

# FOSS4G.NL 2017

## QGIS Feature Frenzy



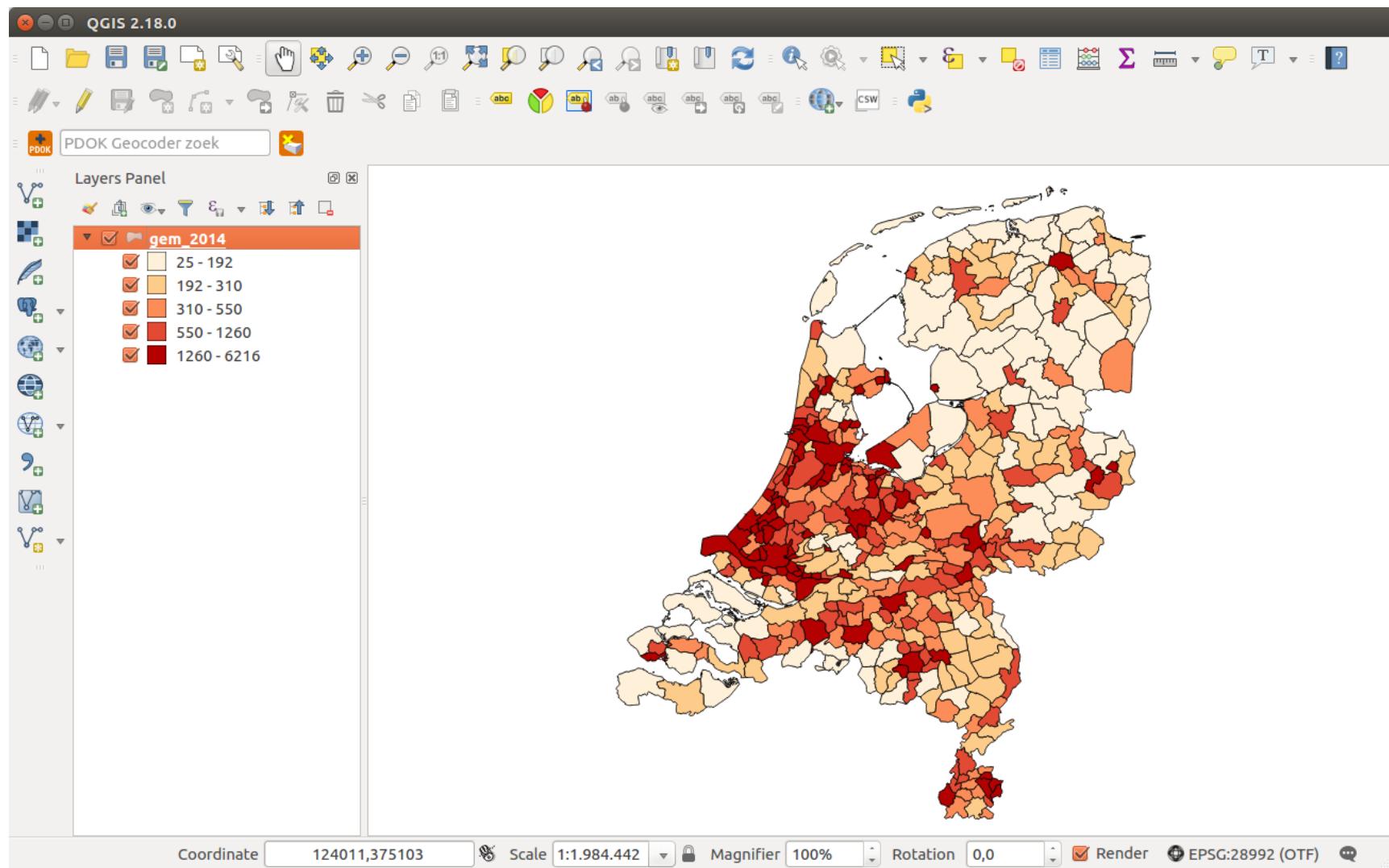
# **FOSS4G.NL 2017**

# **QGIS Feature Frenzy**



Raymond Nijssen  
[r.nijssen@terglobo.nl](mailto:r.nijssen@terglobo.nl)

# QGIS



# Sinds Geobuzz 2015



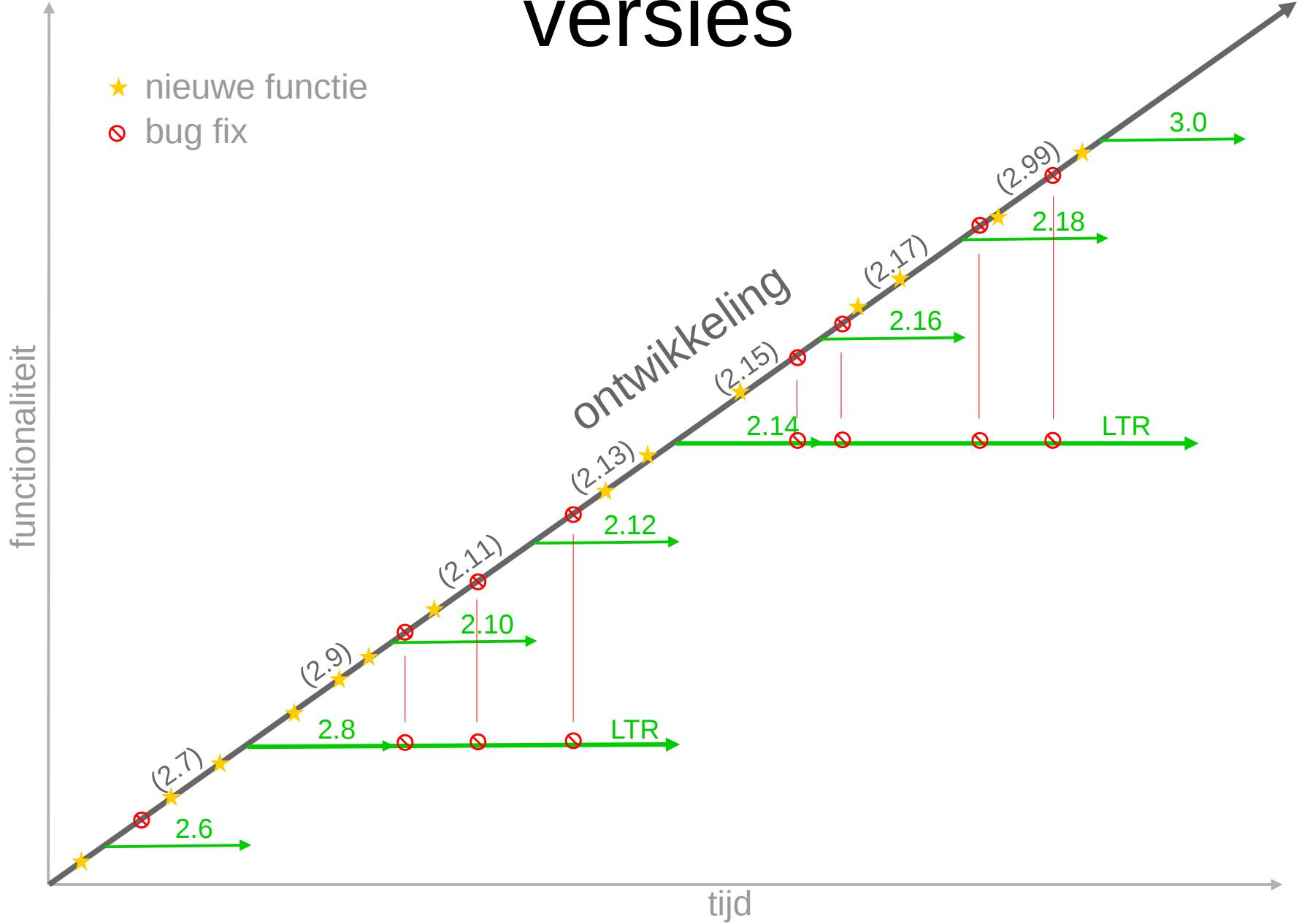
# Sinds Geobuzz 2016

# Logo

QGIS



# versies



# Cartographic algorithms

Topological coloring

Parameters Log Run as batch process...

Input layer: data Locality\_Boundaries MultiPolygon [EPSG:4283]

Minimum number of colors: 4

Minimum distance between features: 0.000000

Balance color assignment: By feature count

Colored: [Create temporary layer]

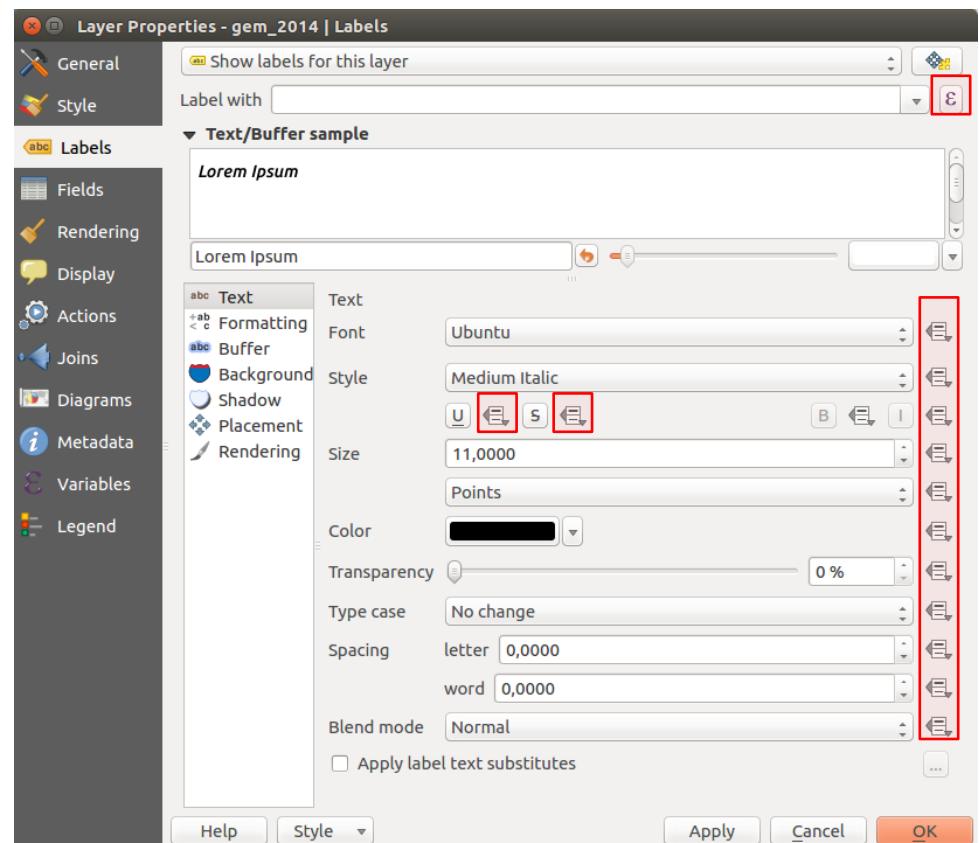
Open output file after running algorithm

0%

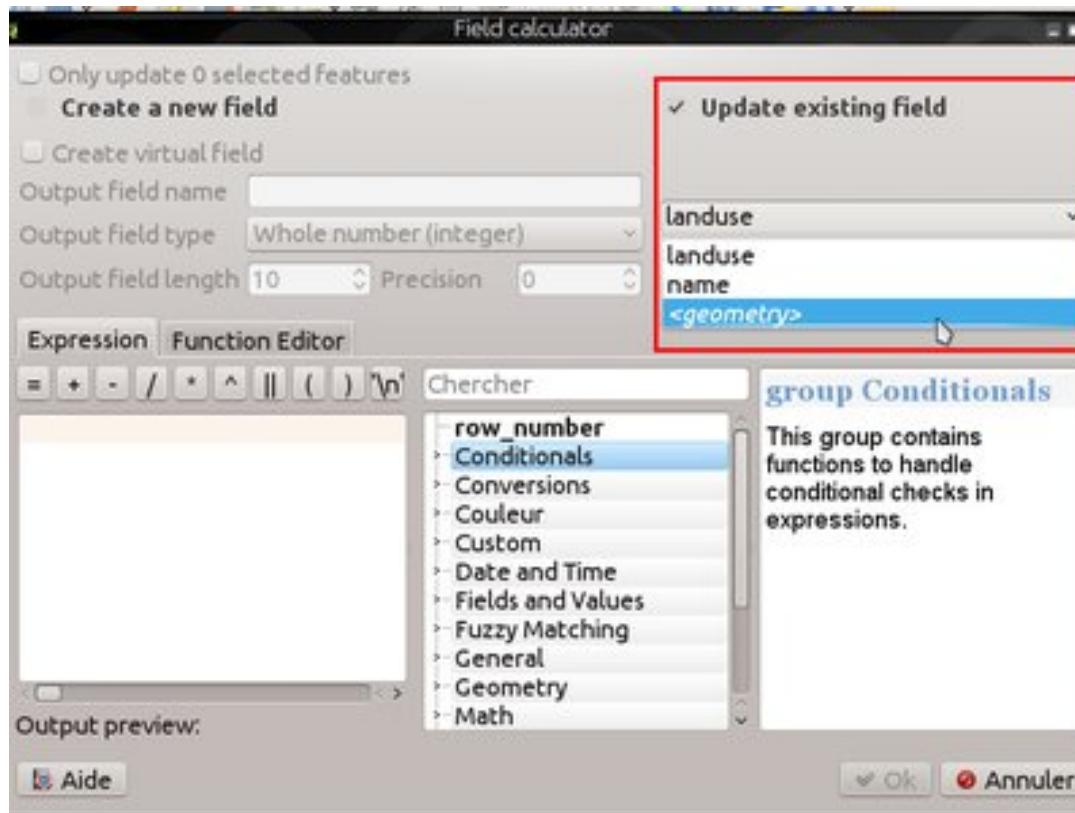


# Expressies

- Meer “data defined”
- Uitbreidingen op expressies
- Dus meer “on-the-fly” visualiseren!

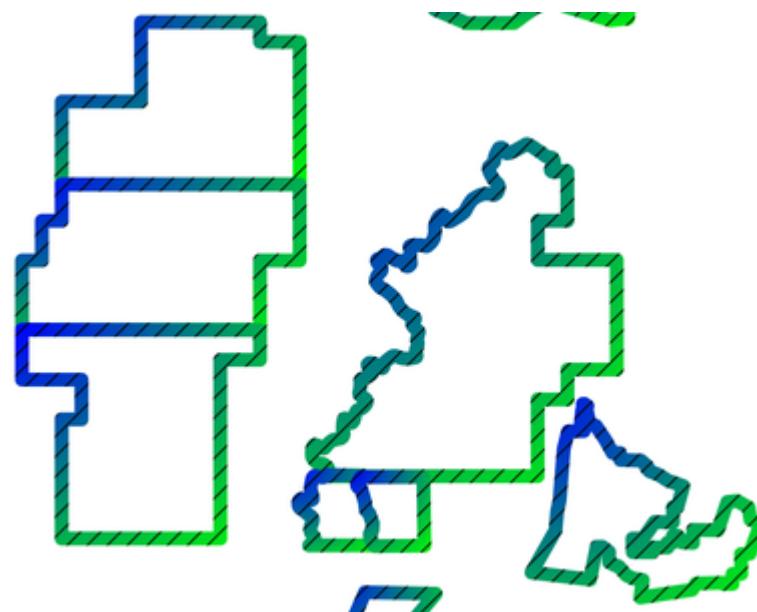


# Expressies



Wijzig ook geometrie in veldcalculator

# Expressies



Geometrie-generator symbologie  
`difference( buffer( $geometry , 250 ), buffer( $geometry, -250 ) )`

# Expressies



## Geometrie-generator symbologie

# Visualisatie



Placement

Cartographic    Around point    Offset from point

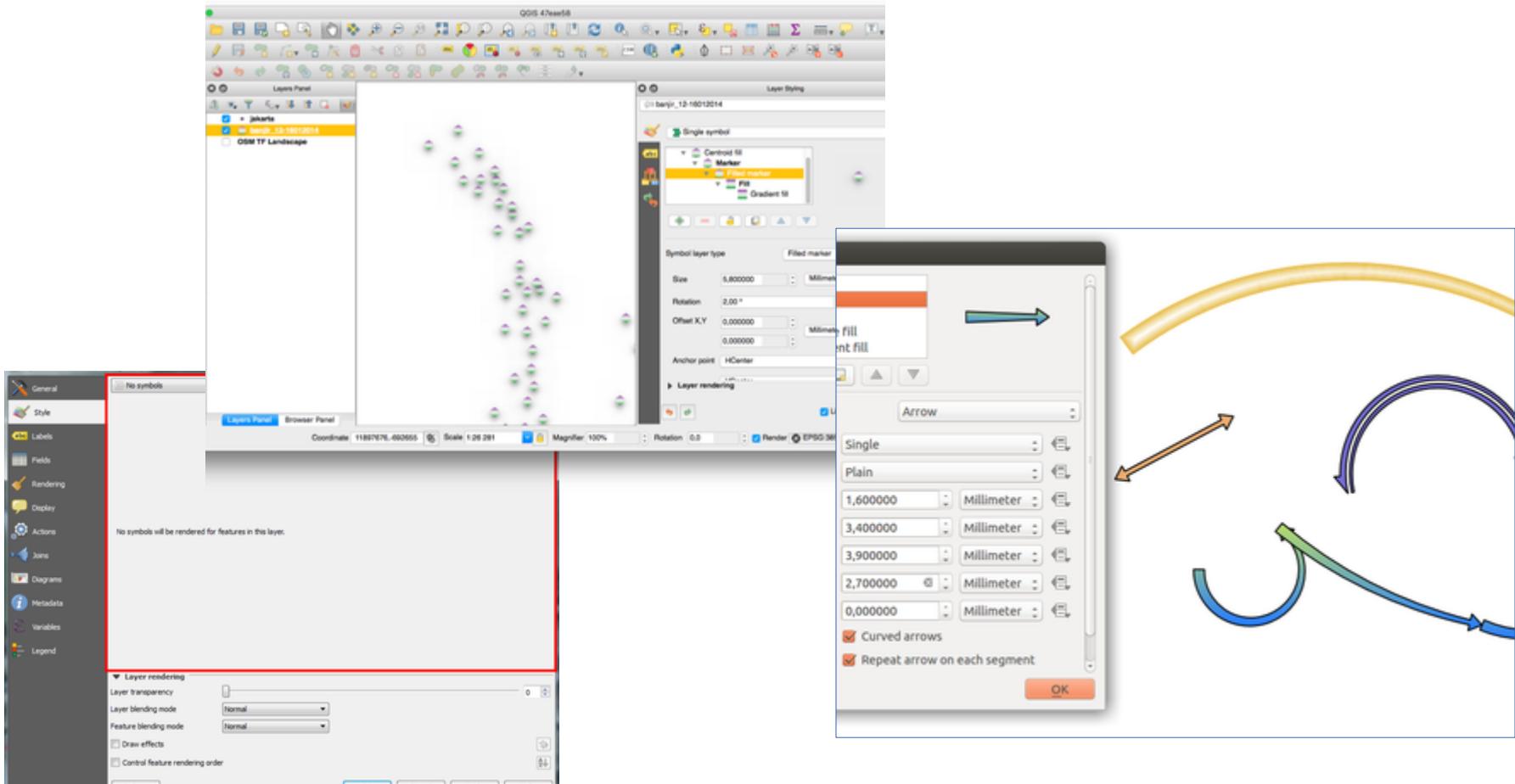
Distance

Millimeter

Distance offset from  From point

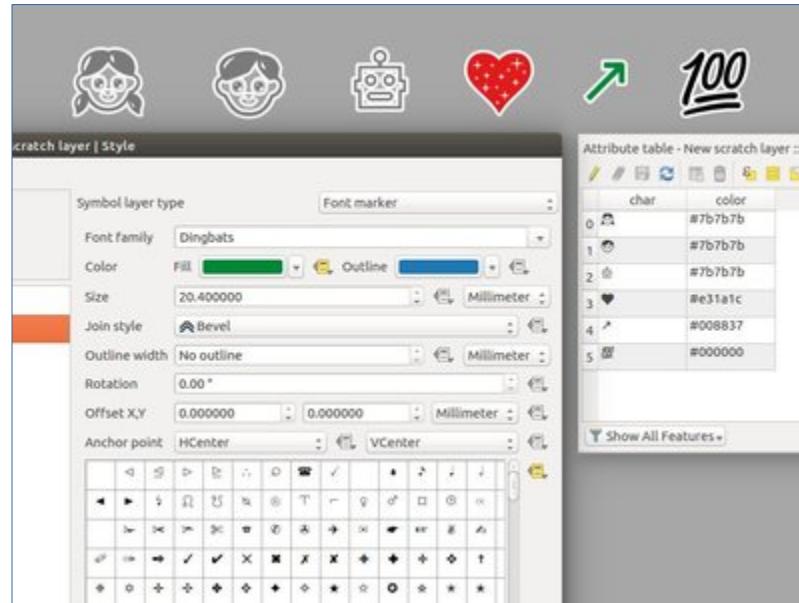
Veel nieuwe opties voor het labelen van punten en lijnen

# Visualisatie



Nieuwe symbologie-types:  
Pijlen, gevulde puntsymbolen, “geen symbolen”

# Visualisatie



Puntsymbool “buffer”

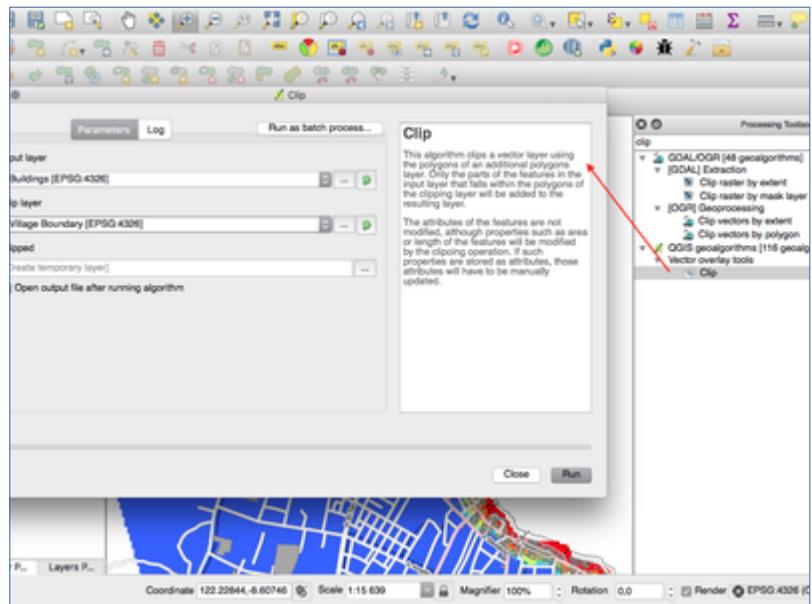
# Data

`http://c.tile.openstreetmap.org/{z}/{x}/{y}.png`

XYZ-kaartlagen

# Speed up

- Objectselecties
- WMS Capabilities  
gecached
- OTF-generalisatie
- Clip-algoritme



# Scripts

- Wie kunnen er programmeren?
- Wie in Python?
- Wie in QGIS?
- Scripts?
- Plugins?

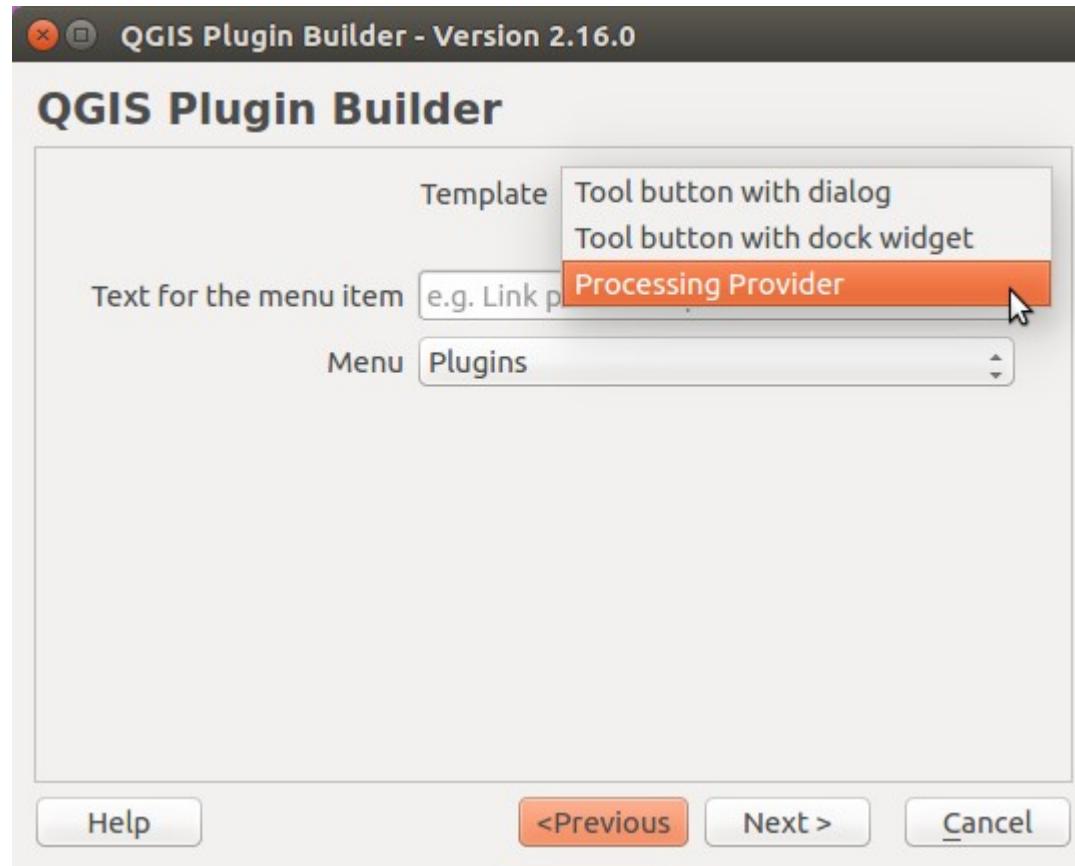
```
gPnt = QgsGeometry.fromPoint(QgsPoint(1,1))
gLine = QgsGeometry.fromPolyline([QgsPoint(1, 1), QgsPoint(2, 2)])
gPolygon = QgsGeometry.fromPolygon([[QgsPoint(1, 1), QgsPoint(2, 2),
                                     QgsPoint(2, 1)]])
```

# Scripts

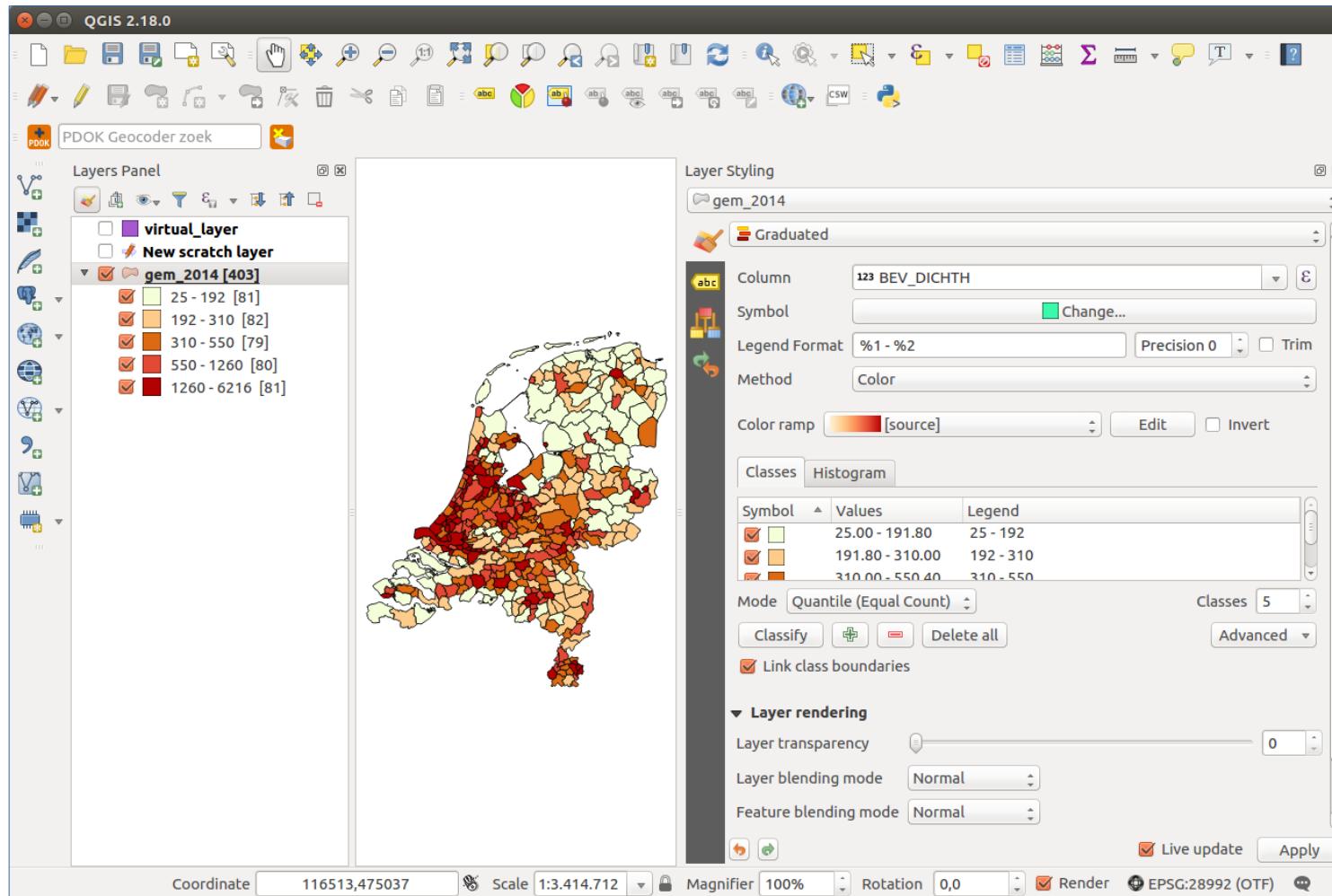
```
1 | for feature in vector_layer.getFeatures():
2 |     if not feature.id() in vector_layer.selectedFeatureIds():
3 |         continue
4 |
5 |     # do something with the feature
```

```
1 | request = QgsFeatureRequest().setFilterExpression('my_field > 20')
2 | for feature in vector_layer.getFeatures(request):
3 |     # do something with the feature
```

# Processing Provider Plugin



# Cool stuff



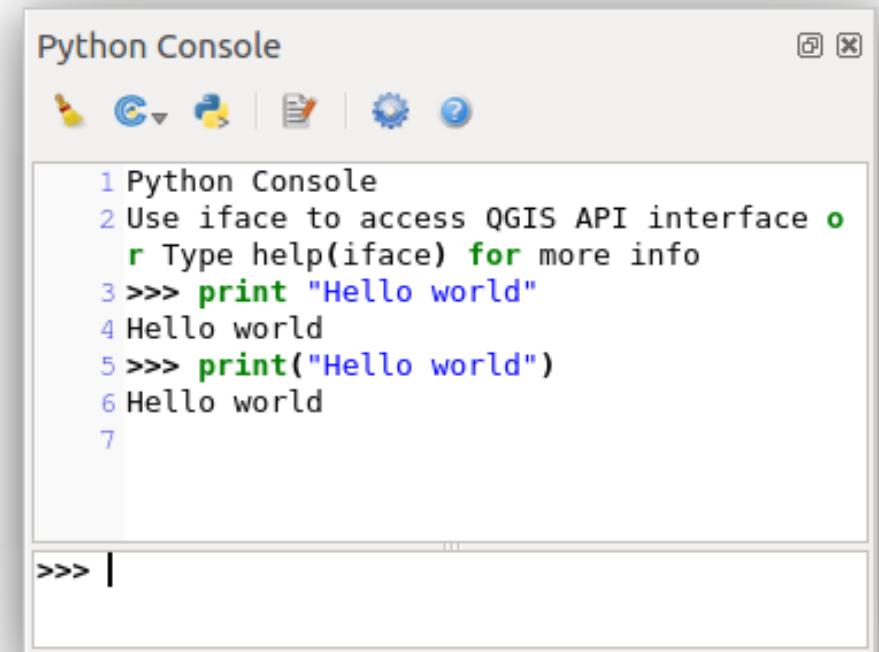
Paneel “Layer Styling” met voorvertoning

# Bekijk visual change logs

- <https://qgis.org/en/site/forusers/visualchangelog214/>
- <https://qgis.org/en/site/forusers/visualchangelog216/>
- <https://qgis.org/en/site/forusers/visualchangelog218/>

# QGIS 3

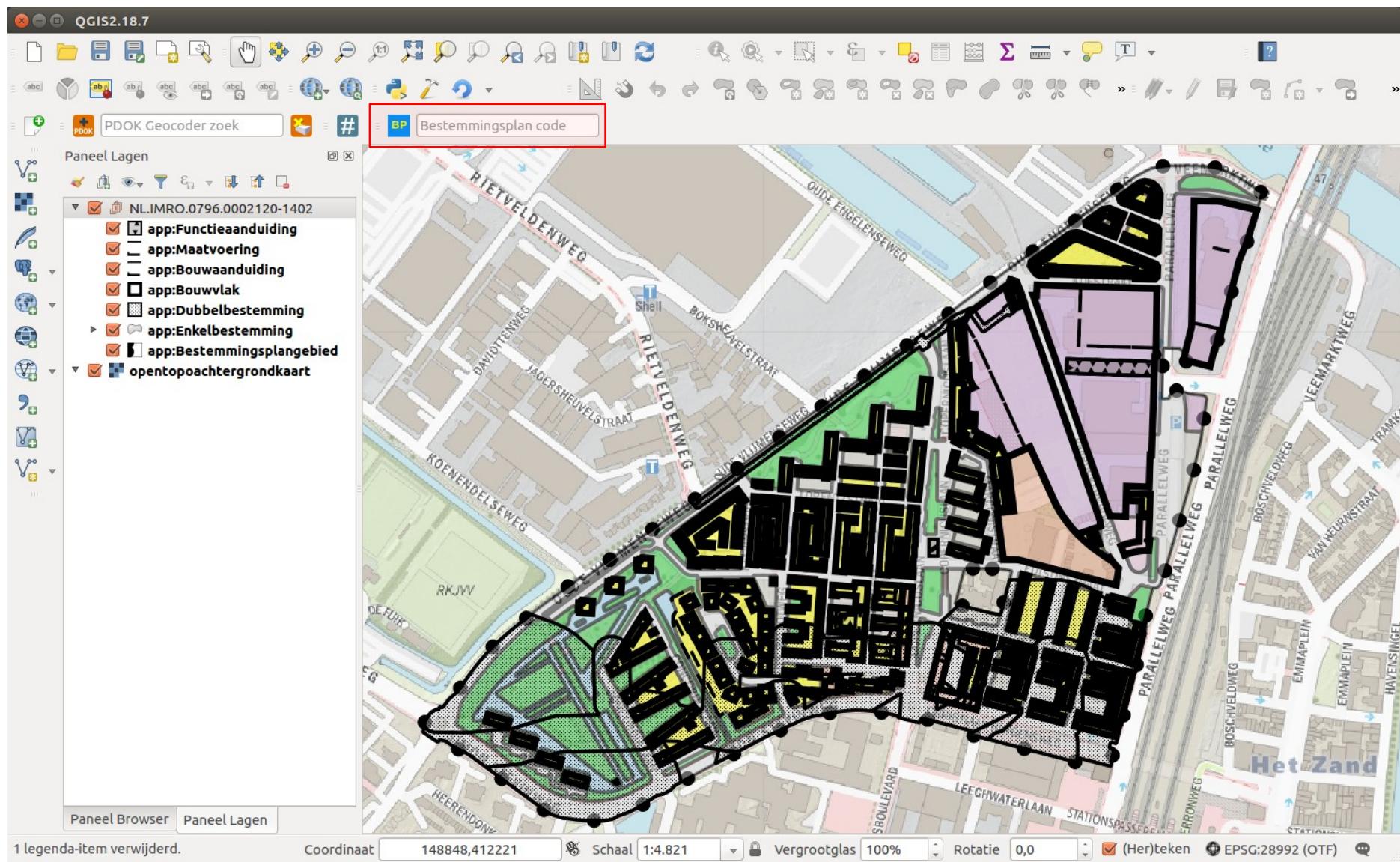
- QT5 + Python3
- September 2017
- Niet backwards compatible
- Python plugins werken niet meer

A screenshot of the QGIS Python Console window. The title bar says "Python Console". The main area contains the following text:

```
1 Python Console
2 Use iface to access QGIS API interface •
   r Type help(iface) for more info
3 >>> print "Hello world"
4 Hello world
5 >>> print("Hello world")
6 Hello world
7
```

The bottom part of the window shows an empty command line input field starting with ">>> |".

# Ruimtelijke Plannen plugin



# Community!



# Ervaringen



# Ervaringen



# Vragen

