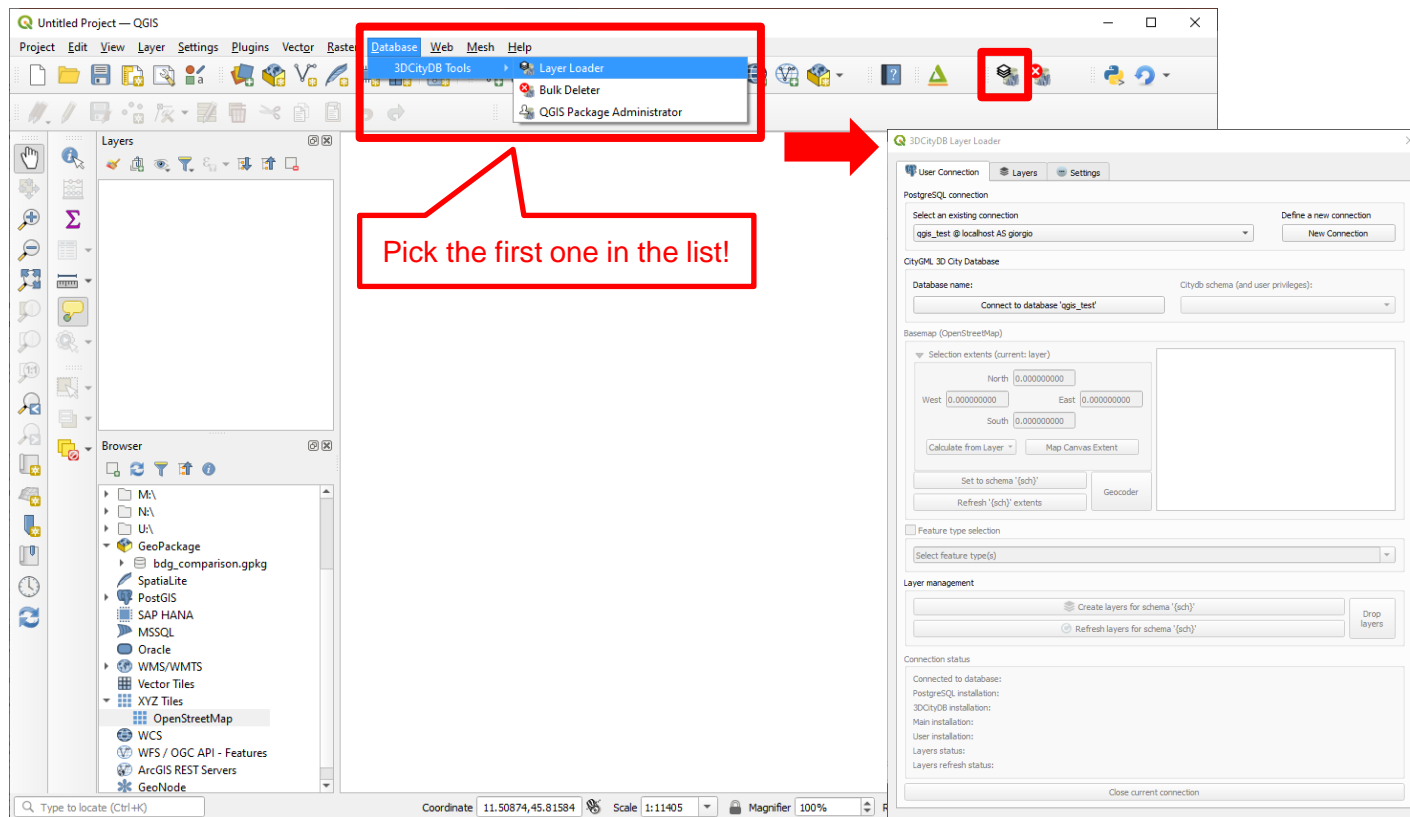
Software uninstall

Current limitations

QGIS Package

Resources

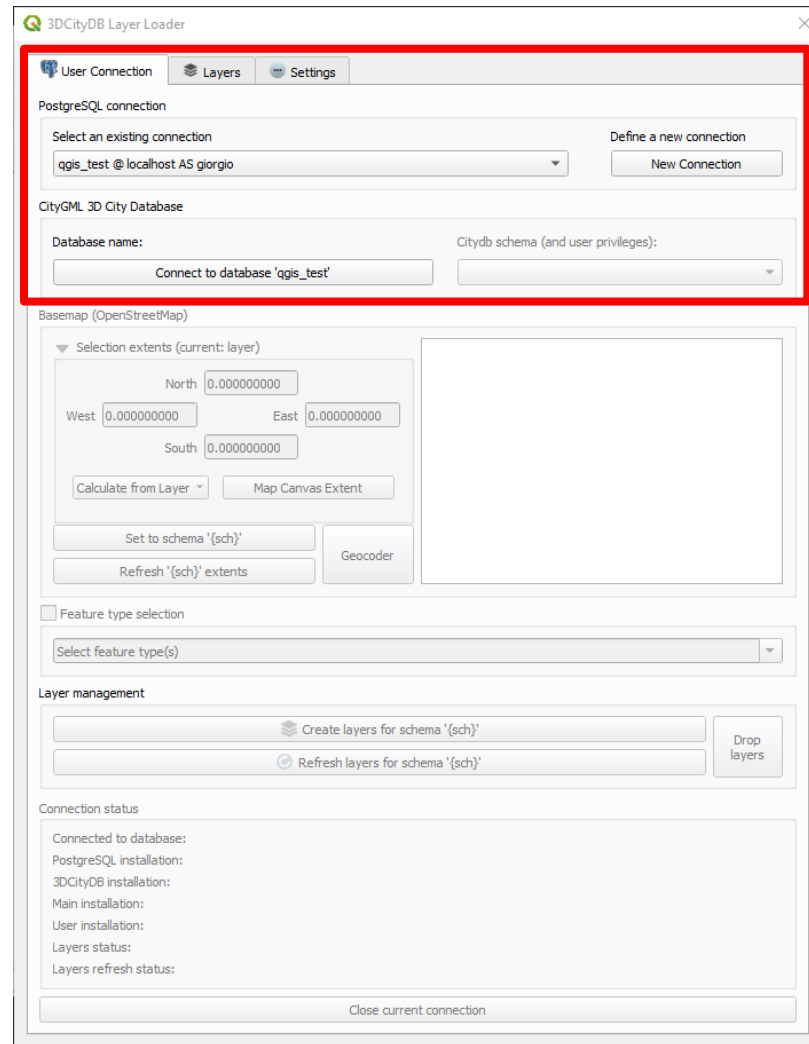
Changelog



Layer Loader

In the "User Connection" tab

- 1) Create a new connection or use an existing one to the desired 3DCityDB instance (here: "qgis_test")
- 2) Use the credentials of:
 - The default users **qgis_user_ro** or **qgis_user_rw** (if previously installed)
 - Your own credentials (if the administrator has set up your *usr_schema* before)



3DCityDB Layer Loader

User Connection Layers Settings

PostgreSQL connection

Select an existing connection: qgis_test @ localhost AS giorgio Define a new connection: New Connection

CityGML 3D City Database

Database name: Connect to database 'qgis_test' Citydb schema (and user privileges):

Basemap (OpenStreetMap)

Selection extents (current: layer)

North: 0.000000000 West: 0.000000000 East: 0.000000000 South: 0.000000000

Calculate from Layer Map Canvas Extent

Set to schema '{sch}' Refresh '{sch}' extents Geocoder

Feature type selection: Select feature type(s)

Layer management

Create layers for schema '{sch}' Refresh layers for schema '{sch}' Drop layers

Connection status

Connected to database: PostgreSQL installation: 3DCityDB installation: Main installation: User installation: Layers status: Layers refresh status:

Close current connection

Layer Loader

Motivation

Plugin overview

Software installation

Software use

- Layer Loader
- Bulk Deleter

Advanced options

Software uninstall

Current limitations

QGIS Package

Resources

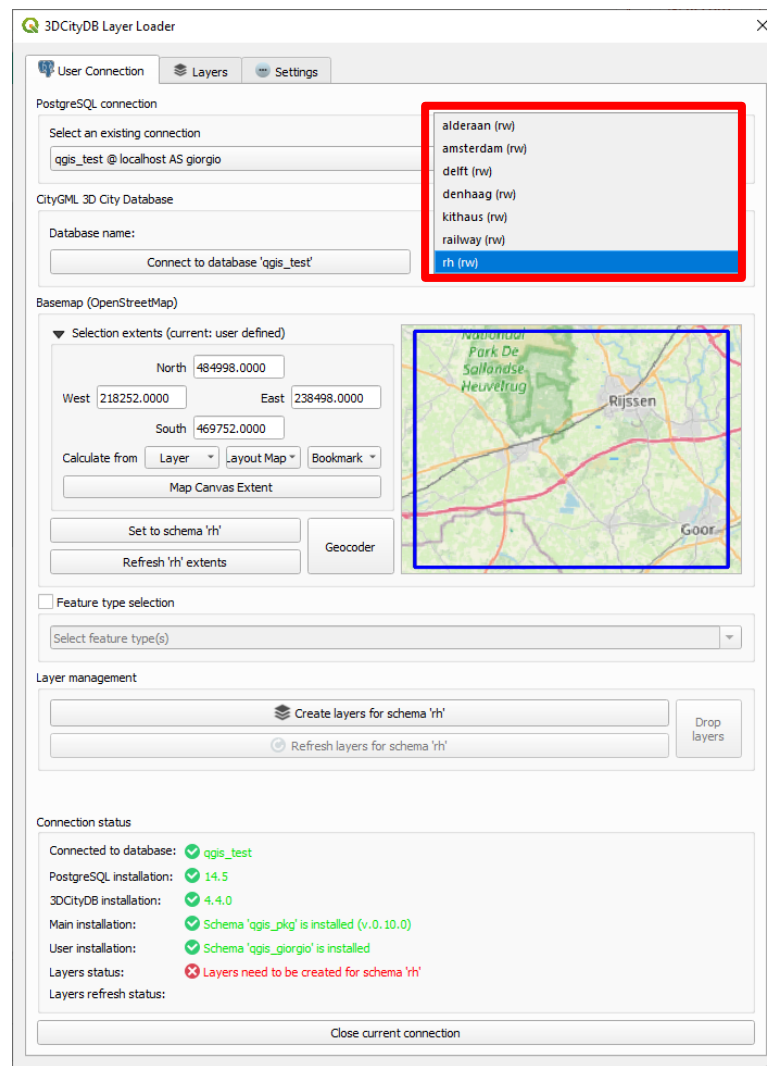
Changelog

3) Once connected, choose one of the existing citydb schemas. If they contain CityGML data, they will be listed.

You will also see your privileges for that citydb schema ("ro" or "rw").

Nota bene: Generally, "citydb" is the default, and, very often, the only one citydb schema! Nevertheless, the next slides refer to the "rh" schema

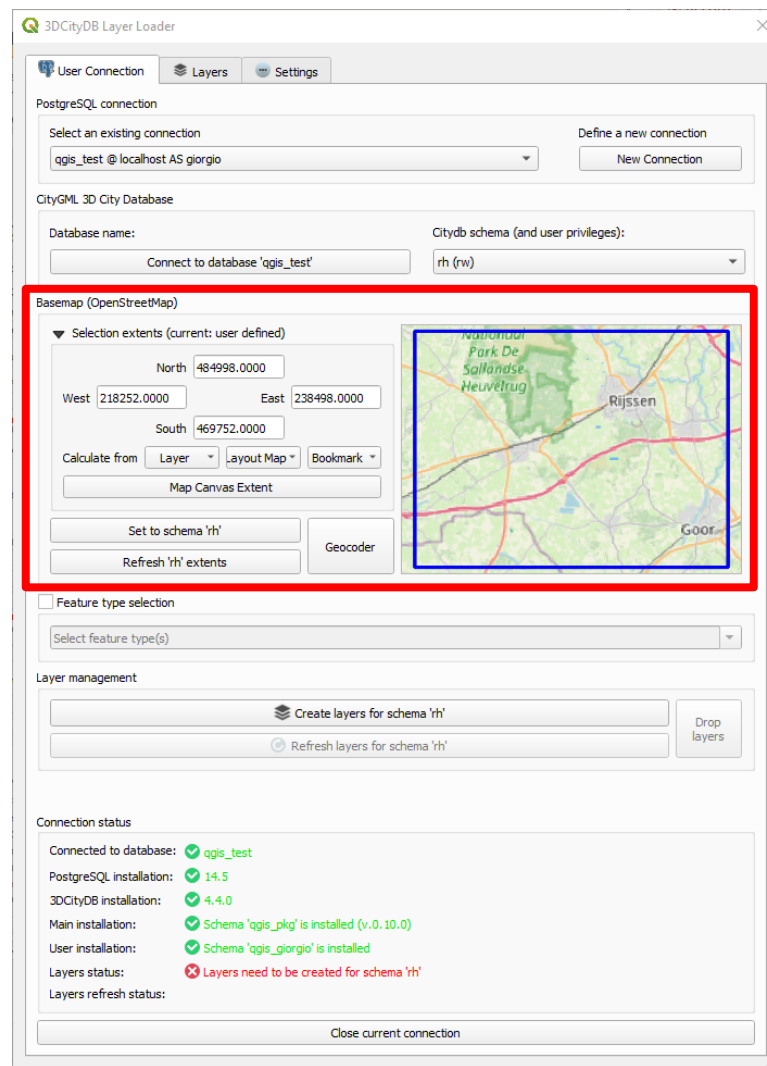
How to create additional citydb schemas
<https://3dcitydb-docs.readthedocs.io/en/latest/3dcitydb/multi-schema.html>



Layer Loader

4a) Upon selection of the citydb schema, you will see the extents of the dataset. They correspond to the extents of all currently loaded data in the selected citydb schema (here, for example, schema "rh")

Please note: the very first time you load a citydb schema, and depending on the size of the city model, it might take a while to load as the bounding boxes are being computed. From the second time onwards, it will load nearly instantly.

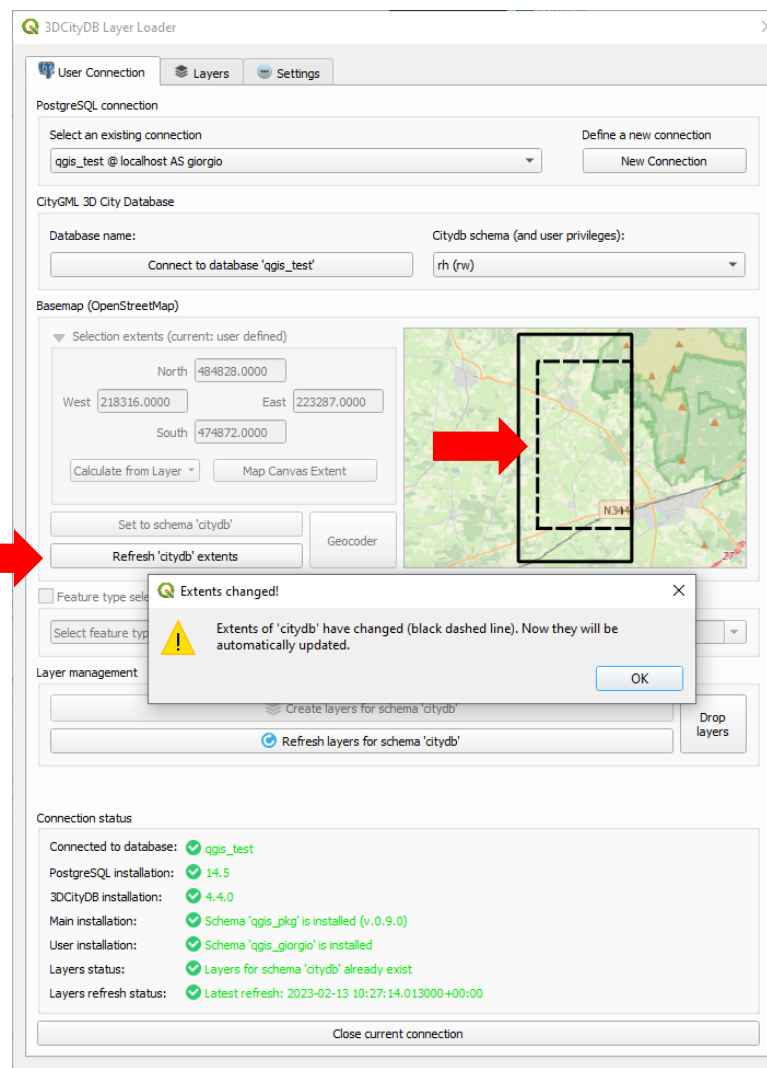


Layer Loader

4b) If data has been added or removed in the current citydb schema, you can refresh the extents by pressing the **Refresh {cdb_schema} extents button**.

The new extents will be temporarily shown with a **black dashed line**, before being updated.

Note bene: Depending on how the extents have changed, you may have to recreate, refresh and reload the layers in QGIS (see next slides)



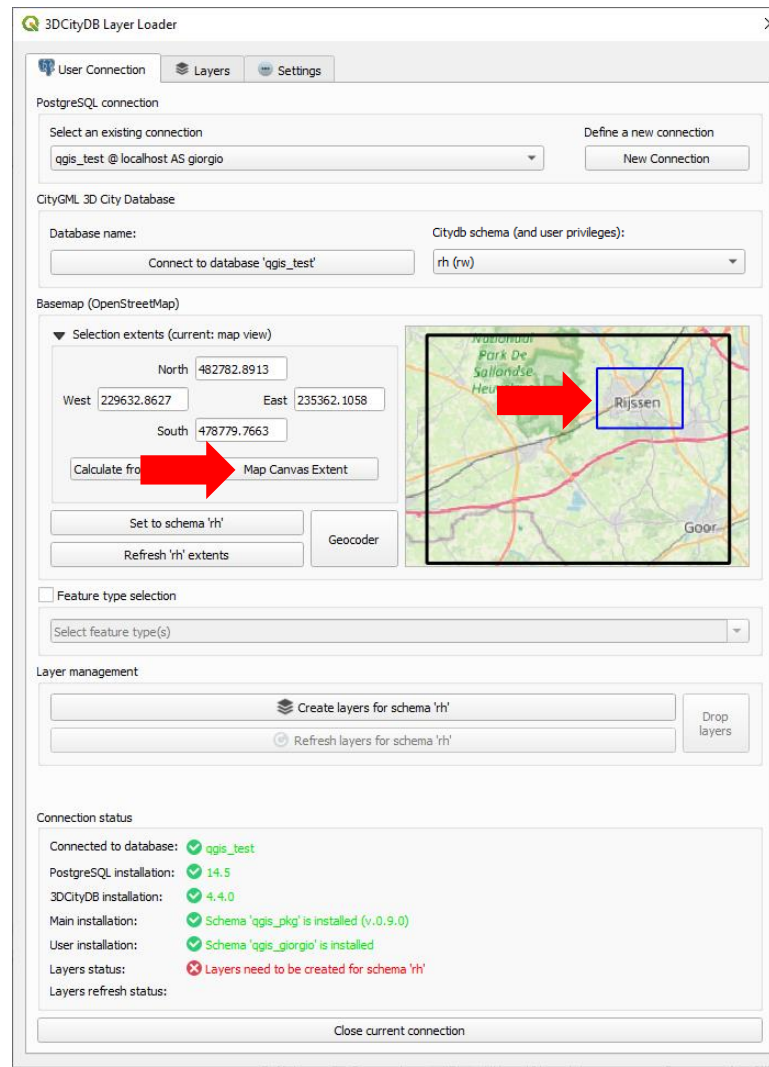
Layer Loader

5) Depending on your needs, you can set the spatial extents of the study area for which the layers will be created

- Default: same size of the whole dataset
- Otherwise: zoom in the map and choose your own area by clicking the **Map Canvas Extent** button. The **blue bounding box** shows the layers extents.

Beware: The bigger the size, the more time it will take to populate the layers!

Behind the scenes: In the database, materialised views of the geometries will be generated according to the selected extents. In case of very large cities, it might take a long time (and a lot of space on the server)!

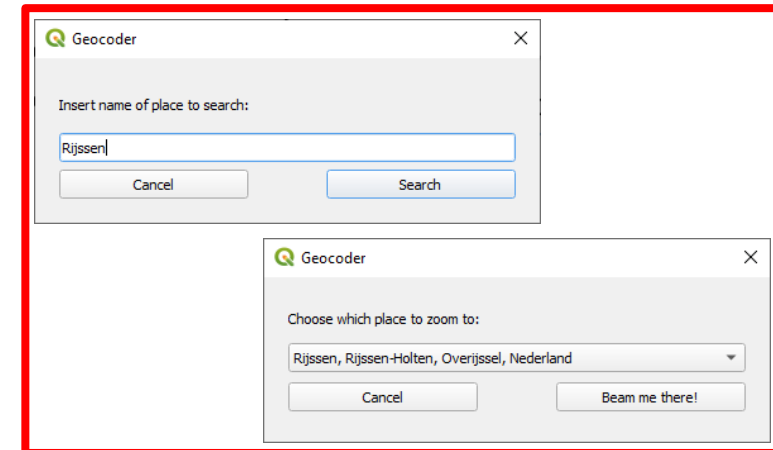
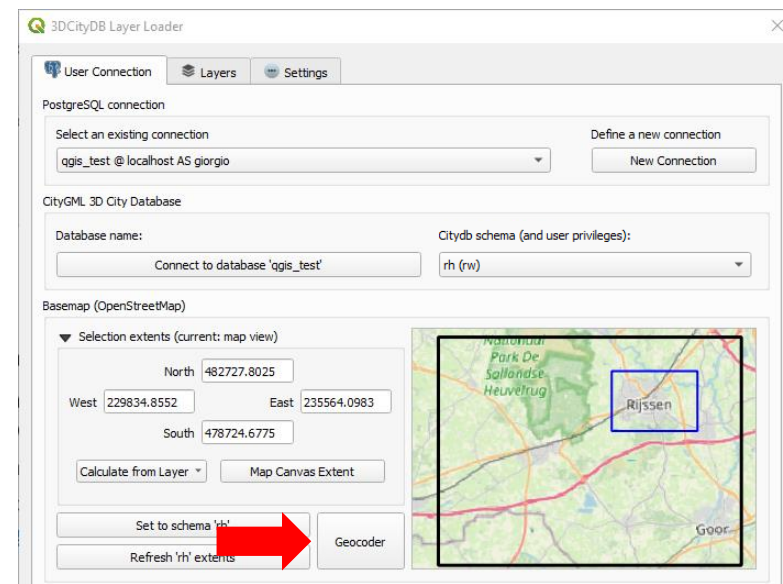


Layer Loader

5) Depending on your needs, you can set the spatial extents of the study area for which the layers will be created

- Default: same size of the whole dataset
- Otherwise: zoom in the map and choose your own area by clicking the **Map Canvas Extent** button. The **blue bounding box** shows the layers extents.

If you are looking for a specific place inside the citydb extents, you can also use the Geocoder that will zoom you directly there. Simply click the **Geocoder button**.



Layer Loader

Black: database schema extents (i.e. extents of the whole city model/dataset)

Blue: database-side layers extents (i.e. extents of the materialised views)

Motivation

Plugin overview

Software installation

Software use

- Layer Loader
- Bulk Deleter

Advanced options

Software uninstall

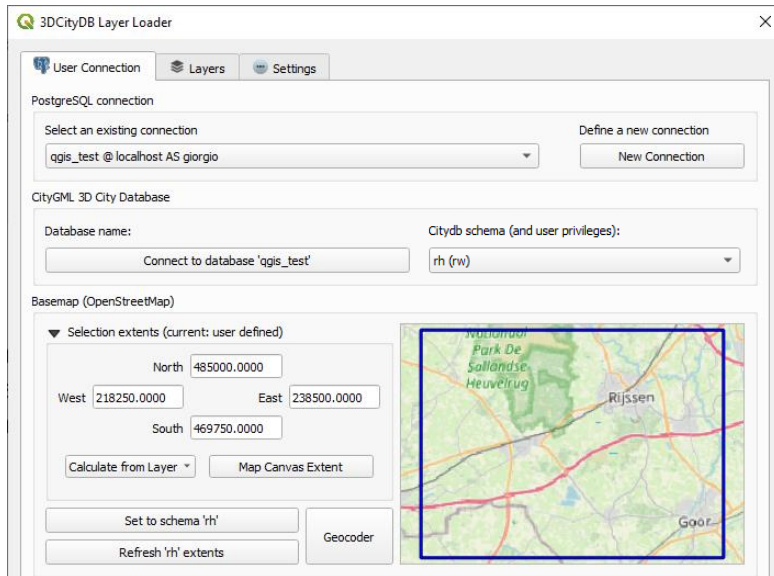
Current limitations

QGIS Package

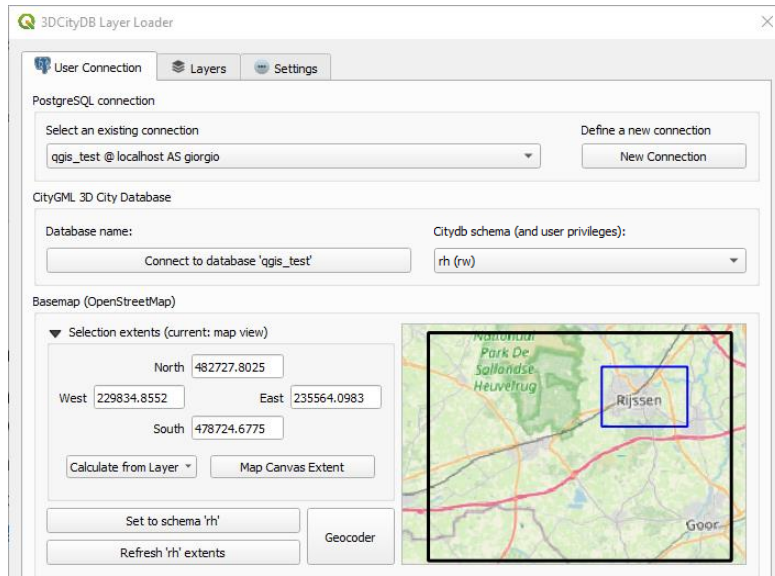
Resources

Changelog

Remember: **Layers extents** \subseteq **City model extents**



Both areas coincide (default)



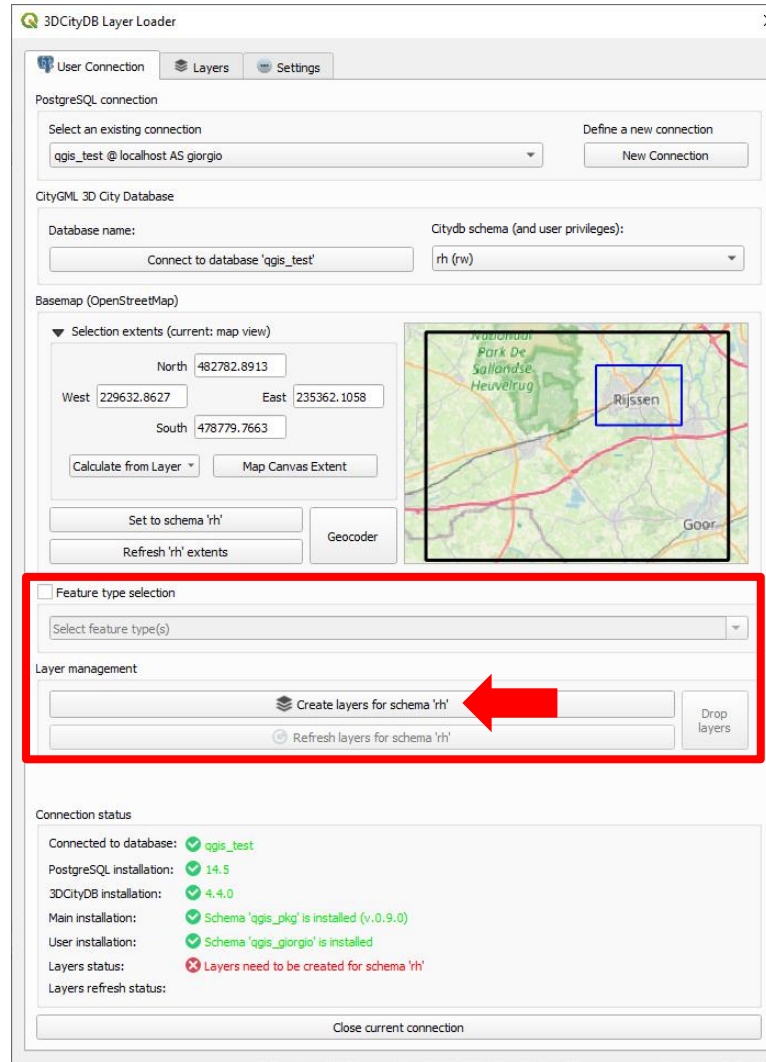
User-selected layers extents

Layer Loader

6a) Create the layers

Layers for all CityObjects available
withing the Layer extents will be created

Behind the scenes: In the database, (empty)
materialised views of the geometries and
views will be created as layers.
Only layers for *existing* data will be created.



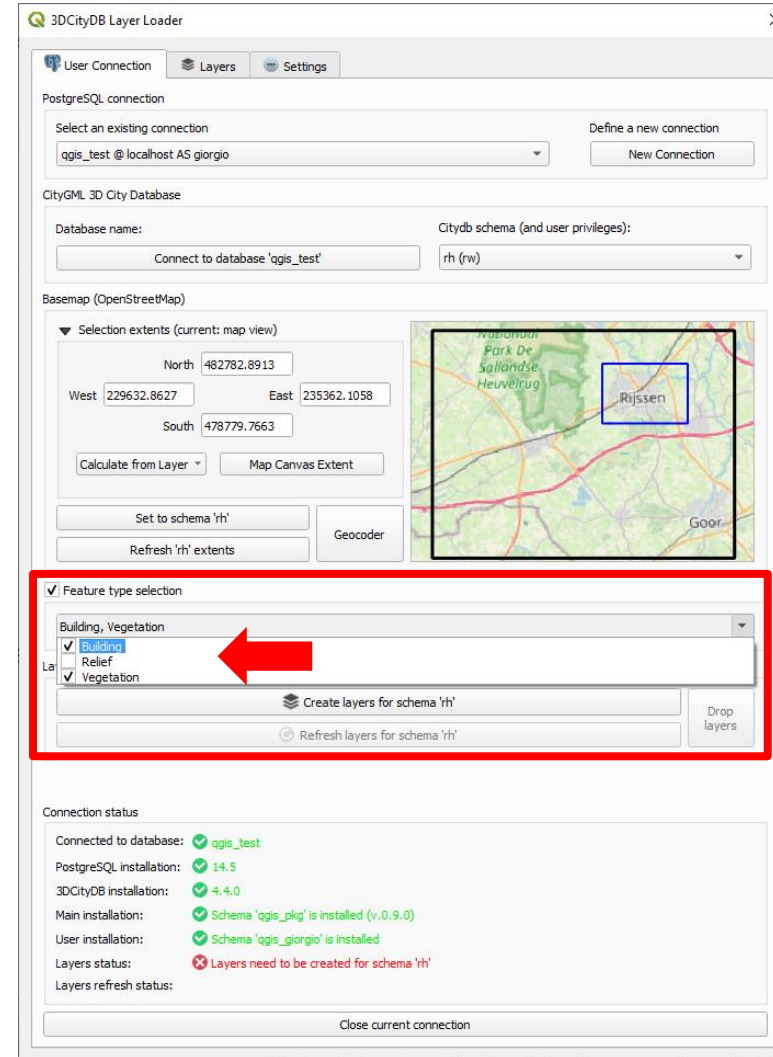
Layer Loader

6a) Create the layers

Layers for all CityObjects available withing the Layer extents will be created.

Optionally, you can further refine your selection and choose for which Feature Types the layers will be generated. Open the **Feature type selection box** and check the desired Feature types.

Note bene: Feature Types correspond the the CityGML modules (Building, Vegetation, Transportaton, LandUse, Relief, etc.)



Layer Loader

6b) Create the layers

Upon successful creation of the layers,
you will be notified in the **Connection**
Status box

Motivation

Plugin overview

Software installation

Software use

- Layer Loader
- Bulk Deleter

Advanced options

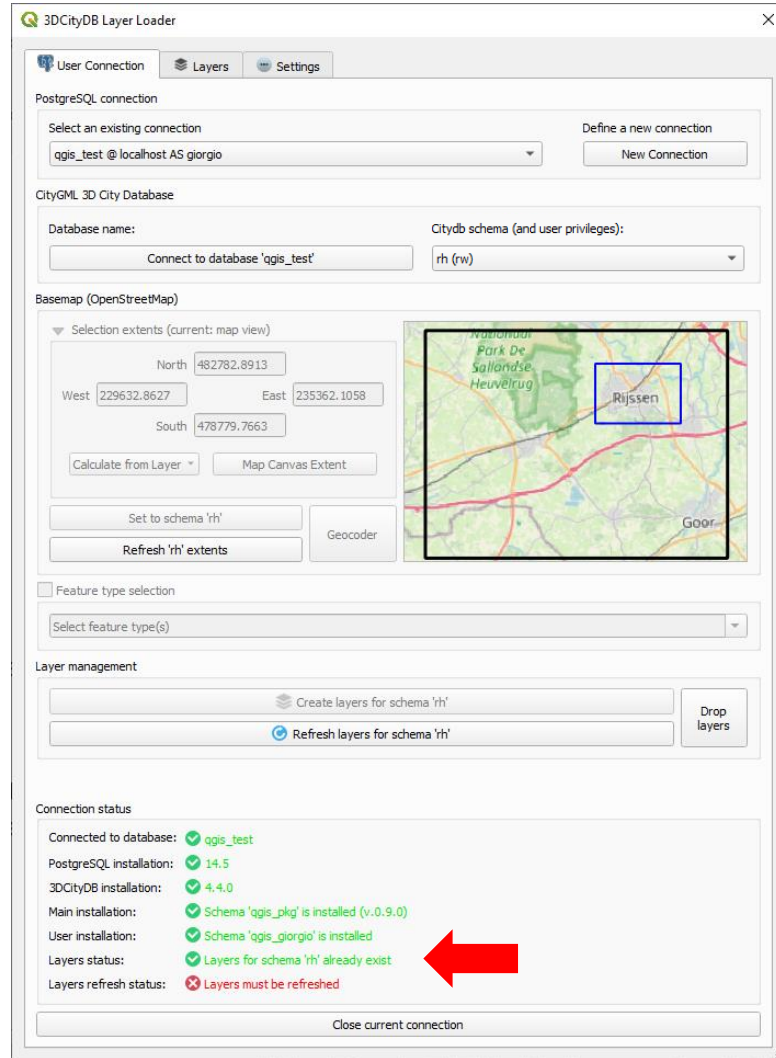
Software uninstall

Current limitations

QGIS Package

Resources

Changelog



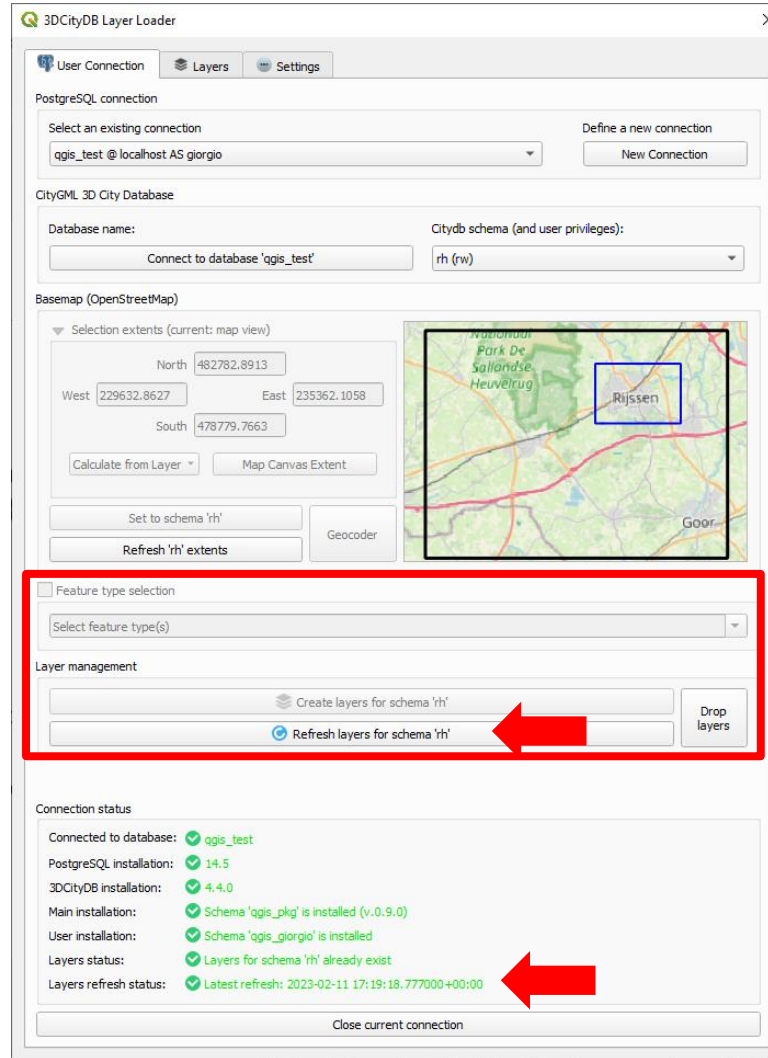
Layer Loader

Motivation
 Plugin overview
 Software installation
Software use
 • Layer Loader
 • Bulk Deleter
 Advanced options
 Software uninstall
 Current limitations
 QGIS Package
 Resources
 Changelog

7) Populate/refresh the layers
 Click on the **Refresh layers button**. Once the operation is complete, you will be notified in the **Connection status box**.
 The following "Layers" tab is now activated and you can open it.

Alternatively, layers created in a previous session may be used (and/or refreshed again), or dropped.

Beware: Depending on the size of the selected area and the amount of data in the city model, this operation might take long.



Motivation

Plugin overview

Software installation

Software use

- Layer Loader
- Bulk Deleter

Advanced options

Software uninstall

Current limitations

QGIS Package

Resources

Changelog

Behind the scenes:

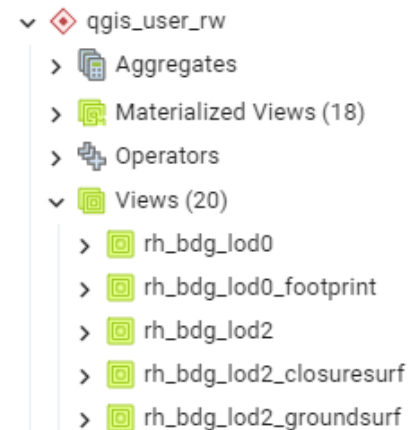
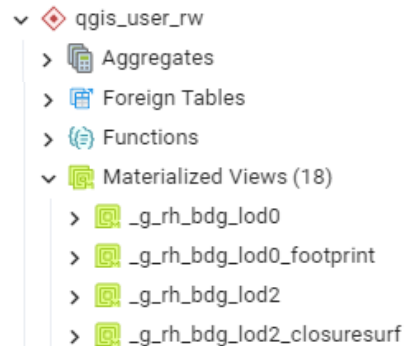
In the database user schema, both materialised views (containing the feature geometries) and the layers (as updatable views, containing the feature attributes linked to the corresponding geometries in the materialised views) can be accessed.

Materialised views name coding:

- "_g_" prefix + citydb schema name + feature name + lodx + (optional) semantic details

Views name coding:

- citydb schema name + feature name + lodx + (optional) semantic details
- Linked via column co_id (PK and FK to the materialised views)



Layer Loader

In the "Layers" tab

Motivation

Plugin overview

Software installation

Software use

- Layer Loader
- Bulk Deleter

Advanced options

Software uninstall

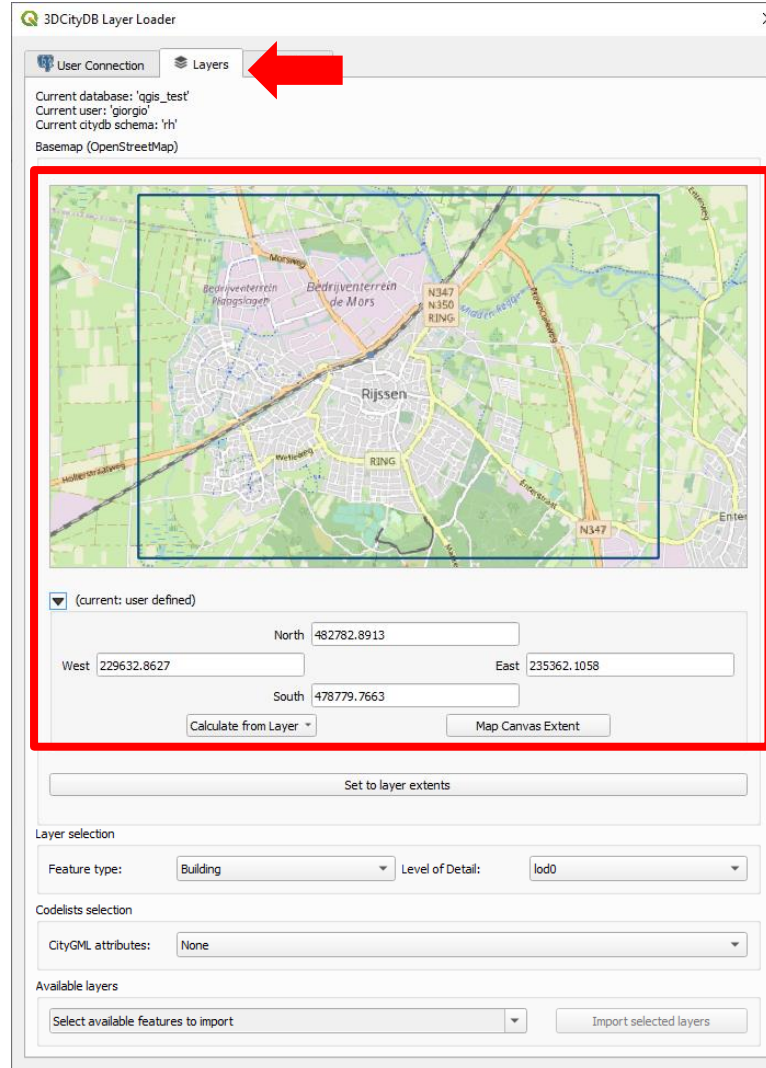
Current limitations

QGIS Package

Resources

Changelog

8) You are now directly zoomed to the layers extents (**blue bounding box**)



Layer Loader

Motivation

Plugin overview

Software installation

Software use

- Layer Loader
- Bulk Deleter

Advanced options

Software uninstall

Current limitations

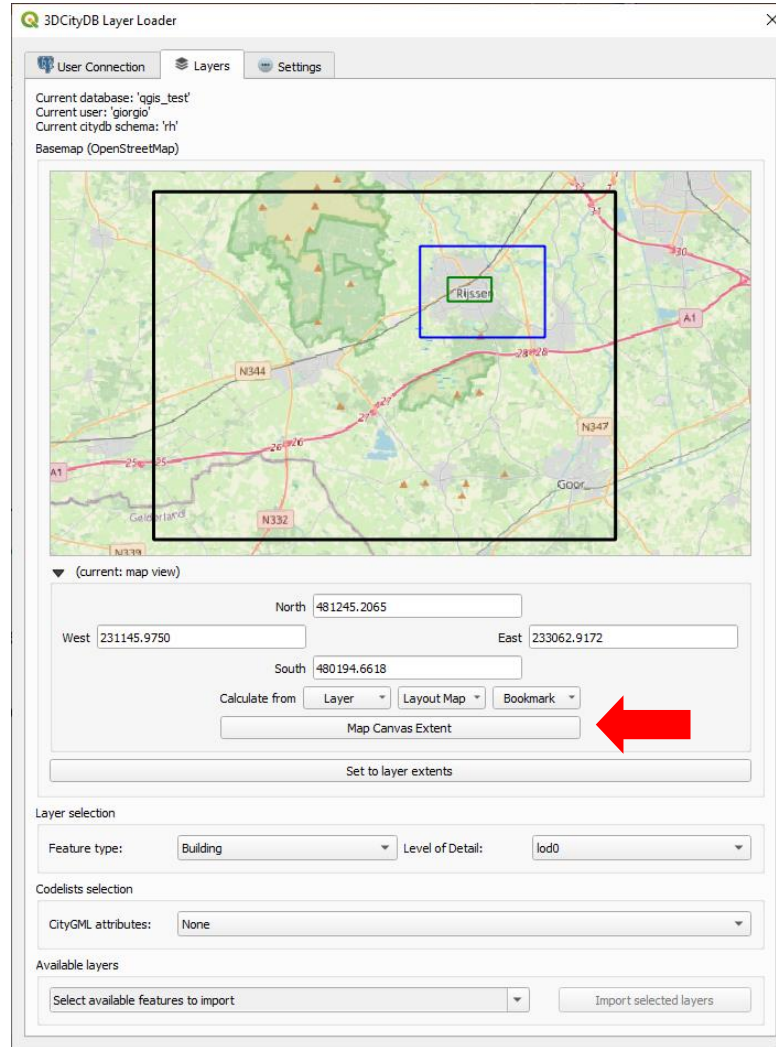
QGIS Package

Resources

Changelog

9) Depending on your needs, you can further reduce the extents of the layers to be loaded into QGIS by pressing the **Map Canvas Extent** button. The new extents are represented by the **green bounding box**.

- Default: same size of the layer extents (**blue bounding box**)
- Otherwise: zoom in and choose your own area



Layer Loader

Black: database schema extents (i.e. extents of the whole city model/dataset)

Blue: database-side layers extents (i.e. extents of the materialised views)

Green: QGIS-side layers extents (i.e. extents of the data loaded into QGIS)

Remember: **QGIS extents** \subseteq **Layers extents** \subseteq **City model extents**

Motivation

Plugin overview

Software installation

Software use

- Layer Loader
- Bulk Deleter

Advanced options

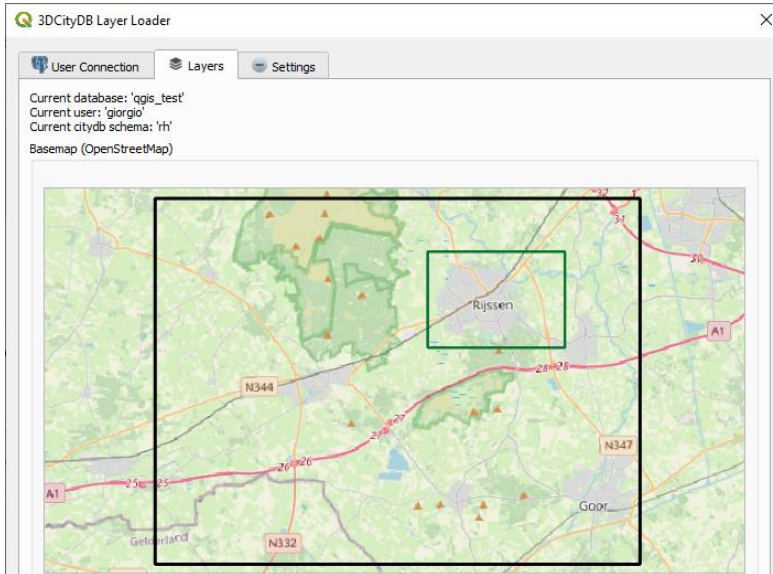
Software uninstall

Current limitations

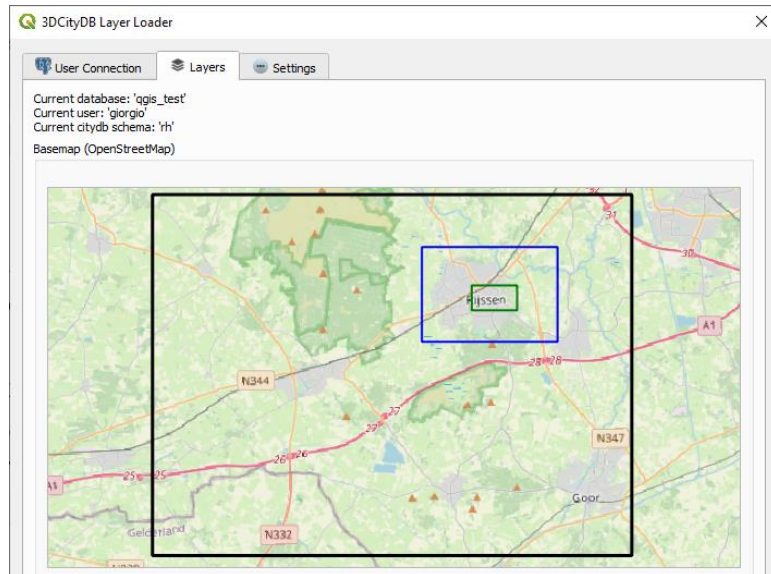
QGIS Package

Resources

Changelog



Database- and QGIS-side extents coincide (default)



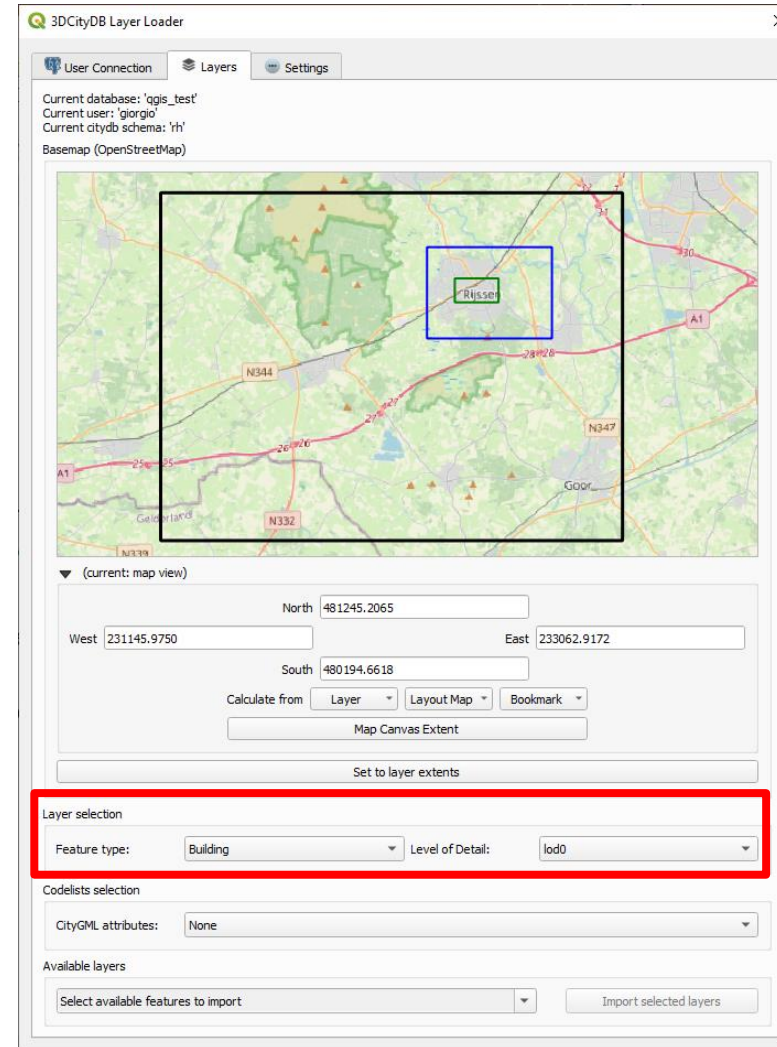
User-selected QGIS-layers extents

Layer Loader

10) Select available Feature type and LoD

- Layers are grouped according to the CityGML Feature Types (e.g. "Bridge", "Building", "Tunnel", "Relief", etc.)

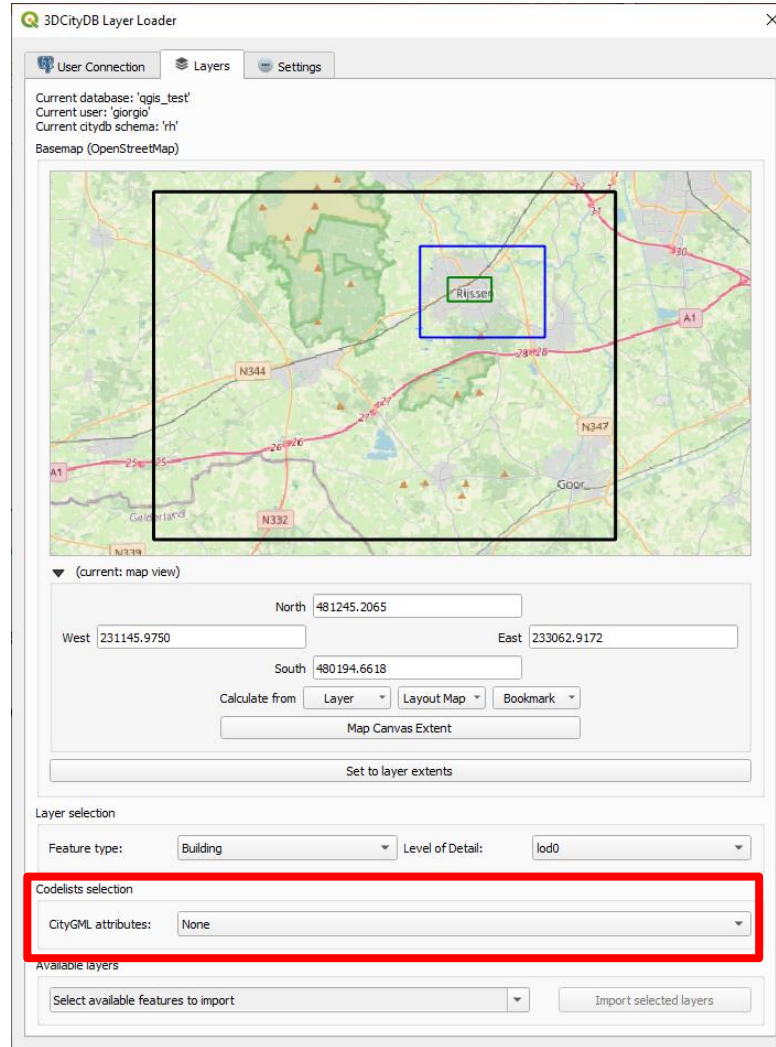
Behind the scenes: The plugin shows only the available Feature types and LoDs of data *within* the QGIS extents (**green bounding box**).



Layer Loader

11) The **Codelist selecton** allows to optionally load codelists and set up the attribute forms accordingly (see next slides about "Use in QGIS" for more details)

Motivation
 Plugin overview
 Software installation
Software use
 • Layer Loader
 • Bulk Deleter
 Advanced options
 Software uninstall
 Current limitations
 QGIS Package
 Resources
 Changelog

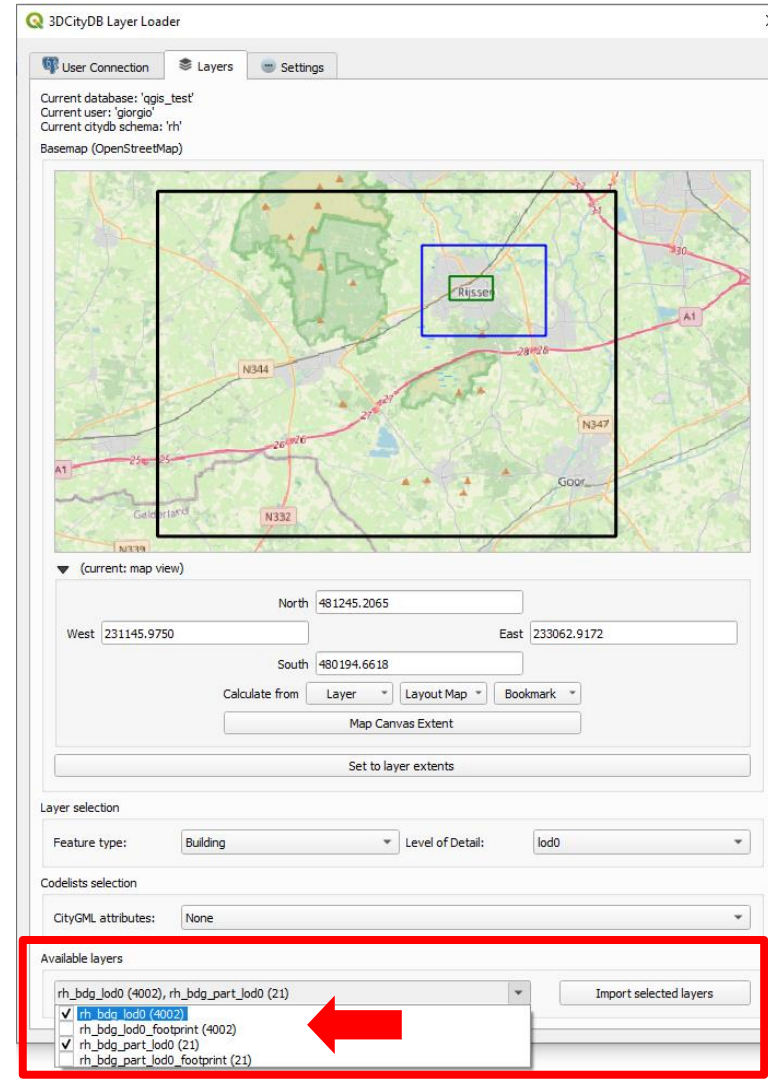


Layer Loader

12) Select the layer(s) to import into QGIS

- The number of available features is shown next to the layer name

Behind the scenes: The plugin shows only the available Layers *within* the QGIS extents (**green bounding box**).



Layer Loader

Motivation

Plugin overview

Software installation

Software use

- Layer Loader
- Bulk Deleter

Advanced options

Software uninstall

Current limitations

QGIS Package

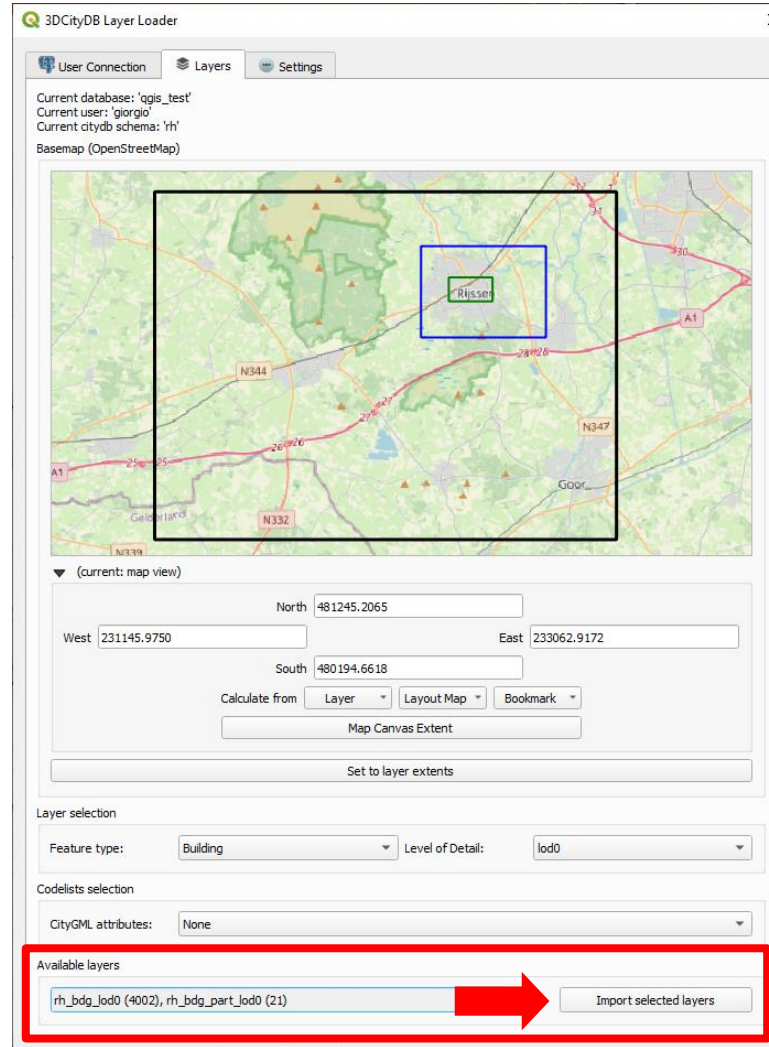
Resources

Changelog

13) Import the selected layers to QGIS by clicking on the **Import selected layers button**

- The import operation can be repeated with different layers
- The layers will be automatically added to the QGIS Layers Tree / Table of Contents
- The Plugin window can be closed, the connection parameters and settings will be kept until the connection is intentionally closed by the user (in the "User Connection" tab)

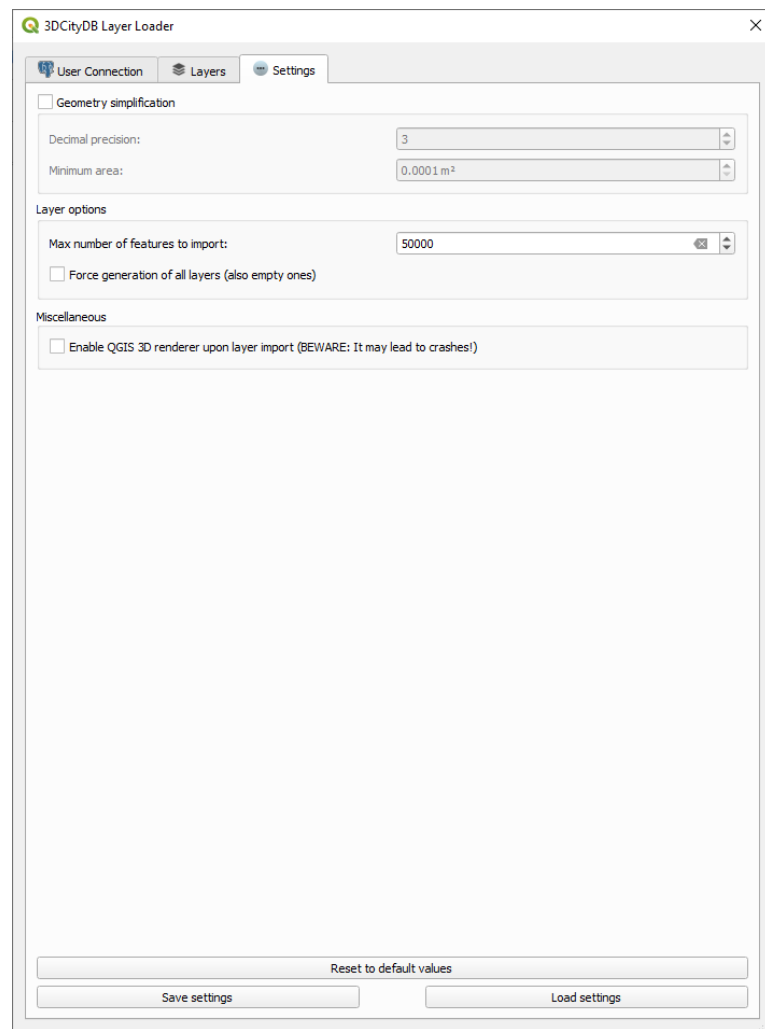
Nota bene: Layers that have already been loaded can be selected, but won't be loaded again



Layer Loader

The "Settings" tab allows to enable specific options.

- The **Geometry simplification box** contains details about the coordinates precisions and the minimum area of the geometries to be generated in the materialized views
- The **Layer options box** allows to set the maximum number of features to be imported in each import action and to force the generation of the empty layers
- In the **Miscellaneous box**, the user can force the 3D rendered to be enabled upon import of the selected layers, although this might lead to instabilities (see next slides)
- Settings can be saved, (re)loaded and reset to the default values.



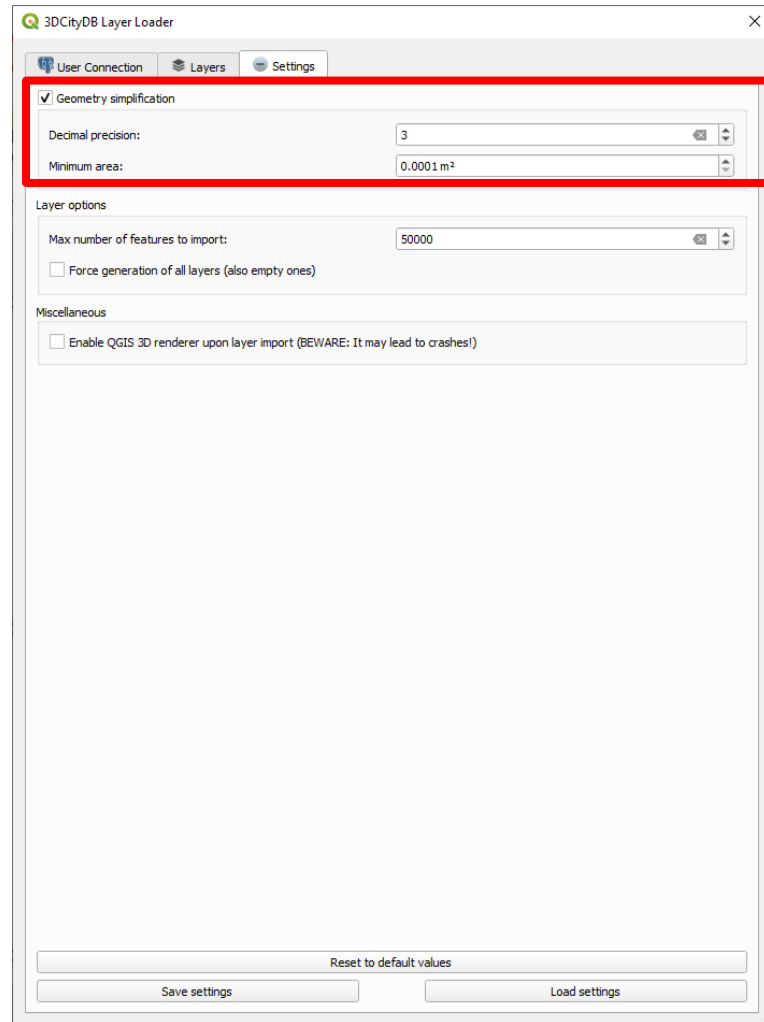
Geometry simplification

In order to simplify geometries and (partially) cope with the 3D visualisation issues of the 3D View Map in QGIS, the user can set some simplification parameters *before* generating the layers.

All polygons composing the geometries will be checked. The user can set the number of decimal positions in the coordinates. Resulting degenerate geometries are filtered out. The second parameter filters out all polygons smaller than the chosen threshold.

Beware! This operation:

- can significantly increase the time needed to refresh the layers
- does NOT change the original data in the database!



Use in QGIS: Layers

Motivation

Plugin overview

Software installation

Software use

- Layer Loader
- Bulk Deleter

Advanced options

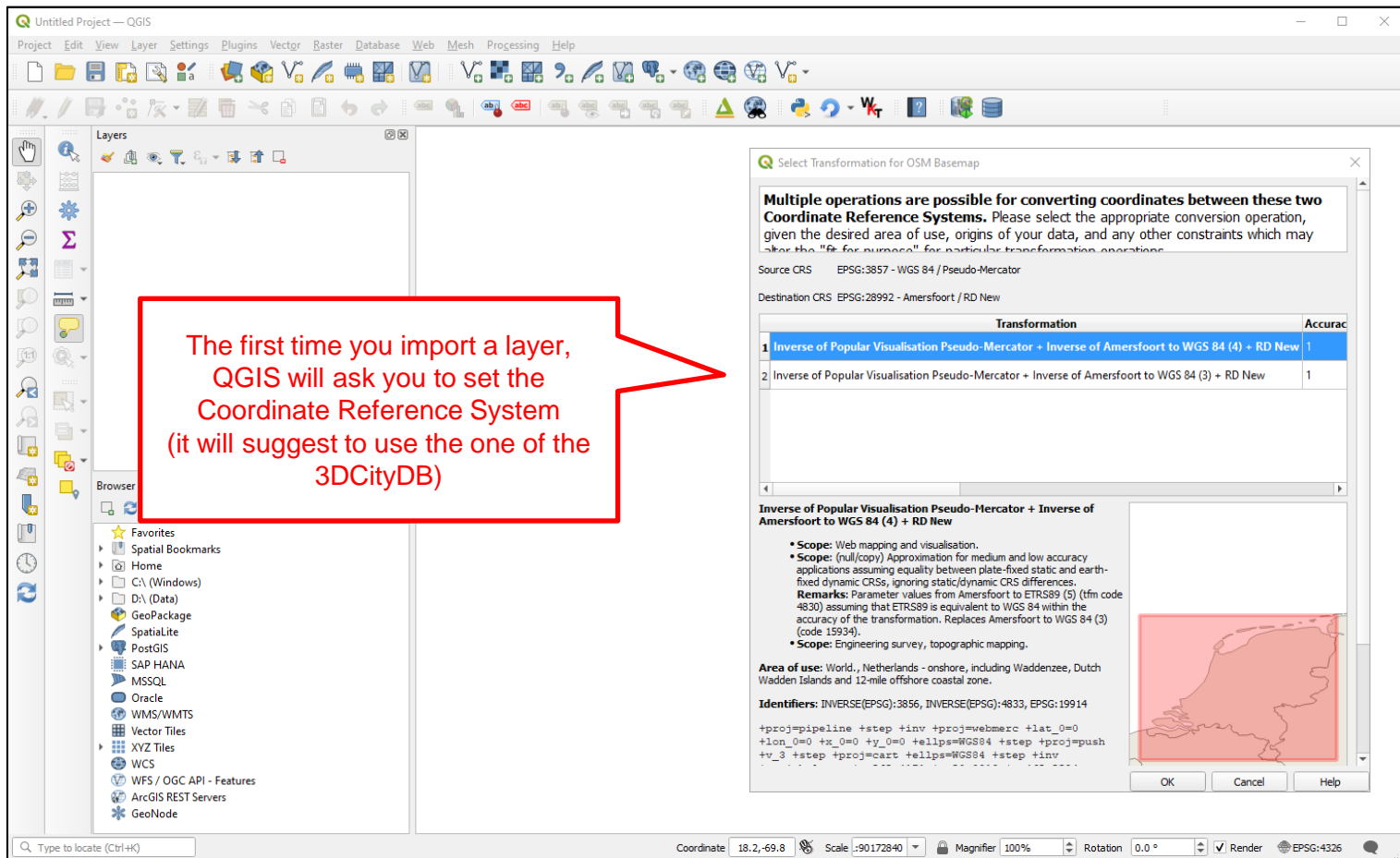
Software uninstall

Current limitations

QGIS Package

Resources

Changelog



Untitled Project — QGIS

Project Edit View Layer Settings Plugins Vector Raster Database Web Mesh Processing Help

Layers

Browser

Favorites

- Spatial Bookmarks
- Home
- C:\ (Windows)
- D:\ (Data)
- GeoPackage
- SpatiaLite
- PostGIS
- SAP HANA
- MSSQL
- Oracle
- WMS/WMTS
- Vector Tiles
- XYZ Tiles
- WCS
- WFS / OGC API - Features
- ArcGIS REST Servers
- GeoNode

Select Transformation for OSM Basemap

Multiple operations are possible for converting coordinates between these two **Coordinate Reference Systems**. Please select the appropriate conversion operation, given the desired area of use, origins of your data, and any other constraints which may alter the "fit for purpose" for particular transformation operations.

Source CRS EPSG:3857 - WGS 84 / Pseudo-Mercator

Destination CRS EPSG:28992 - Amersfoort / RD New

	Transformation	Accuracy
1	Inverse of Popular Visualisation Pseudo-Mercator + Inverse of Amersfoort to WGS 84 (4) + RD New	1
2	Inverse of Popular Visualisation Pseudo-Mercator + Inverse of Amersfoort to WGS 84 (3) + RD New	1

Inverse of Popular Visualisation Pseudo-Mercator + Inverse of Amersfoort to WGS 84 (4) + RD New

- **Scope:** Web mapping and visualisation.
- **Scope:** (null/copy) Approximation for medium and low accuracy applications assuming equality between plate-fixed static and earth-fixed dynamic CRSs, ignoring static/dynamic CRS differences.
- Remarks:** Parameter values from Amersfoort to ETRS89 (5) (tfm code 4830) assuming that ETRS89 is equivalent to WGS 84 within the accuracy of the transformation. Replaces Amersfoort to WGS 84 (3) (code 15934).
- **Scope:** Engineering survey, topographic mapping.

Area of use: World, Netherlands - onshore, including Waddenzee, Dutch Wadden Islands and 12-mile offshore coastal zone.

Identifiers: INVERSE(EPSG):3856, INVERSE(EPSG):4833, EPSG:19914

+proj=pipeline +step +inv +proj=webmerc +lat_0=0
+lon_0=0 +x_0=0 +y_0=0 +ellps=WGS84 +step +proj=push
+v_3 +step +proj=cart +ellps=WGS84 +step +inv

OK Cancel Help

Coordinate 18.2, -69.8 Scale : 90172840 Magnifier 100% Rotation 0.0 ° Render EPSG:4326

Use in QGIS: Layers

Motivation

Plugin overview

Software installation

Software use

- Layer Loader
- Bulk Deleter

Advanced options

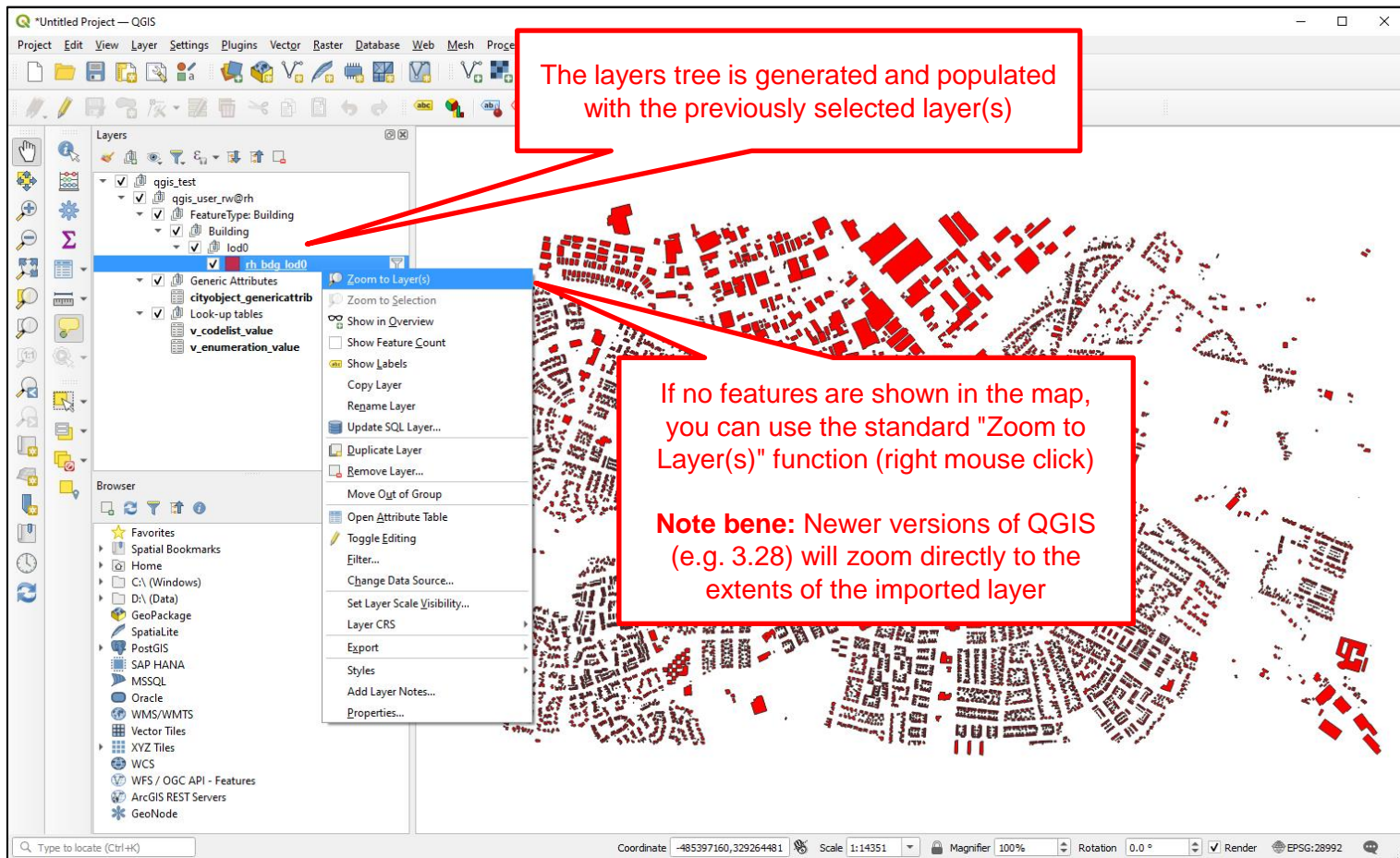
Software uninstall

Current limitations

QGIS Package

Resources

Changelog



Use in QGIS: Layers

Motivation

Plugin overview

Software installation

Software use

- Layer Loader
- Bulk Deleter

Advanced options

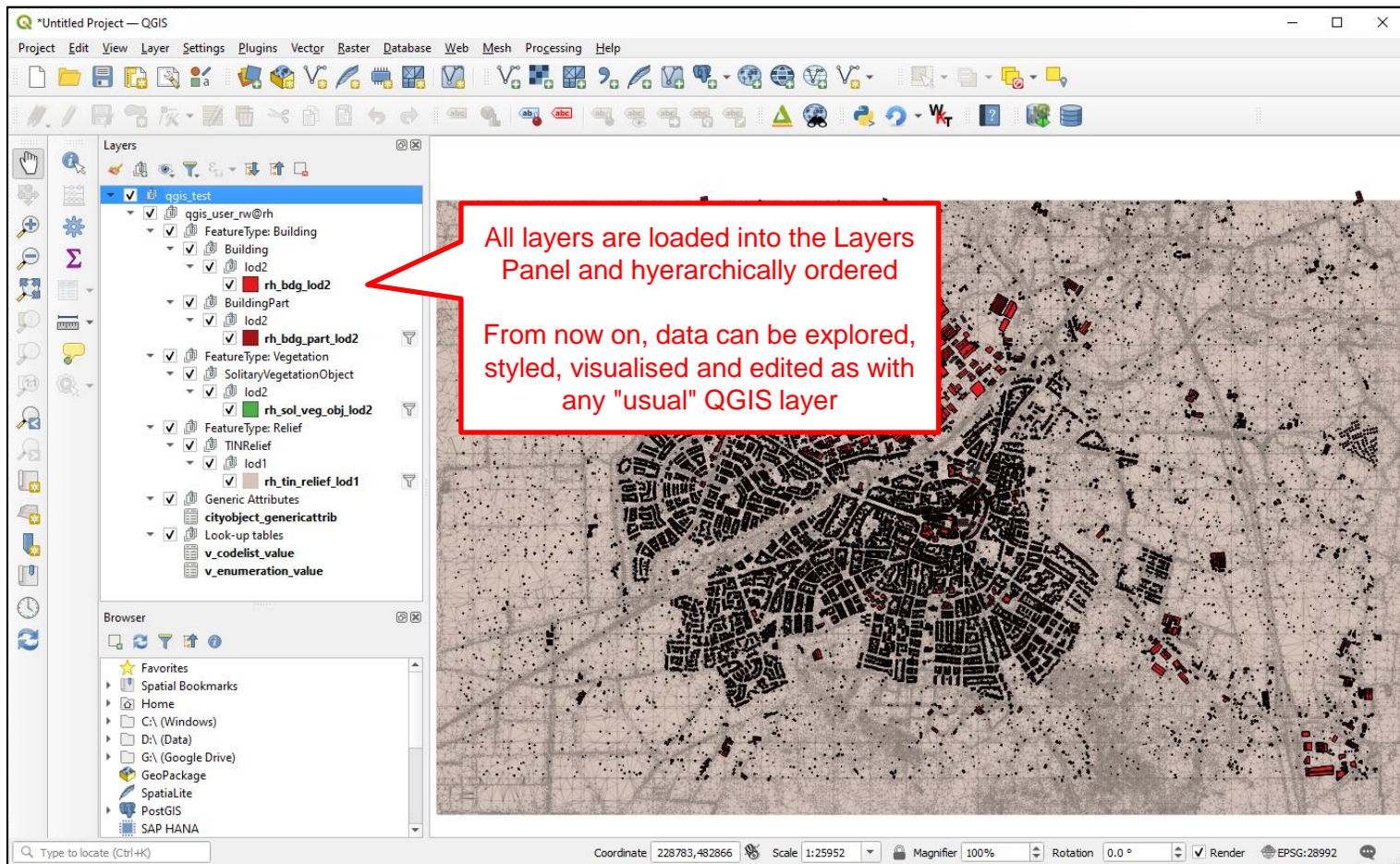
Software uninstall

Current limitations

QGIS Package

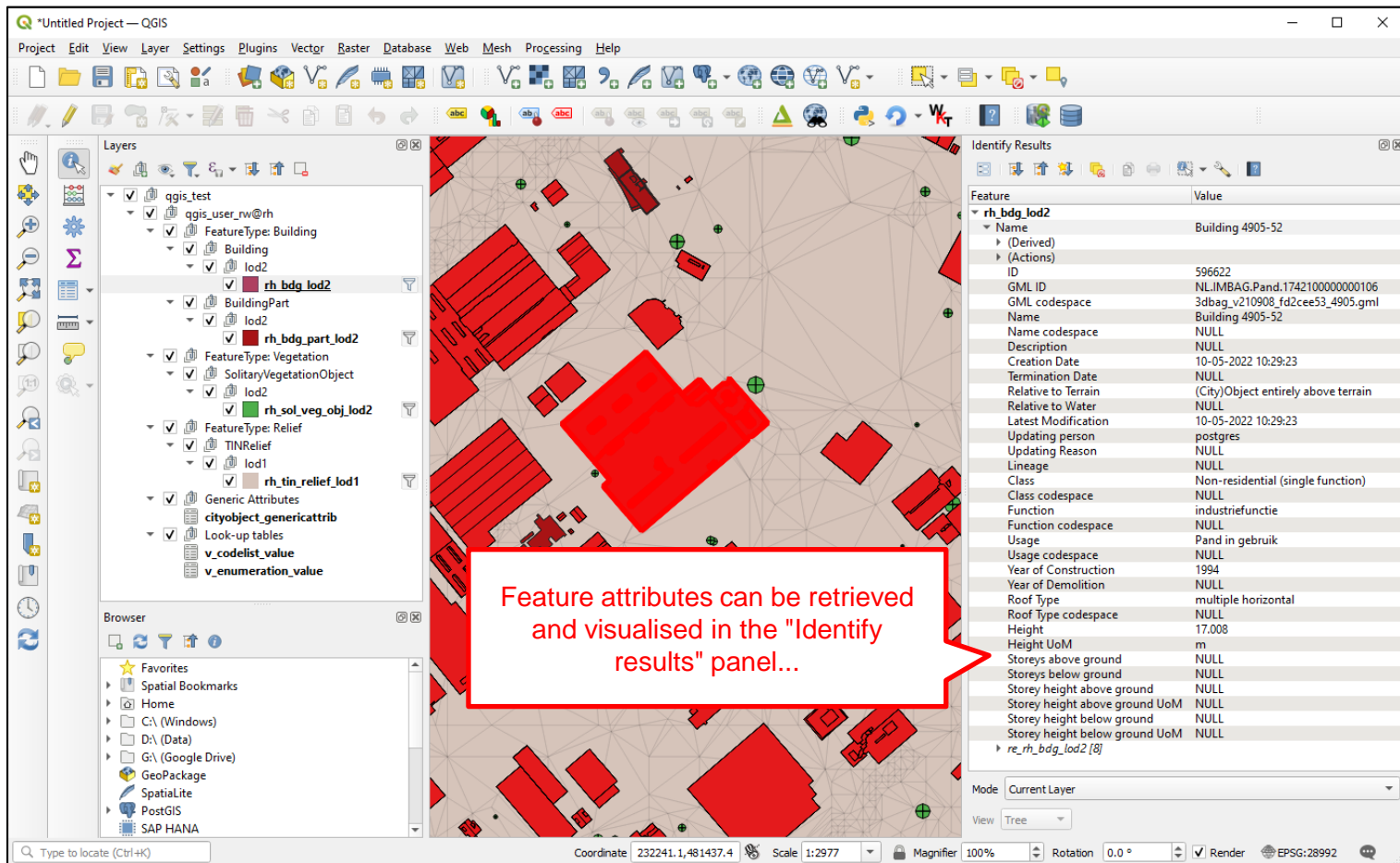
Resources

Changelog



Use in QGIS: Attributes

Motivation
 Plugin overview
 Software installation
Software use
 • Layer Loader
 • Bulk Deleter
 Advanced options
 Software uninstall
 Current limitations
 QGIS Package
 Resources
 Changelog

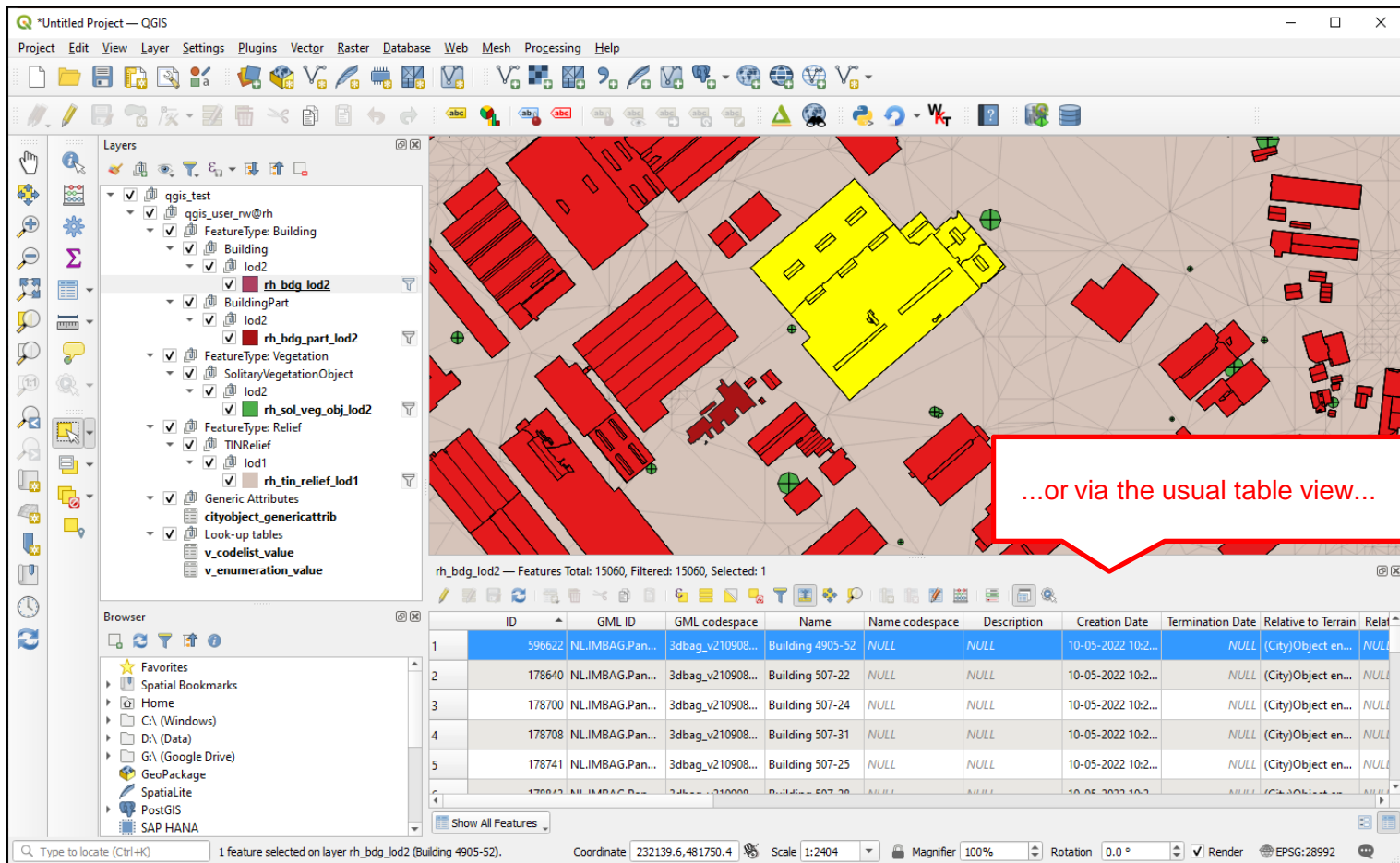


The screenshot shows the QGIS interface with a 3D map view. The 'Layers' panel on the left shows a tree structure of layers, including 'qgis_test', 'qgis_user_nw@rh', 'Building', 'lod2', 'rh_bdg_lod2', 'BuildingPart', 'lod2', 'rh_bdg_part_lod2', 'FeatureType: Vegetation', 'SolitaryVegetationObject', 'lod2', 'rh_sol_veg_obj_lod2', 'FeatureType: Relief', 'TINRelief', 'lod1', 'rh_tin_relief_lod1', 'Generic Attributes', 'cityobject_genericattrib', 'Look-up tables', 'v_codelist_value', and 'v_enumeration_value'. The 'Identify Results' panel on the right shows a table of attributes for the selected feature 'rh_bdg_lod2'. A red callout box points to the 'Identify Results' panel with the text: 'Feature attributes can be retrieved and visualised in the "Identify results" panel...'

Feature	Value
rh_bdg_lod2	
Name	Building 4905-52
(Derived)	
(Actions)	
ID	596622
GML ID	NLIMBAG.Pand.174210000000106
GML codespace	3dbag_v210908_fd2cee53_4905.gml
Name	Building 4905-52
Name codespace	NULL
Description	NULL
Creation Date	10-05-2022 10:29:23
Termination Date	NULL
Relative to Terrain	(City)Object entirely above terrain
Relative to Water	NULL
Latest Modification	10-05-2022 10:29:23
Updating person	postgres
Updating Reason	NULL
Lineage	NULL
Class	Non-residential (single function)
Class codespace	NULL
Function	industriefunctie
Function codespace	NULL
Usage	Pand in gebruik
Usage codespace	NULL
Year of Construction	1994
Year of Demolition	NULL
Roof Type	multiple horizontal
Roof Type codespace	NULL
Height	17.008
Height UoM	m
Storeys above ground	NULL
Storeys below ground	NULL
Storey height above ground	NULL
Storey height above ground UoM	NULL
Storey height below ground	NULL
Storey height below ground UoM	NULL
re_rh_bdg_lod2 [8]	

Use in QGIS: Attributes

Motivation
 Plugin overview
 Software installation
Software use
 • Layer Loader
 • Bulk Deleter
 Advanced options
 Software uninstall
 Current limitations
 QGIS Package
 Resources
 Changelog



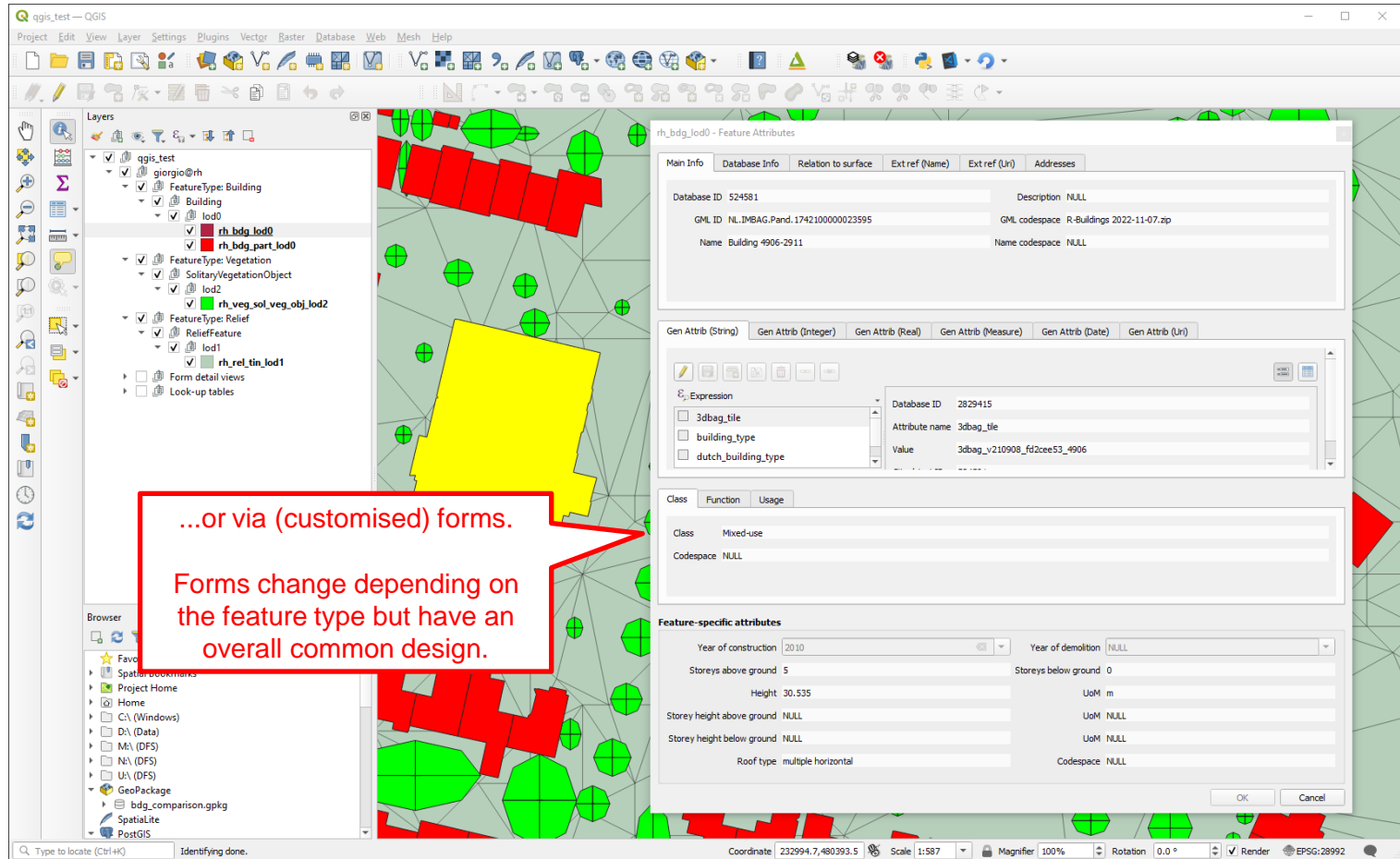
The screenshot shows the QGIS interface with a 3D map view of buildings. A red callout box points to the table view, which displays the attributes of the selected building (Building 4905-52).

...or via the usual table view...

ID	GML ID	GML codespace	Name	Name codespace	Description	Creation Date	Termination Date	Relative to Terrain	Relat
1	596622	NL.IMBAG.Pan...	Building 4905-52	NULL	NULL	10-05-2022 10:2...	NULL	(City)Object en...	NULL
2	178640	NL.IMBAG.Pan...	Building 507-22	NULL	NULL	10-05-2022 10:2...	NULL	(City)Object en...	NULL
3	178700	NL.IMBAG.Pan...	Building 507-24	NULL	NULL	10-05-2022 10:2...	NULL	(City)Object en...	NULL
4	178708	NL.IMBAG.Pan...	Building 507-31	NULL	NULL	10-05-2022 10:2...	NULL	(City)Object en...	NULL
5	178741	NL.IMBAG.Pan...	Building 507-25	NULL	NULL	10-05-2022 10:2...	NULL	(City)Object en...	NULL

Use in QGIS: Attributes

Motivation
 Plugin overview
 Software installation
Software use
 • Layer Loader
 • Bulk Deleter
 Advanced options
 Software uninstall
 Current limitations
 QGIS Package
 Resources
 Changelog



...or via (customised) forms.
 Forms change depending on the feature type but have an overall common design.

Use in QGIS: Attributes

Attributes are grouped into tabs.

rh_bdg_lod0 - Feature Attributes

Main Info Database Info Relation to surface Ext ref (Name) Ext ref (Uri) Addresses

Database ID 524581 Description NULL

GML ID NL.IMBAG.Pand.1742100000023595 GML codespace R-Buildings 2022-11-07.zip

Name Building 4906-2911 Name codespace NULL

Gen Attrib (String) Gen Attrib (Integer) Gen Attrib (Real) Gen Attrib (Measure) Gen Attrib (Date) Gen Attrib (Uri)

Expression

☐ 3dbag_tile

☐ building_type

☐ dutch_building_type

Database ID 2829415

Attribute name 3dbag_tile

Value 3dbag_v210908_fd2cee53_4906

Class Function Usage

Class Mixed-use

Codespace NULL

Feature-specific attributes

Year of construction 2010 Year of demolition NULL

Storeys above ground 5 Storeys below ground 0

Height 30.535 UoM m

Storey height above ground NULL UoM NULL

Storey height below ground NULL UoM NULL

Roof type multiple horizontal Codespace NULL

OK Cancel

Motivation

Plugin overview

Software installation

Software use

- Layer Loader
- Bulk Deleter

Advanced options

Software uninstall

Current limitations

QGIS Package

Resources

Changelog

Use in QGIS: Attributes

Additional related tables are connected (e.g. External References, Addresses, Generic Attributes)

rh_bdg_lod0 - Feature Attributes

Main Info Database Info Relation to surface Ext ref (Name) Ext ref (Uri) Addresses

Database ID 524581 Description NULL

GML ID NL.IMBAG.Pand.1742100000023595 GML codespace R-Buildings 2022-11-07.zip

Name Building 4906-2911 Name codespace NULL

Gen Attrib (String) Gen Attrib (Integer) Gen Attrib (Real) Gen Attrib (Measure) Gen Attrib (Date) Gen Attrib (Uri)

Expression

☐ 3dbag_tile

☐ building_type

☐ dutch_building_type

Database ID 2829415

Attribute name 3dbag_tile

Value 3dbag_v210908_fd2cee53_4906

Class Function Usage

Class Mixed-use

Codespace NULL

Feature-specific attributes

Year of construction 2010 Year of demolition NULL

Storeys above ground 5 Storeys below ground 0

Height 30.535 UoM m

Storey height above ground NULL UoM NULL

Storey height below ground NULL UoM NULL

Roof type multiple horizontal Codespace NULL

OK Cancel

Use in QGIS: Attributes

Motivation

Plugin overview

Software installation

Software use

- Layer Loader
- Bulk Deleter

Advanced options

Software uninstall

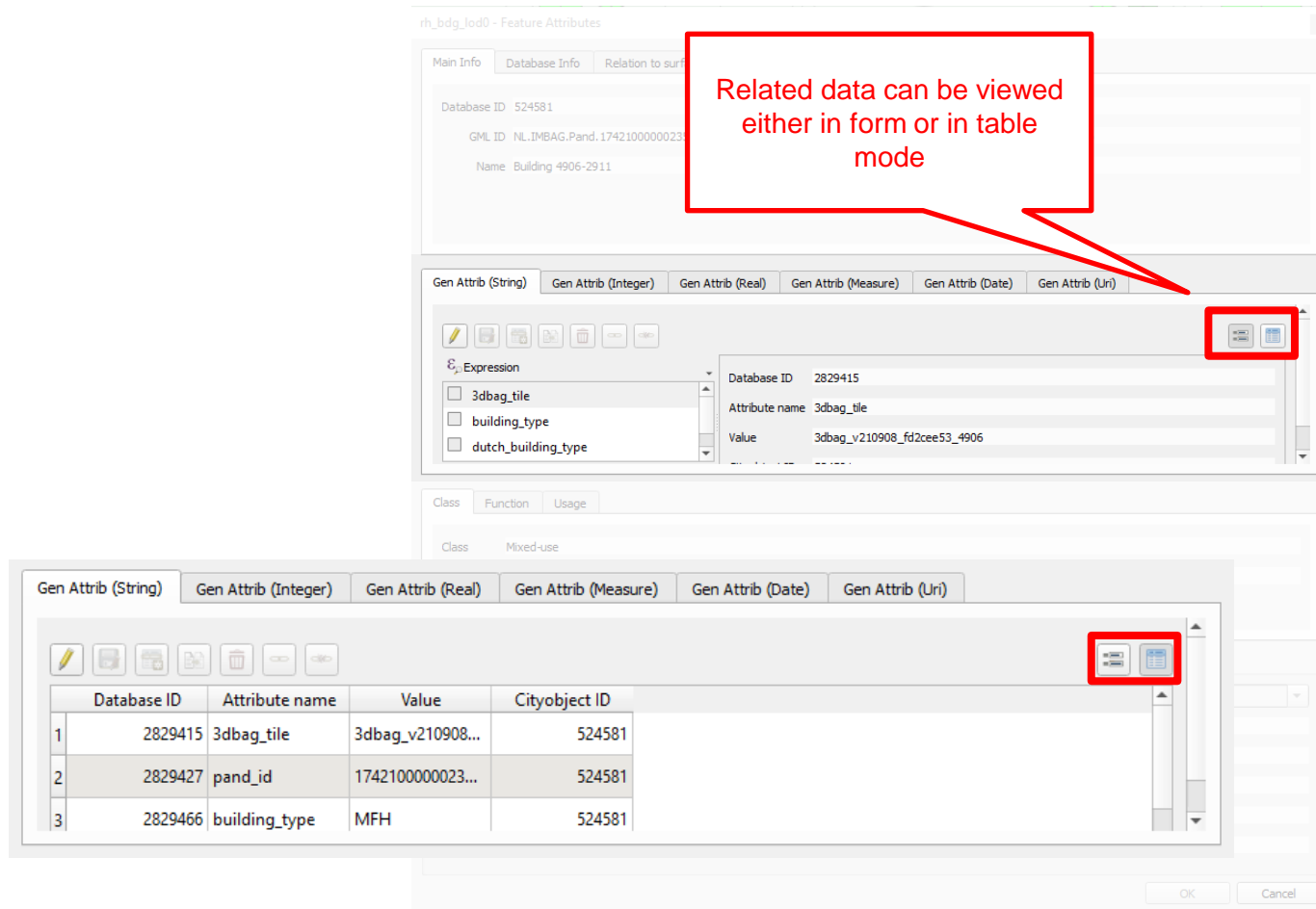
Current limitations

QGIS Package

Resources

Changelog

Related data can be viewed either in form or in table mode



rh_bdg_lod0 - Feature Attributes

Main Info Database Info Relation to surf

Database ID 524581

GML ID NL.IMBAG.Pand.1742100000023

Name Building 4906-2911

Gen Attrib (String) Gen Attrib (Integer) Gen Attrib (Real) Gen Attrib (Measure) Gen Attrib (Date) Gen Attrib (Uri)

Expression

☐ 3dbag_tile

☐ building_type

☐ dutch_building_type

Database ID 2829415

Attribute name 3dbag_tile

Value 3dbag_v210908_fd2cee53_4906

Class Function Usage

Class Mixed-use

Gen Attrib (String) Gen Attrib (Integer) Gen Attrib (Real) Gen Attrib (Measure) Gen Attrib (Date) Gen Attrib (Uri)

	Database ID	Attribute name	Value	Cityobject ID
1	2829415	3dbag_tile	3dbag_v210908...	524581
2	2829427	pand_id	1742100000023...	524581
3	2829466	building_type	MFH	524581

Use in QGIS: Attributes

Motivation

Plugin overview

Software installation

Software use

- Layer Loader
- Bulk Deleter

Advanced options

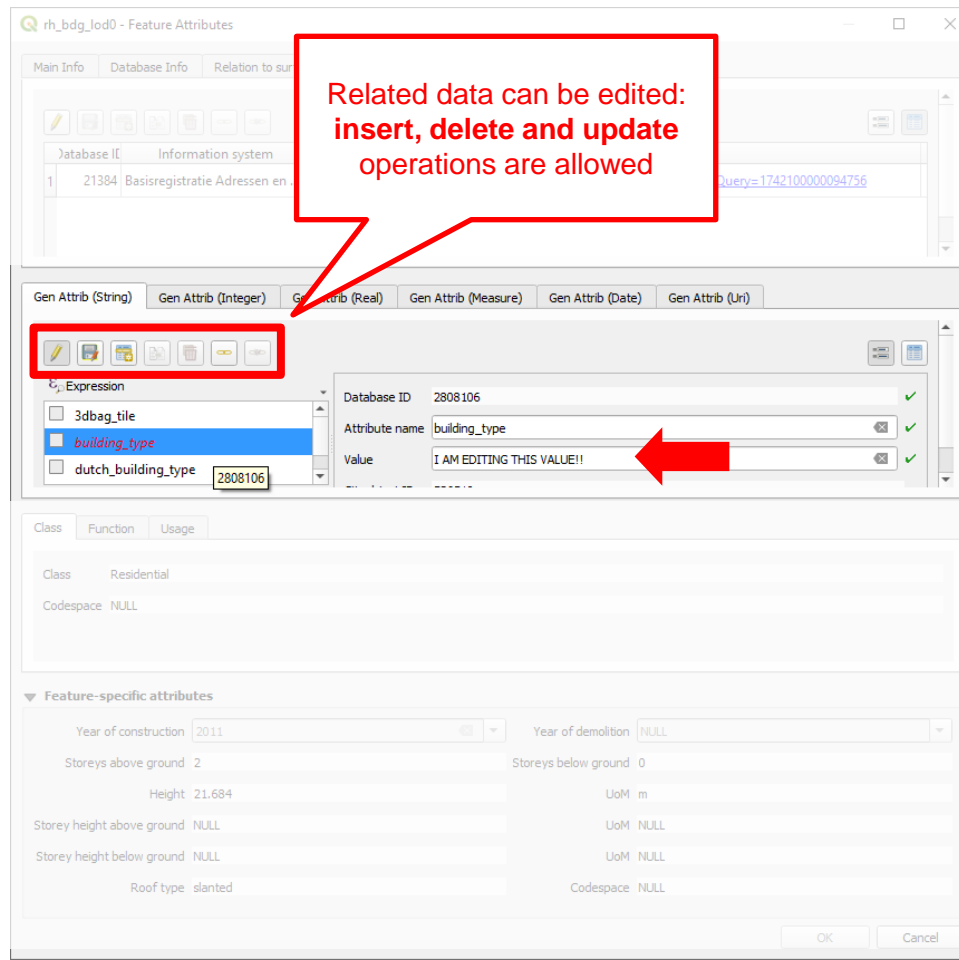
Software uninstall

Current limitations

QGIS Package

Resources

Changelog



Use in QGIS: Attributes

Motivation

Plugin overview

Software installation

Software use

- Layer Loader
- Bulk Deleter

Advanced options

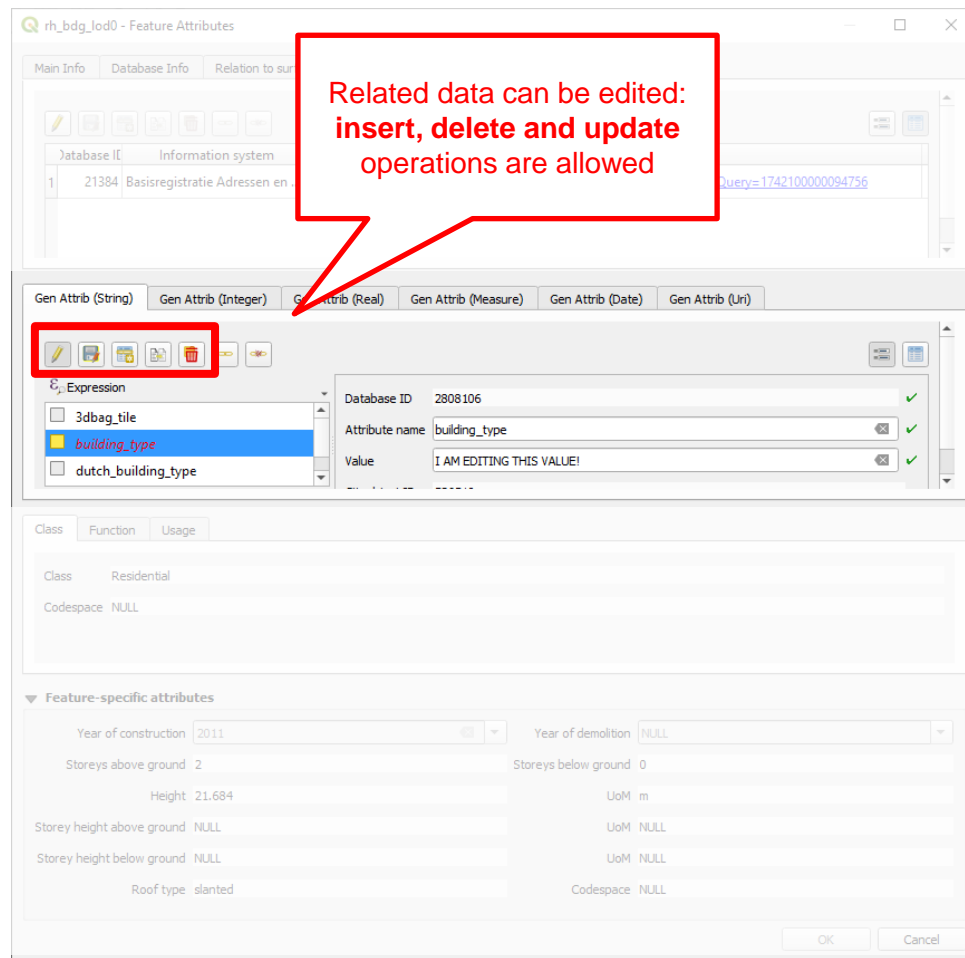
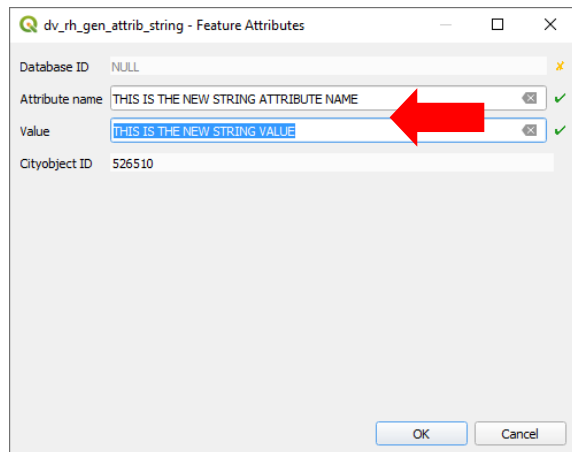
Software uninstall

Current limitations

QGIS Package

Resources

Changelog



Use in QGIS: Attributes

Motivation

Plugin overview

Software installation

Software use

- Layer Loader
- Bulk Deleter

Advanced options

Software uninstall

Current limitations

QGIS Package

Resources

Changelog

Feature-specific attributes
are grouped in the lower
part of the form.

rh_bdg_lod0 - Feature Attributes

Main Info Database Info Relation to surface Ext ref (Name) Ext ref (Uri) Addresses

Database ID 524581 Description NULL

GML ID NL.IMBAG.Pand.1742100000023595 GML codespace R-Buildings 2022-11-07.zip

Name Building 4906-2911 Name codespace NULL

Gen Attrib (String) Gen Attrib (Integer) Gen Attrib (Real) Gen Attrib (Measure) Gen Attrib (Date) Gen Attrib (Uri)

Expression

☐ 3dbag_tile

☐ building_type

☐ dutch_building_type

Database ID 2829415

Attribute name 3dbag_tile

Value 3dbag_v210908_fd2cee53_4906

Class Function Usage

Class Mixed-use

Codespace NULL

Feature-specific attributes

Year of construction 2010 Year of demolition NULL

Storeys above ground 5 Storeys below ground 0

Height 30.535 UoM m

Storey height above ground NULL UoM NULL

Storey height below ground NULL UoM NULL

Roof type multiple horizontal Codespace NULL

OK Cancel

Use in QGIS: Attributes

Motivation

Plugin overview

Software installation

Software use

- Layer Loader
- Bulk Deleter

Advanced options

Software uninstall

Current limitations

QGIS Package

Resources

Changelog

rh_bdg_lod0 - Feature Attributes

Main Info Database Info Relation to surface Ext ref (Name) Ext ref (Uri) Addresses

Database ID 524581 Description NULL

GML ID NL.IMBAG.Pand.1742100000023595 GML codespace R-Buildings 2022-11-07.zip

Name Building 4906-2911 Name codespace NULL

Gen Attrib (String) Gen Attrib (Integer) Gen Attrib (Real) Gen Attrib (Measure) Gen Attrib (Date) Gen Attrib (Uri)

Expression

3dbag_tile

building_type

dutch_building_type

Database ID 2829415

Attribute name 3dbag_tile

Value 3dbag_v210908_fd2cee53_4906

Class Function Usage

Function overige gebruiksfunctie

woonfunctie

Feature-specific attributes

Year of construction 2010 Year of demolition NULL

Storeys above ground 5 Storeys below ground 0

Height 30.535 UoM m

Storey height above ground NULL UoM NULL

Storey height below ground NULL UoM NULL

Roof type multiple horizontal Codespace NULL

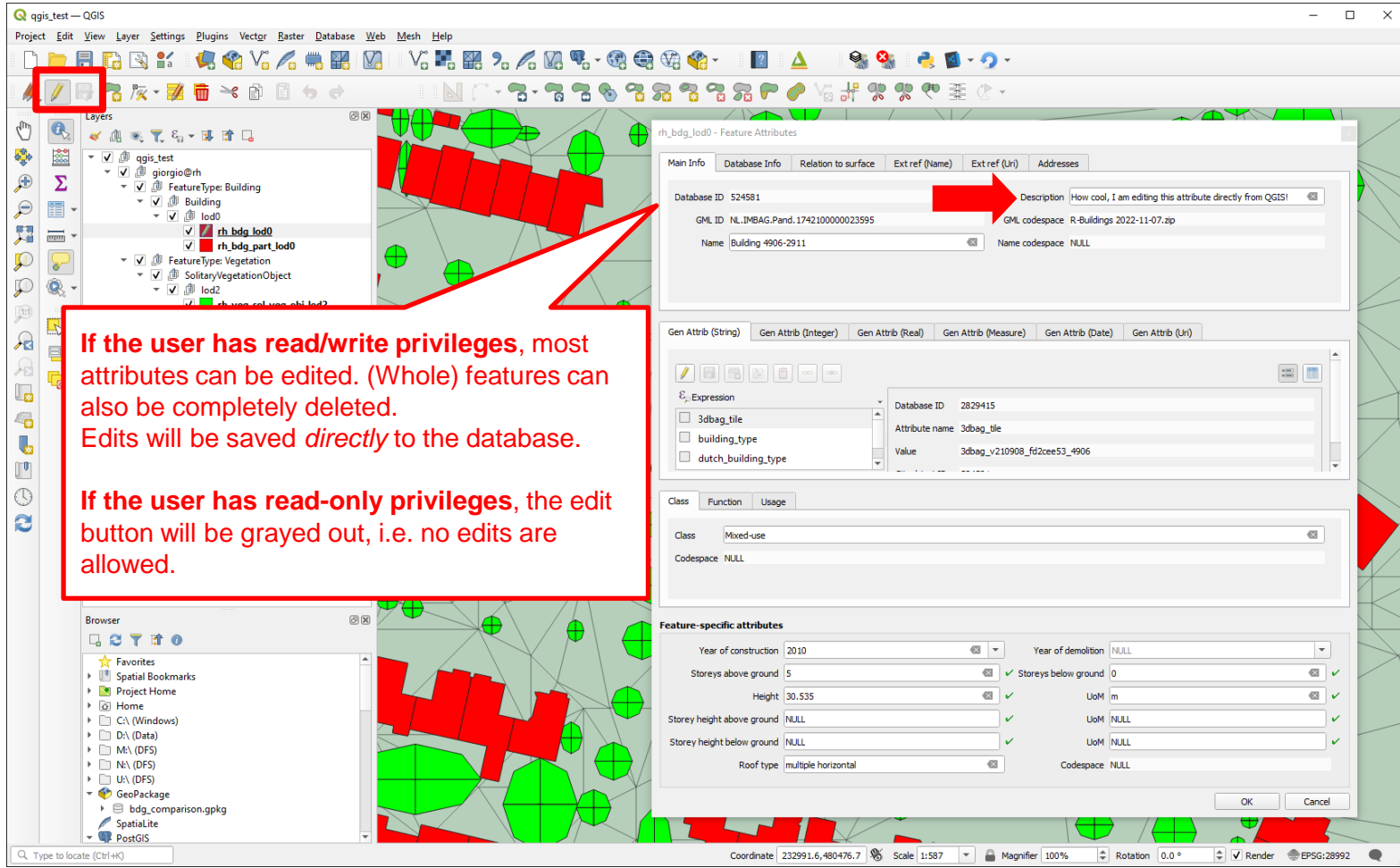
OK Cancel

Attributes containing [0..*] entries (e.g. function, usage, etc.) are presented as lists.

External codelists can also be loaded and visualised as look-up tables (see later on)

Use in QGIS: Attribute editing

Motivation
 Plugin overview
 Software installation
Software use
 • Layer Loader
 • Bulk Deleter
 Advanced options
 Software uninstall
 Current limitations
 QGIS Package
 Resources
 Changelog



If the user has read/write privileges, most attributes can be edited. (Whole) features can also be completely deleted. Edits will be saved *directly* to the database.

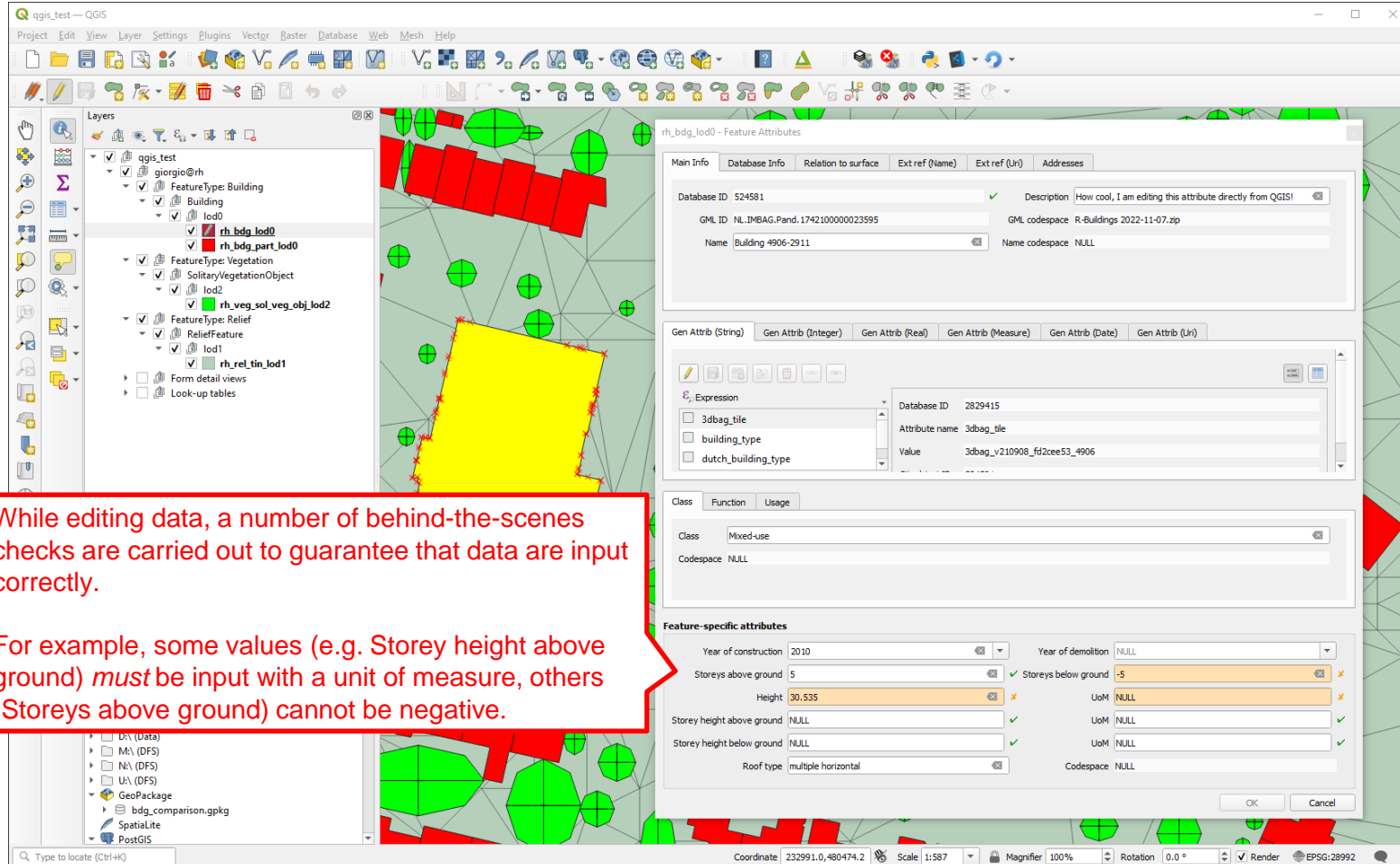
If the user has read-only privileges, the edit button will be grayed out, i.e. no edits are allowed.

Use in QGIS: Attribute editing

Motivation
 Plugin overview
 Software installation
Software use
 • Layer Loader
 • Bulk Deleter
 Advanced options
 Software uninstall
 Current limitations
 QGIS Package
 Resources
 Changelog

While editing data, a number of behind-the-scenes checks are carried out to guarantee that data are input correctly.

For example, some values (e.g. Storey height above ground) *must* be input with a unit of measure, others (Storeys above ground) cannot be negative.



Use in QGIS: Attribute editing

While editing data, input of enumeration values is made via drop-down lists

Motivation

Plugin overview

Software installation

Software use

- Layer Loader
- Bulk Deleter

Advanced options

Software uninstall

Current limitations

QGIS Package

Resources

Changelog

rh_bdg_lod0 - Feature Attributes

Main Info Database Info Relation to surface Ext ref (Name) Ext ref (Uri) Addresses

(no selection)
 (CityObject entirely above terrain
 (CityObject entirely below terrain
 (CityObject substantially above and below terrain
 (CityObject substantially above terrain
 (CityObject substantially below terrain

Relative to terrain
 Relative to water

Gen Attrib (String) Gen Attrib (Integer) Gen Attrib (Real) Gen Attrib (Measure) Gen Attrib (Date) Gen Attrib (Uri)

Expression
 3dbag_tile
 building_type
 dutch_building_type

Database ID 2829415
 Attribute name 3dbag_tile
 Value 3dbag_v210908_fd2cee53_4906

Class Function Usage

Class Mixed-use
 Codespace NULL

Feature-specific attributes

Year of construction	2010	Year of demolition	NULL
Storeys above ground	5	Storeys below ground	0
Height	30.535	UoM	m
Storey height above ground	NULL	UoM	NULL
Storey height below ground	NULL	UoM	NULL
Roof type	multiple horizontal	Codespace	NULL

OK Cancel

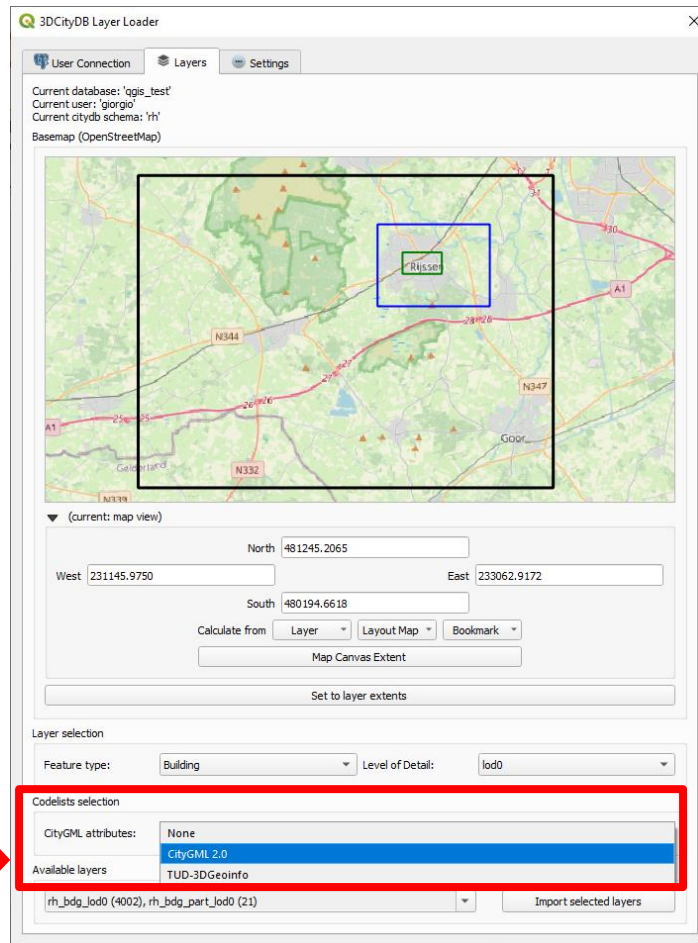
Use in QGIS: Codelist support

Codelists can be selected upon layer import

The attribute forms will be formatted automatically into drop-down lists or multiple-selection lists

Codelists can be added and customised either by the database administrator or by the user (see **Advanced options**)

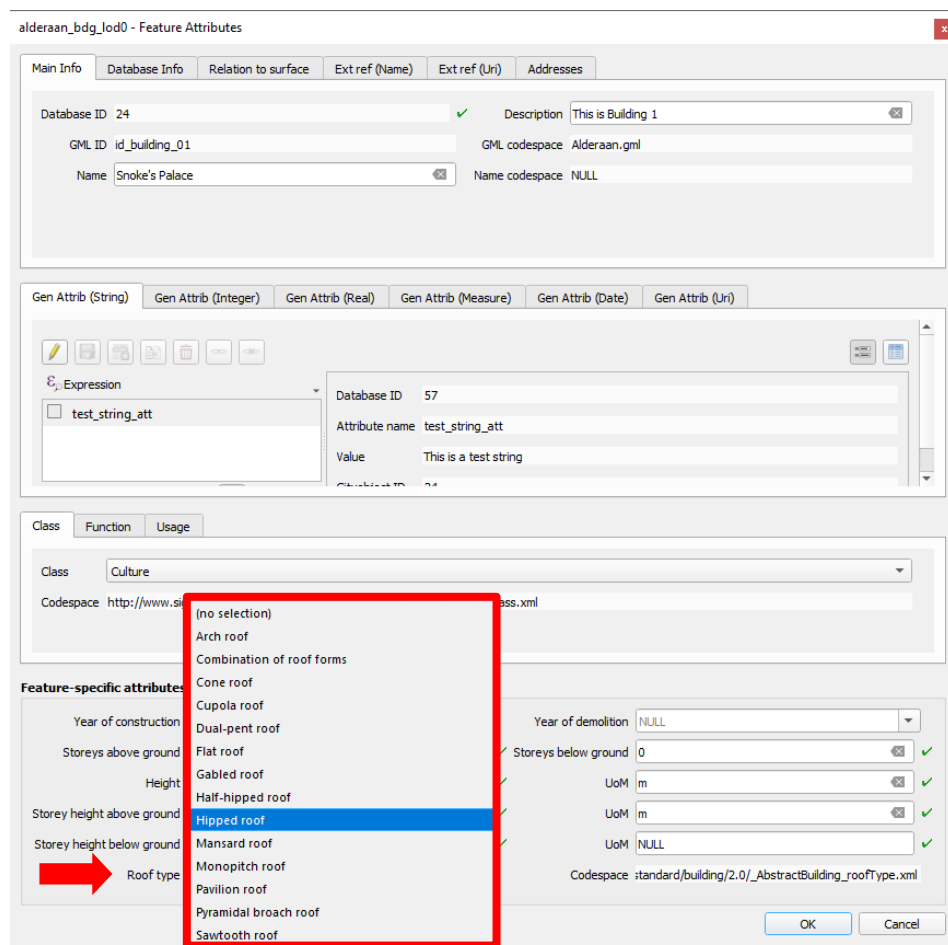
Motivation
 Plugin overview
 Software installation
Software use
 • Layer Loader
 • Bulk Deleter
 Advanced options
 Software uninstall
 Current limitations
 QGIS Package
 Resources
 Changelog



Use in QGIS: Codelist support

In the case of a CityGML property with cardinality **[0..1]**, the associated codelist values are presented as a **drop-down list**

Example: **Roof type**



alderaan_bdg_lod0 - Feature Attributes

Main Info Database Info Relation to surface Ext ref (Name) Ext ref (Un) Addresses

Database ID 24 ✓ Description This is Building 1

GML ID id_building_01 GML codespace Alderaan.gml

Name Snoke's Palace Name codespace NULL

Gen Attrib (String) Gen Attrib (Integer) Gen Attrib (Real) Gen Attrib (Measure) Gen Attrib (Date) Gen Attrib (Un)

Expression

test_string_att

Database ID 57

Attribute name test_string_att

Value This is a test string

Class Function Usage

Class Culture

Codespace http://www.s...

Feature-specific attributes

Year of construction

Storeys above ground

Height

Storey height above ground

Storey height below ground

Roof type

(no selection)

Arch roof

Combination of roof forms

Cone roof

Cupola roof

Dual-pent roof

Flat roof

Gabled roof

Half-hipped roof

Hipped roof

Mansard roof

Monopitch roof

Pavillion roof

Pyramidal broach roof

Sawtooth roof

Year of demolition NULL

Storeys below ground 0 ✓

UoM m ✓

UoM m ✓

UoM NULL ✓

Codespace standard/building/2.0/_AbstractBuilding_roofType.xml

OK Cancel

Use in QGIS: Codelist support

In the case of a CityGML property with cardinality **[0..*]**, the associated codelist values are presented as a **multiple-selection list**

Example: property (Building) **function**

alderaan_bdg_lod0 - Feature Attributes

Main Info Database Info Relation to surface Ext ref (Name) Ext ref (Un) Addresses

Database ID 24 ✓ Description This is Building 1

GML ID id_building_01 GML codespace Alderaan.gml

Name Snoke's Palace Name codespace NULL

Gen Attrib (String) Gen Attrib (Integer) Gen Attrib (Real) Gen Attrib (Measure) Gen Attrib (Date) Gen Attrib (Un)

Expression

test_string_att

Database ID 57

Attribute name test_string_att

Value This is a test string

Class Function

Function

☒ Activity building ☐ Administration building ☐ Agrarian and forestry building

☒ Airport building ☒ Alpine cabin ☐ Arboretum

☐ Asylum seekers home ☐ Barn ☒ Barracks

Feature-specific attributes

Year of construction 1955 Year of demolition NULL

Storeys above ground 3 ✓ Storeys below ground 0 ✓

Height 15 ✓ UoM m ✓

Storey height above ground 3.0 ✓ UoM m ✓

Storey height below ground NULL ✓ UoM NULL ✓

Roof type Arch roof Codespace standard/building/2.0/_AbstractBuilding_roofType.xml

OK Cancel

Use in QGIS: 3D visualisation

Motivation

Plugin overview

Software installation

Software use

- Layer Loader
- Bulk Deleter

Advanced options

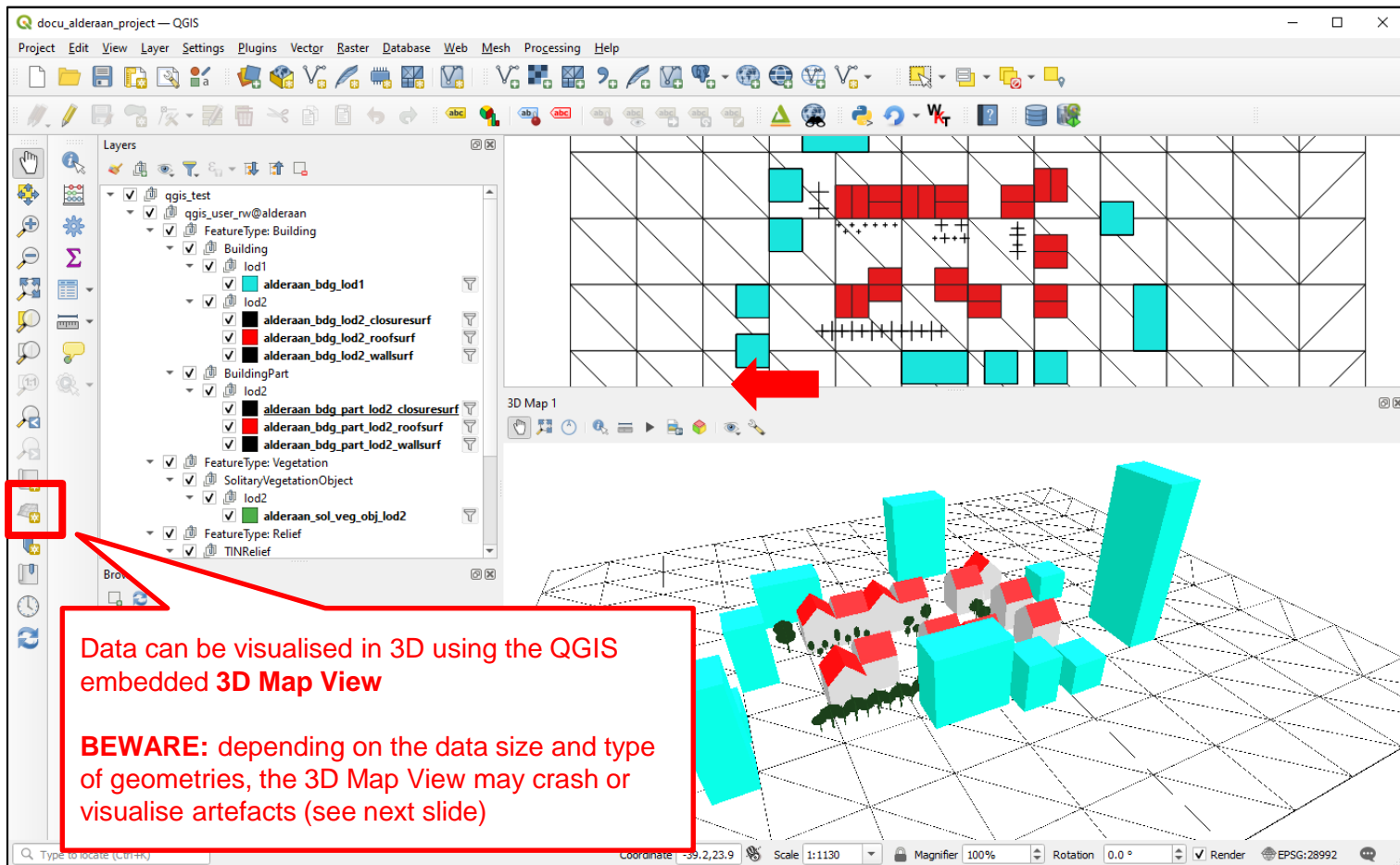
Software uninstall

Current limitations

QGIS Package

Resources

Changelog



docu_aldeeraan_project — QGIS

Project Edit View Layer Settings Plugins Vector Raster Database Web Mesh Processing Help

Layers

- qgis_test
 - qgis_user_nw@aldeeraan
 - FeatureType: Building
 - Building
 - lod1
 - aldeeraan_bdg_lod1
 - lod2
 - aldeeraan_bdg_lod2_closuresurf
 - aldeeraan_bdg_lod2_roofsurf
 - aldeeraan_bdg_lod2_wallsurf
 - BuildingPart
 - lod2
 - aldeeraan_bdg_part_lod2_closuresurf
 - aldeeraan_bdg_part_lod2_roofsurf
 - aldeeraan_bdg_part_lod2_wallsurf
 - FeatureType: Vegetation
 - SolitaryVegetationObject
 - lod2
 - aldeeraan_sol_veg_obj_lod2
 - FeatureType: Relief
 - TINRelief

3D Map 1

Data can be visualised in 3D using the QGIS embedded **3D Map View**

BEWARE: depending on the data size and type of geometries, the 3D Map View may crash or visualise artefacts (see next slide)

Type to locate (Ctrl+F)

Coordinate: -99.2, 23.9 Scale: 1:1130 Magnifier: 100% Rotation: 0.0 ° Render EPSG:28992

Use in QGIS: 3D visualisation

Motivation

Plugin overview

Software installation

Software use

- Layer Loader
- Bulk Deleter

Advanced options

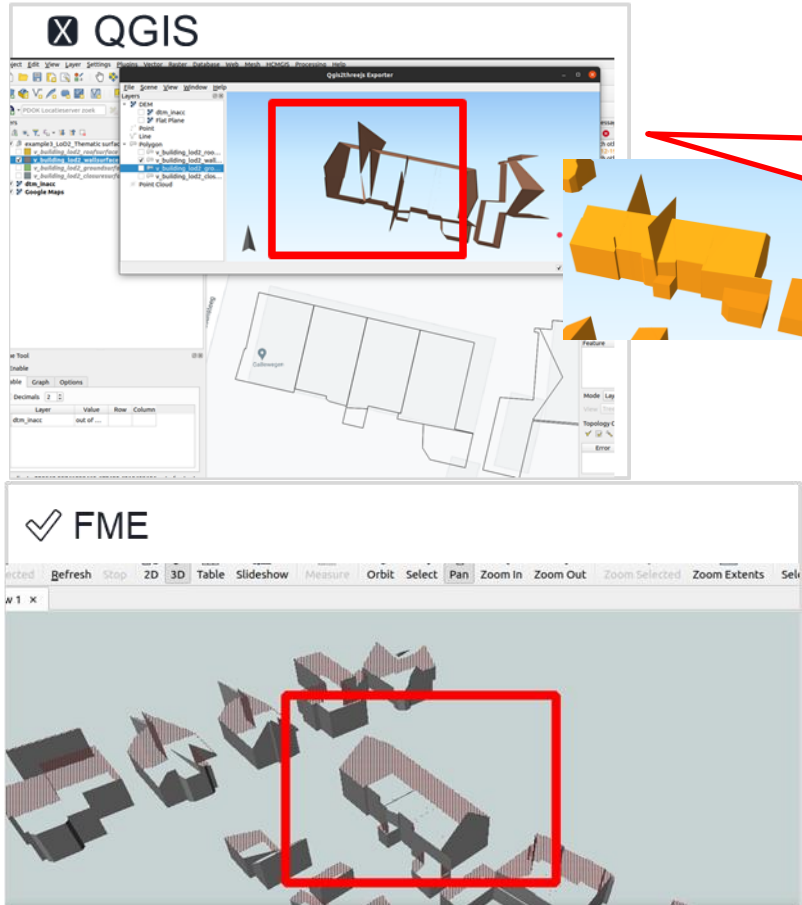
Software uninstall

Current limitations

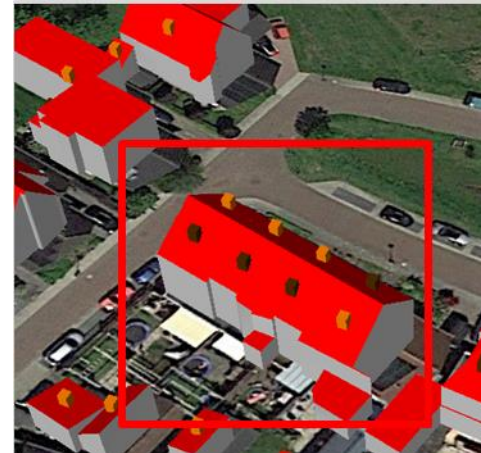
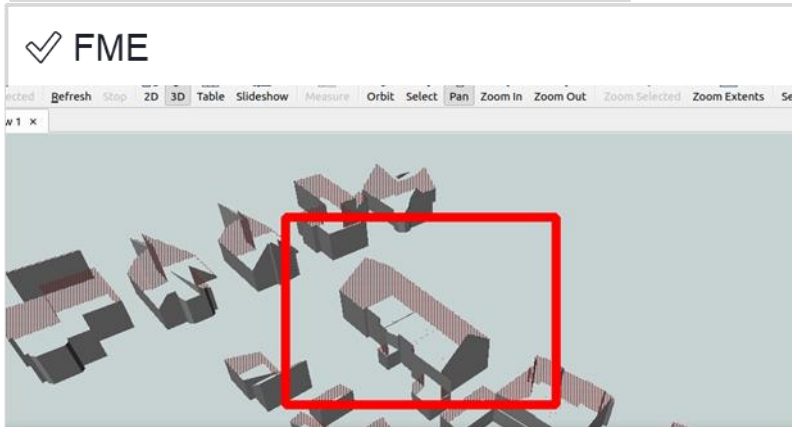
QGIS Package

Resources

Changelog



In our tests, such artefacts are however a 3D visualisation issue (QGIS 3D renderer?) and *not* related to the actual data. 3D visualisation in FME and in Google Earth show indeed correct results.



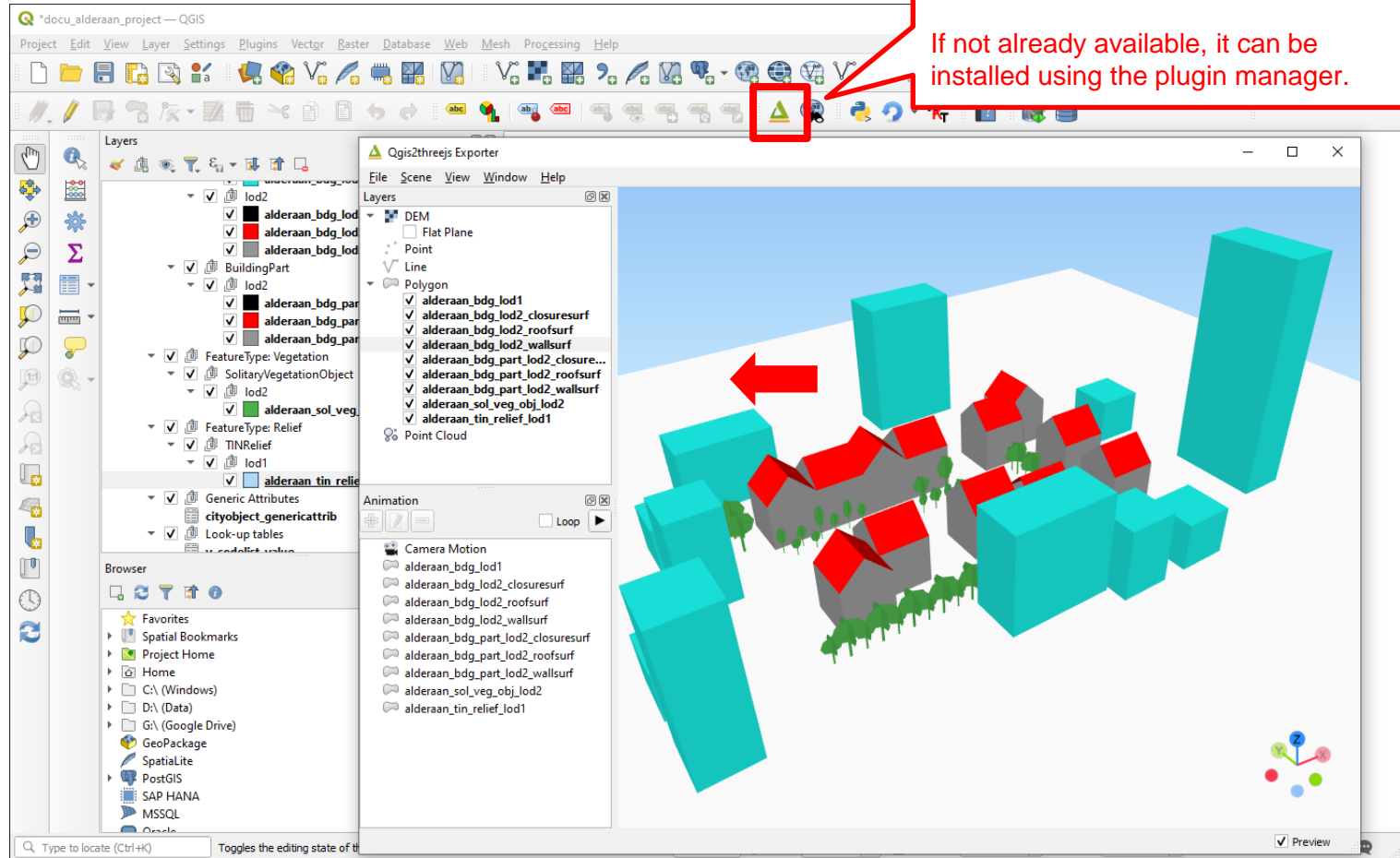
✓
Google
earth
(as KML)

Use in QGIS: 3D visualisation

Motivation
 Plugin overview
 Software installation
Software use
 • Layer Loader
 • Bulk Deleter
 Advanced options
 Software uninstall
 Current limitations
 QGIS Package
 Resources
 Changelog

Alternatively, the **Qgis2threejs** plugin can be used for 3D visualisation.

If not already available, it can be installed using the plugin manager.



Bulk Deleter

Open the **Bulk Deleter** from the menu or by clicking on the corresponding icon



Motivation

Plugin overview

Software installation

Software use

- Layer Loader
- **Bulk Deleter**

Advanced options

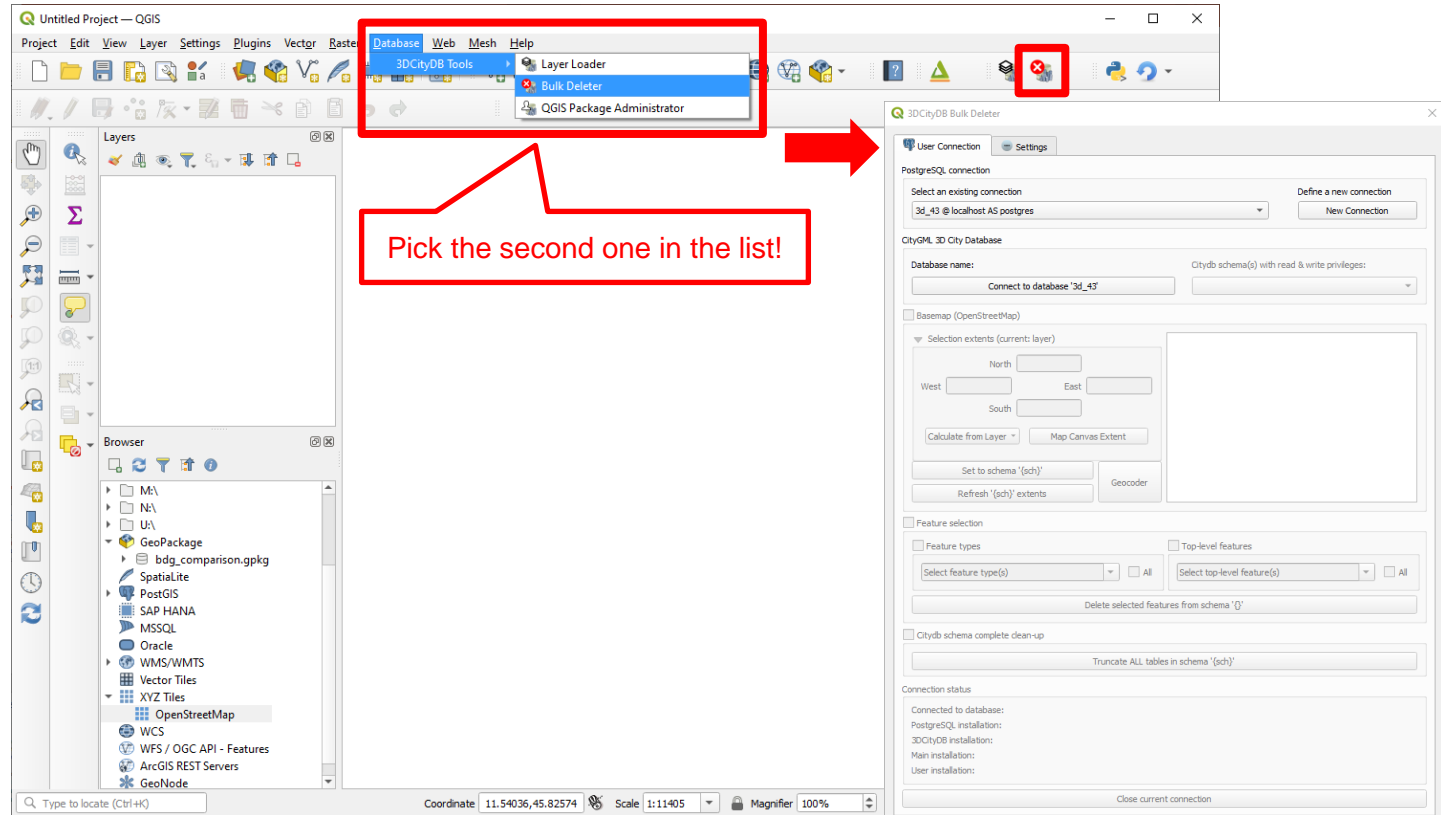
Software uninstall

Current limitations

QGIS Package

Resources

Changelog



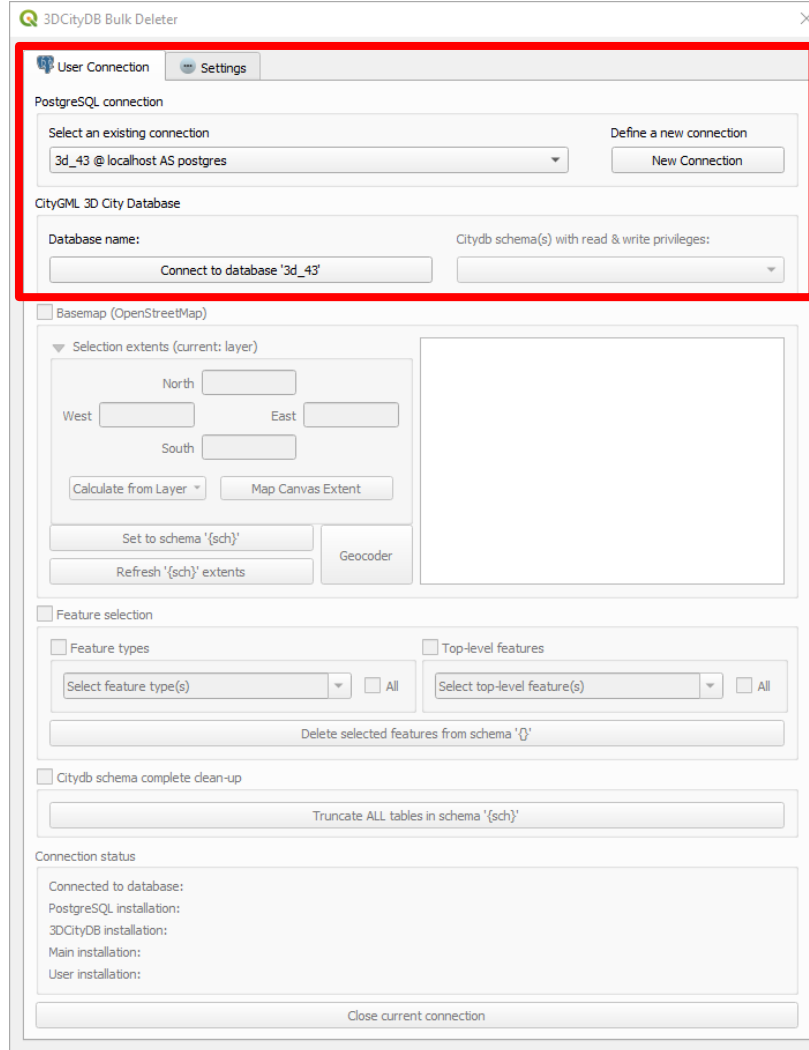
The screenshot shows the QGIS interface with the 'Database' menu open, highlighting the 'Bulk Deleter' option. A red box and arrow point to this option with the text 'Pick the second one in the list!'. The '3D CityDB Bulk Deleter' dialog is also shown, with the 'User Connection' tab selected. The dialog includes fields for 'PostgreSQL connection', 'CityGML 3D City Database', 'Basemap (OpenStreetMap)', 'Selection extents', 'Feature selection', and 'CityGML schema complete clean-up'. The 'Connection status' section at the bottom shows the current connection details.

Bulk Deleter

In the "User Connection" tab

1) Create a new connection or use an existing one to the desired 3DCityDB instance (here: "qgis_test")

2) Connect to the chosen database



3DCityDB Bulk Deleter

User Connection Settings

PostgreSQL connection

Select an existing connection: 3d_43 @ localhost AS postgres Define a new connection: New Connection

CityGML 3D City Database

Database name: Connect to database '3d_43' Citydb schema(s) with read & write privileges:

Basemap (OpenStreetMap)

Selection extents (current: layer)

North West East South

Calculate from Layer Map Canvas Extent

Set to schema '{sch}' Geocoder

Refresh '{sch}' extents

Feature selection

Feature types Top-level features

Select feature type(s) All Select top-level feature(s) All

Delete selected features from schema '{sch}'

Citydb schema complete clean-up

Truncate ALL tables in schema '{sch}'

Connection status

Connected to database:
PostgreSQL installation:
3DCityDB installation:
Main installation:
User installation:

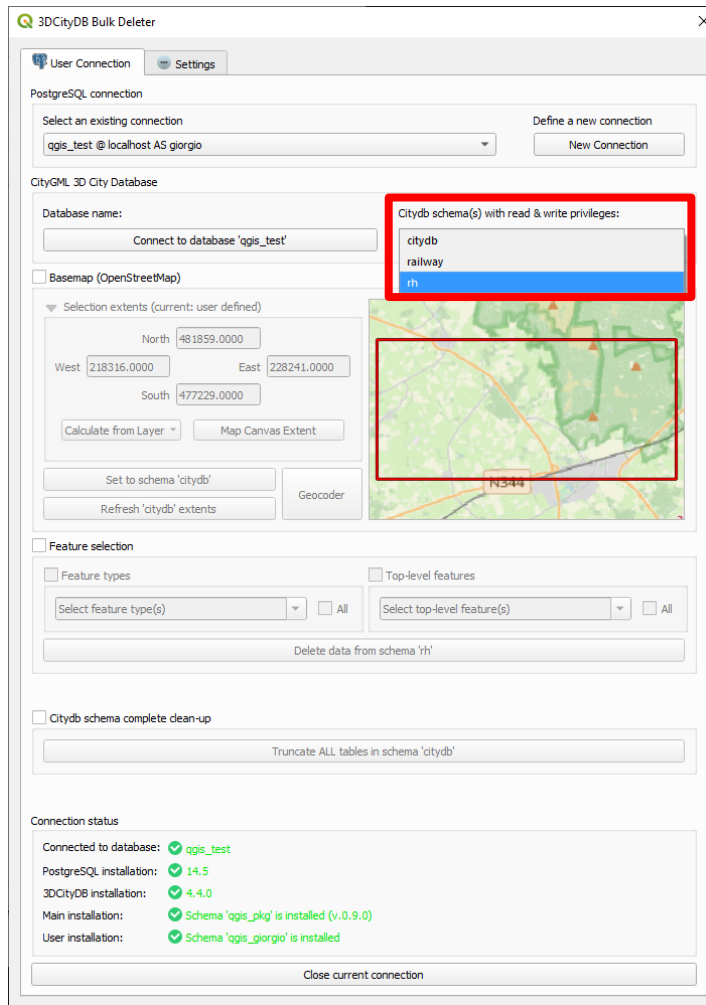
Close current connection

Bulk Deleter

3) Once connected, choose one of the available citydb schemas

Nota bene: Only the citydb schemas for **which you have read & write ("rw") privileges** are listed

Motivation
 Plugin overview
 Software installation
Software use
 • Layer Loader
 • **Bulk Deleter**
 Advanced options
 Software uninstall
 Current limitations
 QGIS Package
 Resources
 Changelog

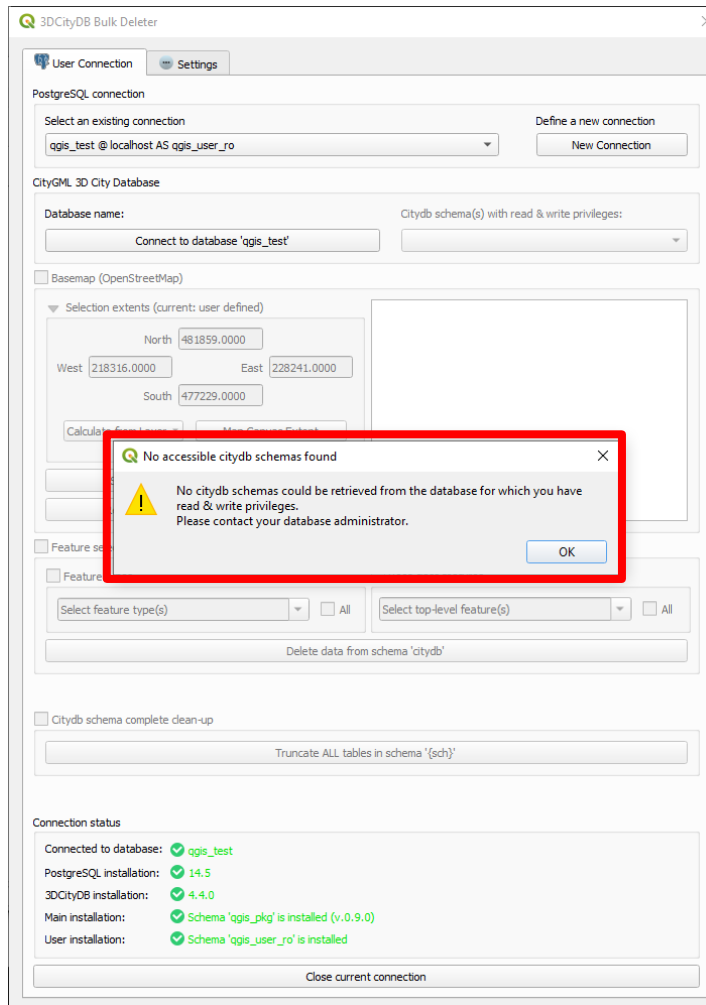


Bulk Deleter

3) Once connected, choose one of the available citydb schemas

Nota bene: Only the citydb schemas for **which you have read & write ("rw") privileges** are listed

If there are no citydb schemas for which you have "rw" privileges, you will be notified before the connection is closed and the GUI completely disabled.



Bulk Deleter

Motivation

Plugin overview

Software installation

Software use

- Layer Loader
- **Bulk Deleter**

Advanced options

Software uninstall

Current limitations

QGIS Package

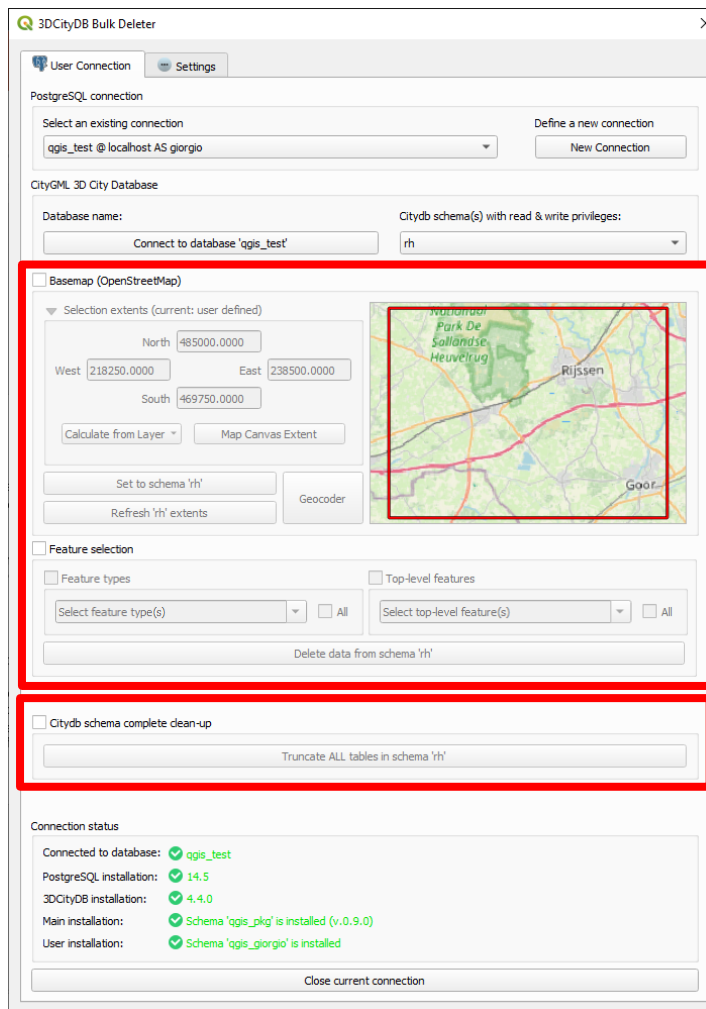
Resources

Changelog

4) Once you have selected the citydb schema, you can perform 2 types of actions:

- Select specific features to delete (and optionally define a spatial filter, too)
- Clean up the whole schema, i.e. truncate all tables of the selected citydb schema

The GUI will prevent you from choosing both at the same time



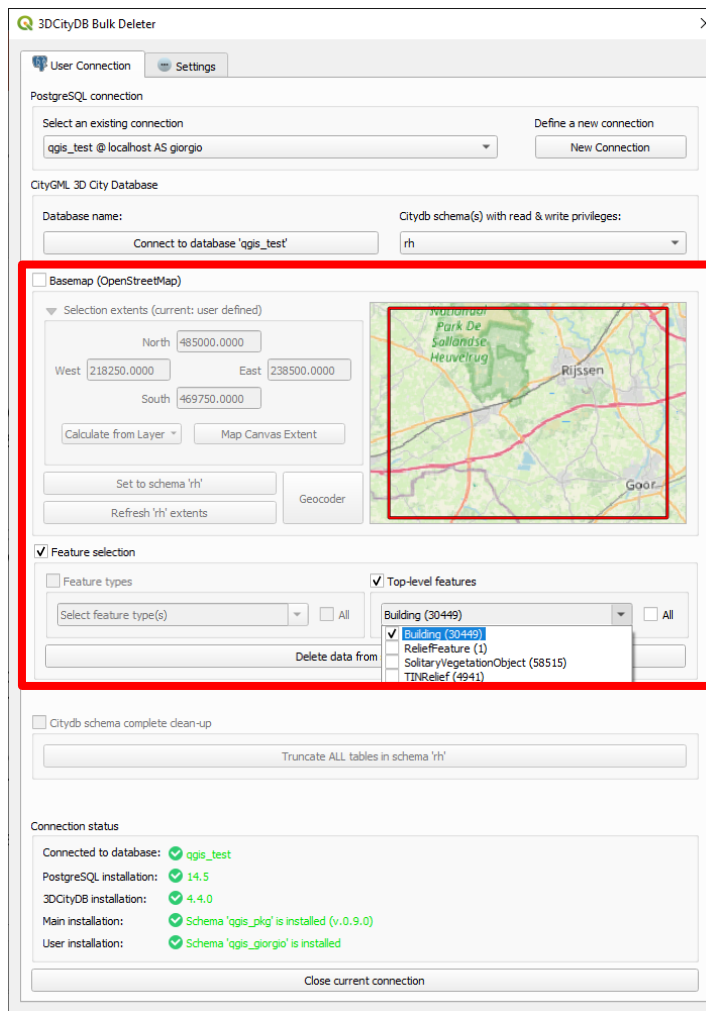
Bulk Deleter

5a) Select the features to delete. Activate the **Feature selection box**. You can now select:

- either CityGML Features types
- or top-level features

Remember: Feature Types correspond to CityGML modules, i.e. they may contain multiple top-level features. For example:

- Feature Type "Vegetation" includes "Solitary Vegetation Object" and "PlantCover" top-level features
- Feature Type "Relief" includes "TINRelief", "BreakLineRelief" and "MassPointRelief" top-level features



Bulk Deleter

Motivation

Plugin overview

Software installation

Software use

- Layer Loader
- **Bulk Deleter**

Advanced options

Software uninstall

Current limitations

QGIS Package

Resources

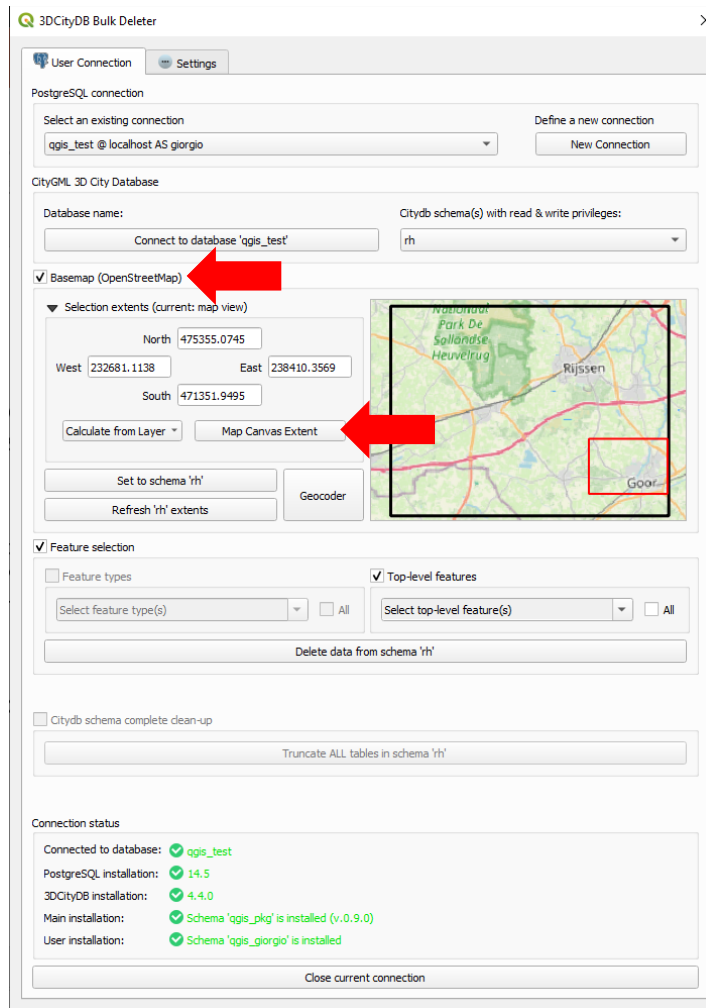
Changelog

5a) Select the features to delete. Activate the **Feature selection box**. You can now select:

- either CityGML Features types
- or top-level features
- and, optionally, define the extents of the area where to delete the selected feature. You must then also activate the **Basemap** box and press the **Map Canvas Extent** button

The delete extents are represented by the **red bounding box**.

Please note: The Set to schema {cdb_schema}, Refresh {cdb_schema} and GeoCoder buttons follow the same logic as in the Layer Loader

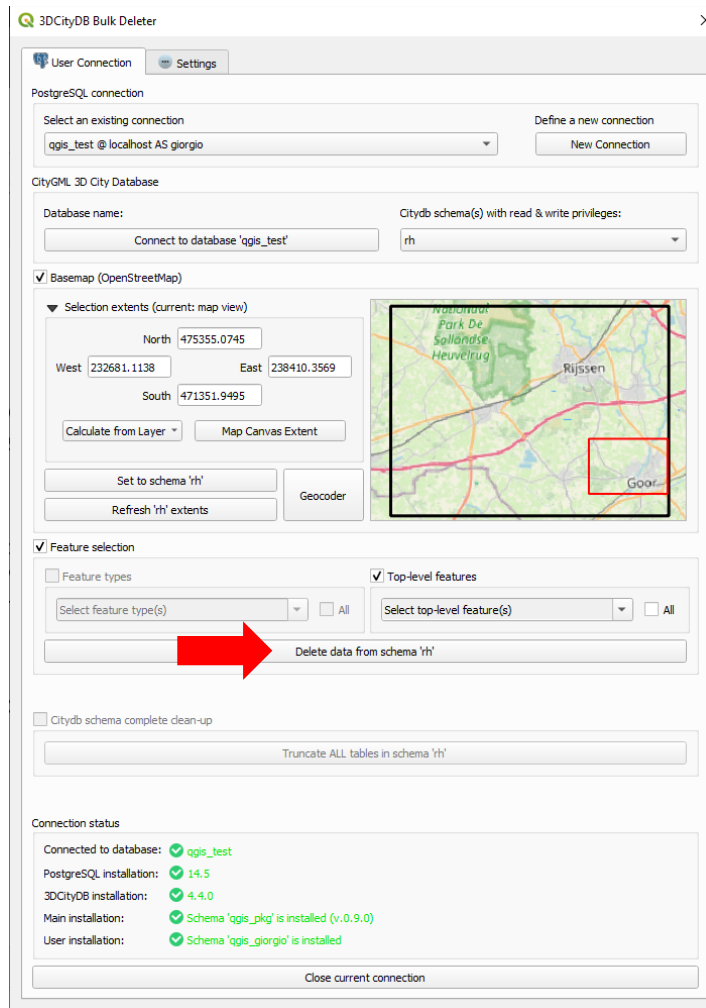


Bulk Deleter

6a) Press the **Delete data from schema {cdb_schema}** button

Beware: Depending on the quantity of selected features, the operation might take some time.

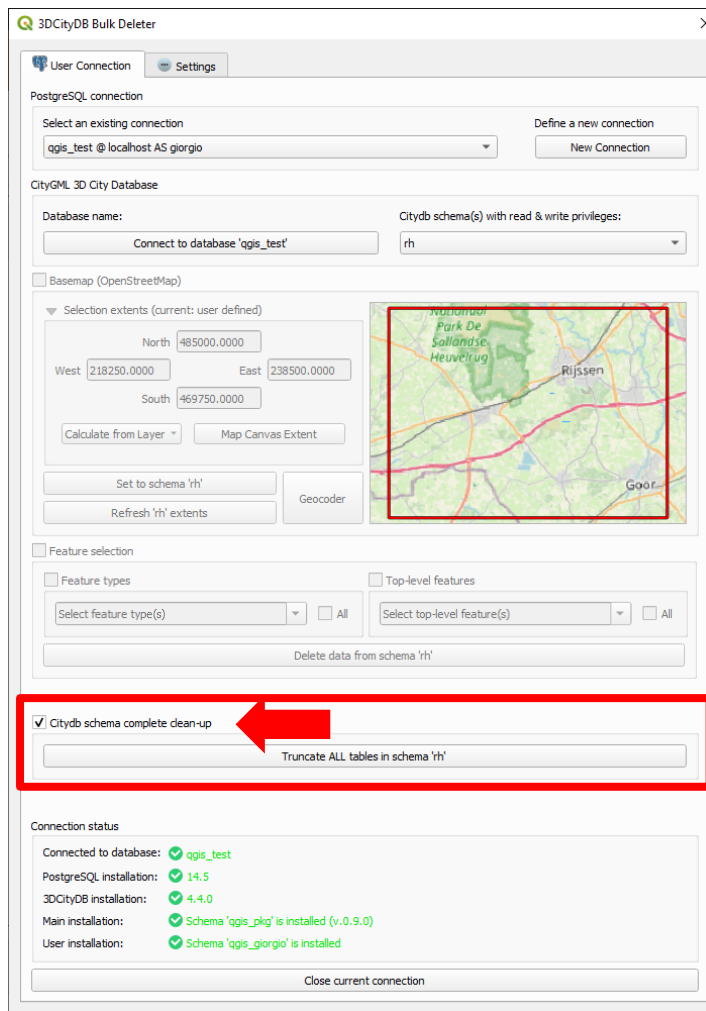
If you want to completely delete the database, you are recommended to use the other option (see next slides) which is **much** faster!



Bulk Deleter

5b) Enable the **Citydb schema complete clean-up box** and press the **Truncate ALL tables in schema {cdb_schema}** button.

Beware: The selected citydb schema will be completely emptied and reset to its initial state. In addition, all preexisting privileges (also of other users) will be reset to "None"



3DCityDB Bulk Deleter

User Connection Settings

PostgreSQL connection

Select an existing connection: Define a new connection:

CityGML 3D City Database

Database name: Citydb schema(s) with read & write privileges:

☐ Basemap (OpenStreetMap)

Selection extents (current: user defined)

North: East:
 West: South:

☐ Feature selection

☐ Feature types ☐ Top-level features

Select feature type(s): ☐ All Select top-level feature(s): ☐ All

☒ Citydb schema complete clean-up

Connection status

Connected to database: ☒ qgis_test
 PostgreSQL installation: ☒ 14.5
 3DCityDB installation: ☒ 4.4.0
 Main installation: ☒ Schema 'qgis_pkg' is installed (v.0.9.0)
 User installation: ☒ Schema 'qgis_giorgio' is installed

Motivation

Plugin
overview

Software
installation

Software
use

Advanced
options

Software
uninstall

Current
limitations

QGIS
Package

Resources

Customised codelists

For certain CityGML properties (e.g. class, function, usage, roof type, etc.) the QGIS attribute forms in the Layer Loader can be linked to look-up tables containing

- Codelist values from the non-normative CityGML specifications
- Codelist values optionally defined by the user

This applies to properties containing single (e.g. class, roof type) or, possibly, multiple values (e.g. function, usage)

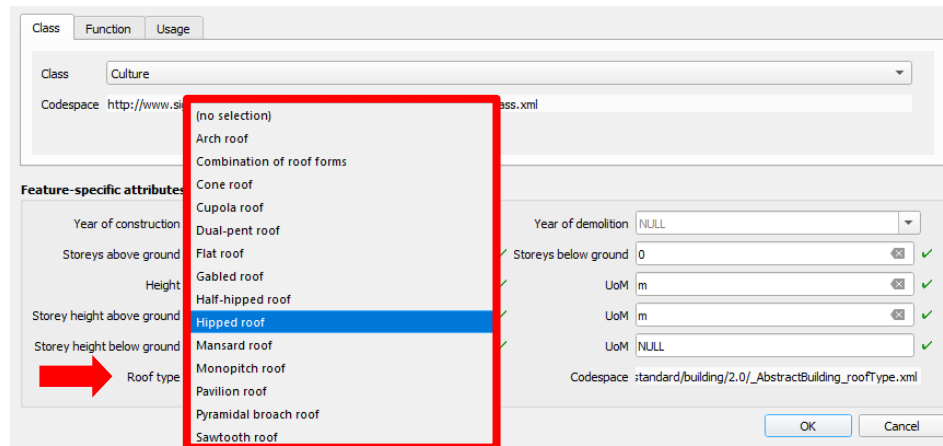
In this way the user does not have to "remember" specific codes when typing, thus reducing the chances of wrong data input

The 3DCityDB-Tools plugin already contains the codelists from the CityGML 2.0 specifications.

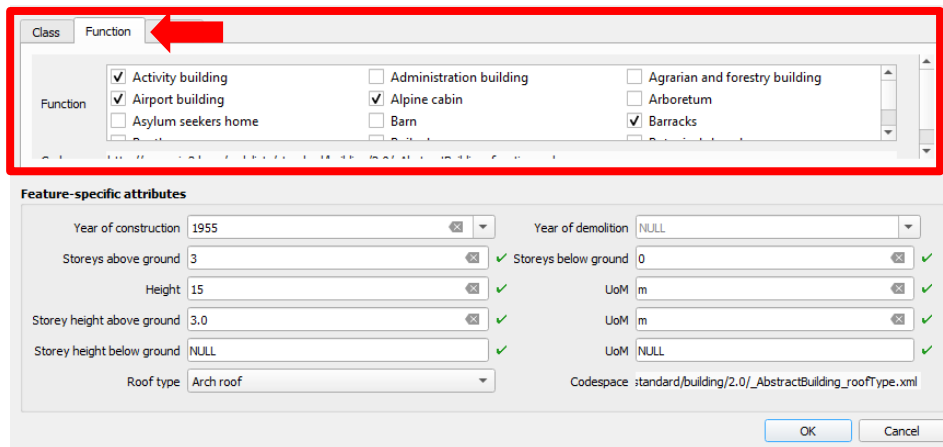
Customised codelists

Motivation
 Plugin overview
 Software installation
 Software use
Advanced options
 Software uninstall
 Current limitations
 QGIS Package
 Resources
 Changelog

Property with cardinality [0..1]:
 drop-down list



Property with cardinality [0..*]:
 multiple-selection list



Customised codelists

Adding customised codelists requires **two steps**:

- 1) Load the actual codelist data** (codelist entries and codelist metadata). Such data must be stored in two predefined tables of the QGIS Package in the 3DCityDB
- 2) Add mapping rules** for automatic setup of the lists in the attribute forms. Such data must be stored in a predefined table of the QGIS Package in the 3DCityDB

Both operations can be carried out either by the database administrator or by the user

1) Administrator:

- The predefined tables are in schema **qgis_pkg**
- All codelists and rules are made available to each newly created `qgis_{usr}` schema

2) User:

- The predefined tables are in schema **qgis_{usr_name}**
- All codelists and rules are available only to user

Customised codelists

Adding customised codelists requires **two steps**:

1) Load the actual codelist data (codelist entries and codelist metadata). Such data must be stored in two predefined tables of the QGIS Package in the 3DCityDB

- The tables are named **CODELIST** and **CODELIST_VALUE** in the `qgis_{usr_name}` and **CODELIST_TEMPLATE** and **CODELIST_VALUE_TEMPLATE** in the `qgis_pkg` schema, respectively. Their structure is exactly the same
- The following examples are based on tables `CODELIST` and `CODELIST_VALUE` but the procedure is the same for the `*_TEMPLATE` ones

Customised codelists

Table **CODELIST** contains some metadata values such as the codelist name, its name_space and data_model.

It is referenced by table **CODELIST_VALUE** which contains the actual values

Tables (8)

- > **codelist**
- > **codelist_lookup_config**
- > **codelist_value**
- > enumeration
- > enumeration_value
- > extents
- > layer_metadata
- > settings

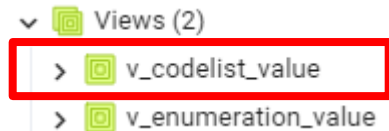
Table **CODELIST** (excerpt)

id [PK] bigint	data_model character varying	name character varying	name_space character varying	description text
1	CityGML 2.0	MimeType	https://www.sig3d.org/codelists/standard/core/2.0/ImplicitGeometry_mimeType.xml	[null]
2	CityGML 2.0	_AbstractBridgeClass	https://www.sig3d.org/codelists/standard/bridge/2.0/_AbstractBridge_class.xml	[null]
3	CityGML 2.0	_AbstractBridgeFunctionUsage	https://www.sig3d.org/codelists/standard/bridge/2.0/_AbstractBridge_function.xml	[null]
4	CityGML 2.0	_AbstractBuildingClass	https://www.sig3d.org/codelists/standard/building/2.0/_AbstractBuilding_class.xml	[null]
5	CityGML 2.0	_AbstractBuildingFunctionUsage	https://www.sig3d.org/codelists/standard/building/2.0/_AbstractBuilding_function.xml	[null]
6	CityGML 2.0	_AbstractBuildingRoofType	https://www.sig3d.org/codelists/standard/building/2.0/_AbstractBuildingRoofType.xml	[null]
7	CityGML 2.0	RoomClass	https://www.sig3d.org/codelists/standard/building/2.0/RoomClass.xml	[null]
8	CityGML 2.0	RoomFunctionUsage	https://www.sig3d.org/codelists/standard/building/2.0/RoomFunctionUsage.xml	[null]
9	CityGML 2.0	BuildingFurnitureClass	https://www.sig3d.org/codelists/standard/building/2.0/BuildingFurnitureClass.xml	[null]
10	CityGML 2.0	BuildingFurnitureFunctionUsage	https://www.sig3d.org/codelists/standard/building/2.0/BuildingFurnitureFunctionUsage.xml	[null]

Table **CODELIST_VALUE** (excerpt)

id [PK] bigint	code_id integer	value character varying	description text
1	1	model/vrml	VRML97
2	1	application/x-3ds	3ds max
3	1	application/dxf	AutoCad DXF
4	1	application/x-autocad	AutoCad DXF
5	1	application/x-dxf	AutoCad DXF
6	1	application/acad	AutoCad DWG
13	1	image/tiff	*.tiff, *.tif images
14	1	image/bmp	*.bmp images
15	2	1000	Arced bridge
16	2	1010	Cable-stayed bridge
17	2	1020	Deck bridge
18	2	1030	Cable-stayed overpass

Customised codelists



In QGIS, all codelists values are retrieved from view **v_codelist_value** in the user schema of every user (e.g. "qgis_giorgio")

View V_CODELISTS_VALUE (excerpt)

id bigint	data_model character varying	name character varying	value character varying	description text	name_space character varying
1	CityGML 2.0	MimeType	model/vrml	VRML97	https://www.sig3d.org/codelists/stan...
2	CityGML 2.0	MimeType	application/x-3ds	3ds max	https://www.sig3d.org/codelists/stan...
3	CityGML 2.0	MimeType	application/dxf	AutoCad DXF	https://www.sig3d.org/codelists/stan...
4	CityGML 2.0	MimeType	application/x-autocad	AutoCad DXF	https://www.sig3d.org/codelists/stan...
39	CityGML 2.0	_AbstractBuildingClass	1110	Maintenance and waste mana...	https://www.sig3d.org/codelists/stan...
40	CityGML 2.0	_AbstractBuildingClass	1120	Healthcare	https://www.sig3d.org/codelists/stan...
41	CityGML 2.0	_AbstractBuildingClass	1130	Communicating	https://www.sig3d.org/codelists/stan...
42	CityGML 2.0	_AbstractBuildingClass	1140	Security	https://www.sig3d.org/codelists/stan...
43	CityGML 2.0	_AbstractBuildingClass	1150	Storage	https://www.sig3d.org/codelists/stan...
44	CityGML 2.0	_AbstractBuildingClass	1160	Industry	https://www.sig3d.org/codelists/stan...
45	CityGML 2.0	_AbstractBuildingClass	1170	Traffic	https://www.sig3d.org/codelists/stan...
46	CityGML 2.0	_AbstractBuildingClass	1180	Other function	https://www.sig3d.org/codelists/stan...
47	CityGML 2.0	_AbstractBuildingClass	9999	Unknown	https://www.sig3d.org/codelists/stan...
48	CityGML 2.0	_AbstractBuildingFunct...	1000	Residential building	https://www.sig3d.org/codelists/stan...
49	CityGML 2.0	_AbstractBuildingFunct...	1010	Tenement	https://www.sig3d.org/codelists/stan...

Customised codelists

To add values to the **CODELIST** and **CODELIST_VALUE** tables, the user can for example issue a SQL statement such as:

```
-- Optionally, delete previously loaded values belonging to the same codelist in user schema "qgis_giorgio"
DELETE FROM qgis_giorgio.codelist WHERE data_model = 'NL-BAG';
-- Insert first the codelist metadata and then the values in one single SQL statement
WITH cl AS (
    INSERT INTO qgis_giorgio.codelist (data_model, name, name_space, description)
    VALUES
        ('NL-BAG', 'BAG', 'https://..some_url_here.....', 'Codelist containing the values of the Dutch Basisregistratie
        Adressen en Gebouwen')
    RETURNING id)
INSERT INTO qgis_giorgio.codelist_value (code_id, value, description)
SELECT cl.id, v.value, v.description FROM cl, (VALUES
('apple' , 'Codelist value for "apple"' ),
('orange' , 'Codelist value for "orange"' ),
('pear' , 'Codelist value for "pear"' ),
('banana' , 'Codelist value for "banana"' )
) AS v(value, description);
```

This SQL statement can be adapted
by changing only the parts in **red**


Customised codelists









Adding customised codelists requires **two steps**:

- 1) **Load the actual codelist data** (codelist entries and codelist metadata). Such data must be stored in two predefined tables of the QGIS Package in the 3DCityDB
- 2) **Add mapping rules** for automatic setup of the lists in the attribute forms. Such data must be stored in a predefined table of the QGIS Package in the 3DCityDB
 - The table is named **CODELIST_LOOKUP_CONFIG** in the `qgis_{usr_name}` and **CODELIST_LOOKUP_CONFIG_TEMPLATE** in the `qgis_pkg` schema, respectively. Their structure is exactly the same
 - The following examples are based on tables `CODELIST_LOOKUP_CONFIG` but the procedure is the same for the `*_TEMPLATE` one

Customised codelists

Table **CODELIST_LOOKUP_CONFIG** contains fields used to map Codelist values to the corresponding form attributes in QGIS.
Explanation of the necessary fields is given in the next slide.

▼  Tables (8)

- >  codelist
- >  **codelist_lookup_config**
- >  codelist_value
- >  enumeration
- >  enumeration_value
- >  extents
- >  layer_metadata
- >  settings

id	name	ade_prefix	source_class	source_table	source_column	target_table	key_column	value_column	filter_expression
[PK] integer	character varying	character var	character varying	character varying	character varying	character varyin	character var	character varyin	character varying
1	CityGML 2.0	[null]	Bridge	bridge	class	v_codelist	value	description	data_model = 'CityGML 2.0' AND name = '_AbstractBridgeClass'
2	CityGML 2.0	[null]	Bridge	bridge	function	v_codelist	value	description	data_model = 'CityGML 2.0' AND name = '_AbstractBridgeFunctionUsage'
3	CityGML 2.0	[null]	Bridge	bridge	usage	v_codelist	value	description	data_model = 'CityGML 2.0' AND name = '_AbstractBridgeFunctionUsage'
4	CityGML 2.0	[null]	BridgePart	bridge	class	v_codelist	value	description	data_model = 'CityGML 2.0' AND name = '_AbstractBridgeClass'
5	CityGML 2.0	[null]	BridgePart	bridge	function	v_codelist	value	description	data_model = 'CityGML 2.0' AND name = '_AbstractBridgeFunctionUsage'
6	CityGML 2.0	[null]	BridgePart	bridge	usage	v_codelist	value	description	data_model = 'CityGML 2.0' AND name = '_AbstractBridgeFunctionUsage'
7	CityGML 2.0	[null]	Building	building	class	v_codelist	value	description	data_model = 'CityGML 2.0' AND name = '_AbstractBuildingClass'
8	CityGML 2.0	[null]	Building	building	function	v_codelist	value	description	data_model = 'CityGML 2.0' AND name = '_AbstractBuildingFunctionUsage'
9	CityGML 2.0	[null]	Building	building	usage	v_codelist	value	description	data_model = 'CityGML 2.0' AND name = '_AbstractBuildingFunctionUsage'
10	CityGML 2.0	[null]	Building	building	roof_type	v_codelist	value	description	data_model = 'CityGML 2.0' AND name = '_AbstractBuildingRoofType'
11	CityGML 2.0	[null]	BuildingPart	building	class	v_codelist	value	description	data_model = 'CityGML 2.0' AND name = '_AbstractBuildingClass'
12	CityGML 2.0	[null]	BuildingPart	building	function	v_codelist	value	description	data_model = 'CityGML 2.0' AND name = '_AbstractBuildingFunctionUsage'
13	CityGML 2.0	[null]	BuildingPart	building	usage	v_codelist	value	description	data_model = 'CityGML 2.0' AND name = '_AbstractBuildingFunctionUsage'
14	CityGML 2.0	[null]	BuildingPart	building	roof_type	v_codelist	value	description	data_model = 'CityGML 2.0' AND name = '_AbstractBuildingRoofType'
15	CityGML 2.0	[null]	BuildingRoom	room	class	v_codelist	value	description	data_model = 'CityGML 2.0' AND name = 'RoomClass'
16	CityGML 2.0	[null]	BuildingRoom	room	function	v_codelist	value	description	data_model = 'CityGML 2.0' AND name = 'RoomFunctionUsage'
17	CityGML 2.0	[null]	BuildingRoom	room	usage	v_codelist	value	description	data_model = 'CityGML 2.0' AND name = 'RoomFunctionUsage'
18	CityGML 2.0	[null]	BuildingFurnit...	building_furnit...	class	v_codelist	value	description	data_model = 'CityGML 2.0' AND name = 'BuildingFurnitureClass'
19	CityGML 2.0	[null]	BuildingFurnit...	building_furnit...	function	v_codelist	value	description	data_model = 'CityGML 2.0' AND name = 'BuildingFurnitureFunctionUsage'
20	CityGML 2.0	[null]	BuildingFurnit...	building_furnit...	usage	v_codelist	value	description	data_model = 'CityGML 2.0' AND name = 'BuildingFurnitureFunctionUsage'

Customised codelists

Those in yellow are the fields the must be added by the user. The other ones will be added automatically

NAME: The name indicating the set of mapping rules. It must be the same for all rules belonging to the same group. This label will be the selectable one in the QGIS GUI.

SOURCE_CLASS: The CityGML/ADE* class the CodeList will be associated to (*ADEs are currently not supported)

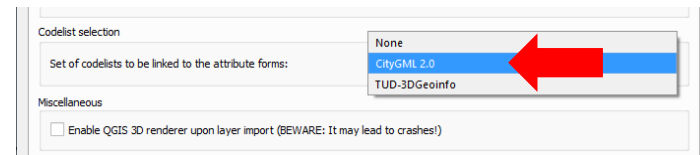
SOURCE_TABLE: The corresponding citydb table which contains the column to be associated to a codelist

SOURCE_COLUMN: The column to be associated to a codelist

ALLOW_MULTI: FALSE if the cardinality is 0..1, TRUE if it is 0..*

NUM_COLUMNS: Number of column presented in the widget and containing look-up values. Default: 1 when ALLOW_MULTI is FALSE, 3 when ALLOW_MULTI is TRUE.

FILTER_EXPRESSION: String containing the expression to filter the values of the desired codelist in the GUI form. It refers to the values of the columns DATA_MODEL and NAME of view V_CODELIST (see previous slides).



eger	name character varyi	ade_prefix character var	source_class character varyin	source_table character varying	source_column character var	target_table character var	key_column character var	value_column character vary	filter_expression character varying	num_columns integer	allow_multi boolean	allow_null_value boolean	order_by_value boolean	use_comp boolean
1	CityGML 2.0	[null]	Bridge	bridge	class	v_codelist	value	description	data_model = 'CityGML 2.0' AND name = '_AbstractBridgeClass'	1	false	true	true	false
2	CityGML 2.0	[null]	Bridge	bridge	function	v_codelist	value	description	data_model = 'CityGML 2.0' AND name = '_AbstractBridgeFunctionUsage'	3	true	true	true	false
3	CityGML 2.0	[null]	Bridge	bridge	usage	v_codelist	value	description	data_model = 'CityGML 2.0' AND name = '_AbstractBridgeFunctionUsage'	3	true	true	true	false
4	CityGML 2.0	[null]	BridgePart	bridge	class	v_codelist	value	description	data_model = 'CityGML 2.0' AND name = '_AbstractBridgeClass'	1	false	true	true	false
5	CityGML 2.0	[null]	BridgePart	bridge	function	v_codelist	value	description	data_model = 'CityGML 2.0' AND name = '_AbstractBridgeFunctionUsage'	3	true	true	true	false
6	CityGML 2.0	[null]	BridgePart	bridge	usage	v_codelist	value	description	data_model = 'CityGML 2.0' AND name = '_AbstractBridgeFunctionUsage'	3	true	true	true	false

Codelists and look-up tables

To add values to the **CODELIST_LOOKUP_CONFIG** table, the user can for example issue a SQL statement such as:

```
-- Optionally, delete previously loaded values belonging to the same codelist group in user schema "qgis_giorgio"
DELETE FROM qgis_giorgio.codelist_lookup_config WHERE name = 'StarWarsCodelist';

-- Insert the mapping rules in one single SQL statement
INSERT INTO qgis_giorgio.codelist_lookup_config
(name, source_class, source_table, source_column, allow_multi, num_columns, filter_expression)
VALUES
('StarWarsCodelist', 'Building' , 'building', 'class' , FALSE, 1, 'data_model = "StarWarsCoruscant" AND name =
"CoruscantBdgClass"'),
('StarWarsCodelist', 'Building' , 'building', 'function' , TRUE , 3, 'data_model = "StarWarsCoruscant" AND name =
"CoruscantBdgFunction"'),
('StarWarsCodelist', 'Building' , 'building', 'usage' , TRUE , 3, 'data_model = "StarWarsCoruscant" AND name =
"CoruscantBdgUsage"'),
('StarWarsCodelist', 'Building' , 'building', 'roof_type' , FALSE, 1, 'data_model = "StarWarsCoruscant" AND name =
"CoruscantBdgRoofType"');
```

This SQL statement can be adapted by changing only the parts in **red**. For the qgis_pkg, use table CODELIST_LOOKUP_CONFIG_TEMPLATE instead.

Motivation

Plugin
overview

Software
installation

Software
use

Advanced
options

Software
uninstall

Current
limitations

QGIS
Package

Resources

Software uninstall

Uninstallation consists of two parts:

1) Partial/complete **removal of the QGIS Package** from PostgreSQL

- This operation can be carried out only by the database administrator
- The administrator can choose to drop only the schema of a selected user (e.g. "qgis_giorgio")
- The administrator can remove all user schemas AND the qgis_pkg schema

2) **Removal of the 3DCityDB-Tools plugin** from QGIS

- This operation can be carried out by any user
- The plugin can be uninstalled from the \Plugins\Manage and Install Plugins... Menu in QGIS
- Alternatively, it can be uninstalled manually by simply removing the plugin folder

Drop user schema

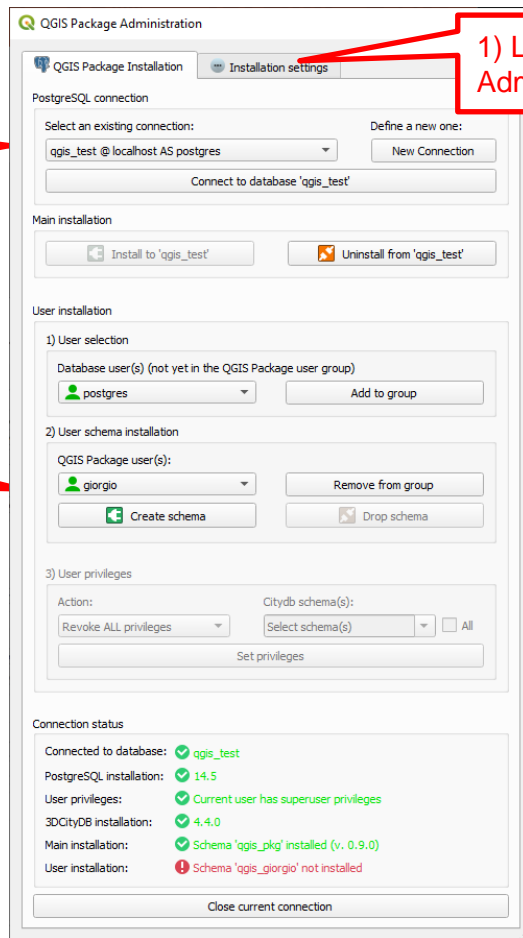
Motivation
 Plugin overview
 Software installation
 Software use
 Advanced options
Software uninstall
 Current limitations
 QGIS Package
 Resources
 Changelog

2) Connect as administrator

3) Select the user

1) Load the "QGIS Package Administration" GUI

4) Drop the schema of the selected user.
 All privileges will be automatically reset to "None"



QGIS Package Administration

QGIS Package Installation | Installation settings

PostgreSQL connection

Select an existing connection: qgis_test @ localhost AS postgres Define a new one: New Connection

Connect to database 'qgis_test'

Main installation

Install to 'qgis_test' Uninstall from 'qgis_test'

User installation

1) User selection

Database user(s) (not yet in the QGIS Package user group)

postgres Add to group

2) User schema installation

QGIS Package user(s):

giorgio Remove from group

Create schema Drop schema

3) User privileges

Action: Revoke ALL privileges Citydb schema(s): Select schema(s) ☐ All

Set privileges

Connection status

Connected to database: ✓ qgis_test

PostgreSQL installation: ✓ 14.5

User privileges: ✓ Current user has superuser privileges

3DCityDB installation: ✓ 4.4.0

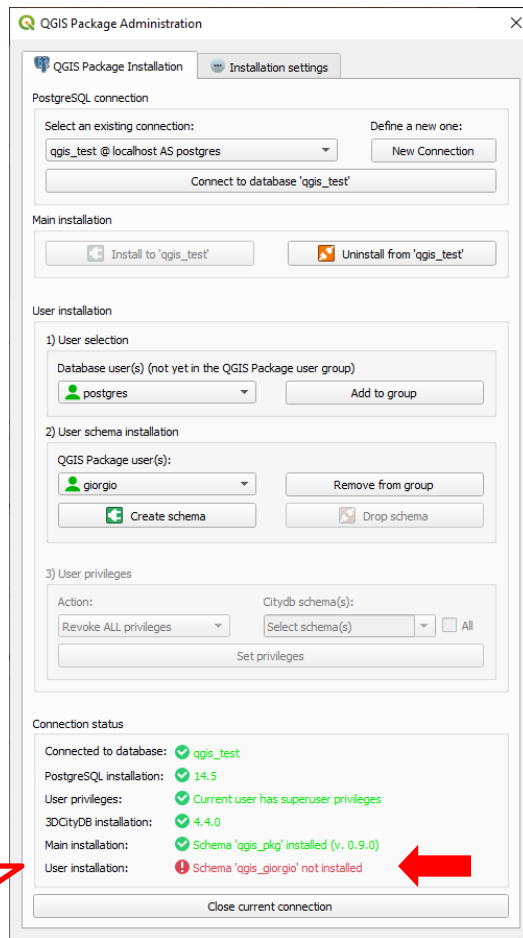
Main installation: ✓ Schema 'qgis_pkg' installed (v. 0.9.0)

User installation: ✗ Schema 'qgis_giorgio' not installed

Close current connection

Drop user schema

Motivation
 Plugin overview
 Software installation
 Software use
 Advanced options
Software uninstall
 Current limitations
 QGIS Package
 Resources
 Changelog



QGIS Package Administration

QGIS Package Installation Installation settings

PostgreSQL connection

Select an existing connection: qgis_test @ localhost AS postgres Define a new one: New Connection

Connect to database 'qgis_test'

Main installation

Install to 'qgis_test' Uninstall from 'qgis_test'

User installation

1) User selection

Database user(s) (not yet in the QGIS Package user group)

postgres Add to group

2) User schema installation

QGIS Package user(s): giorgio Remove from group

Create schema Drop schema

3) User privileges

Action: Revoke ALL privileges Citydb schema(s): Select schema(s) All

Set privileges

Connection status

Connected to database: ✓ qgis_test

PostgreSQL installation: ✓ 14.5

User privileges: ✓ Current user has superuser privileges

3DCityDB installation: ✓ 4.4.0

Main installation: ✓ Schema 'qgis_pkg' installed (v. 0.9.0)

User installation: ! Schema 'qgis_giorgio' not installed

Close current connection

5) The user installation field is now red again

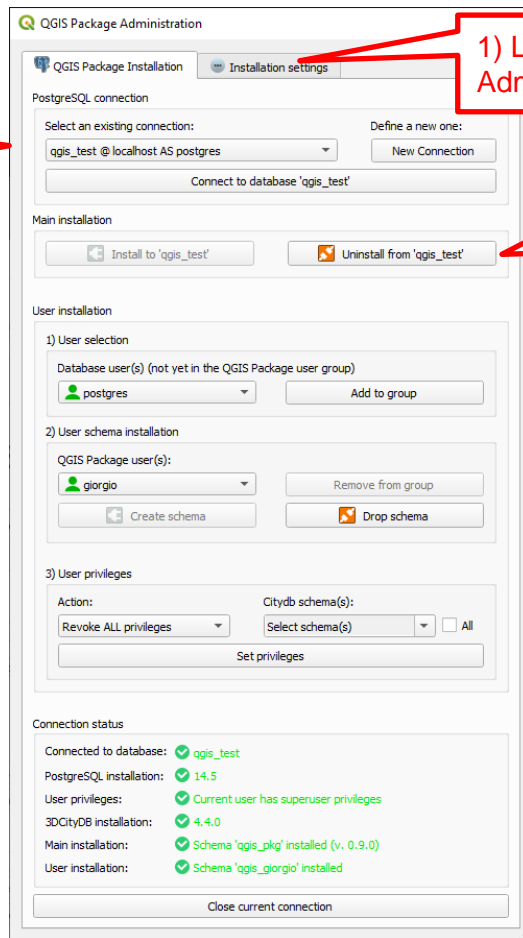
Software uninstall: Uninstall QGIS Package

Motivation
 Plugin overview
 Software installation
 Software use
 Advanced options
Software uninstall
 Current limitations
 QGIS Package
 Resources
 Changelog

2) Connect as administrator

1) Load the "QGIS Package Administration" GUI

3) Uninstall the QGIS Package from the current database



The screenshot shows the "QGIS Package Administration" window with the "Installation settings" tab selected. The "PostgreSQL connection" section shows a dropdown menu with "qgis_test @ localhost AS postgres" selected and a "New Connection" button. Below it is a "Connect to database 'qgis_test'" button. The "Main installation" section has two buttons: "Install to 'qgis_test'" and "Uninstall from 'qgis_test'". The "User installation" section is divided into three parts: 1) User selection, where "postgres" is selected as the database user; 2) User schema installation, where "giorgio" is selected as the QGIS Package user; and 3) User privileges, where "Revoke ALL privileges" is selected for the "Citydb" schema. At the bottom, the "Connection status" section shows a list of installed components and their versions, all with green checkmarks indicating successful installation.

QGIS Package Administration

QGIS Package Installation Installation settings

PostgreSQL connection

Select an existing connection: qgis_test @ localhost AS postgres Define a new one: New Connection

Connect to database 'qgis_test'

Main installation

Install to 'qgis_test' Uninstall from 'qgis_test'

User installation

1) User selection

Database user(s) (not yet in the QGIS Package user group)

postgres Add to group

2) User schema installation

QGIS Package user(s):

giorgio Remove from group

Create schema Drop schema

3) User privileges

Action: Revoke ALL privileges Citydb schema(s): Select schema(s) All

Set privileges

Connection status

Connected to database: ✓ qgis_test

PostgreSQL installation: ✓ 14.5

User privileges: ✓ Current user has superuser privileges

3DCityDB installation: ✓ 4.4.0

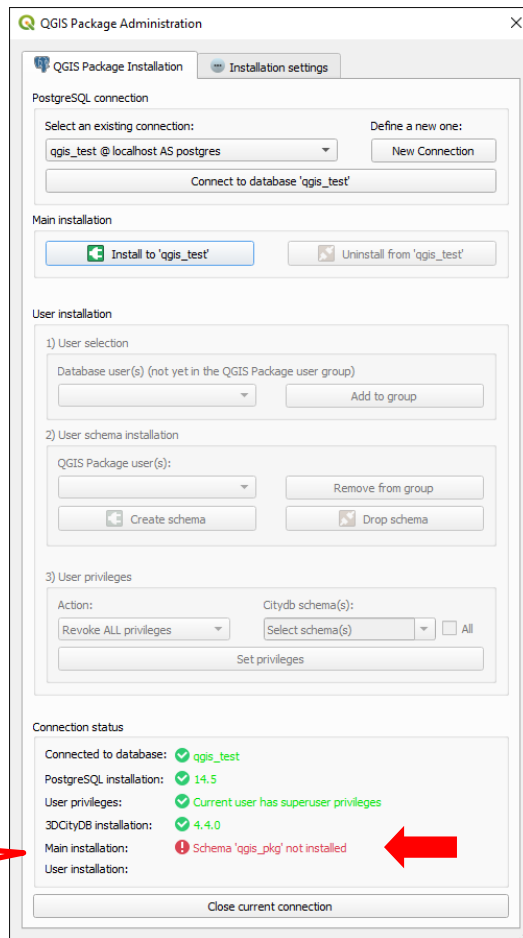
Main installation: ✓ Schema 'qgis_pkg' installed (v. 0.9.0)

User installation: ✓ Schema 'qgis_giorgio' installed

Close current connection

Uninstall QGIS Package

Motivation
 Plugin overview
 Software installation
 Software use
 Advanced options
Software uninstall
 Current limitations
 QGIS Package
 Resources
 Changelog



The screenshot shows the 'QGIS Package Administration' window with the 'Installation settings' tab selected. The 'PostgreSQL connection' section shows a dropdown for 'qgis_test @ localhost AS postgres' and a 'New Connection' button. The 'Main installation' section has 'Install to 'qgis_test'' and 'Uninstall from 'qgis_test'' buttons. The 'User installation' section has three sub-sections: '1) User selection', '2) User schema installation', and '3) User privileges'. The 'Connection status' section at the bottom shows the following status:

- Connected to database: ✓ qgis_test
- PostgreSQL installation: ✓ 14.5
- User privileges: ✓ Current user has superuser privileges
- 3DCityDB installation: ✓ 4.4.0
- Main installation: ✗ Schema 'qgis_pkg' not installed
- User installation:

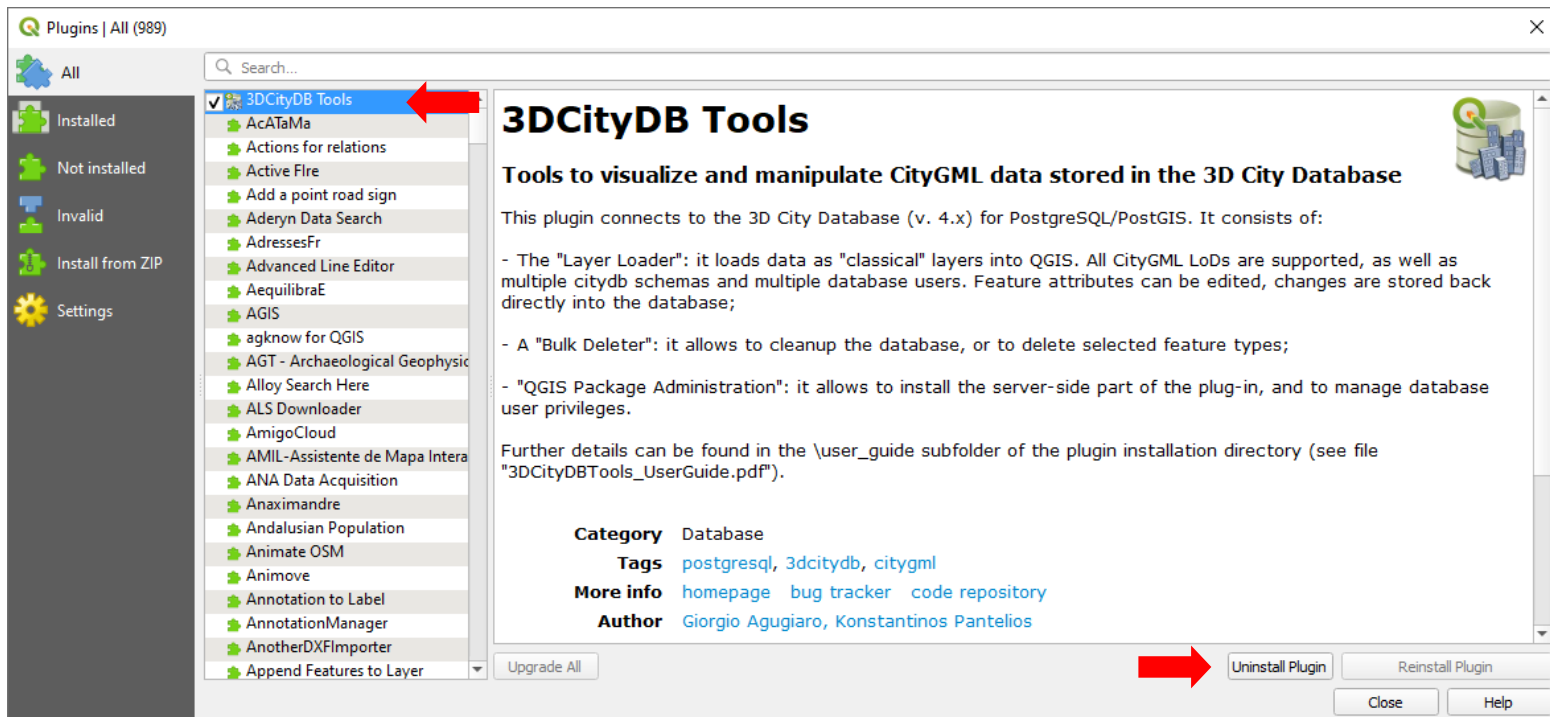
A red arrow points to the 'Main installation' status, which is highlighted in red.

4) The main installation field is now red again

Uninstall 3DCityDB-Tools

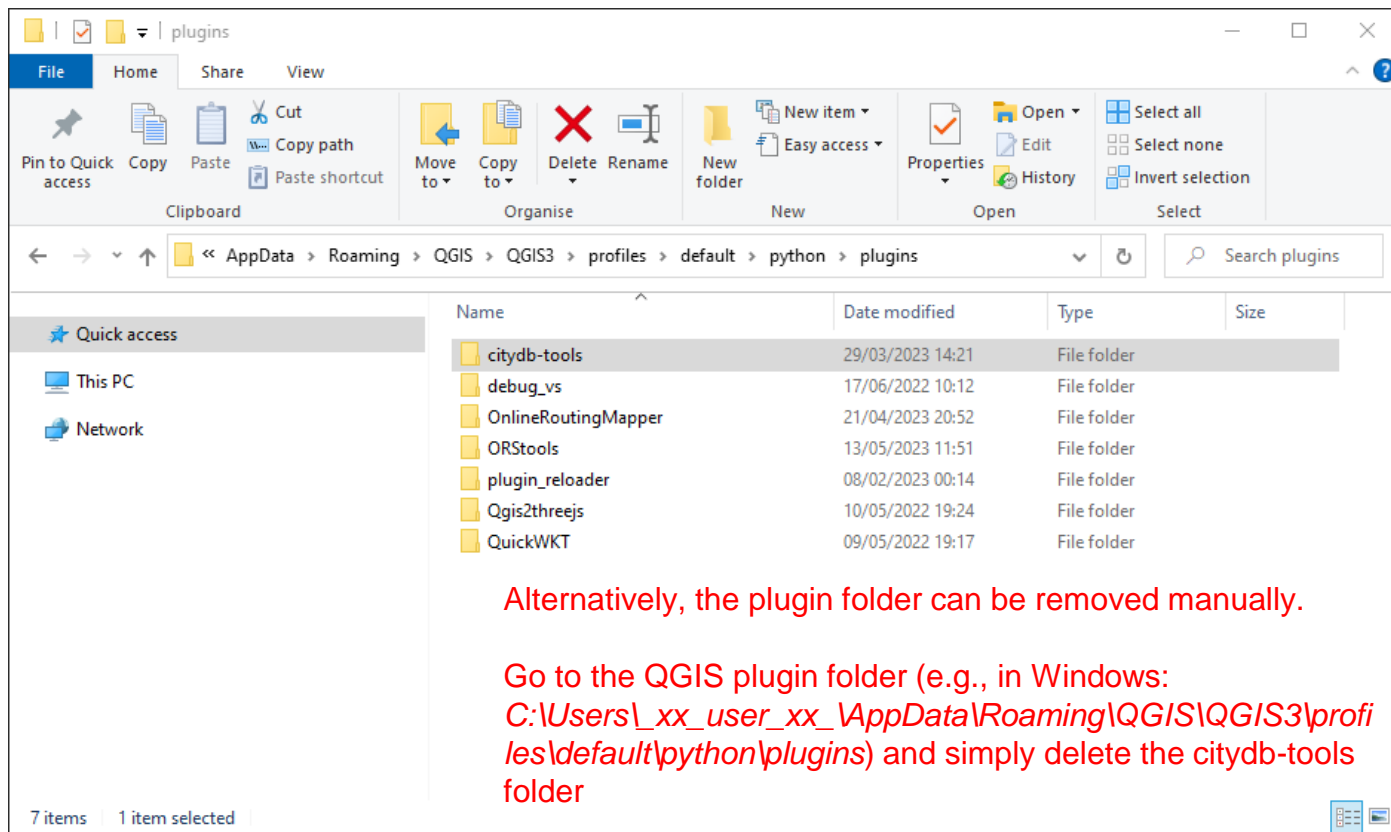
The plugin can be uninstalled from the \Plugins\Manage and Install Plugins... Menu in QGIS

Motivation
 Plugin overview
 Software installation
 Software use
 Advanced options
Software uninstall
 Current limitations
 QGIS Package
 Resources
 Changelog



Uninstall 3DCityDB-Tools

Motivation
 Plugin overview
 Software installation
 Software use
 Advanced options
Software uninstall
 Current limitations
 QGIS Package
 Resources
 Changelog



Alternatively, the plugin folder can be removed manually.

Go to the QGIS plugin folder (e.g., in Windows:
C:\Users_xx_user_xx\AppData\Roaming\QGIS\QGIS3\profiles\default\python\plugins) and simply delete the citydb-tools folder

Motivation

Plugin
overview

Software
installation

Software
use

Advanced
options

Software
uninstall

Current
limitations

QGIS
Package

Resources

Current limitations

Motivation

Plugin overview

Software installation

Software use

Advanced options

Software uninstall

Current limitations

QGIS Package

Resources

Changelog

In general:

- CityGML appearances are not supported
- The Layer Loader does not support CityObjectGroups
- CityGML ADEs (Application Domain Extensions) are not supported

The QGIS Package does not support:

- Raster-based Relief features
- Generation of layers for CityObjectGroups

Motivation

Plugin
overview

Software
installation

Software
use

Advanced
options

Software
uninstall

Current
limitations

QGIS
Package

Resources

QGIS Package in a nutshell

Motivation

Plugin overview

Software installation

Software use

Advanced options

Software uninstall

Current limitations

QGIS Package

Resources

Changelog

- It represents the **server-side part** of the QGIS plugin
 - Most of the actions carried out from the QGIS GUI can be actually performed also by interacting directly with the database (e.g. using **PgAdmin**)
- It can be installed and used independently from the QGIS client-side part
 - E.g. with FME, or programmatically via Python, SQLAlchemy, etc.
- It requires
 - PostgreSQL **v. 10 or higher**
 - An existing installation of the 3DCityDB **v. 4.x**
- All relevant entities are installed in the "**qgis_pkg**" database schema
 - Database types
 - Triggers and trigger functions
 - Functions
 - Tables, mostly used as templates for the user schemas

QGIS Package in a nutshell

Motivation

Plugin overview

Software installation

Software use

Advanced options

Software uninstall

Current limitations

QGIS Package

Resources

Changelog

The database administrator can:

- Create user schemas
- Grant/revoke privileges per user and per citydb schema
- Create, refresh, drop layers
- Drop user schemas

The required SQL functions are all available in schema **qgis_pkg**.

QGIS Package: Create user schemas

Motivation

Plugin overview

Software installation

Software use

Advanced options

Software uninstall

Current limitations

QGIS Package

Resources

Changelog

The database administrator can create user schemas for specific database users.

For example:

- User "giorgio" -> schema "qgis_giorgio"
- User "konstantinos" -> schema "qgis_konstantinos"
- User "postgres" -> schema "qgis_postgres"

Each user schema will be used only by the corresponding user

Each user schema is accessible only by the corresponding user (and the database superusers)

In a user schema all necessary tables, updatable views, materialized views etc. will be created

SQL example

-- Create the schema for user "giorgio". It will create schema "qgis_giorgio" in the current database

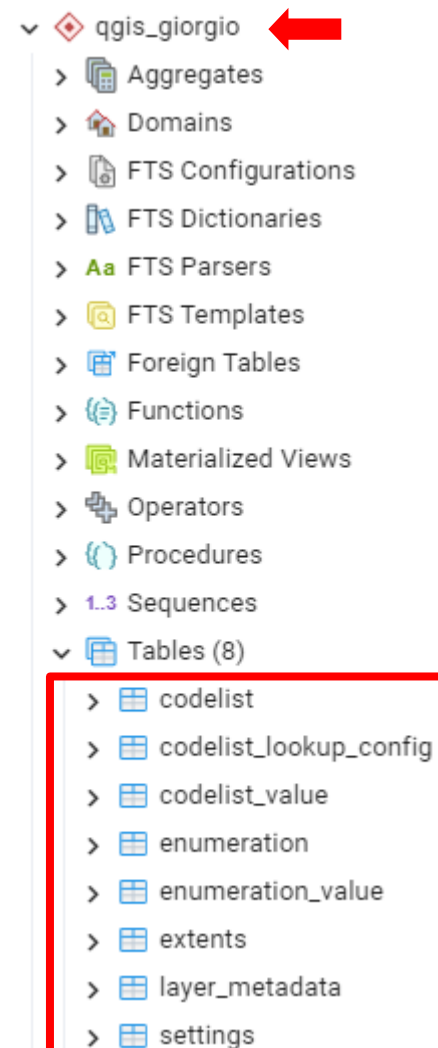
SELECT qgis_pkg.create_qgis_usr_schema('giorgio');

QGIS Package: Create user schemas

User schema overview

In each user schema (e.g. "qgis_giorgio") some tables are generated upon schema creation:

- Tables **CODELIST**, **CODELIST_LOOKUP_CONFIG** and **CODELIST_VALUE** are used to store codelists and related settings. See slides on **advanced options** for more details
- Tables **ENUMERATION** and **ENUMERATION_VALUE** are used to store enumerations
- Table **EXTENTS** contains the bounding boxes of the citydb schemas and those of the materialized views
- Table **LAYER_METADATA** contains information about generated and refreshed layers
- Table **SETTINGS** is used to store the user's settings (from the QGIS GUI)



QGIS Package: User management

The database administrator can grant read-only (ro) or read-write (rw) privileges to the users.

- Privileges can be granted to a specific citydb schema. Please note the **cdb_schema** parameter, requiring a single varchar value with the specific schema name

SQL examples

-- Database user "giorgio" is added to group "qgis_pkg_usrgroup_qgis_test", can access data in citydb schema "citydb" of database "qgis_test" with read-only privileges

```
SELECT qgis_pkg.grant_qgis_usr_privileges(usr_name := 'giorgio', priv_type := 'ro', cdb_schema := 'citydb');
```

-- Database user "konstantinos" is added to group "qgis_pkg_usrgroup_qgis_db", can access data in citydb schema "citydb_2" of database "qgis_db" with read-write privileges

```
SELECT qgis_pkg.grant_qgis_usr_privileges(usr_name := 'konstantinos', priv_type := 'rw', cdb_schema := 'citydb_2');
```

QGIS Package: User management

The database administrator can grant read-only (ro) or read-write (rw) privileges to the users

- Alternatively, privileges can be granted to multiple (or all) citydb schemas at the same time; Please note the **cdb_schemas** (with s!) parameter, requiring an array of varchar values containing the schema names, or NULL for all schemas

SQL examples

-- Database user "camilo" is added to group "qgis_pkg_usrgroup_starwars", can access data in citydb schemas "alderaan" and "hoth" of the current database "starwars" with read-only privileges

```
SELECT qgis_pkg.grant_qgis_usr_privileges(usr_name := 'camilo', priv_type := 'ro', cdb_schemas :=  
ARRAY['alderaan', 'hoth']);
```

-- Database user "giorgio" is added to group "qgis_pkg_usrgroup_starwars", can access data in ALL citydb schemas of the current database "starwars" with read-write privileges

```
SELECT qgis_pkg.grant_qgis_usr_privileges(usr_name := 'giorgio', priv_type := 'rw', cdb_schemas := NULL);
```

QGIS Package: User management

The database administrator can revoke privileges from the users

SQL examples

-- Database user "giorgio" cannot access anymore data in citydb schema "citydb" of the current database

```
SELECT qgis_pkg.revoke_qgis_usr_privileges('giorgio', 'citydb');
```

-- Database user "camilo" cannot access anymore ANY citydb schemas of the current database

```
SELECT qgis_pkg.revoke_qgis_usr_privileges('camilo');
```

IMPORTANT: The database users are NOT automatically removed from the group "qgis_pkg_usrgroup_{cdb_schema}". If required, the administrator must remove them manually (or use the QGIS plug-in GUI).

-- Database user "giorgio" is removed from group "qgis_pkg_usrgroup_starwars" and won't be able to use the QGIS plugin anymore for the database "starwars"

```
REVOKE qgis_pkg_usrgroup_starwars FROM giorgio;
```

QGIS Package: Layer management

Motivation

Plugin overview

Software installation

Software use

Advanced options

Software uninstall

Current limitations

QGIS Package

Resources

Changelog

The database administrator can create layers with function **qgis_pkg.create_layers(...)**.

- All materialized views and updatable views are created, but only if corresponding data exist in the database
- The user can create layers only for selected CityGML modules using the similar functions:
 - **qgis_pkg.create_layers_bridge(...)**
 - **qgis_pkg.create_layers_building(...)**
 - ...
 - **qgis_pkg.create_layers_waterbody(...)**
- All functions are in schema **qgis_pkg** and have the same signature (see next slide)

QGIS Package: Layer management

Function

qgis_pkg.create_layers(usr_schema, cdb_schema [, perform_snapping] [, digits] [, area_poly_min] [, bbox_corners_array] [, is_geographic] [, force_layer_creation])

Parameter	Type	Description
usr_schema	varchar	The database user schema, e.g. "qgis_giorgio".
cdb_schema	varchar	the citydb schema where data are stored, e.g. "citycb", or "citydb2".
perform_snapping	integer	DEFAULT 0 (i.e. disabled). If 1, geometry simplification is performed. Unused if perform_snapping is set to 0.
digits	integer	DEFAULT 3. Number of decimal positions to keep during geometry simplification. Unused if perform_snapping is set to 0.
area_poly_min	numeric	DEFAULT 0.001 [m ²]. Minimum polygon area during geometry simplification. Unused if perform_snapping is set to 0.
bbox_corners_array	numeric[]	DEFAULT Null, i.e. the extents of the whole <i>cdb_schema</i> . Otherwise, extents of the materialized views to be created, e.g. {x_min, y_min, x_max, y_max}. Coordinates must be in the same SRS as the cdb_schema!
is_geographic	boolean	DEFAULT False. True is the coordinate system of the citydb schema has geographic coordinates, False if it has projected coordinates.
force_layer_creation	boolean	DEFAULT False. Otherwise: force creation of all layers, also the empty ones.

QGIS Package: Layer management

SQL examples

-- For user "giorgio", create all layers for existing data in citydb schema "citydb"

```
SELECT qgis_pkg.create_layers('giorgio', 'citydb');
```

-- For user "giorgio", create all layers for existing data in citydb schema "citydb2" and perform geometry simplification with 2 decimal places and 0.01 m² minimum area for polygons

```
SELECT qgis_pkg.create_layers('giorgio', 'citydb', 1, 2, 0.01);
```

-- For user "camilo", create all building module layers for existing data in citydb schema "vienna"

```
SELECT qgis_pkg.create_layers_building('camilo', 'vienna');
```

-- For user "konstantinos", create all waterbody module layers for existing data in citydb schema "alderaan" within a certain bounding box. The bounding box is defined by the lower-left and upper-right coordinates (i.e. x_min, y_min, x_max, y_max)

```
SELECT qgis_pkg.create_layers_waterbody('konstantinos', 'alderaan', bbox_corners_array := '{10, 20, 110, 220}');
```

QGIS Package: Layer management

SQL examples

-- The following query works, but actually it is not written in a user-friendly way.

```
SELECT qgis_pkg.create_layers('giorgio', 'citydb', 1, 2, 0.01);
```

-- In general, therefore, it is always a good habit to use **named parameters** when calling functions!

```
SELECT qgis_pkg.create_layers(  
    usr_name := 'giorgio',  
    cdb_schema := 'citydb',  
    perform_snapping := 1,  
    digits := 2,  
    area_poly_min := 0.01)
```


QGIS Package: Layer management

After creating the layers, you may (optionally) want to register also their bounding box in the EXTENTS table of the respective qgis_schema. This will enable the plugin in QGIS to "see" and represent the extents also from the plugin GUI.

SQL examples

-- In user schema "qgis_giorgio", update the extents for the whole dataset in citydb schema "citydb"

```
SELECT qgis_pkg.upsert_extents('qgis_giorgio', 'citydb', 'db_schema');
```

-- In user schema "qgis_giorgio", update the extents for the user-defined bounding box in citydb schema "citydb"

```
SELECT qgis_pkg.upsert_extents(  
    usr_schema := 'qgis_giorgio',  
    cdb_schema := 'citydb',  
    cdb_bbox_type := 'm_view', -- must be one of ('db_schema', 'm_view', 'qgis')  
    cdb_envelope := ST_Envelope('LINESTRING(232038 480366, 232600 480856)::geometry),  
    is_geographic := FALSE);
```

Please note: the *cdb_envelope* parameter requires a PostGIS geometry that is a polygon. As long as this requirement is fulfilled, other PostGIS functions can be used, for example:

- ST_GeomfromText('POLYGON(...))')
- ST_MakePolygon(...)

The SRID can be omitted as it is automatically set based on the one from the selected citydb schema. The coordinates must be however in the same CRS of the selected citydb!

QGIS Package: Layer management

Motivation

Plugin overview

Software installation

Software use

Advanced options

Software uninstall

Current limitations

QGIS Package

Resources

Changelog

The database administrator can refresh the materialized views with function **qgis_pkg.refresh_layers(...)**.

- All materialized views created before will be refreshed. This is necessary every time the layers are generated (or re-generated using a different bounding box).
- The user can refresh the materialized views only for selected CityGML modules using the similar functions:
 - **qgis_pkg.refresh_layers_bridge(...)**
 - **qgis_pkg.refresh_layers_building(...)**
 - ...
 - **qgis_pkg.refresh_layers_waterbody(...)**
- All functions are in schema `qgis_pkg` and have the same signature (see next slide)

QGIS Package: Layer management

- Motivation
- Plugin overview
- Software installation
- Software use
- Advanced options
- Software uninstall
- Current limitations
- QGIS Package**
- Resources
- Changelog

Function

qgis_pkg.refresh_layers(usr_schema, cdb_schema)

Parameter	Type	Description
usr_schema	varchar	The database user schema, e.g. "qgis_giorgio".
cdb_schema	varchar	the citydb schema where data are stored, e.g. "citycb", or "citydb2".

SQL examples

-- In user schema "qgis_giorgio", refresh all materialized views in citydb schema "citydb"

SELECT qgis_pkg.refresh_layers('qgis_giorgio', 'citydb');

-- In user schema "qgis_konstantinos", refresh all waterbody module materialized views in citydb schema "alderaan"

SELECT qgis_pkg.refresh_layers_waterbody('qgis_konstantinos', 'alderaan');

QGIS Package: Layer management

Motivation

Plugin overview

Software installation

Software use

Advanced options

Software uninstall

Current limitations

QGIS Package

Resources

Changelog

The database administrator can drop layers with function **qgis_pkg.drop_layers(...)**.

- All existing layers in the selected user schema and related to the selected citydb schema will be dropped
- The user can drop layers only for selected CityGML modules using the similar functions:
 - **qgis_pkg.drop_layers_bridge(...)**
 - **qgis_pkg.drop_layers_building(...)**
 - ...
 - **qgis_pkg.drop_layers_waterbody(...)**
- All functions are in schema **qgis_pkg** and have the same signature (see next slide)

QGIS Package: Layer management

- Motivation
- Plugin overview
- Software installation
- Software use
- Advanced options
- Software uninstall
- Current limitations
- QGIS Package**
- Resources
- Changelog

Function

qgis_pkg.drop_layers(usr_schema, cdb_schema)

Parameter	Type	Description
usr_schema	varchar	The database user schema, e.g. "qgis_giorgio".
cdb_schema	varchar	the citydb schema where data are stored, e.g. "citycb", or "citydb2".

SQL examples

-- In user schema "qgis_giorgio", drop all layers related to citydb schema "citydb"

SELECT qgis_pkg.drop_layers('qgis_giorgio', 'citydb');

-- In user schema "qgis_konstantinos", drop all waterbody module layers related to citydb schema "alderaan"

SELECT qgis_pkg.drop_layers_waterbody('qgis_konstantinos', 'alderaan');

QGIS Package: Detail views

Detail views are children tables containing additional layer data (e.g. generic attributes, external references, etc.)

Optionally, they can be created programmatically and dropped in a similar way as normal layers. If created, they will be used (also) by the plugin GUI in QGIS to link these views to the attribute forms as nested tables.

SQL examples

-- For user "giorgio", create the detail views for citydb schema "citydb"

```
SELECT qgis_pkg.create_details_view('giorgio', 'citydb');
```

-- For user "giorgio", create the detail views for citydb schema "citydb" inside the user-defined bounding box

```
SELECT qgis_pkg.create_detail_view(  
    usr_name := 'giorgio',  
    cdb_schema := 'citydb',  
    bbox_corners_array := ARRAY[ 232038, 480366, 232600, 480856],  
    is_geographic := FALSE);
```

-- In user schema "qgis_giorgio", drop the detail views for citydb schema "citydb"

```
SELECT qgis_pkg.drop_detail_view('qgis_giorgio', 'citydb');
```

QGIS Package: Drop user schema

The database administrator can drop a user schema

FIRST, revoke privileges of the user for all citydb schemas, THEN drop the user schema. Please refer to the previous slides for more details about user privileges.

SQL example

-- First revoke all ro/rw privileges of user "giorgio" for all citydb schemas

```
SELECT qgis_pkg.revoke_qgis_usr_privileges('giorgio');
```

-- Then drop the layers using the drop_layer_x functions

```
SELECT qgis_pkg.drop_layers_building('qgis_giorgio');
```

```
SELECT qgis_pkg.drop_layers_bridge('qgis_giorgio');
```

...

-- Then drop the user schema

```
DROP SCHEMA qgis_giorgio CASCADE;
```

-- Optionally (if necessary) remove user "giorgio" from the "qgis_pkg_usrgroup_qgis_test" associated to database "qgis_test"

```
REVOKE qgis_pkg_usrgroup_qgis_test FROM giorgio;
```

QGIS Package via FME

This is a simple example of how the QGIS package can be used via FME

- Simply connect to the 3D City Database and import the views with **PostGIS** readers

Motivation

Plugin overview

Software installation

Software use

Advanced options

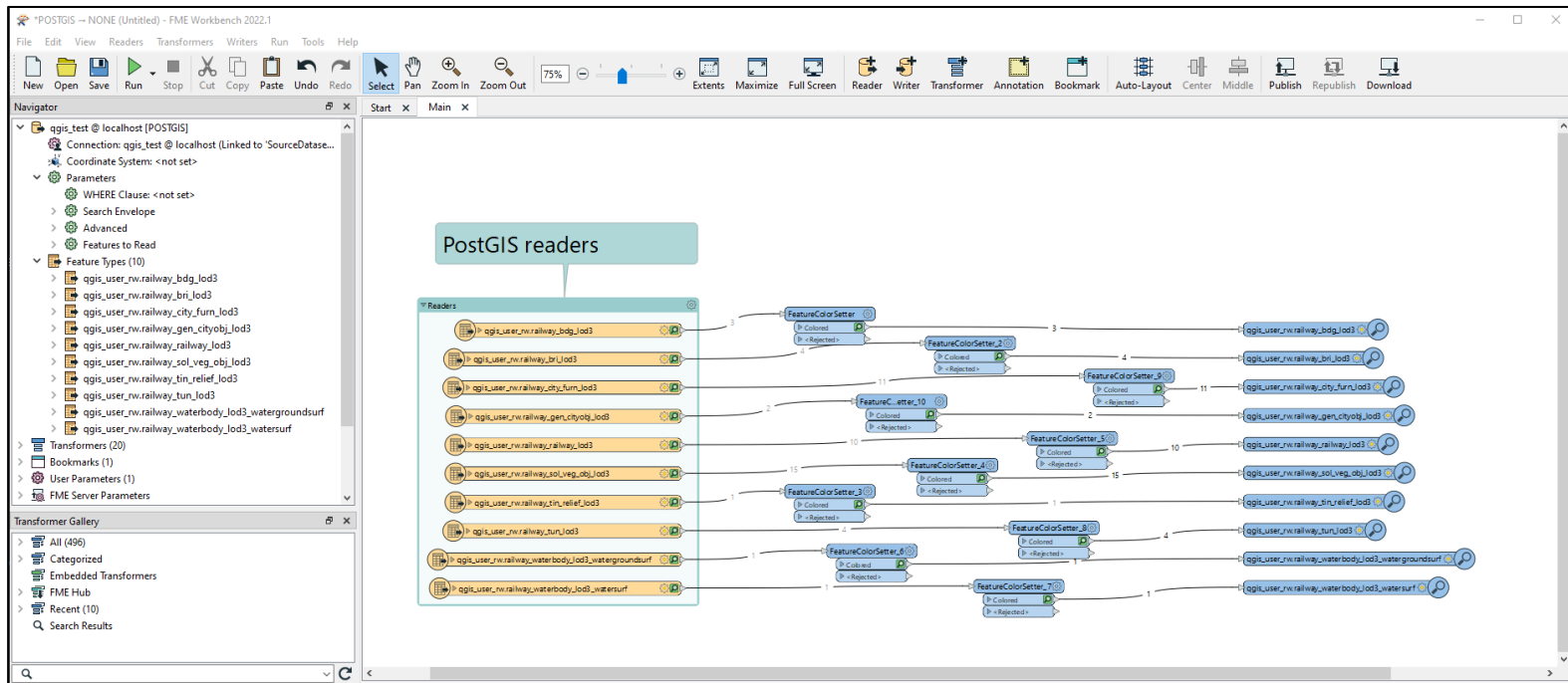
Software uninstall

Current limitations

QGIS Package

Resources

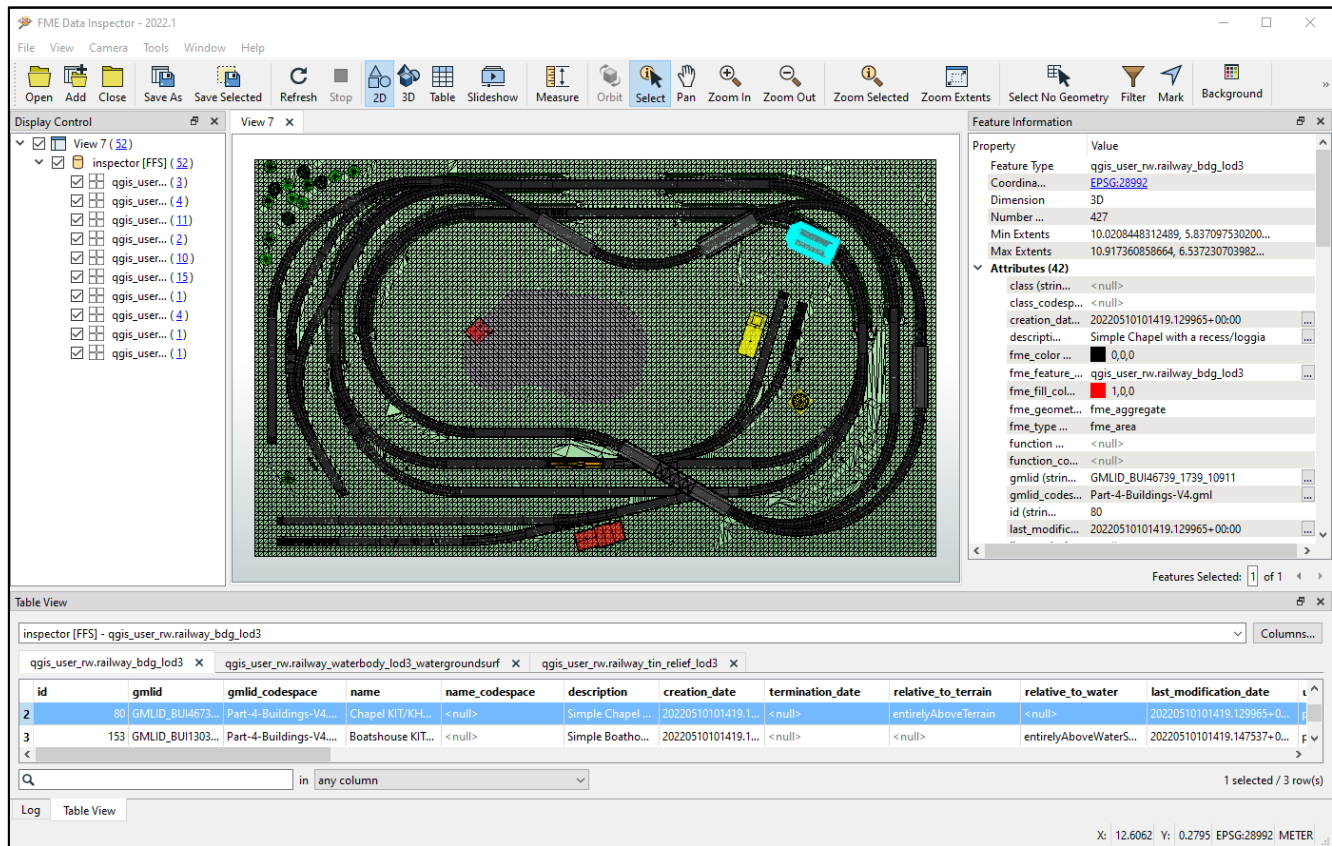
Changelog



QGIS Package via FME

- 2D visualisation via FME Data Inspector

Motivation
 Plugin overview
 Software installation
 Software use
 Advanced options
 Software uninstall
 Current limitations
QGIS Package
 Resources
 Changelog



Feature Information

Property	Value
Feature Type	qgis_user_rw.railway_bdg_lod3
Coordina...	EPSG:28992
Dimension	3D
Number ...	427
Min Extents	10.0208448312489, 5.837097530200...
Max Extents	10.917360858664, 6.537230703982...

Attributes (42)

Property	Value
class (strin...	<null>
class_codesp...	<null>
creation_dat...	20220510101419.129965+00:00
descripti...	Simple Chapel with a recess/loggia
fme_color ...	0,0,0
fme_feature...	qgis_user_rw.railway_bdg_lod3
fme_fill_col...	1,0,0
fme_geomet...	fme_aggregate
fme_type ...	fme_area
function ...	<null>
function_co...	<null>
gmlid (strin...	GMLID_BUI46739_1739_10911
gmlid_codes...	Part-4-Buildings-V4.gml
id (strin...	80
last_modifi...	20220510101419.129965+00:00

Table View

inspector [FFS] - qgis_user_rw.railway_bdg_lod3

id	gmlid	gmlid_codespace	name	name_codespace	description	creation_date	termination_date	relative_to_terrain	relative_to_water	last_modification_date
2	80	GMLID_BUI46739_1739_10911	Chapel KIT/KH...	<null>	Simple Chapel ...	20220510101419.1...	<null>	entirelyAboveTerrain	<null>	20220510101419.129965+00:00
3	153	GMLID_BUI1303...	Boatshouse KIT...	<null>	Simple Boatho...	20220510101419.1...	<null>	<null>	entirelyAboveWaterS...	20220510101419.147537+00:00

1 selected / 3 row(s)

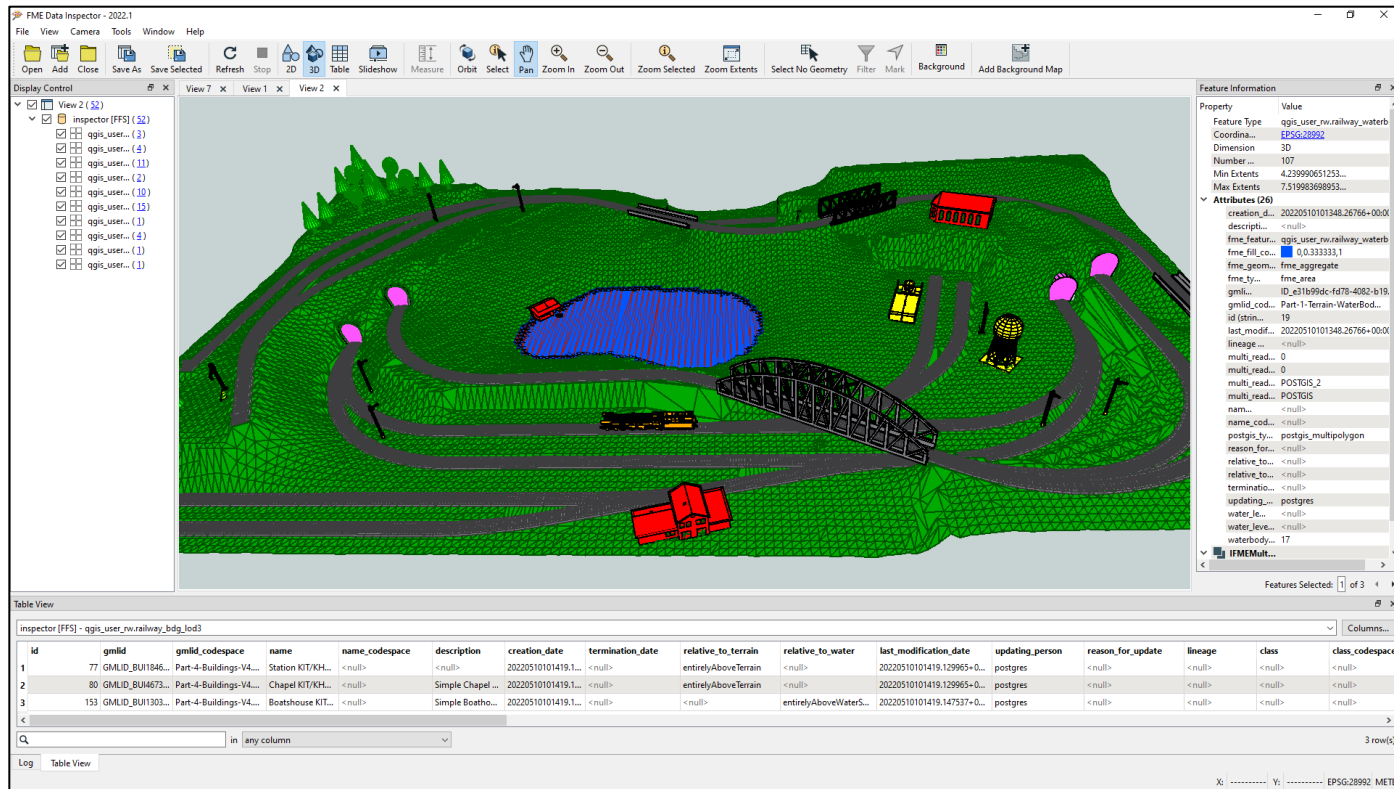
Log Table View

X: 12.6062 Y: 0.2795 EPSG:28992 METER

QGIS Package via FME

- 3D visualisation via FME Data Inspector

Motivation
 Plugin overview
 Software installation
 Software use
 Advanced options
 Software uninstall
 Current limitations
QGIS Package
 Resources
 Changelog



Feature Information

Property	Value
Feature Type	qgis_user_rw_railway_waterb
Coordinate...	EPSG:28992
Dimension	3D
Number	107
Min Extents	4.239990651253...
Max Extents	7.519983689953...
Attributes (26)	
creation_d...	20220510101348.26766+00:00
descript...	<null>
fme_featur...	qgis_user_rw_railway_waterb
fme_fill_co...	0.0333333,1
fme_geom...	fme_aggregate
fme_ty...	fme_area
gmli...	ID_431699dc-fd7b-4082-b19
gmli...	Part-1-Terrain-WaterBod...
id (strin...	19
last_modif...	20220510101348.26766+00:00
lineage...	<null>
multi_read...	0
multi_read...	0
multi_read...	POSTGIS_2
multi_read...	POSTGIS
name...	<null>
name_cod...	<null>
postgis_ty...	postgis_multipolygon
reason_for...	<null>
relative_to...	<null>
relative_to...	<null>
termination...	<null>
updating...	postgres
water_le...	<null>
water_leve...	<null>
waterbody...	17

Table View

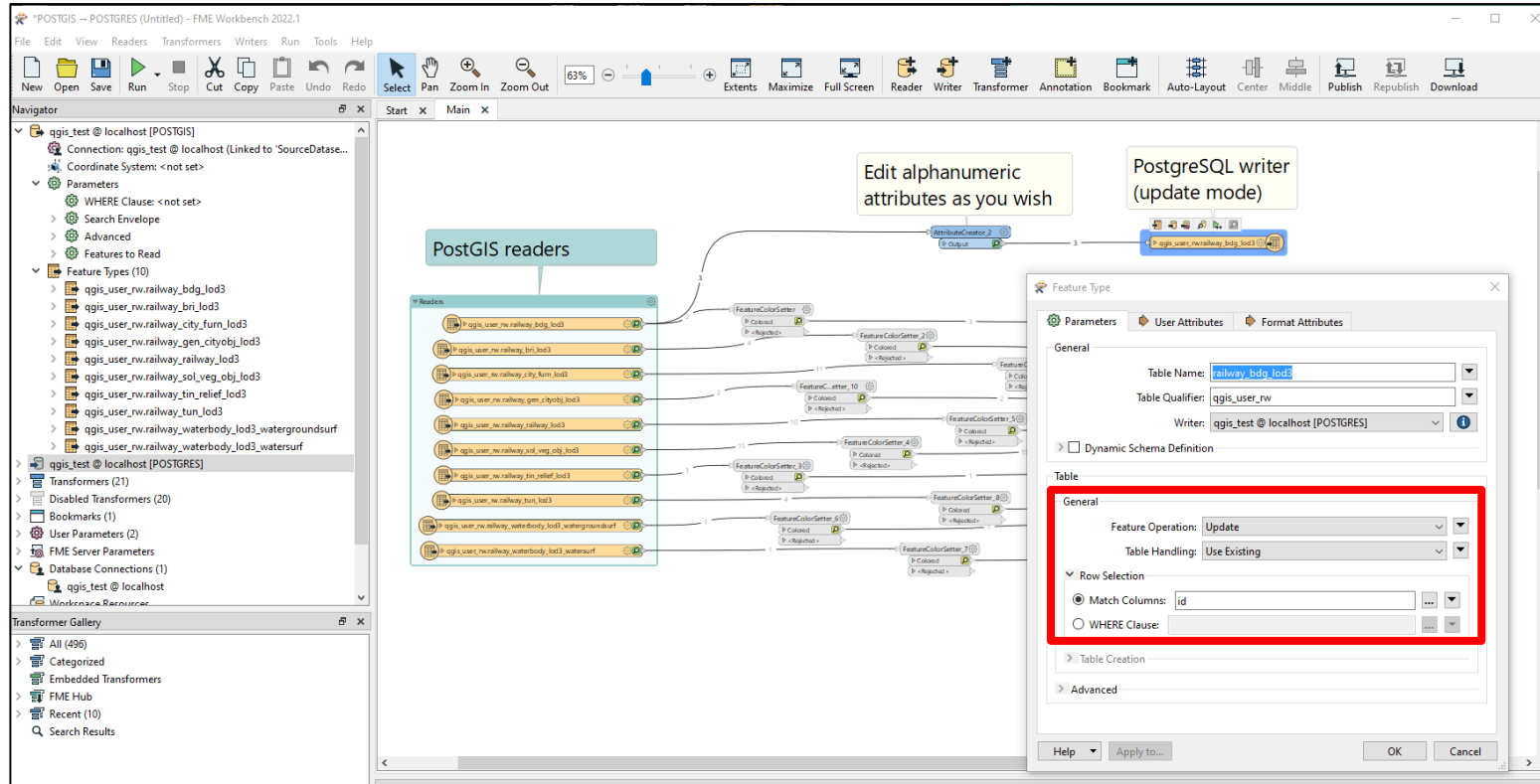
id	gmli...	gmli...	name	name...	description	creation_date	termination_date	relative_to_terrain	relative_to_water	last_modification_date	updating_person	reason_for_update	lineage	class	class_codespace
1	77	GMUID_BU11846...	Part-4-Buildings-V4...	Station KIT/KH...	<null>	20220510101419.1...	<null>	entirelyAboveTerrain	<null>	20220510101419.129965+0...	postgres	<null>	<null>	<null>	<null>
2	80	GMUID_BU4673...	Part-4-Buildings-V4...	Chapel KIT/KH...	<null>	20220510101419.1...	<null>	entirelyAboveTerrain	<null>	20220510101419.129965+0...	postgres	<null>	<null>	<null>	<null>
3	153	GMUID_BU11303...	Part-4-Buildings-V4...	Boathouse KIT...	<null>	20220510101419.1...	<null>	<null>	entirelyAboveWaterS...	20220510101419.147537+0...	postgres	<null>	<null>	<null>	<null>

Log Table View

QGIS Package via FME

- Remember: alphanumeric attributes in the views are updatable! 😊
- You will need a **PostgreSQL writer in update mode**

Motivation
 Plugin overview
 Software installation
 Software use
 Advanced options
 Software uninstall
 Current limitations
QGIS Package
 Resources
 Changelog



The screenshot displays the FME Workbench 2022.1 interface. The main workspace shows a workflow starting with 'PostGIS readers' (labeled 'PostGIS readers' in a blue box) connected to 'AttributeCreator' transformers. These are then connected to a 'PostgreSQL writer' (labeled 'PostgreSQL writer (update mode)' in a yellow box). Annotations include 'Edit alphanumeric attributes as you wish' and 'PostgreSQL writer (update mode)'. The left sidebar shows the 'Navigator' pane with a tree view of the workspace components. The bottom right pane shows the 'Feature Type' dialog for the 'railway_bdg_lod3' table, with the 'Update' operation selected under 'General'.

Feature Type Dialog - General Tab:

- Table Name: `railway_bdg_lod3`
- Table Qualifier: `qgis_user_rw`
- Writer: `qgis_test @ localhost [POSTGRES]`
- Dynamic Schema Definition: ☐
- Feature Operation: **Update**
- Table Handling: **Use Existing**
- Row Selection:
 - ☒ Match Columns: `id`
 - ☐ WHERE Clause:

Motivation

Plugin
overview

Software
installation

Software
use

Advanced
options

Software
uninstall

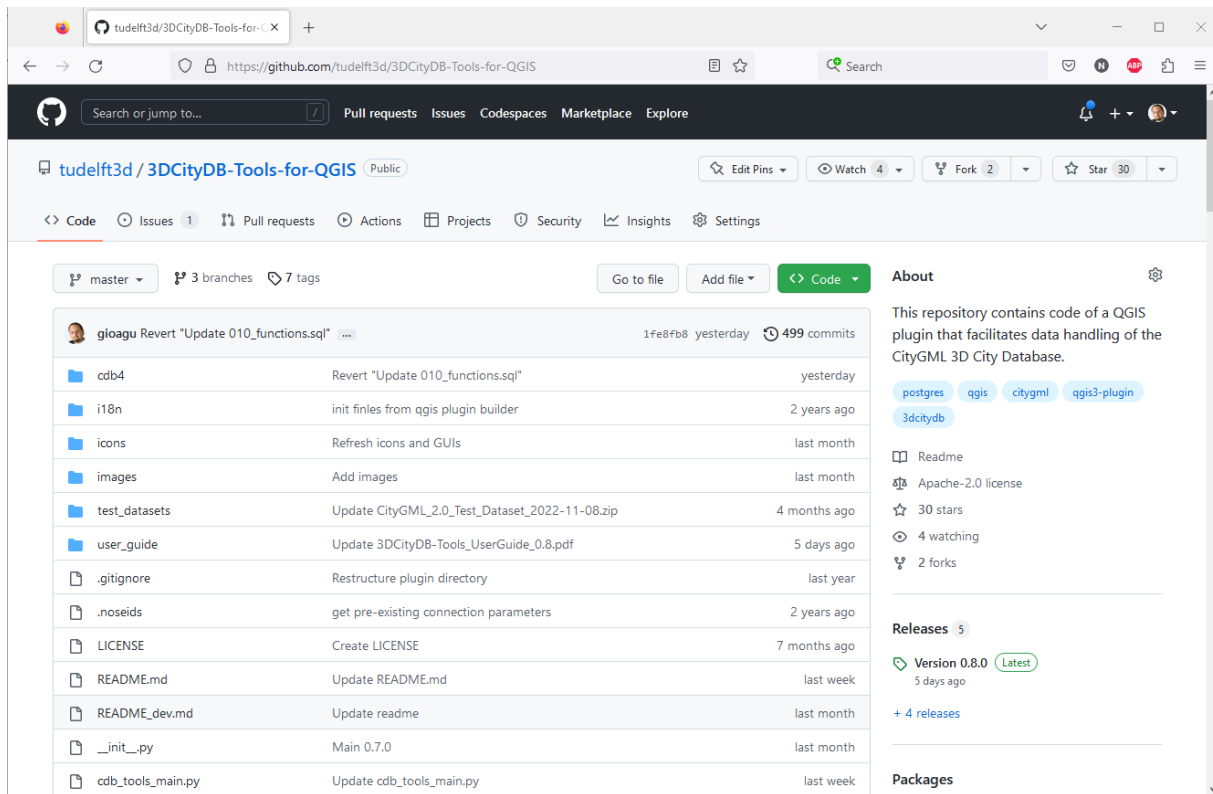
Current
limitations

QGIS
Package

Resources

Source code and GitHub repository

- GitHub: <https://github.com/tudelft3d/3DCityDB-Tools-for-QGIS>



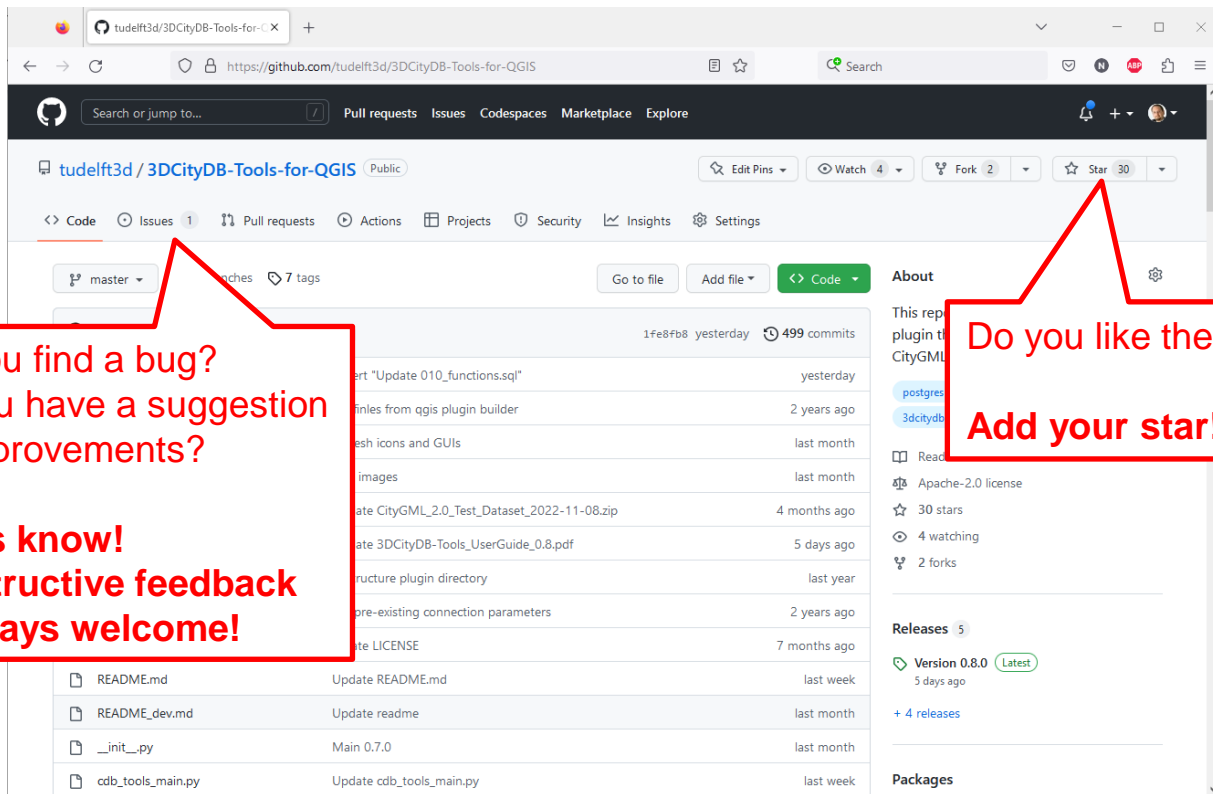
The screenshot shows the GitHub repository page for `tudelft3d/3DCityDB-Tools-for-QGIS`. The repository is public and has 499 commits, 30 stars, and 2 forks. The file list includes:

File	Description	Commit Date
<code>cdb4</code>	Revert "Update 010_functions.sql"	yesterday
<code>i18n</code>	init finles from qgis plugin builder	2 years ago
<code>icons</code>	Refresh icons and GUIs	last month
<code>images</code>	Add images	last month
<code>test_datasets</code>	Update CityGML_2.0_Test_Dataset_2022-11-08.zip	4 months ago
<code>user_guide</code>	Update 3DCityDB-Tools_UserGuide_0.8.pdf	5 days ago
<code>.gitignore</code>	Restructure plugin directory	last year
<code>.noseids</code>	get pre-existing connection parameters	2 years ago
<code>LICENSE</code>	Create LICENSE	7 months ago
<code>README.md</code>	Update README.md	last week
<code>README_dev.md</code>	Update readme	last month
<code>__init__.py</code>	Main 0.7.0	last month
<code>cdb_tools_main.py</code>	Update cdb_tools_main.py	last week

The repository description states: "This repository contains code of a QGIS plugin that facilitates data handling of the CityGML 3D City Database." It also mentions the Apache-2.0 license, 30 stars, 4 watching, and 2 forks. The latest release is Version 0.8.0, released 5 days ago.

Source code and GitHub repository

- GitHub: <https://github.com/tudelft3d/3DCityDB-Tools-for-QGIS>



Did you find a bug?
Do you have a suggestion
for improvements?

Let us know!
Constructive feedback
is always welcome!

Do you like the plugin?

Add your star! 😊

Motivation
Plugin overview
Software installation
Software use
Advanced options
Software uninstall
Current limitations
QGIS Package
Resources
Changelog

QGIS Plug-in repository

- <https://plugins.qgis.org/plugins/citydb-tools/>

Motivation
 Plugin overview
 Software installation
 Software use
 Advanced options
 Software uninstall
 Current limitations
 QGIS Package
Resources
 Changelog



The screenshot shows a web browser window displaying the QGIS Python Plugins Repository page for the 3DCityDB Tools plugin. The browser's address bar shows the URL <https://plugins.qgis.org/plugins/citydb-tools/>. The website has a dark header with navigation links: QGIS HOME, ABOUT PLUGINS, PLUGINS, PLANET, HUB, and LOGIN. A search bar is also present in the header.

On the left side of the page, there is a sidebar with a blue button labeled "Upload a plugin" and a list of plugin categories: Featured, All, Stable, New Plugins, Updated Plugins, Experimental, Popular, Most voted, Top downloads, Most rated, and QGIS Server plugins. At the bottom of the sidebar is a "Plugin Tags" section.

The main content area features the title "QGIS Python Plugins Repository" and "3DCityDB Tools". Below the title is a blue button labeled "Download latest". To the right of the title is an illustration of a database cylinder and city buildings. The plugin has a 5-star rating with 52 votes. The description states: "Tools to visualize and manipulate CityGML data stored in the 3D City Database". There are tabs for "About", "Details", and "Versions". The "About" tab is selected, showing the following text:

This plugin connects to the 3D City Database (v. 4.x) for PostgreSQL/PostGIS. It consists of:

- The "Layer Loader": it loads data as "classical" layers into QGIS. All CityGML LoDs are supported, as well as multiple citydb schemas and multiple database users. Feature attributes can be edited, changes are stored back directly into the database;
- A "Bulk Deleter": it allows to cleanup the database, or to delete selected feature types;
- "QGIS Package Administrator": it allows to install the server-side part of the plug-in, and to manage database user privileges.

Further details can be found in the PDF files contained in the 'user_guide' subfolder of the plugin installation directory. On the GitHub repository, some test datasets can be downloaded.

Test datasets

- In the GitHub repository, you will find test datasets that you can import into the 3DCityDB (using the Importer/Exporter) to test the 3DCityDB-Tools plugin. They are located in subfolder \test_datasets
- The test datasets are:
 - CityGML_2.0_Test_Dataset_2022-03-11.zip (aka "Railway")
 - FZK-Haus-LoD-all-KIT-IAI-KHH-B36-V1.zip (aka "Kit House")
 - DenHaag_bdg_lod2.zip
- You can find links to many additional free and open CityGML/CityJSON datasets at:
 - Awesome CityGML: <https://github.com/OloOcki/awesome-citygml>
 - 3D Geoinformation group @ TU Delft: <https://3d.bk.tudelft.nl/opendata/opencities/>

Changelog

ChangeLog for **version 0.8.4**

Main changes

- Minor bug fixes
- Layer Loader
 - Corrected new connections not being updated in Layer Loader dialog
- QGIS Package (version 0.10.2)
 - WaterBody module: Added support for MultiCurve LoD0-1 layers
 - Transportation and Vegetation modules: Corrected minor bugs
 - Added CityGML 2.0 Codelists for WaterBody module
 - Some code refactoring
- Test datasets
 - Updated Railway dataset to contain WaterBody LoD1 MultiCurve
- Installation and user guide (this document)
 - QGIS Package: Corrected/enhanced examples with functions to set user's privileges

Changelog

ChangeLog for **version 0.8.3**

Main changes

- The plugin is not anymore "experimental"; thus, it can be installed by anybody from QGIS
- Minor edits to the Layer Loader dialog GUI
 - Moved Codelist selection combo box from Settings tab to Layers tab
- Updated this document
 - Added installation from QGIS Plugins repository
 - Updated figures with new Layer Loader dialog GUI
 - QGIS Package: Corrected SQL examples to refresh views
 - QGIS Package: Added SQL examples for detail views
 - QGIS Package: Added SQL examples to upsert the user-defined bounding box extents
 - Corrected some typos and other minor errors

Changelog

ChangeLog for **version 0.8.2**

Main changes

- Minor internal changes to comply with the QGIS Plugin online repository

ChangeLog for **version 0.8.1**

Main changes

- Layer loader: Detail View layers are now ro/rw depending on user privileges
- QGIS Package: Version 0.10.1

Bug fixes

- QGIS Package: Fixed bug to set user privileges

ChangeLog for **version 0.8.0**

Client-side: Layer Loader

- Added support for Address features as layers
- Added support for LoD-independent layers (i.e., "LoDx")
- Redesigned and extended attribute forms to support:
 - External References (per type)
 - Generic Attributes (per type)
 - Addresses
 - All subforms allow insert, update and delete operations
- Added support for CodeLists selection via GUI
- Improved support of datasets with geographic coordinates (bbox)

Client side: Bulk Deleter

- Improved support of datasets with geographic coordinates (bbox)
- Renamed root-class to top-level features

Client-side: QGIS Package Administration

- Minor bug fixes and code clean-up

Changelog for **version 0.8.0 (ctd)**

Server-side (QGIS Package):

- New version 0.10
- Updates to layer_metadata table
- Updates to layer creation functions
- Added support for (updatable) Detail Views
- Added support for look-up tables metadata
- Improved support to datasets with geographic coordinates (bbox precision)
- Improved speed to compute the bbox and list cdb_schemas
- Minor code clean-up and bug fixes

Changelog

Changelog for **version 0.7.1**

Main changes

- Code restructuring to allow for better modularization of current (and future) modules
- Allow for concurrent dialogs, but added logic to:
 - Prevent having user dialogs (and connections) open when using the "QGIS Package Administration"
 - Prevent having user dialogs connected to the same database *and* citydb schema at the same time
- Ships with QGIS Package v. 0.9.1

Bug fixes

- When cleaning up the database, sequences are now correctly restarted

Changelog

Changelog for **version 0.7.0**

The 3DCityDB-Loader is growing and gets therefore a **new name: 3DCityDB-Tools**

Server-side (QGIS Package):

- Version 0.9.0
- Each 3DCityDB instance gets not its own database group
- Added support for new client-side functionalities:
 - Refactored layer_metadata table
 - Added functions to manage groups
 - Added functions to check Feature Types availability
- Minor code clean-up and bug fixes

Client-side: Admin GUI

- QGIS Package Installation tab: Major rework, added user management
- Installation settings tab (NEW)
- Minor GUI improvements to the New connection dialog
- QGIS Package uninstall procedure is now much faster
- User schema uninstall procedure is now much faster
- Rewritten all SQL queries to follow psycopg2 syntax
- Minor code refactoring and clean-up

Changelog for **version 0.7.0 (ctd)**

Client-side: Loader GUI

- Connection tab
 - Added Feature Type selection in layer generation
 - Added Geocoding function (based on Nominatim API)
 - Creating and dropping layers is now faster
 - Minor redesign of the GUI
- Settings tab (NEW)
- Editing of layers from read-only citydb schemas is now blocked in QGIS
- QML files are split and loaded separately for forms, 2D styles and (optionally) 3D styles, respectively
- Rewritten all SQL queries to follow psycopg2 syntax
- Major code refactoring and clean-up

Client side: Bulk Deleter GUI (NEW)

- A user with rw privileges can delete features selecting them:
 - via a spatial filter
 - and/or via a Feature Type/Root-class feature filters
- A user with rw privileges can delete ALL DATA from the selected citydb schema (i.e., truncate all tables)
- Geocoding function (based on Nominatim API)

Changelog

Changelog for **version 0.6.0**

Server-side (QGIS Package)

- Version 0.8.0
- Fixed bug in delete trigger functions to extract the name of the cdb_schema
- Rewritten functions to compute and upsert the extents
- Added layer support for (#8):
 - Terrain Intersection Curve geometries (bridge, building, tunnel, generics, city_furniture)
 - MultiCurve geometries (bridge, building, tunnel)
 - MasspointRelief, BreaklinesRelief Features
- Minor code clean-up and bug fixes

Client-side (Admin):

- Reworked the series of checks carried out upon connection
 - Added check and warning if PostgreSQL < 10.0
 - Added check and warning if QGIS Package < 0.8.0
- Changes to the GUI dialog
 - Reshaped connection buttons
 - Minor other visual improvements (labels, tooltips, etc.)
- Minor code refactoring and clean-up

Changelog for **version 0.6.0 (ctd)**

Client-side (User):

- Added compatibility for QGIS v. 3.28 LTR
- Reworked the series of checks carried out upon connection
 - Added check and warning for outdated QGIS Package versions
 - Added check and warning in case there are no accessible citydb schemas
 - Added check and warning in case there are only empty citydb schemas
- Added functionality to update the cdb_extents in case data are added/removed to/from the citydb schemas
- Solved bug of bboxes being wrongly resized (enlarged) after canvases are set/changed
- Improved function to import layers
 - Drop QGIS spatial filter if QGIS bbox = Layers bbox
 - Added check to avoid loading multiple times the same layer
- Changes to the GUI dialog, forms, etc.
 - Reshaped connection buttons
 - Added button "Refresh {sch} extents"
 - In the "Layer" tab, the Feature types in the combo box are now ordered alphabetically
 - Minor other visual improvements (labels, tooltips, etc.)
 - Added forms for new Relief Features, etc.
- Further major code refactoring and clean-up

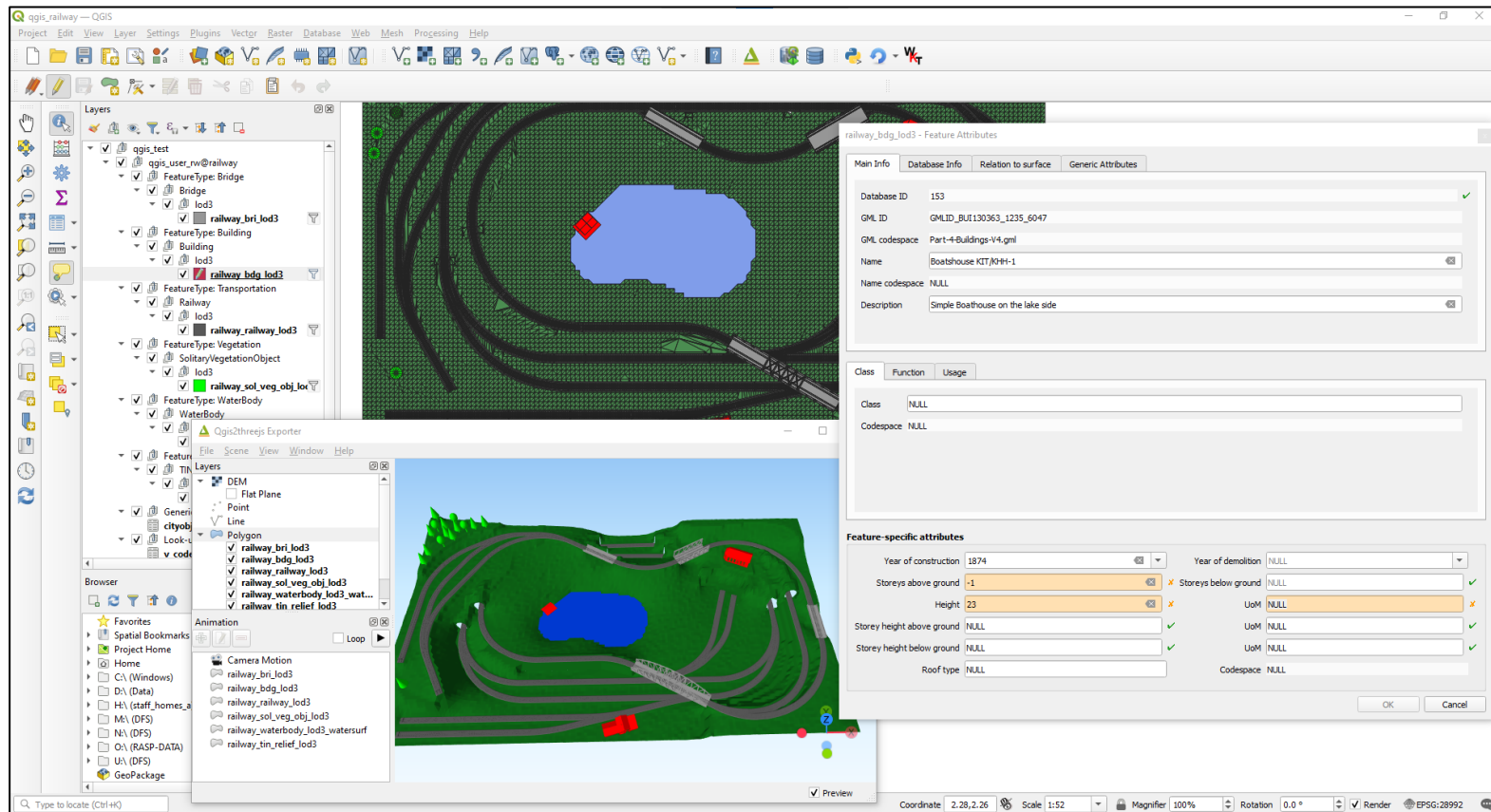
Changelog

Changelog for **version 0.5.0**

First public release of the 3DCityDB-Loader plugin for QGIS

Main changes

- Major code refactoring and cleaning up
- Minor bug fixes
- User guide documentation updated and extended with FME examples
- Switched to TUD GitHub repository



The screenshot shows the QGIS 3D environment. The top panel displays a 2D map view of a railway area with a lake. The bottom panel shows a 3D perspective view of the same area, with the railway tracks and buildings rendered in 3D. The 'Layers' panel on the left lists the loaded layers, including 'railway_brd_lod3', 'railway_bdg_lod3', 'railway_railway_lod3', 'railway_sol_veg_obj_lod3', and 'railway_waterbody_lod3'. The 'Feature Attributes' panel on the right shows the attributes for the selected feature, 'railway_bdg_lod3'. The 'Main Info' tab is active, displaying the following attributes:

Attribute	Value
Database ID	153
GML ID	GMLID_BUI130363_1235_6047
GML codespace	Part-4-Buildings-V4.gml
Name	Boathouse KIT/R0H-1
Name codespace	NULL
Description	Simple Boathouse on the lake side

The 'Feature-specific attributes' tab is also visible, showing the following attributes:

Attribute	Value
Year of construction	1874
Stores above ground	-1
Height	23
Storey height above ground	NULL
Storey height below ground	NULL
Roof type	NULL
Year of demolition	NULL
Stores below ground	NULL
UoM	NULL
UoM	NULL
UoM	NULL
Codespace	NULL

Main developers



Dr. Giorgio Agugiaro

g.agugiaro@tudelft.nl

3D Geoinformation Group

TU Delft

The Netherlands

<https://3d.bk.tudelft.nl/gagugiaro>



Konstantinos Pantelios

konstantinospantelios@yahoo.com

MSc Geomatics graduate

TU Delft

The Netherlands

Acknowledgements / suggestions and contributions by:

Tendai Mbwanda, Camilo León-Sánchez (TU Delft), Claus Nagel and Zhihang Yao (Virtual City Systems GmbH)



3DCityDB Tools

for

QGIS