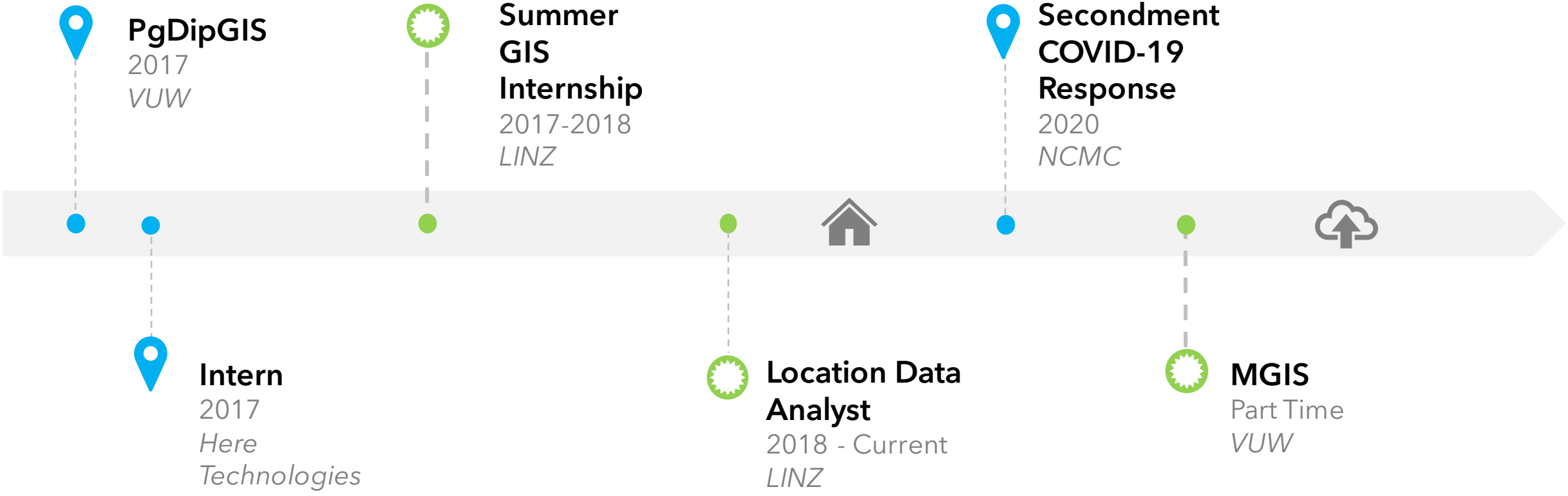
QGIS Plugin Development at Toitū Te Whenua LINZ

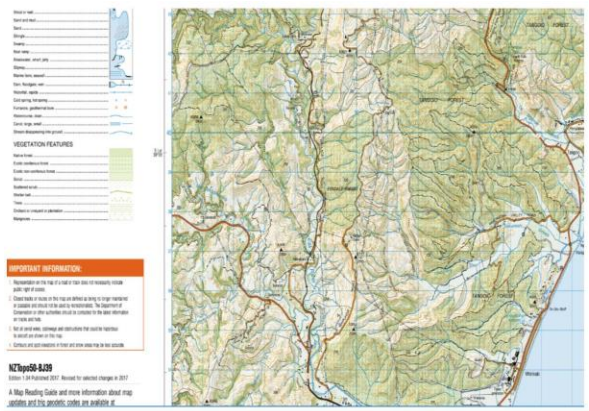
Megan Davidson

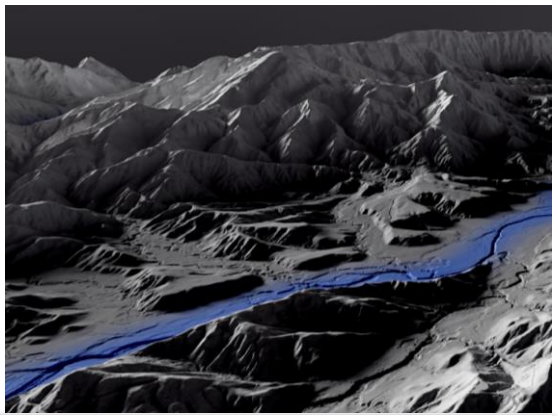
Toitū Te Whenua
Land Information
New Zealand

My Background



Topography Team at LINZ





- Topographic Maps
- Topographic Data
- Aerial Imagery
- LiDAR
- Historical Imagery
- & More...

The Buildings Technology Stack

Database Management

PostgreSQL

Code

python

Code Source Control

GitHub

Spatial Programming

QGIS

PostGIS

Visualisation & User Interfaces

QGIS

Qt

Documentation

Read the Docs

Sphynx

Upload Technology Stack

Database Management

Code

Code Source Control

Spatial Programming

Visualisation & User Interfaces

Documentation

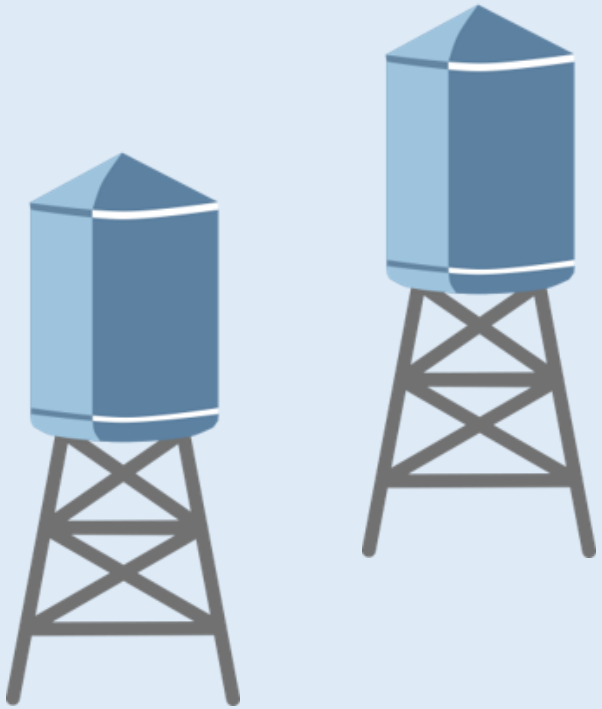
Project – New Zealand Building Outlines

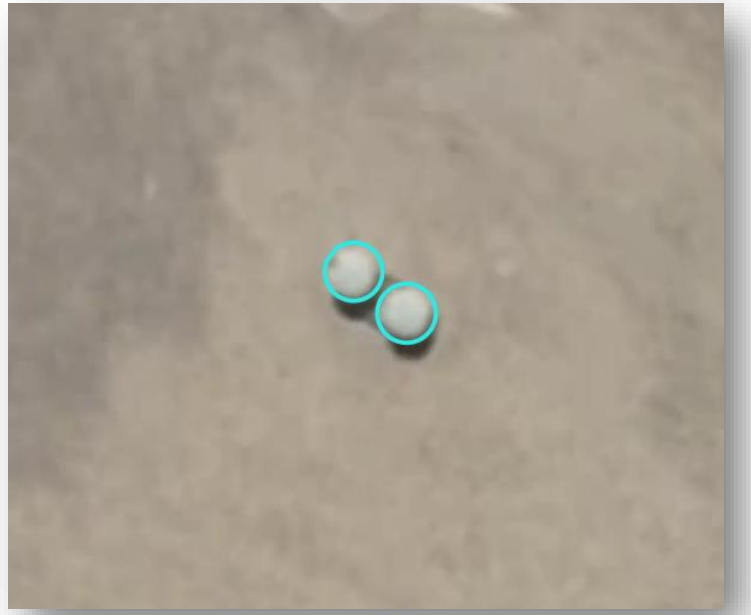
Goal: Publish a dataset containing building outlines extracted from imagery across New Zealand.

Problem: The raw data set contained **numerous** errors.

Solution: Quality control the building outlines dataset using python, SQL and a QGIS plugin.

No. 1: Tanks





Solution:

Find round polygons smaller than $16m^2$

PyQGIS:

The python environment to use QGIS tools with python.

```
from qgis.core import *
import qgis.utils
```

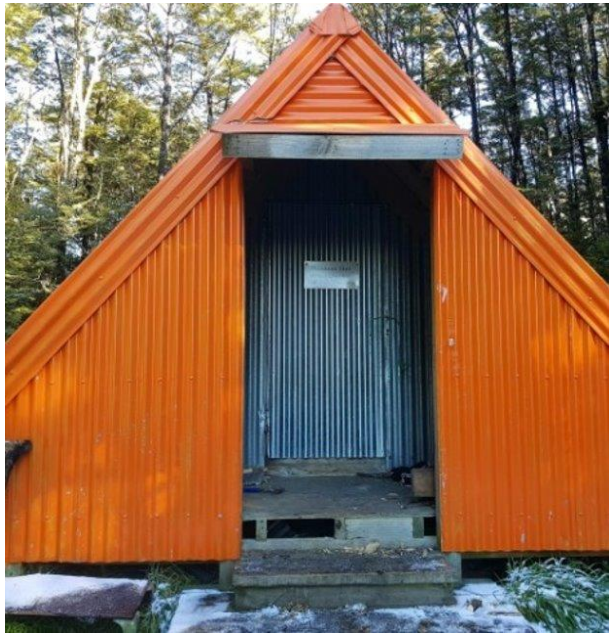
Code:

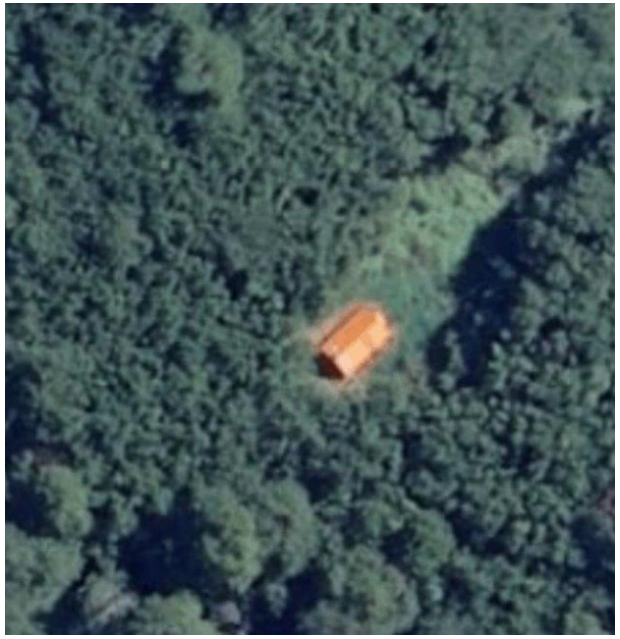
PyQGIS

```
query = "$perimeter < 1.025 * 2 * 3.1416 * sqrt($area/3.1416) and $area <= 16"

layer.getFeatures(QgsFeatureRequest().setFilterExpression(query))
```

No. 2:
Missing Huts





Solution:

Use the existing Topo50 hut points dataset to find huts that don't intersect or aren't within 10m of a polygon.

Code:

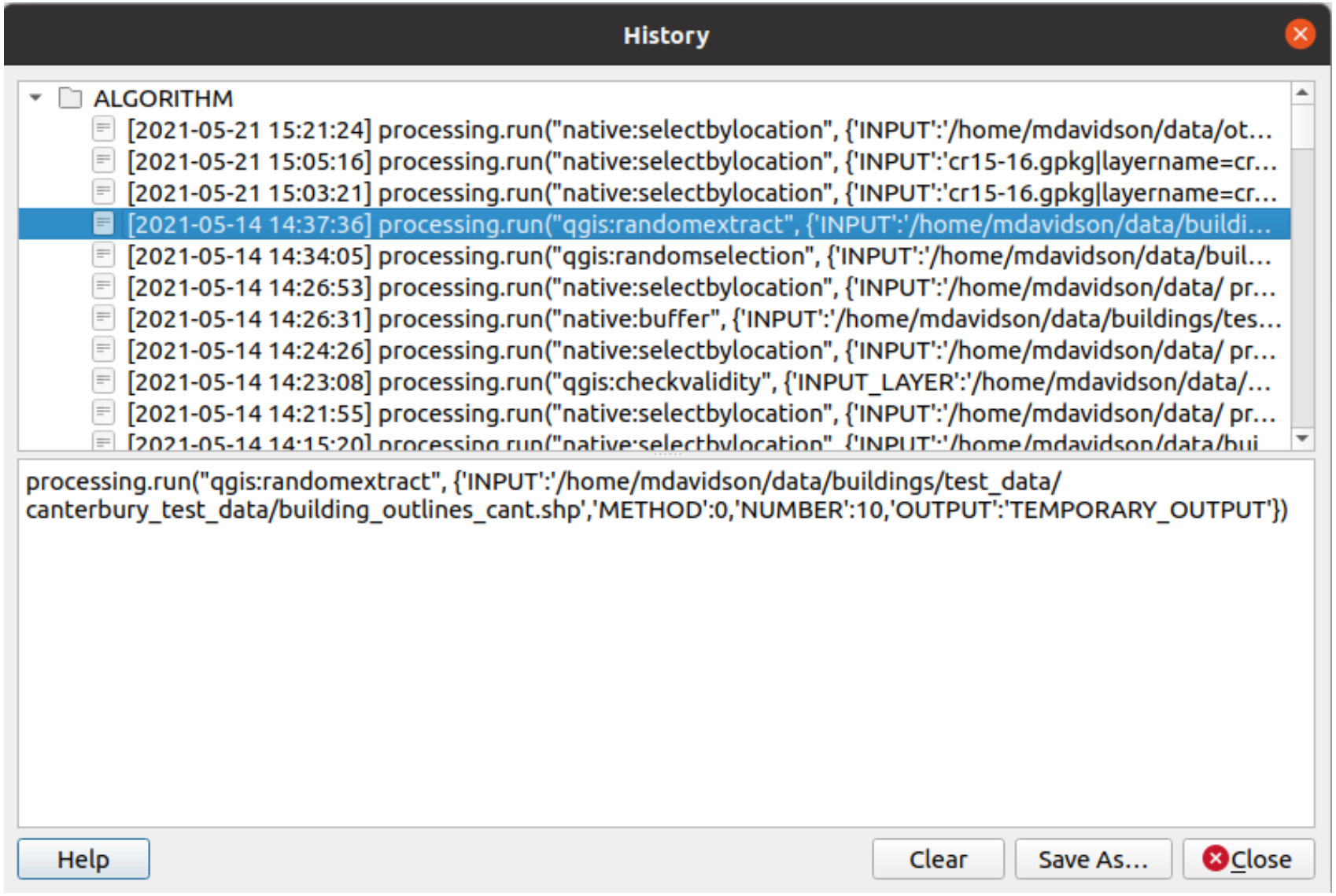
PyQGIS

```
import processing

processing.run("native:buffer",
{'INPUT':'buildings.shp', 'DISTANCE':10,'SEGMENTS':5,'END_CAP_STYLE':0,'JOIN_STYLE':0,'MITER_L
IMIT':2,'DISSOLVE':False,'OUTPUT':'buffer.shp'})

processing.run("native:selectbylocation",
{'INPUT':'topo50huts.shp','PREDICATE':[2],'INTERSECT':'buffer.shp','METHOD':0})
```

QGIS History



No. 3: Rocks, Trees and Unidentifiable Objects

Solution:

- Check a random subset of buildings
- Check all buildings in National Parks

Code:
(Random Subset)

PyQGIS

```
import processing

processing.run("qgis:randomextract",{'INPUT':'buildings.shp','METHOD':0,'NUMBER':100,'OUTPUT':'rando
m_extract.py})
```

Code:
(all buildings in national parks)

PyQGIS

```
import processing

processing.run("native:selectbylocation",{'INPUT':'buildings.shp','PREDICATE':[0],'INTERSECT':'national_parks.shp','METHOD':0})
```

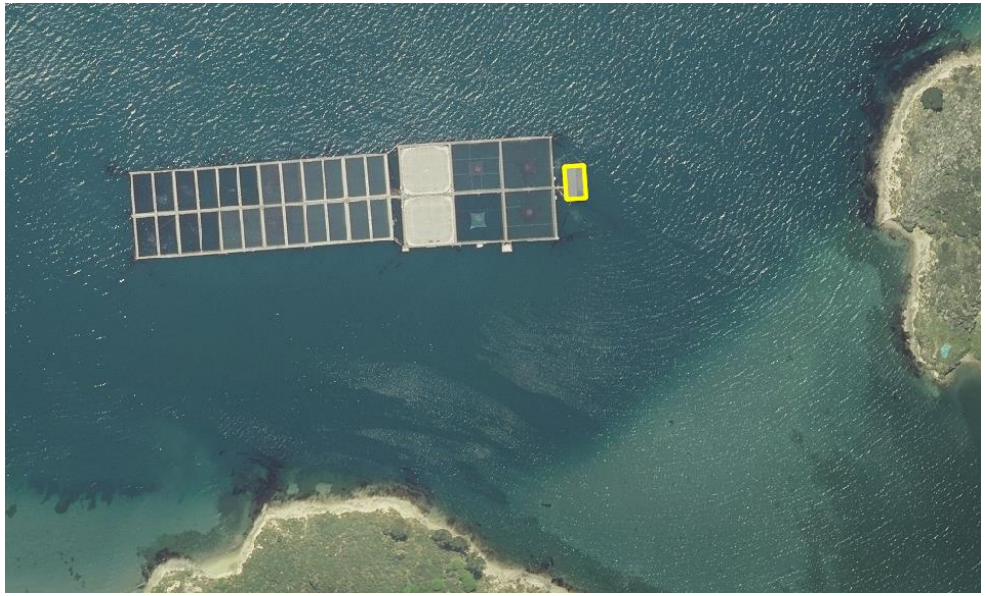
Other QC Checks:

Buildings that intersect water features
- Rivers
- Lakes

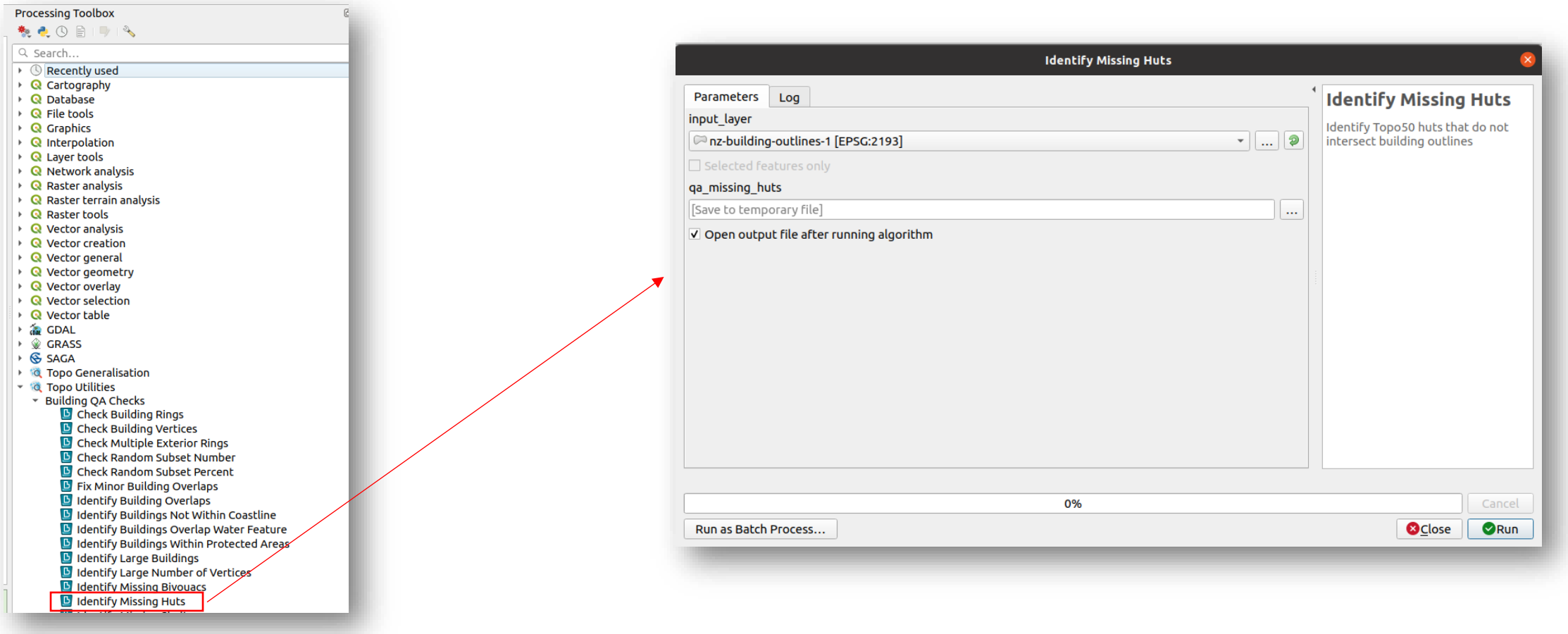
Buildings that weren't within the Coastline

Large Buildings

Geometry Checks
- Rings
- Vertices
- Exterior rings
- Overlaps



Processing Algorithms:

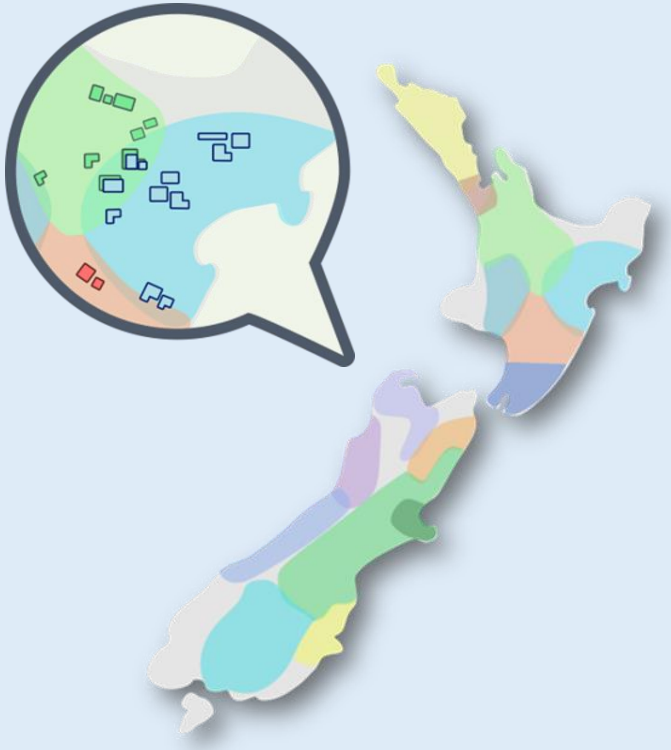


Processing Algorithms:

How to run QGIS processing algorithms using python:
https://docs.qgis.org/3.16/en/docs/user_manual/processing/console.html

How to create your own processing algorithm:
(Includes template)
https://docs.qgis.org/3.4/en/docs/user_manual/processing/scripts.html

No. 4: Overlaps



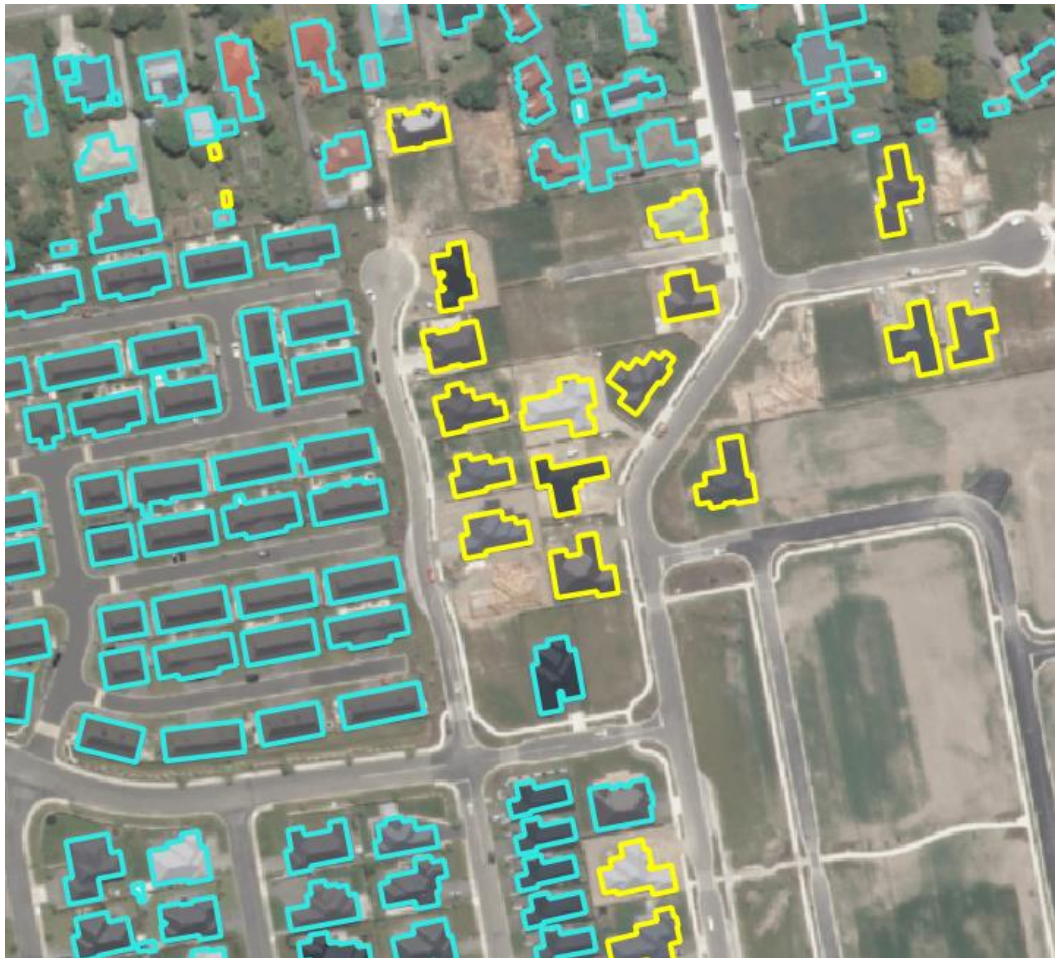
Canterbury 0.3m Rural Aerial Photos (2015-2016)

Canterbury 0.3m Rural Aerial Photos (2014-2015)

***raw data**

New Buildings:

Canterbury 0.3m Rural Aerial Photos (2014-2015)



Canterbury 0.3m Rural Aerial Photos (2015-2016)

Removed Buildings:

Canterbury 0.3m Rural Aerial Photos (2014-2015)

Canterbury 0.3m Rural Aerial Photos (2015-2016)

Matched Buildings:

Canterbury 0.3m Rural Aerial Photos (2014-2015)

Canterbury 0.3m Rural Aerial Photos (2015-2016)

Related Buildings:

Canterbury 0.3m Rural Aerial Photos (2014-2015)

Canterbury 0.3m Rural Aerial Photos (2015-2016)

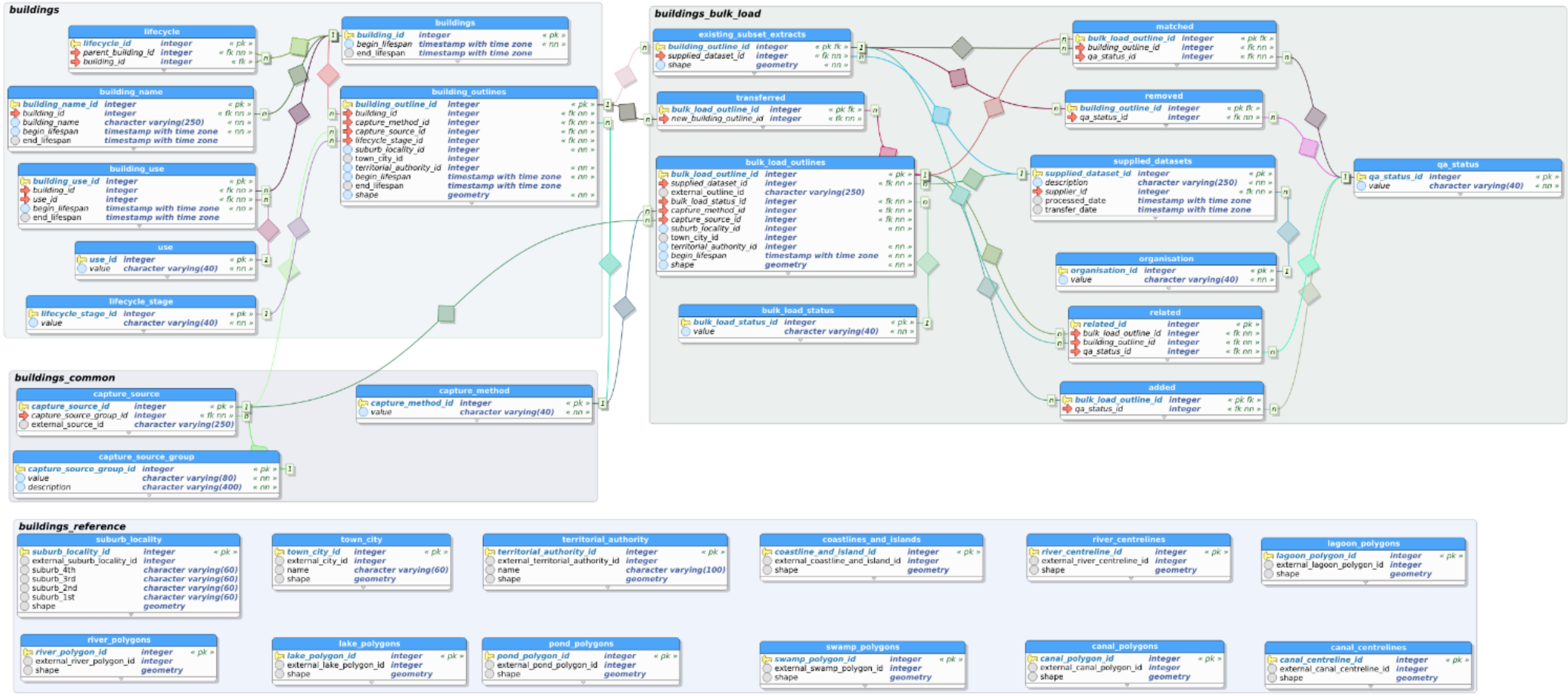
Using SQL:

Written in **SQL** using PostGIS

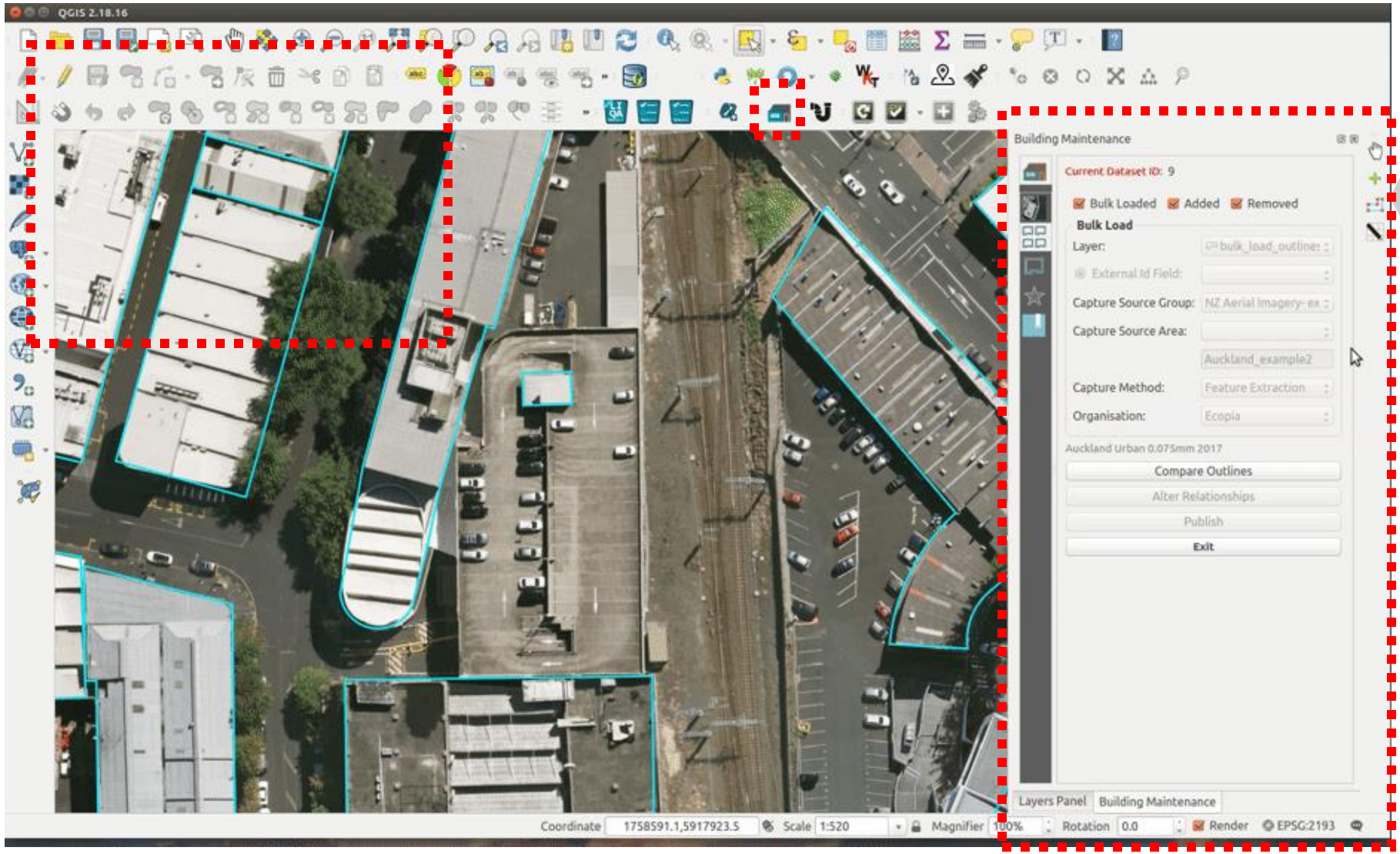
- ST_Intersects(geometry, geometry)
- ST_Area(geometry)

https://github.com/linz/nz-buildings/blob/master/db/sql/deploy/buildings_bulk_load/functions/compare.sql

No. 5: A complex Database



The Solution:
(A QGIS Plugin)



Add Outline

Edit Geometry

Edit Attribute

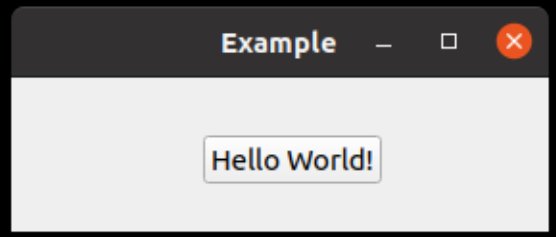
PyQt:

```
import sys
from PyQt5.QtWidgets import QApplication, QPushButton, QWidget

# Create GUI
app = QApplication(sys.argv)
window = QWidget()
window.setWindowTitle("Example")
window.setGeometry(100, 100, 280, 80)
button = QPushButton("Hello World!", parent=window)
button.move(100, 30)

# Show Window
window.show()

# Run
sys.exit(app.exec_())
```

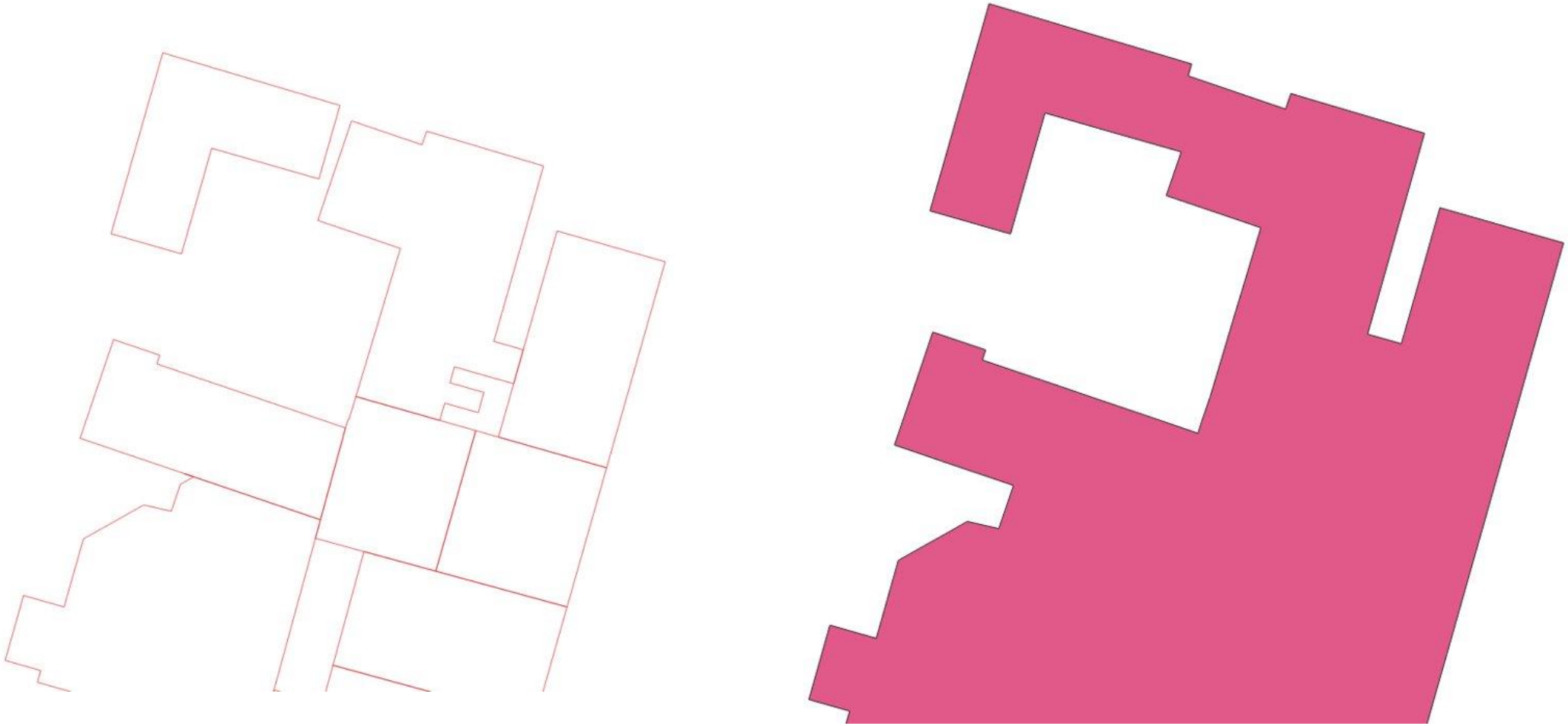


QGIS Plugins:

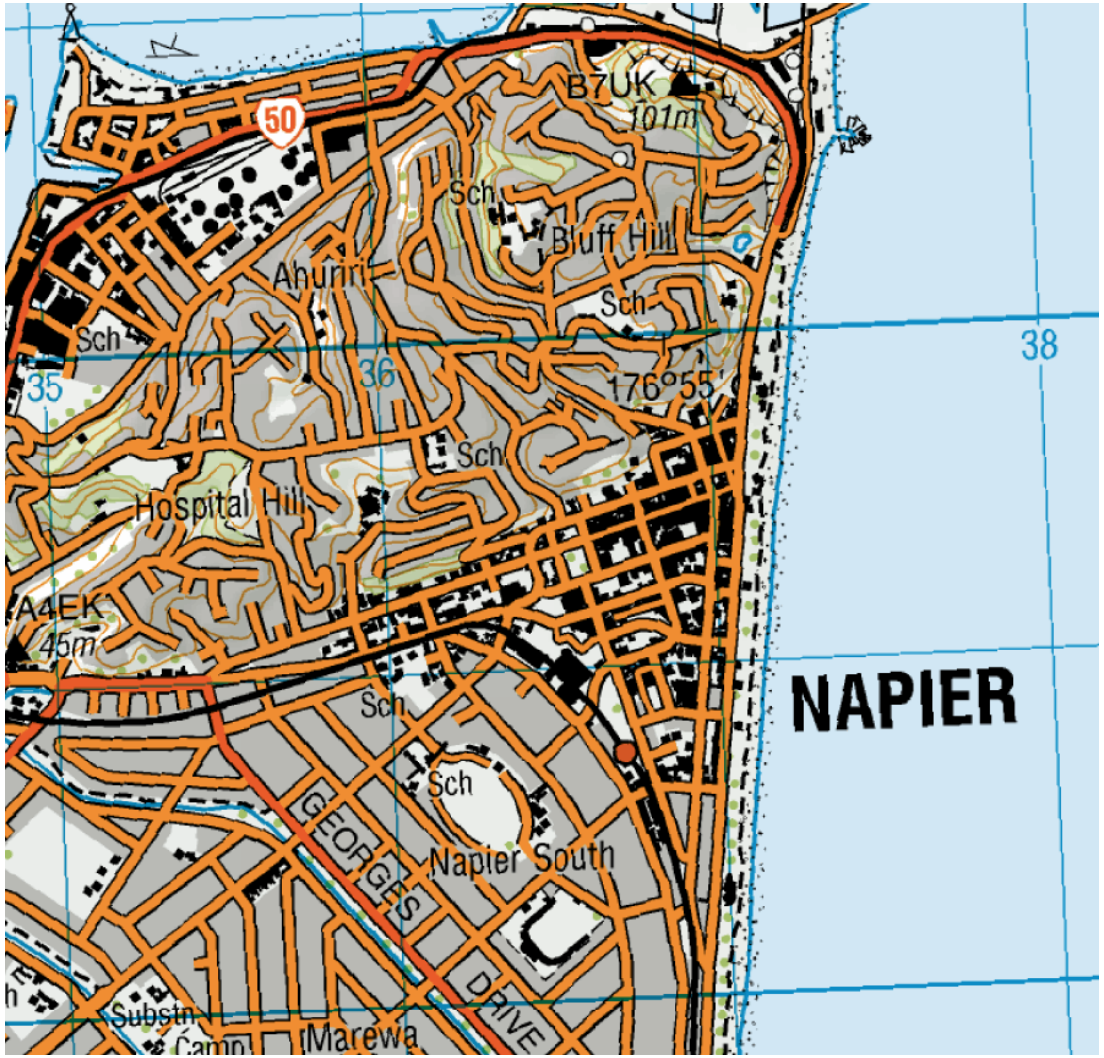
PyQGIS Developer Cookbook:
(Good Starting Point)
https://docs.qgis.org/3.10/en/docs/pyqgis_developer_cookbook/

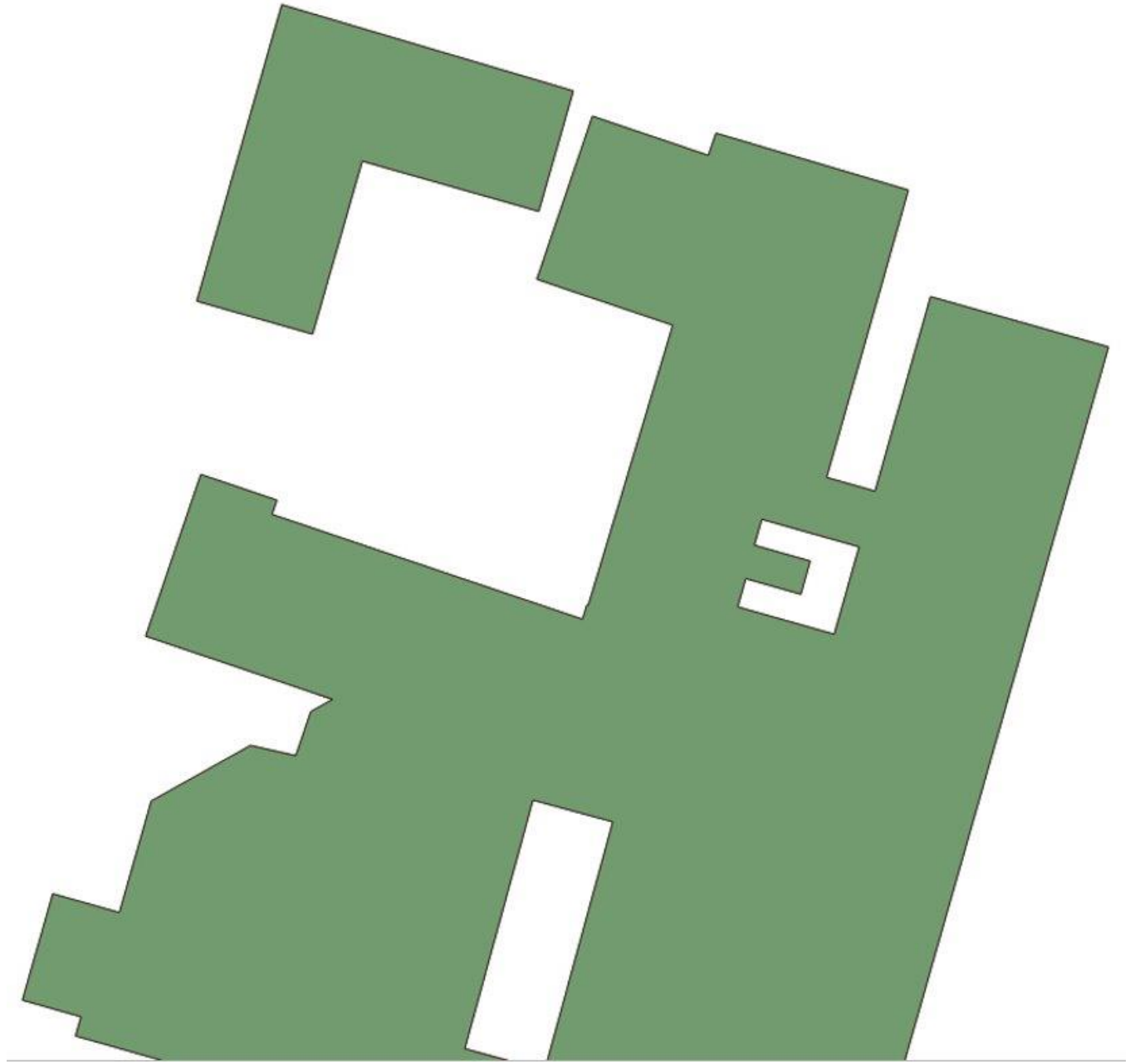
QGIS API Documentation:
(Everything QGIS can do)
https://qgis.org/api/

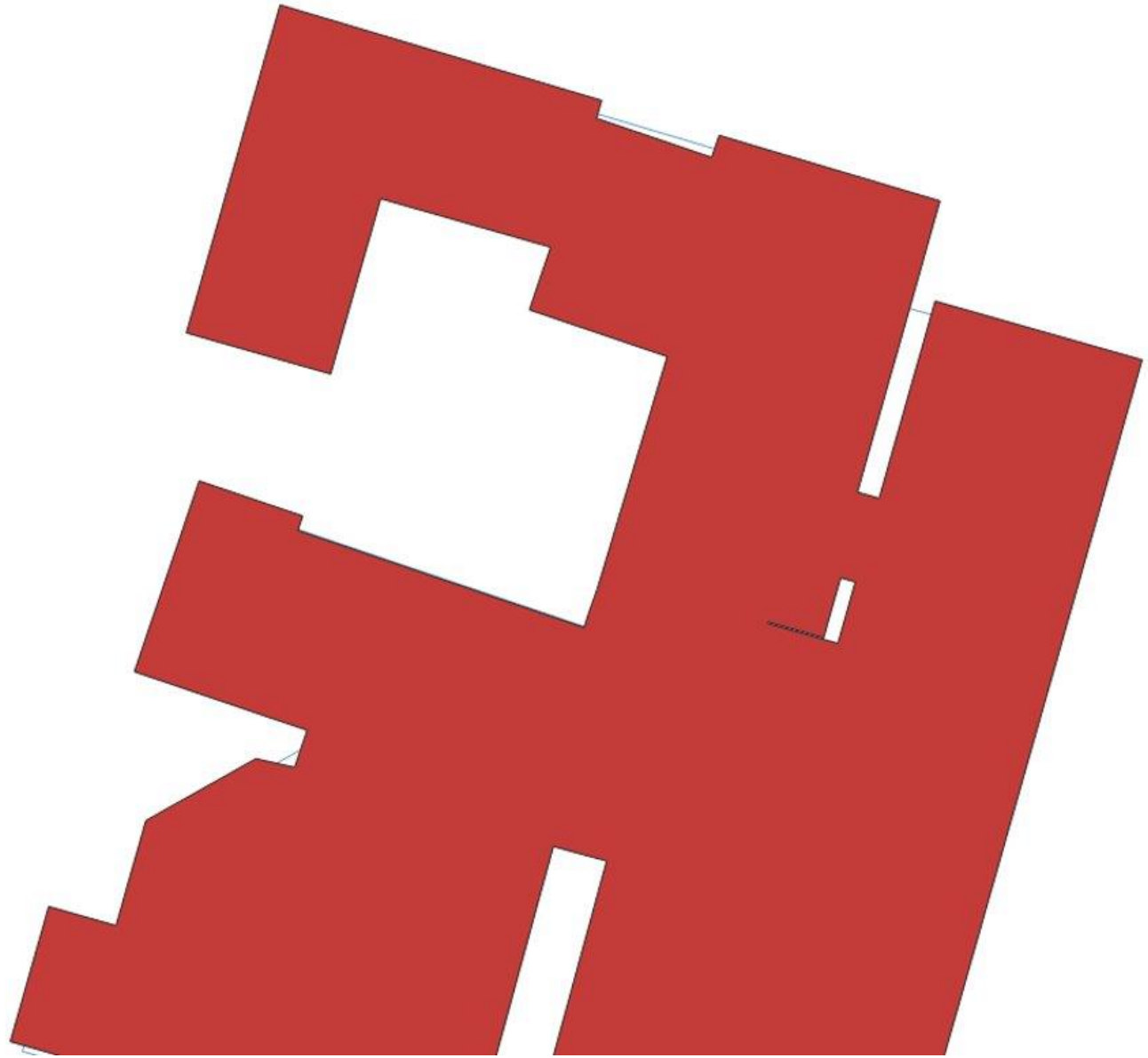
6: Generalised Polygons

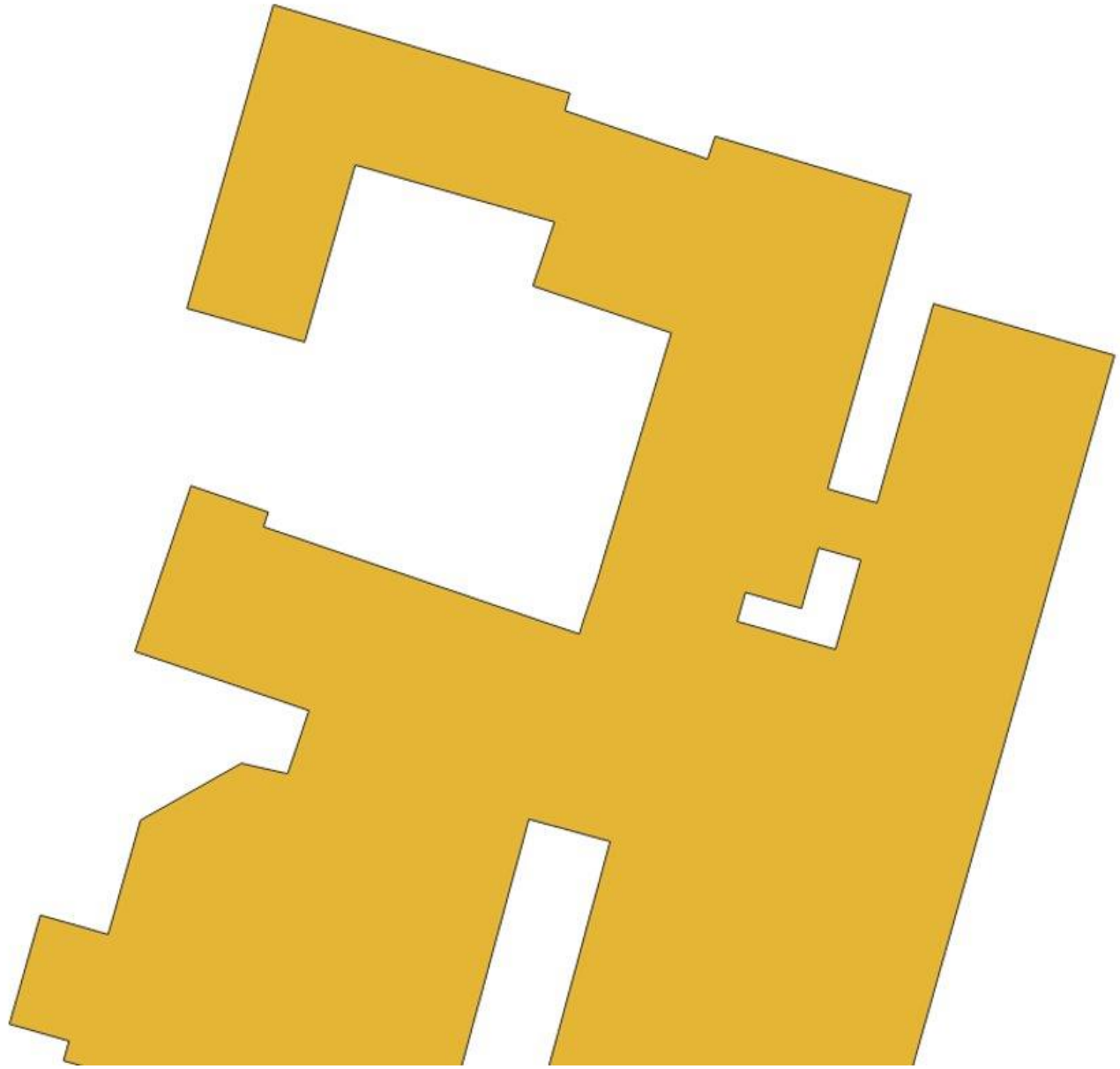


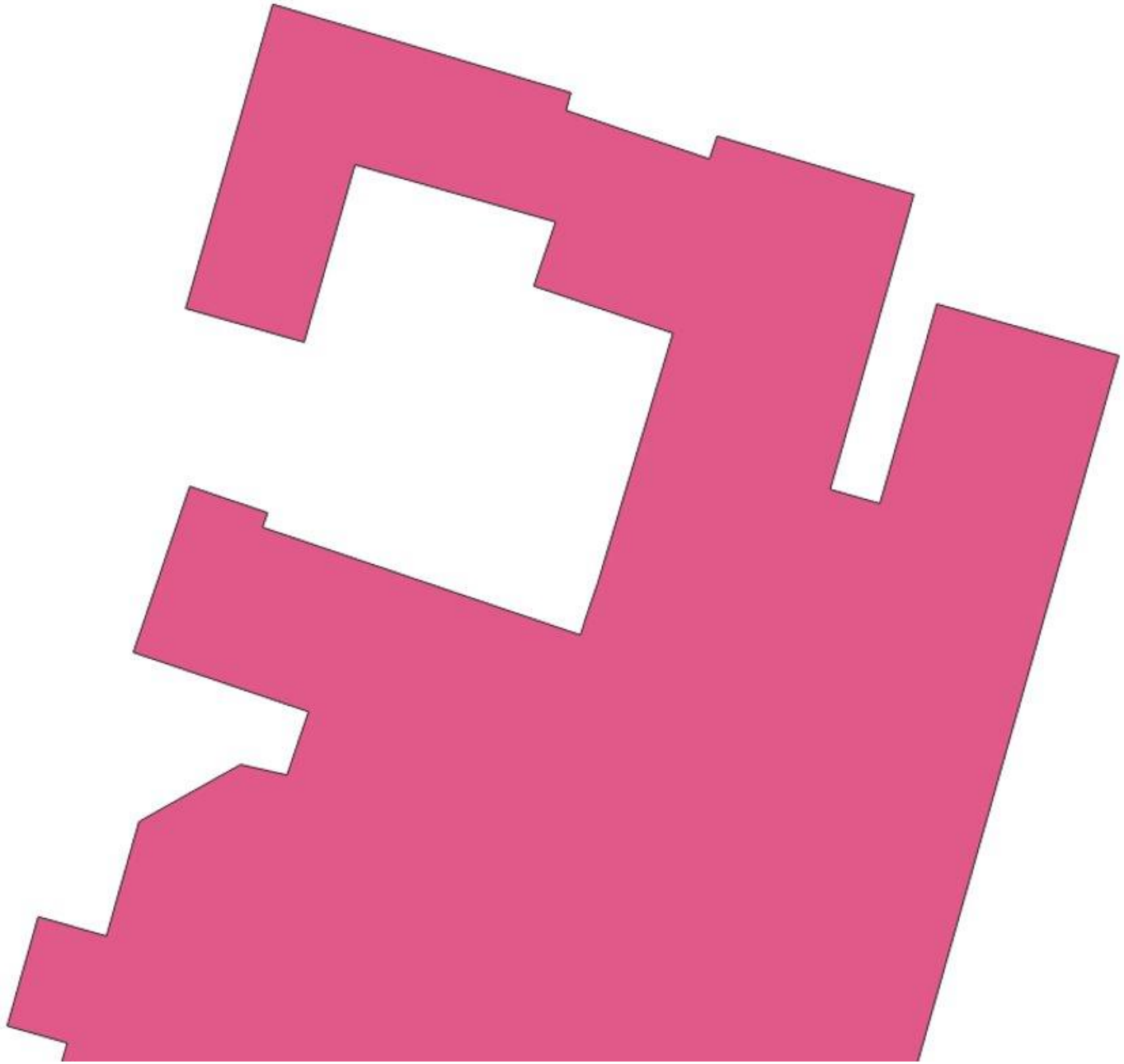
Generalised Outlines:











Code:

PyQGIS

```
import processing


processing.run("native:buffer",{parameters})

processing.run("native:dissolve",{parameters})
```

Links:

https://github.com/linz/nz-buildings

-

https://nz-buildings.readthedocs.io/en/latest/introduction.html

-

https://data.linz.govt.nz/layer/101290-nz-building-outlines/

Thanks!

Any questions?