Nyall Dawson

North Road

NORTH ROAD

Nyall Dawson

North Road

.... also a QGIS developer!

NORTH ROAD

QGIS loves Qt!

Qt development experiences and advice from a massive open-source desktop application

NORTH ROAD

NORTH ROAD

What is QGIS?

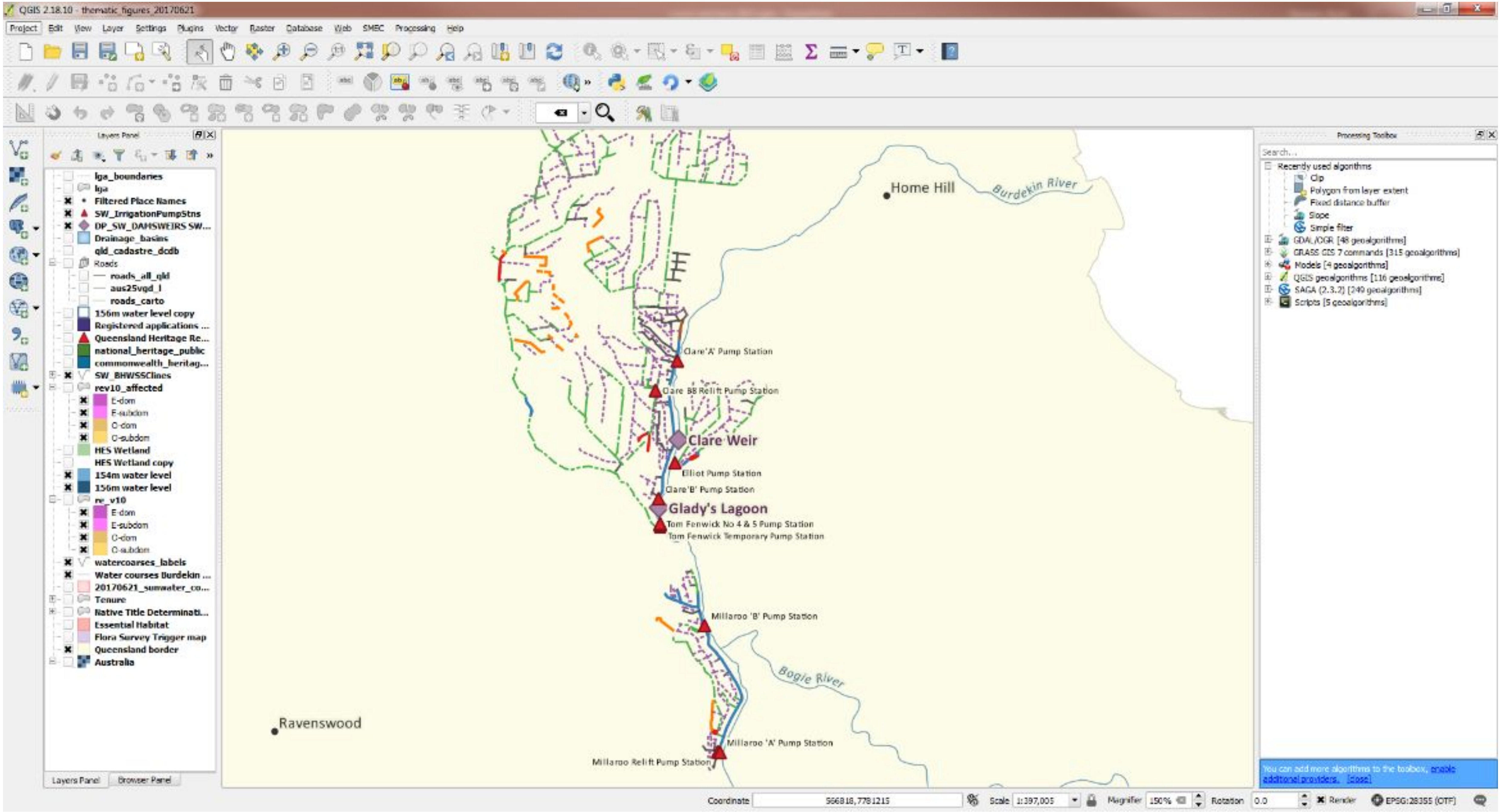
What is a ~~Q~~GIS?

“Geographic Information System”

“Geographic Information System”

(a mapping application)

NORTH ROAD

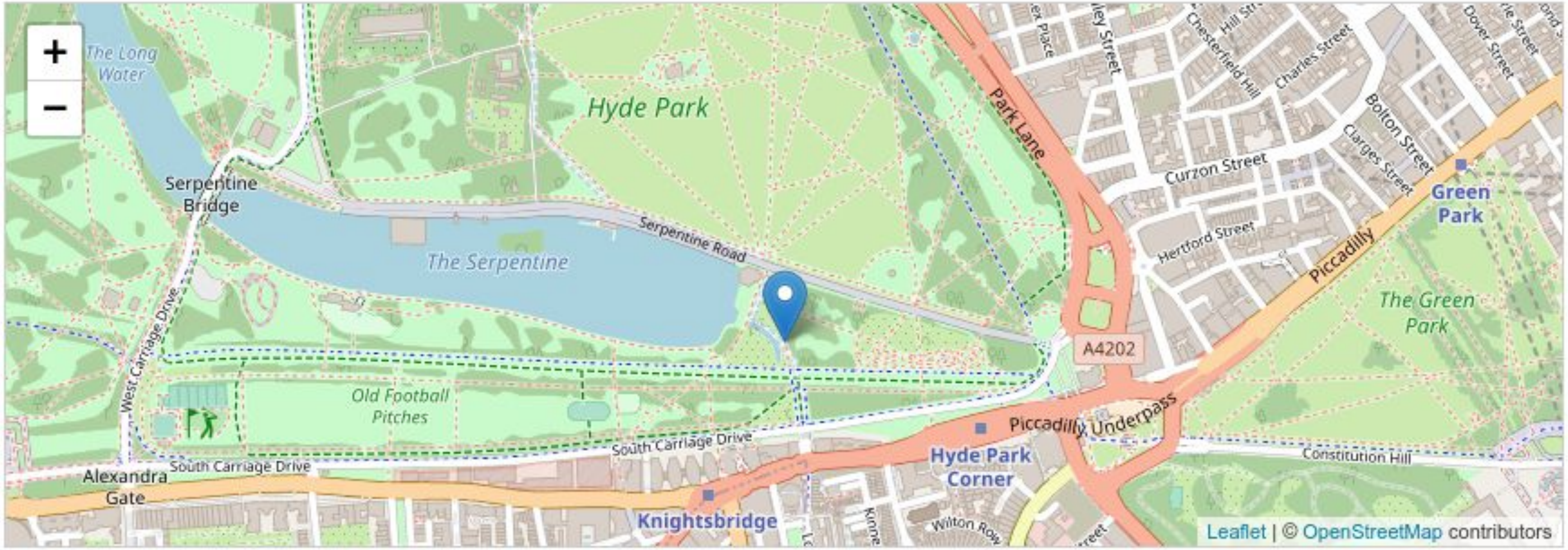


You can add more algorithms to the toolbox, enable additional providers. [close]

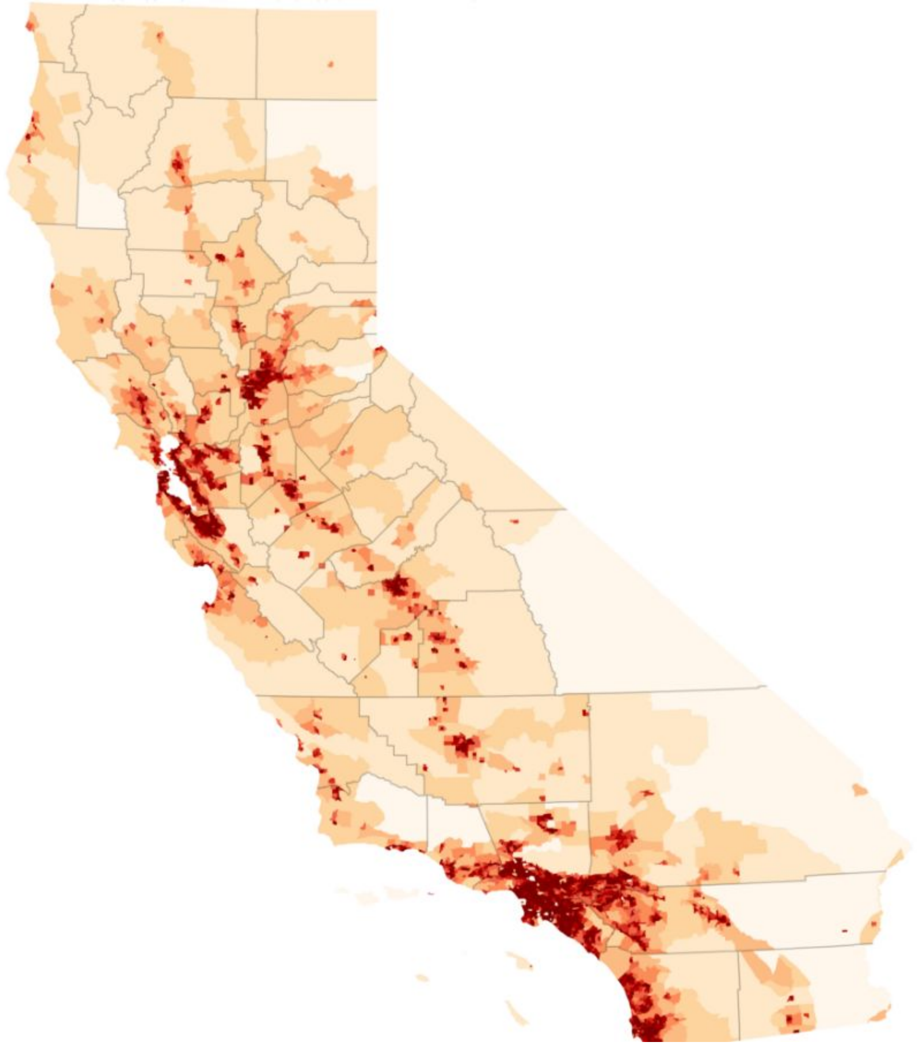
Coordinate 566818,7781215 Scale 1:397,005 Magnifier 150% Rotation 0.0 ☒ Render EPSG:28355 (OTF)

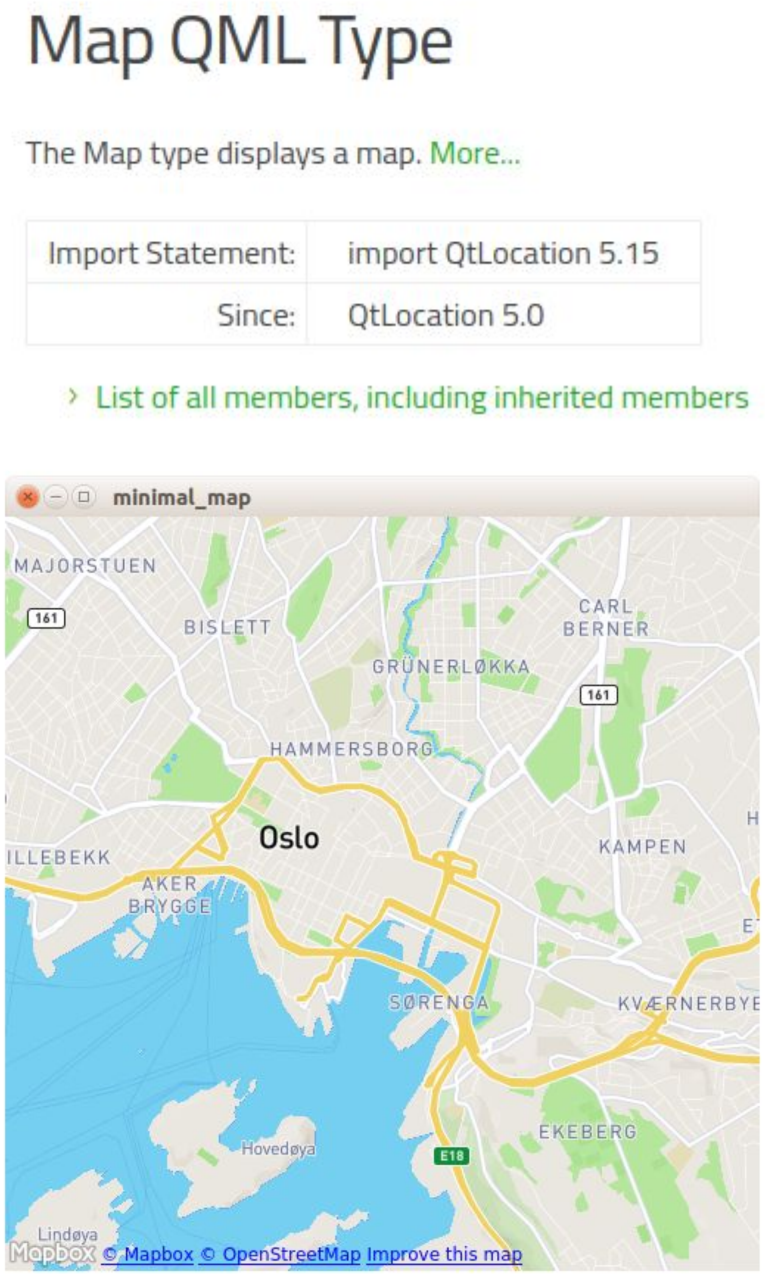
But why?

Leaflet


+
−
The Long Water
Hyde Park
Park Lane
Serpentine Bridge
The Serpentine
Serpentine Road
Curzon Street
Hertford Street
Bolton Street
Clarges Street
Charles Street
Chesterfield Hill
Dover Street
Green Park
Piccadilly
The Green Park
A4202
West Carriage Drive
Old Football Pitches
Piccadilly Underpass
Hyde Park Corner
South Carriage Drive
Constitution Hill
Alexandra Gate
Knightsbridge
Wilton Row
Leaflet | © OpenStreetMap contributors

Data-Driven Documents



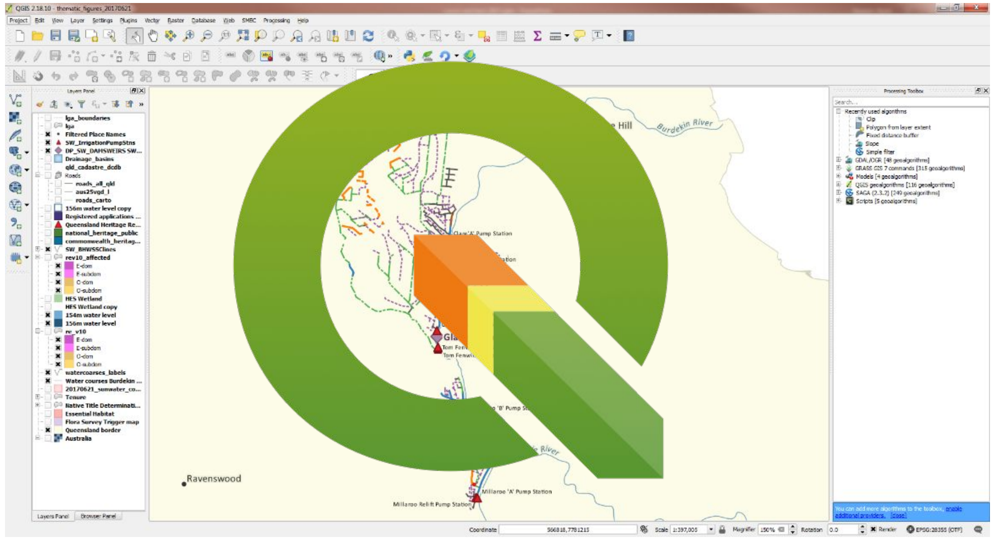


Map QML Type

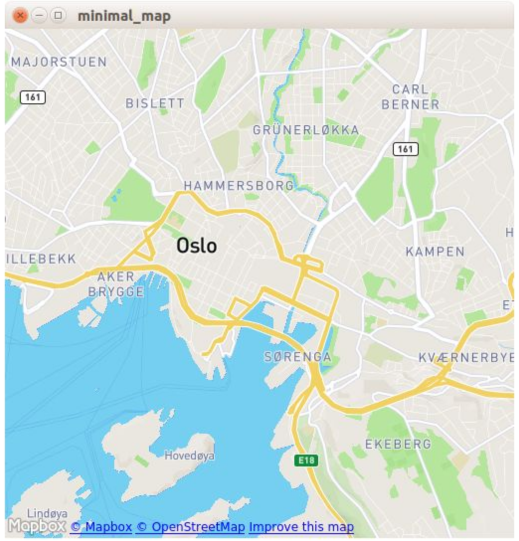
The Map type displays a map. More...

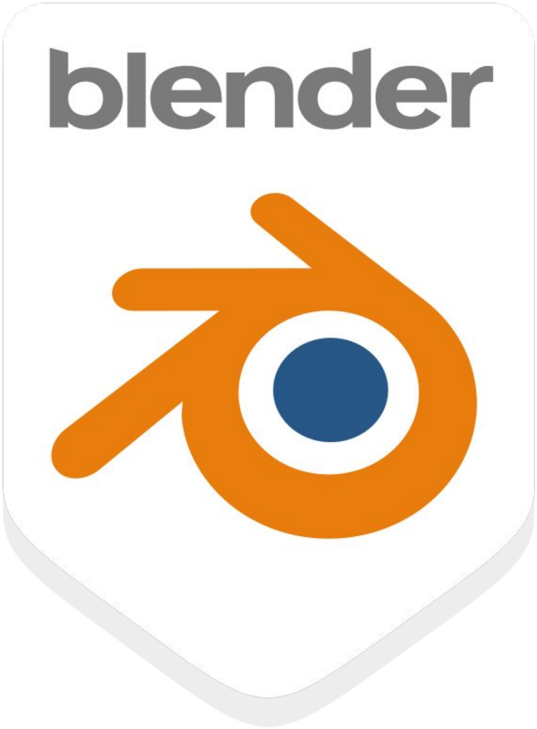
Import Statement:	import QtLocation 5.15
Since:	QtLocation 5.0

› List of all members, including inherited members

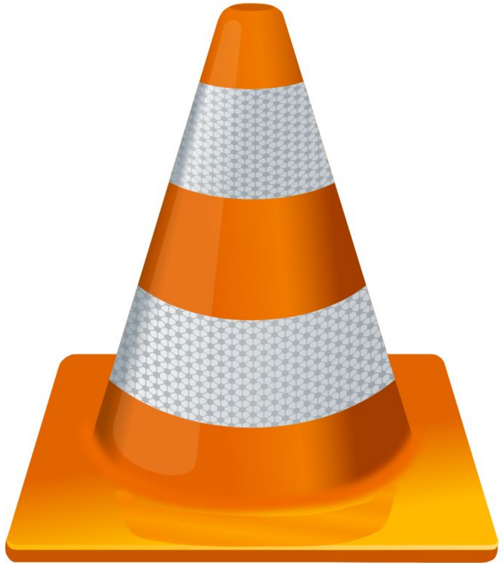

Ravenswood
Burdekin River
Millaroo 'A' Pump Station
Millaroo Relift Pump Station

!=


minimal_map
MAJORSTUEN
BISLETT
CARL BERNER
GRÜNERLØKKA
HAMMERSBORG
Oslo
ILLEBEKK
AKER BRYGGE
KAMPEN
SØRENGA
KVÆRNERBYEN
EKEBERG
Hovedøya
Lindøya
161
E18
Mapbox
© Mapbox © OpenStreetMap Improve this map


blender

!=



What is a ~~Q~~GIS?

Many different tasks

1. Consume spatial data

NORTH ROAD

2. Creation of spatial data

3. Cartography

4. Data analysis

5. Automation/ETL

Some special concepts

Coordinate Reference Systems

NORTH ROAD

Latitude/longitude

“WGS84”

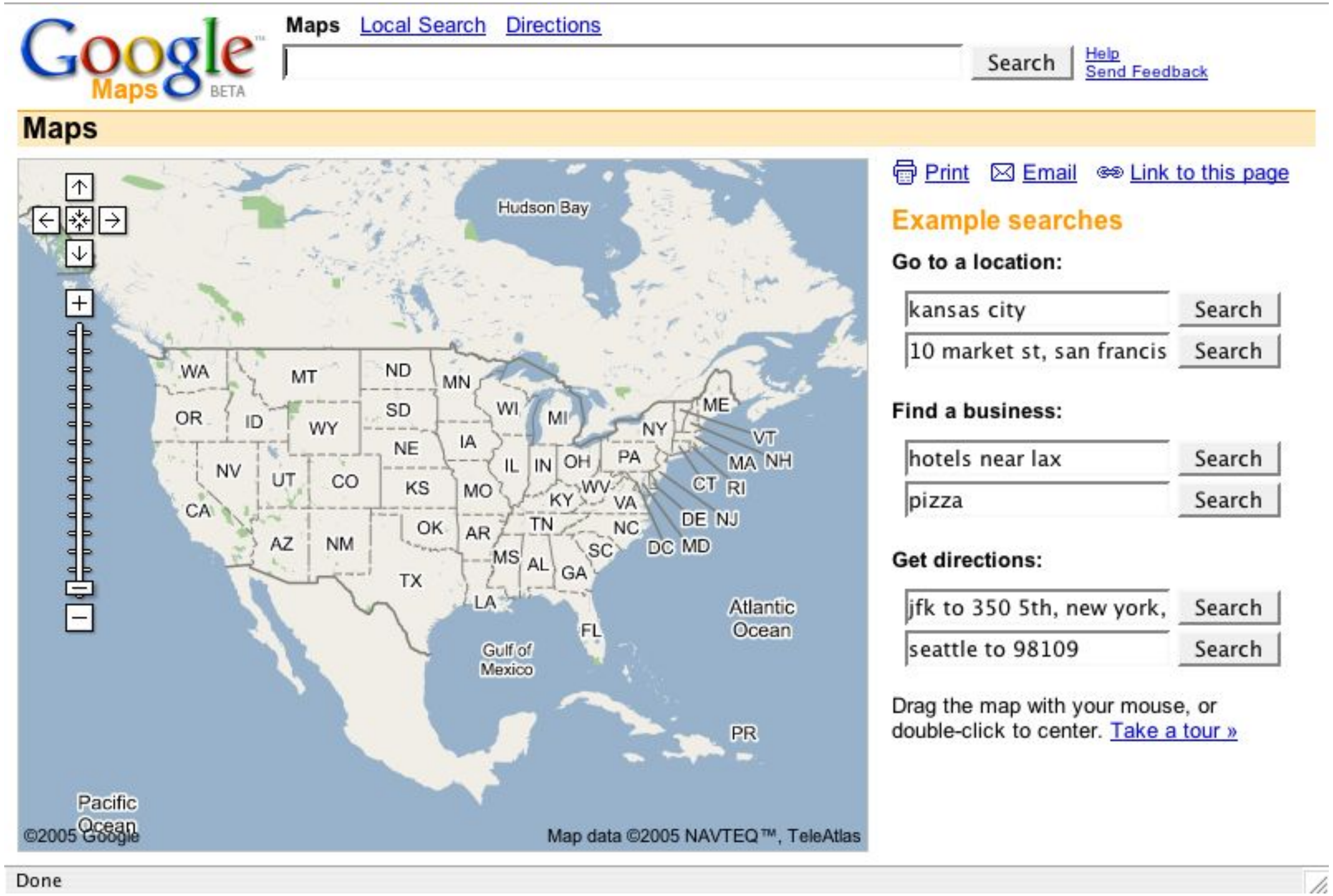
~~WGS84~~

+/- 30m

Projections, e.g.

Web Mercator

NORTH ROAD



Google Maps BETA

Maps Local Search Directions

Search Help Send Feedback

Maps

Print Email Link to this page

Example searches

Go to a location:

kansas city Search

10 market st, san francis Search

Find a business:

hotels near lax Search

pizza Search

Get directions:

jfk to 350 5th, new york, Search

seattle to 98109 Search

Drag the map with your mouse, or double-click to center. Take a tour »

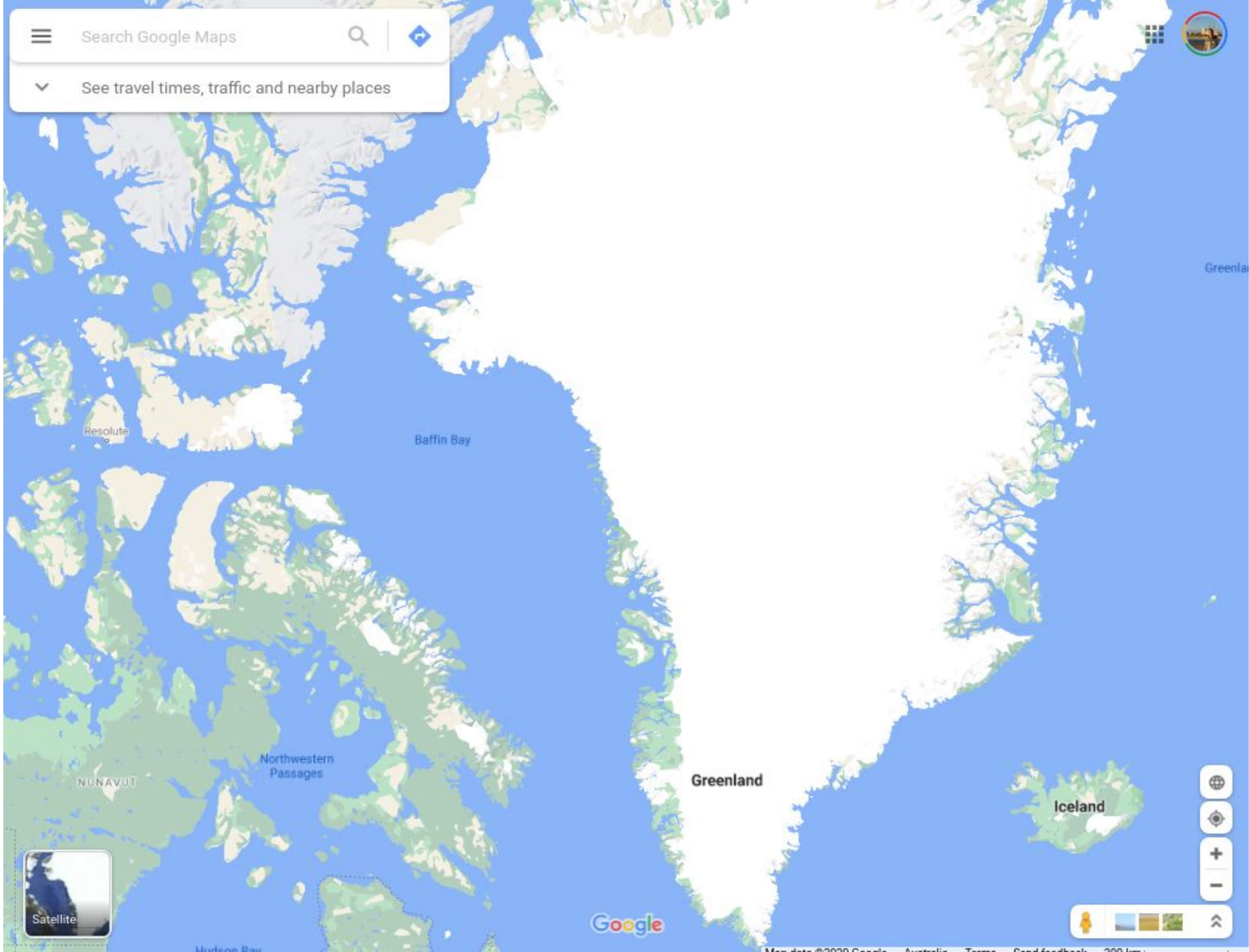
Done

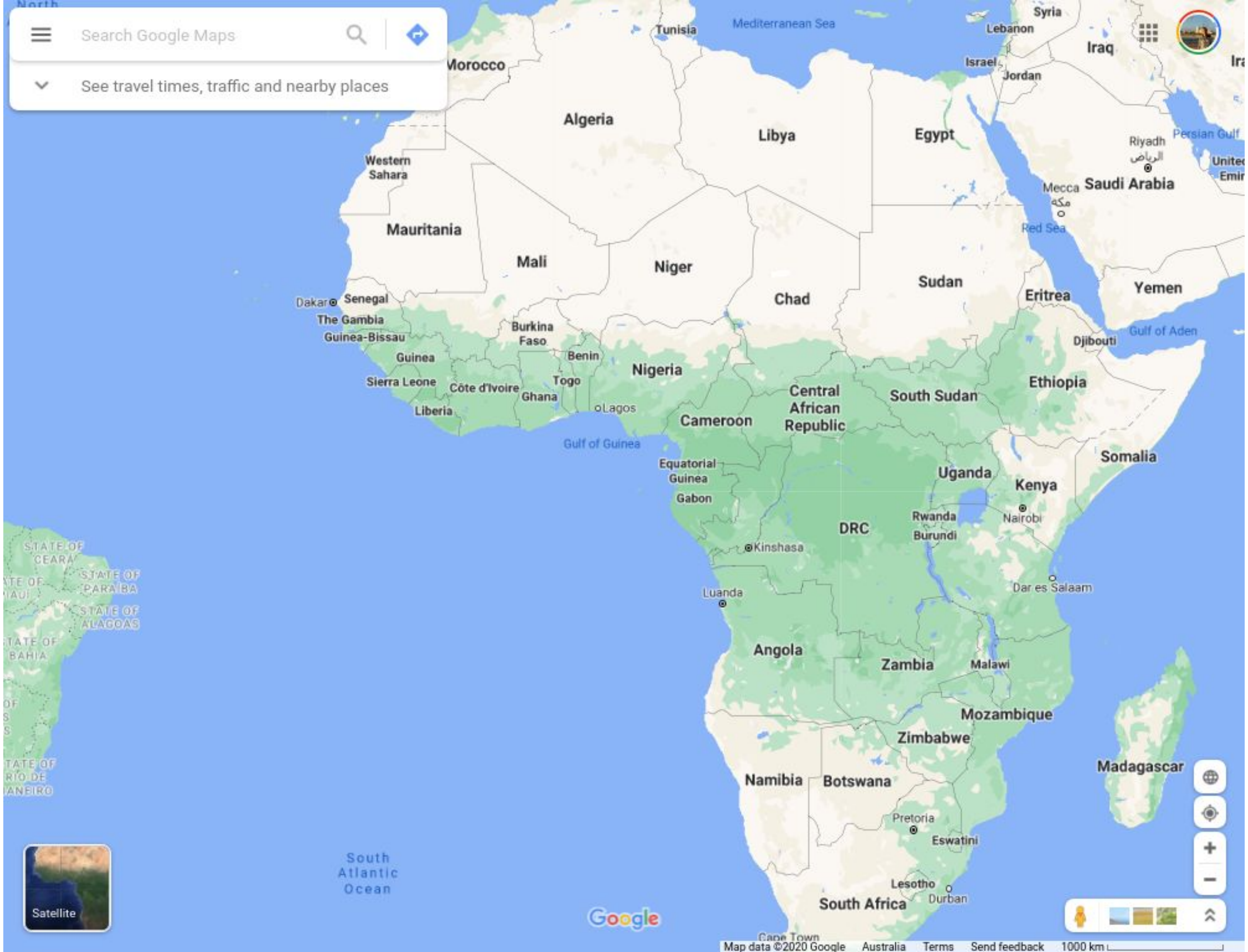
XYZ tiles
Vector tiles (e.g. MapBox)

Web Mercator

NORTH ROAD

Web ~~Mercator~~


Search Google Maps
See travel times, traffic and nearby places
Greenla
Resolute
Baffin Bay
Northwestern Passages
NUNAVUT
Greenland
Iceland
Satellite
Google
Hudson Bay
Map data ©2020 Google Australia Terms Send feedback 200 km


Search Google Maps
See travel times, traffic and nearby places
Mediterranean Sea
Tunisia
Syria
Lebanon
Iraq
Israel
Jordan
Morocco
Algeria
Libya
Egypt
Riyadh
الرياض
Persian Gulf
United Emir
Western Sahara
Mecca
مكة
Saudi Arabia
Red Sea
Mauritania
Mali
Niger
Chad
Sudan
Eritrea
Yemen
Dakar
Senegal
The Gambia
Guinea-Bissau
Burkina Faso
Benin
Gulf of Aden
Djibouti
Guinea
Nigeria
Togo
Sierra Leone
Côte d'Ivoire
Ghana
Liberia
Lagos
Cameroon
Central African Republic
South Sudan
Ethiopia
Gulf of Guinea
Somalia
Equatorial Guinea
Gabon
Uganda
Kenya
Nairobi
Rwanda
Burundi
DRC
Kinshasa
Dar es Salaam
Luanda
Angola
Zambia
Malawi
Mozambique
Zimbabwe
Madagascar
Namibia
Botswana
Pretoria
Eswatini
Lesotho
Durban
South Africa
Cape Town
South Atlantic Ocean
STATE OF CEARÁ
STATE OF PARAÍBA
STATE OF ALAGOAS
STATE OF BAHIA
STATE OF RIO DE JANEIRO
Satellite
Google
Map data ©2020 Google
Australia
Terms
Send feedback
1000 km

Equal area

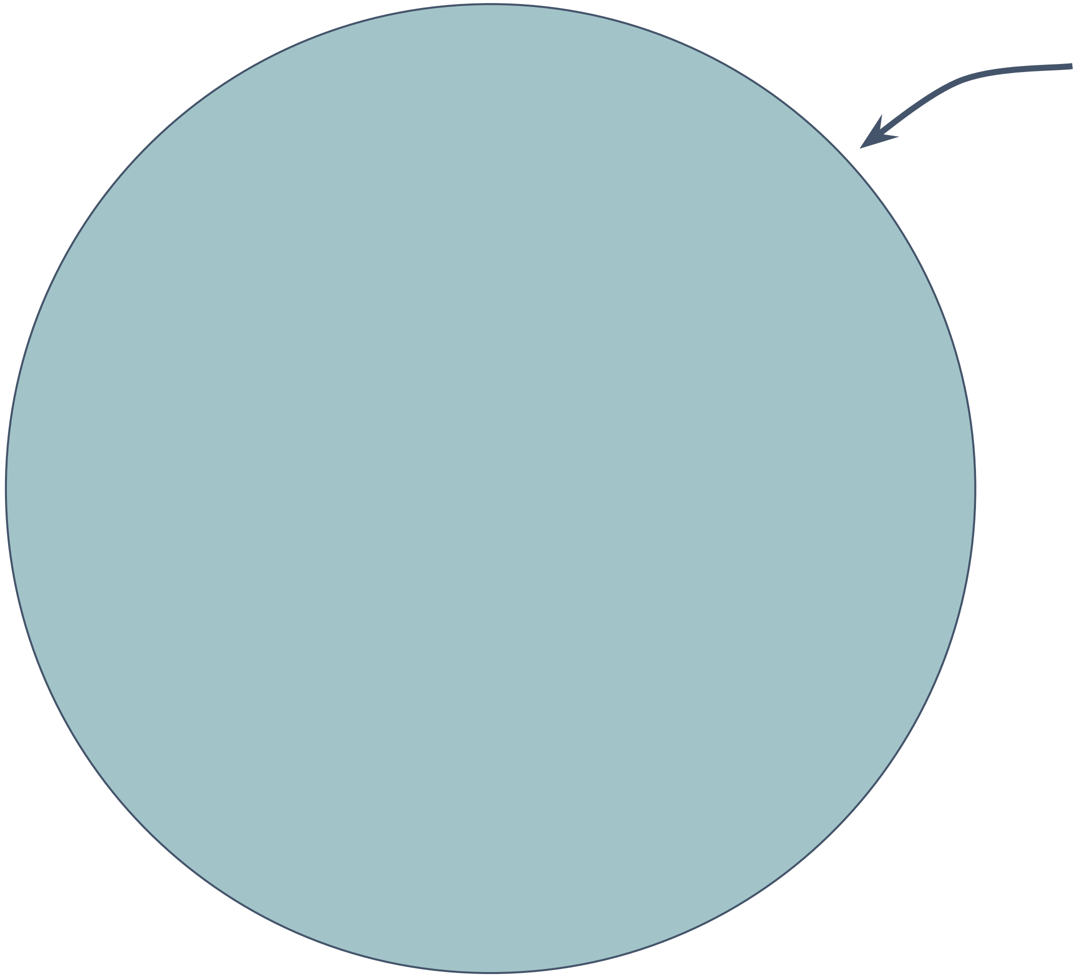
~~Equal area~~

“conformal”

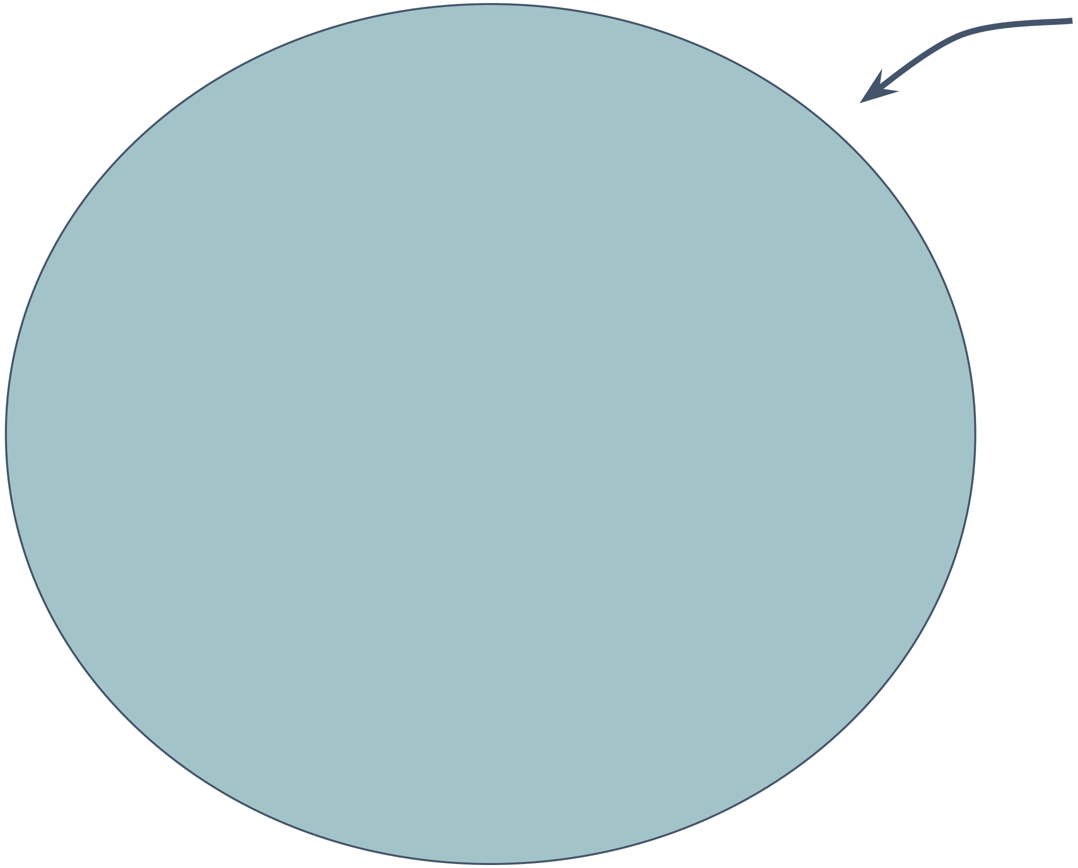
NORTH ROAD

~~“conformal”~~

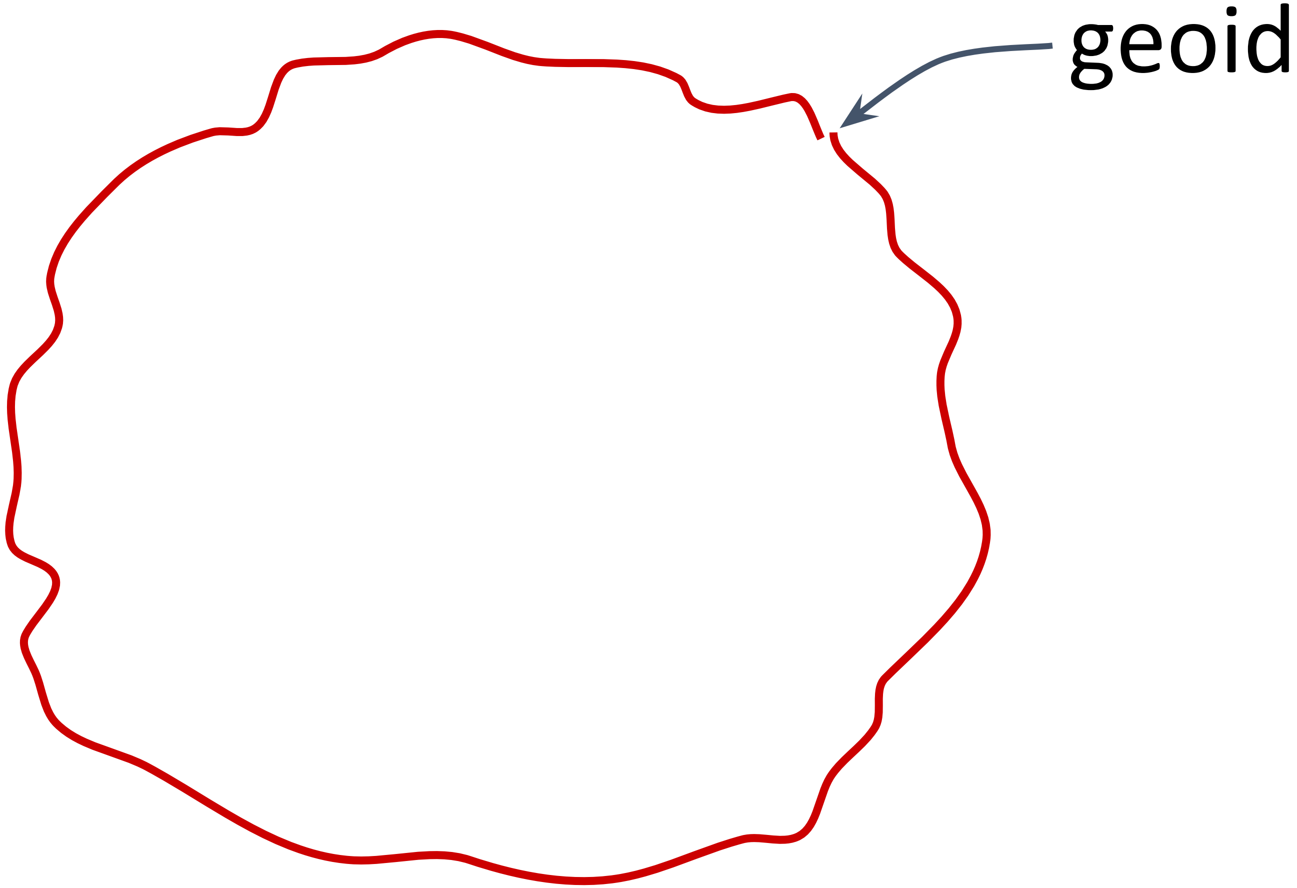
local projections

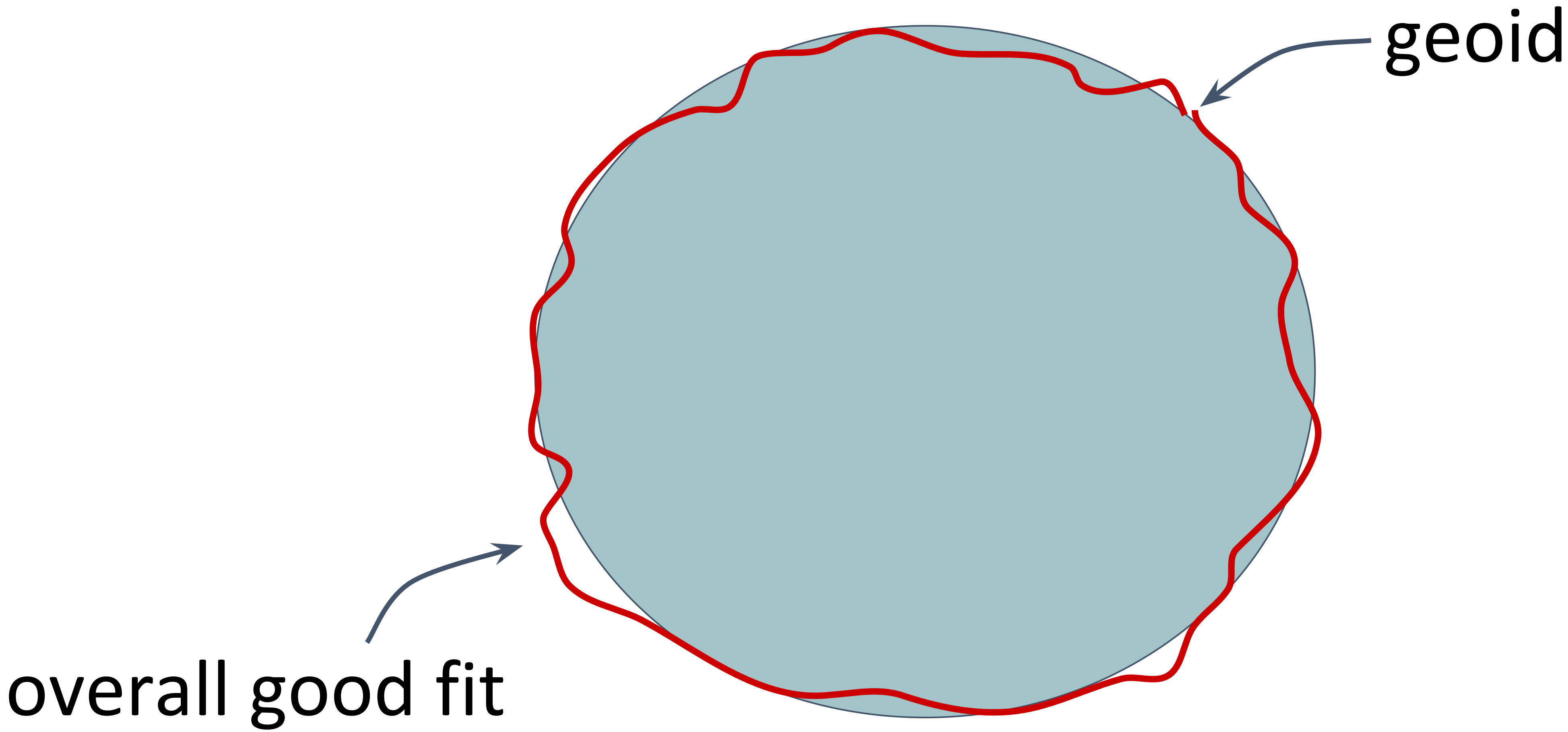


sphere

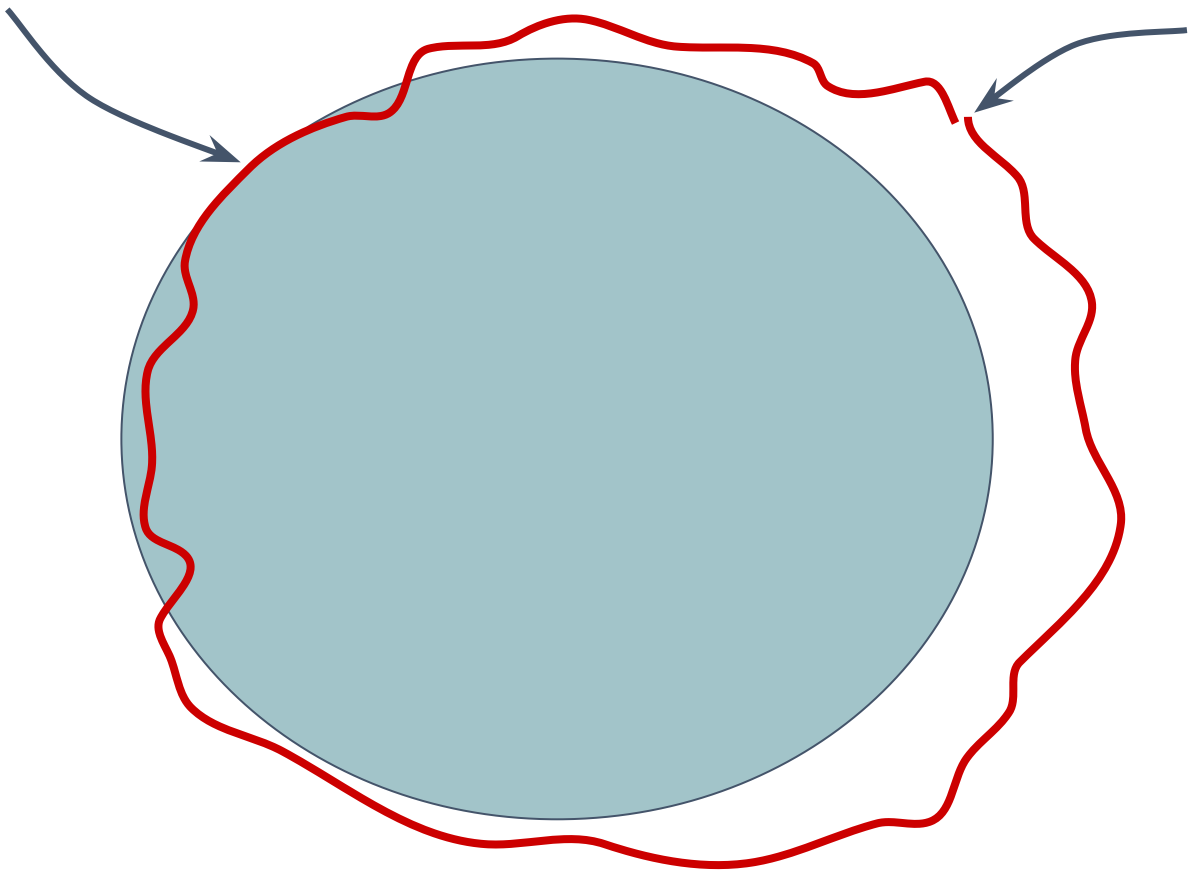


ellipsoid


geoid

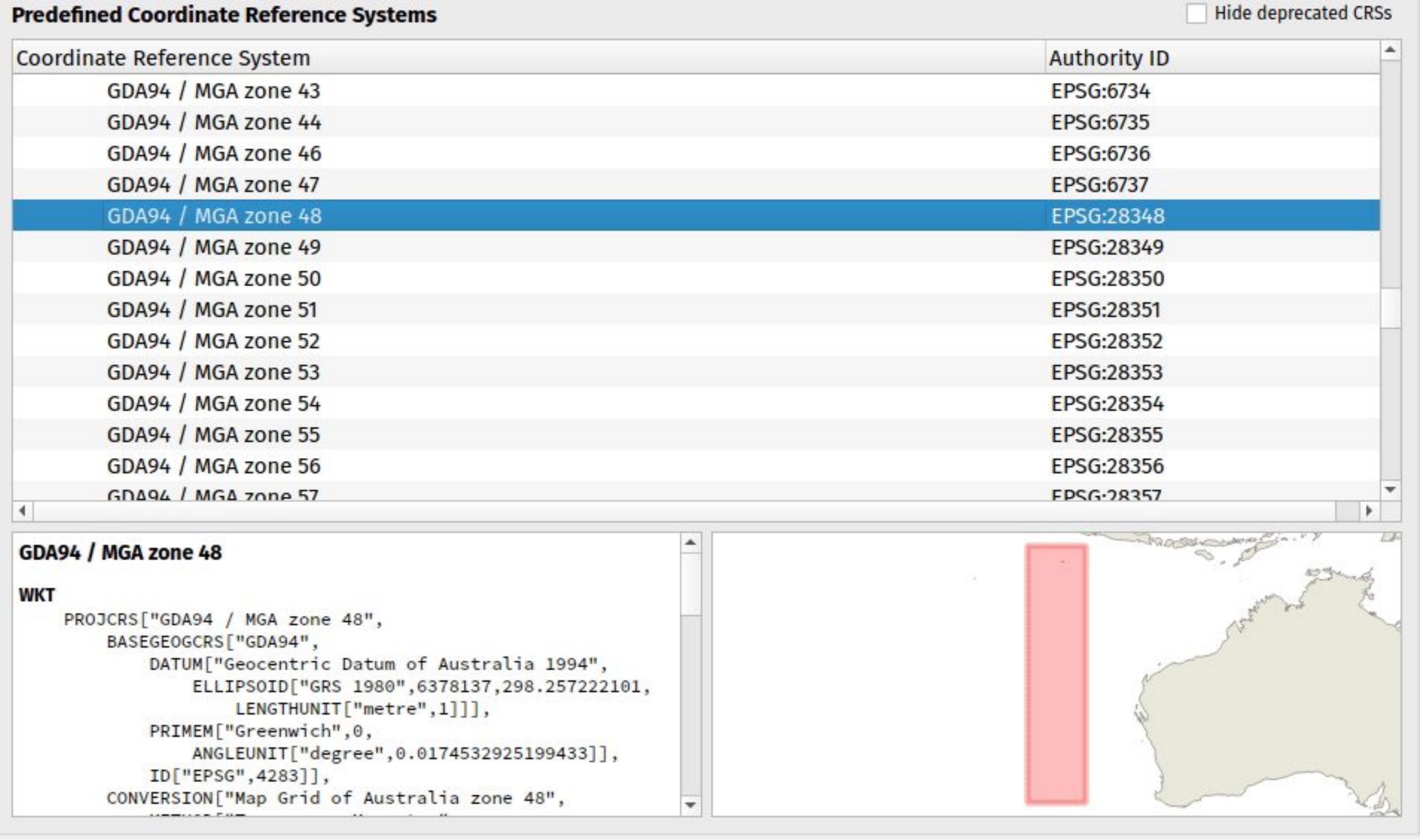

geoid
overall good fit

good fit for local area



geoid

“datums”



Predefined Coordinate Reference Systems

□ Hide deprecated CRSs

Coordinate Reference System	Authority ID
GDA94 / MGA zone 43	EPSG:6734
GDA94 / MGA zone 44	EPSG:6735
GDA94 / MGA zone 46	EPSG:6736
GDA94 / MGA zone 47	EPSG:6737
GDA94 / MGA zone 48	EPSG:28348
GDA94 / MGA zone 49	EPSG:28349
GDA94 / MGA zone 50	EPSG:28350
GDA94 / MGA zone 51	EPSG:28351
GDA94 / MGA zone 52	EPSG:28352
GDA94 / MGA zone 53	EPSG:28353
GDA94 / MGA zone 54	EPSG:28354
GDA94 / MGA zone 55	EPSG:28355
GDA94 / MGA zone 56	EPSG:28356
GDA94 / MGA zone 57	EPSG:28357

GDA94 / MGA zone 48

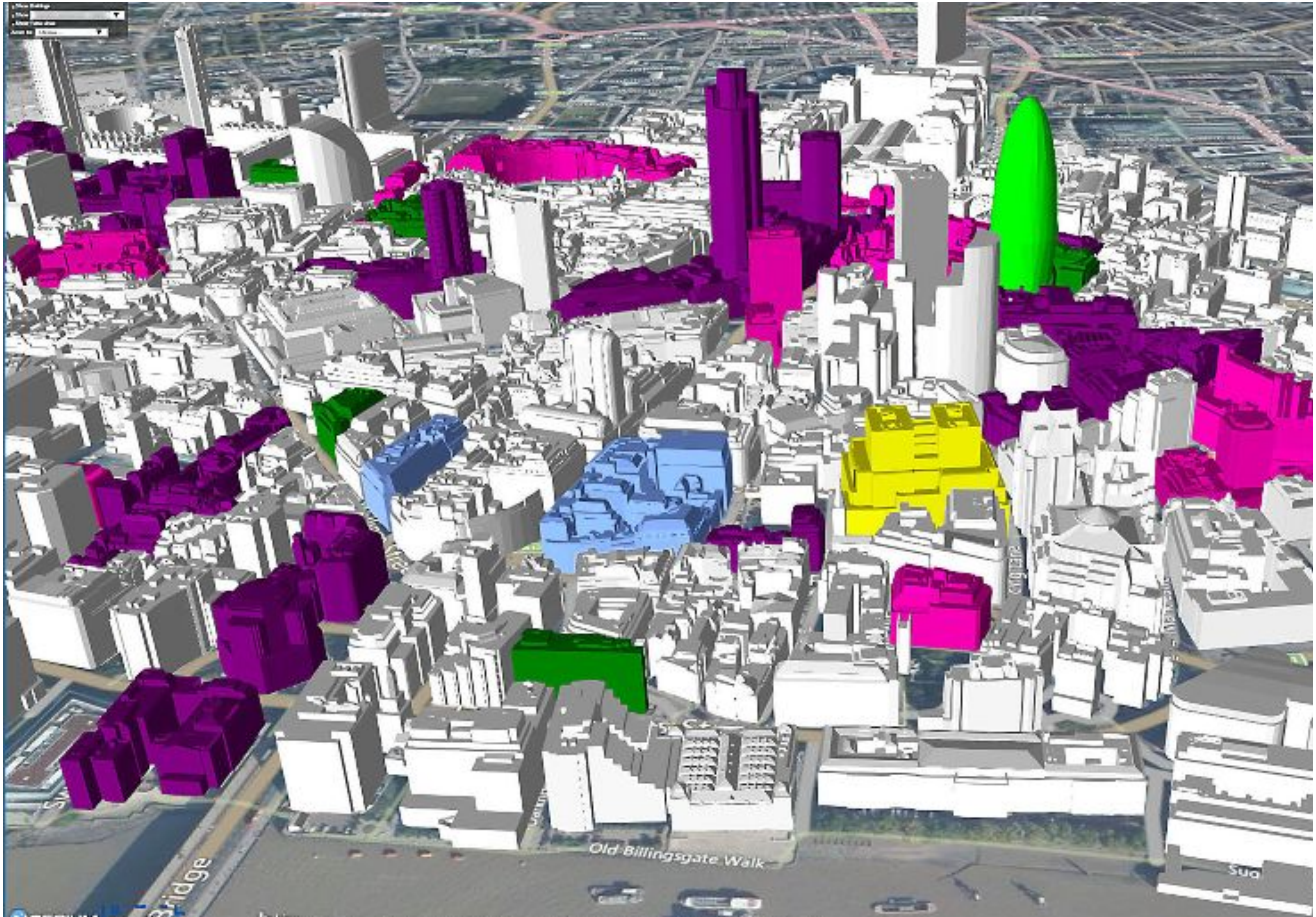
WKT

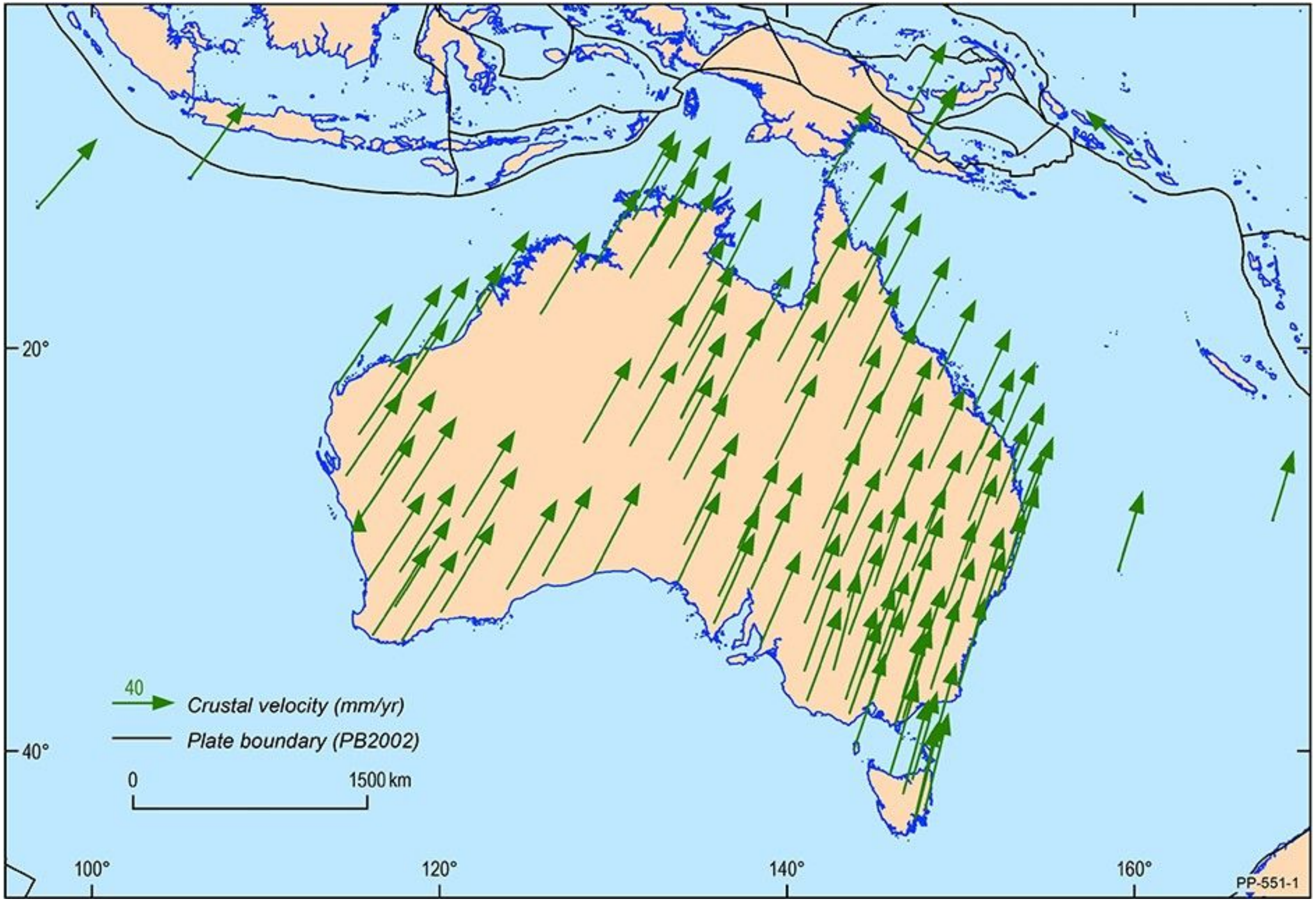
```
PROJCRS["GDA94 / MGA zone 48",
    BASEGEOGCRS["GDA94",
        DATUM["Geocentric Datum of Australia 1994",
            ELLIPSOID["GRS 1980",6378137,298.257222101,
                LENGTHUNIT["metre",1]]],
        PRIMEM["Greenwich",0,
            ANGLEUNIT["degree",0.0174532925199433]],
        ID["EPSG",4283]],
    CONVERSION["Map Grid of Australia zone 48",
```

Complexity of data types

Accuracy requirements

millimeter accuracy!


Old Billingsgate Walk
Bridge
Sua


20°
40°
100°
120°
140°
160°
40
Crustal velocity (mm/yr)
Plate boundary (PB2002)
0
1500 km
PP-551-1

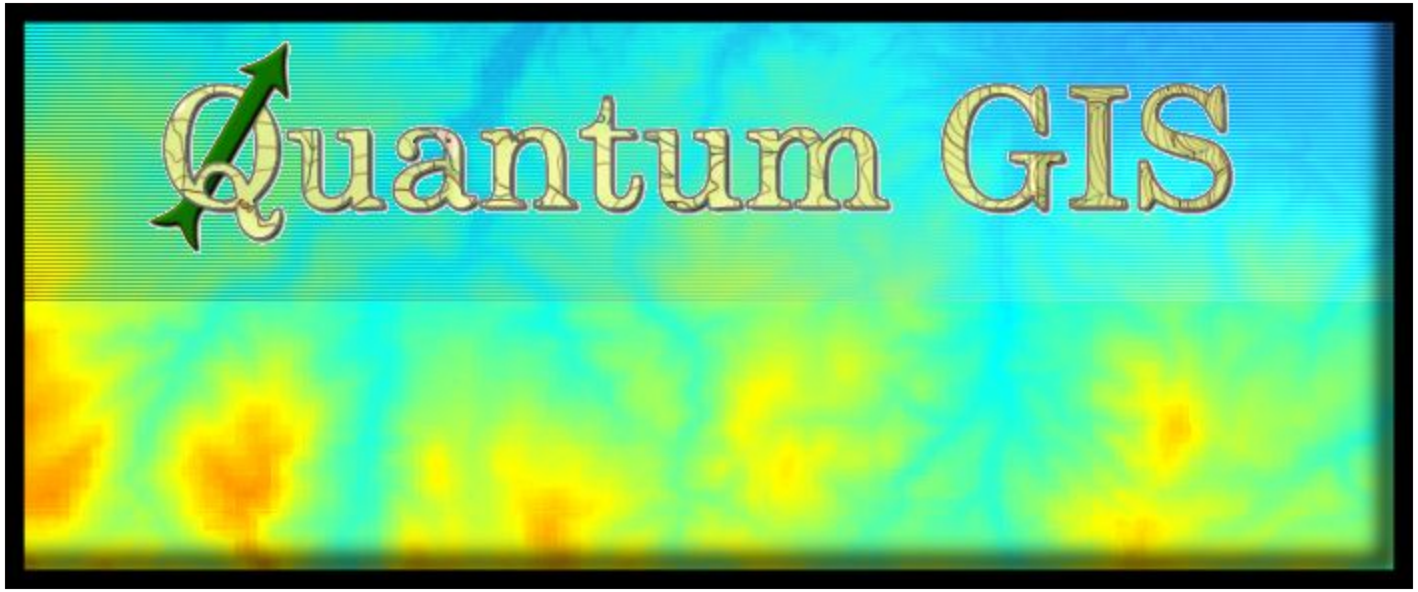
NORTH ROAD

RENSSELAER
HYDRANT.CO
EST 1922
GERN

What is a ~~Q~~GIS?

What is QGIS?

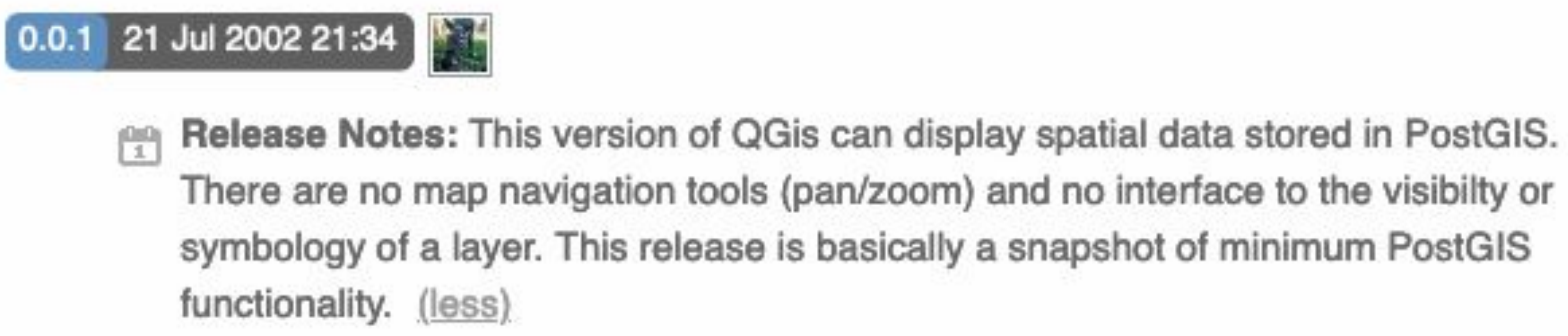
NORTH ROAD


Quantum GIS

Gary Sherman

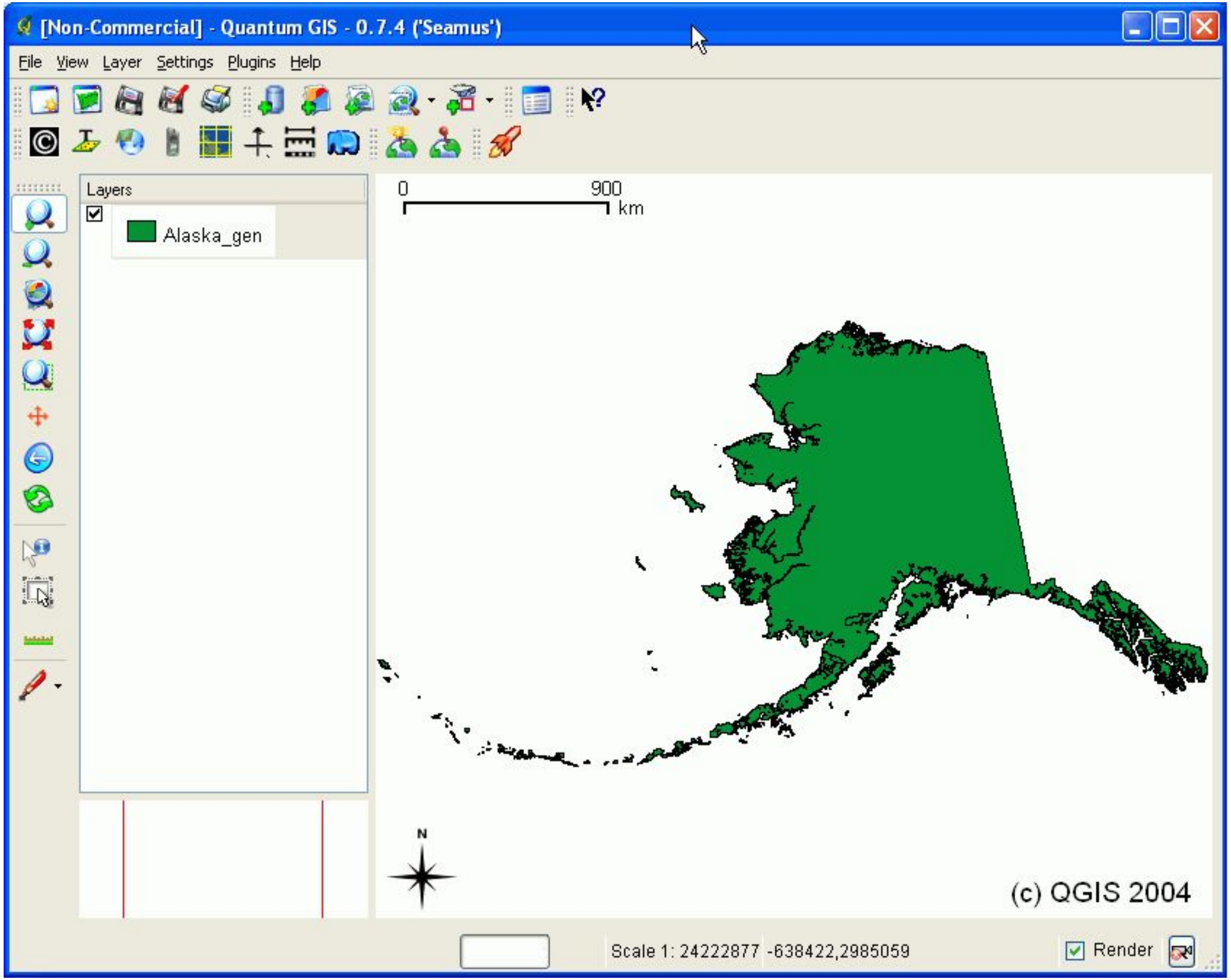
2002

NORTH ROAD



0.0.1 21 Jul 2002 21:34

Release Notes: This version of QGis can display spatial data stored in PostGIS. There are no map navigation tools (pan/zoom) and no interface to the visibilty or symbology of a layer. This release is basically a snapshot of minimum PostGIS functionality. (less)


[Non-Commercial] - Quantum GIS - 0.7.4 ('Seamus')
File View Layer Settings Plugins Help
Layers
☑ 🟩 Alaska_gen
0
900
km
N
(c) QGIS 2004
Scale 1: 24222877 -638422,2985059
☑ Render

GS: QGIS started out as a solo effort in February 2002, driven primarily by my after-hours desire to view PostGIS data on my Linux box. In my day job, I was working on displaying small-tract survey data stored in descriptive XML files. This was a Windows project, and I chose the cross-platform Qt framework to provide the GUI since I was familiar with it from my personal projects. I decided I could do the same at home for PostGIS data on my Linux box, so I started from scratch and began coding up a viewer. So, it really began as a hobby project, using C++ and Qt.

GS: As I said, my primary motivation was to produce a viewer for PostGIS data on Linux. With that in mind, I did choose a cross-platform framework to leave open the ability to compile for other operating systems.

I didn't have grandiose dreams of creating a full-fledged GIS. Ultimately, I hoped to be able to view a number of vector and raster formats with a nice legend and some basic map tools. Obviously, QGIS is way beyond that now, with a good suite of analysis tools and a huge number of plugins.

QGIS Today

… the most popular Open Source Desktop GIS

... the second most popular GIS after ESRI ArcGIS

… translated in 48 languages

… available for Linux, Windows, MacOS and Android

… released every 4 months with LTR releases every year

… available as desktop, server, and mobile clients

NORTH ROAD

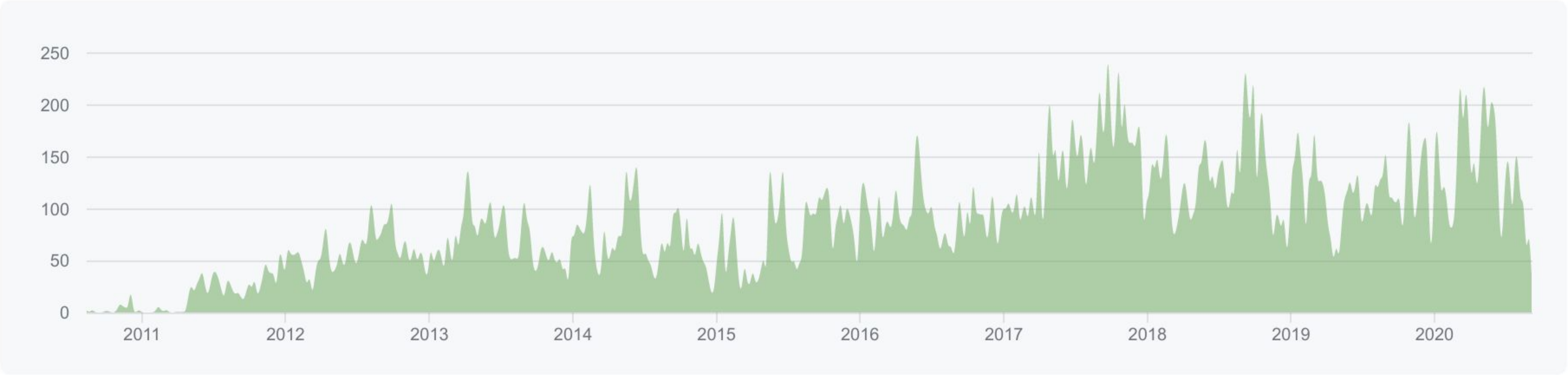
Some stats

Users?

Website visits ~750k/month

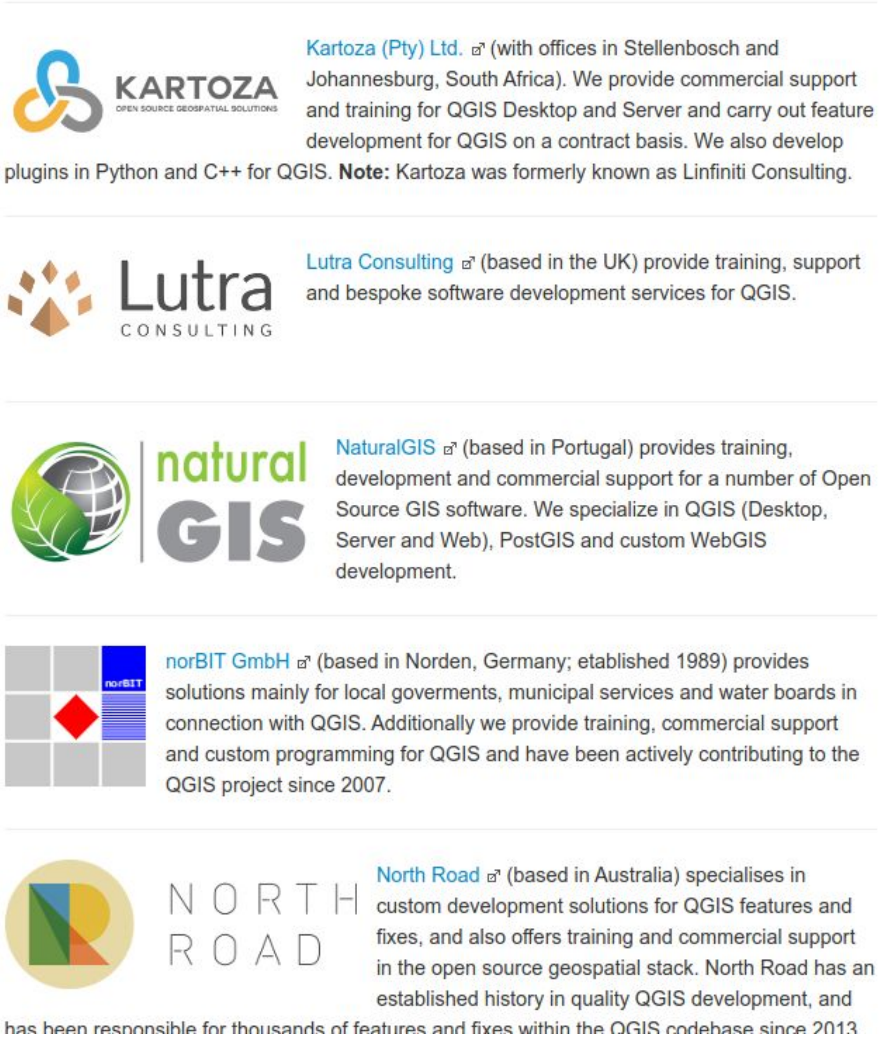
~500 (code) contributors

~1.5-2 million lines of code


250
200
150
100
50
0
2011
2012
2013
2014
2015
2016
2017
2018
2019
2020

NORTH ROAD

~30 Commercial support



Kartoza (Pty) Ltd. (with offices in Stellenbosch and Johannesburg, South Africa). We provide commercial support and training for QGIS Desktop and Server and carry out feature development for QGIS on a contract basis. We also develop plugins in Python and C++ for QGIS. **Note:** Kartoza was formerly known as Linfiniti Consulting.

Lutra Consulting (based in the UK) provide training, support and bespoke software development services for QGIS.

NaturalGIS (based in Portugal) provides training, development and commercial support for a number of Open Source GIS software. We specialize in QGIS (Desktop, Server and Web), PostGIS and custom WebGIS development.

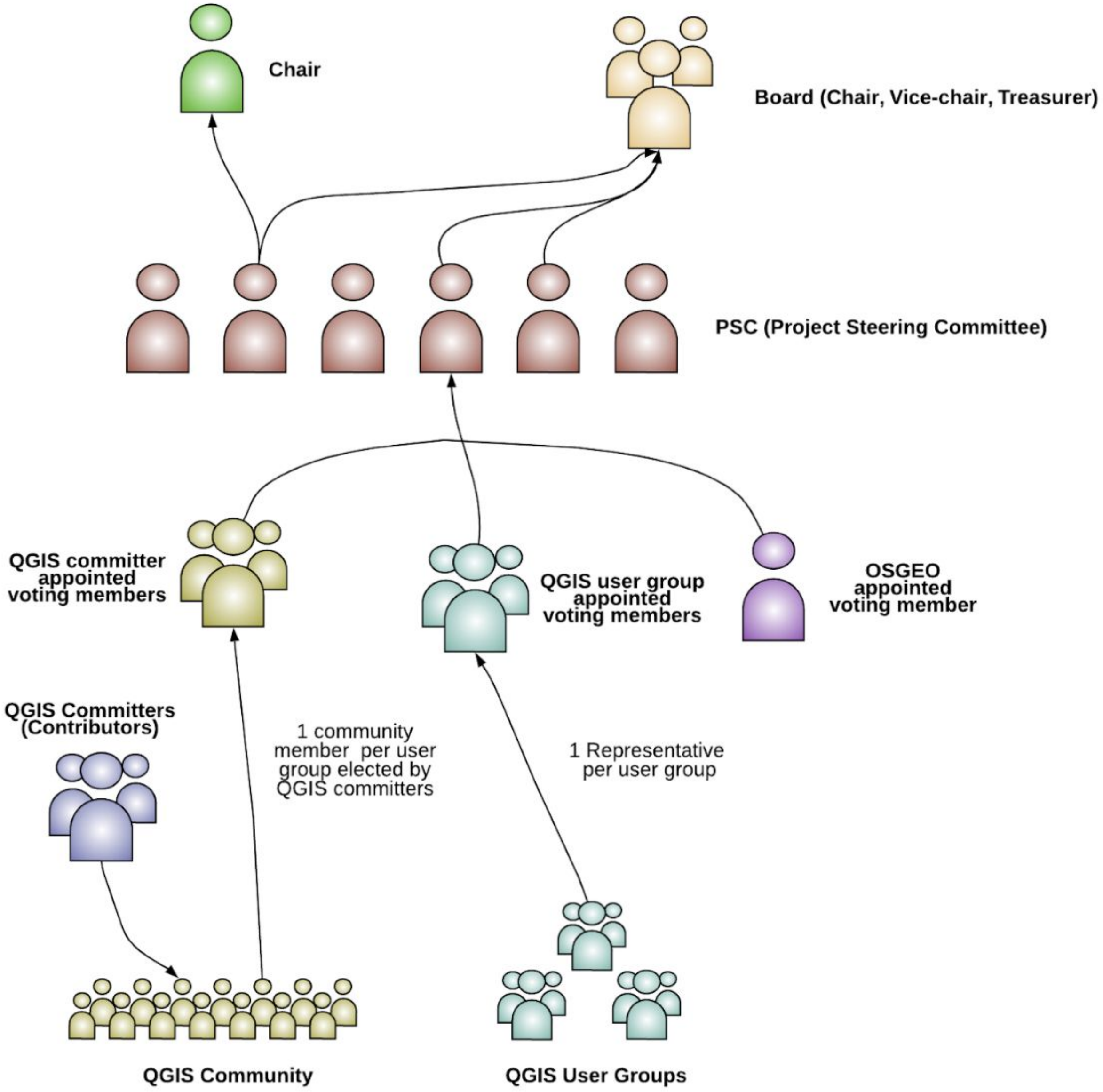
norBIT GmbH (based in Norden, Germany; etablished 1989) provides solutions mainly for local goverments, municipal services and water boards in connection with QGIS. Additionally we provide training, commercial support and custom programming for QGIS and have been actively contributing to the QGIS project since 2007.

North Road (based in Australia) specialises in custom development solutions for QGIS features and fixes, and also offers training and commercial support in the open source geospatial stack. North Road has an established history in quality QGIS development, and has been responsible for thousands of features and fixes within the QGIS codebase since 2013

The QGIS Project

NORTH ROAD

QGIS.ORG ASSOCIATION



Chair

Board (Chair, Vice-chair, Treasurer)

PSC (Project Steering Committee)

QGIS committer appointed voting members

QGIS user group appointed voting members

OSGEO appointed voting member

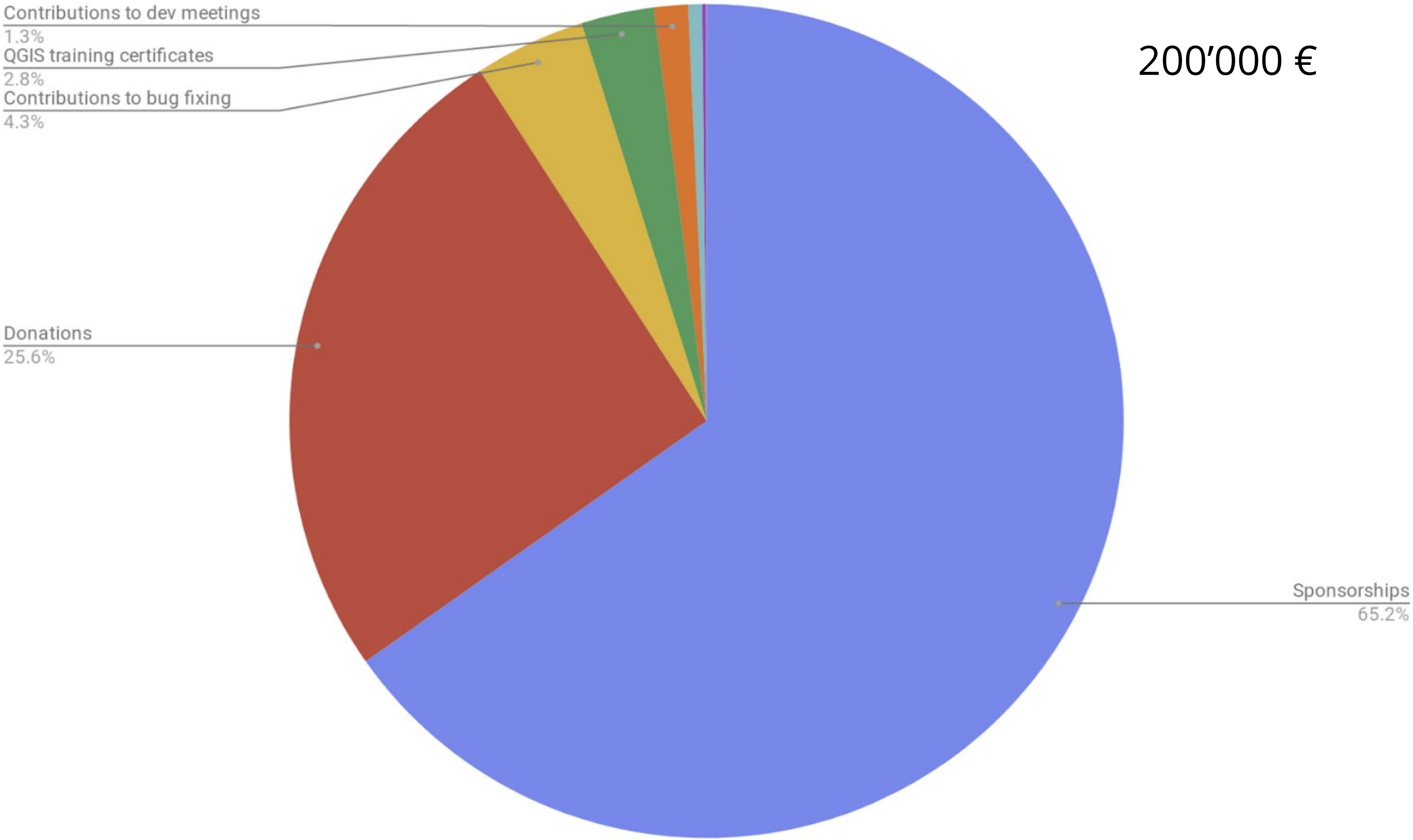
QGIS Committers (Contributors)

1 community member per user group elected by QGIS committers

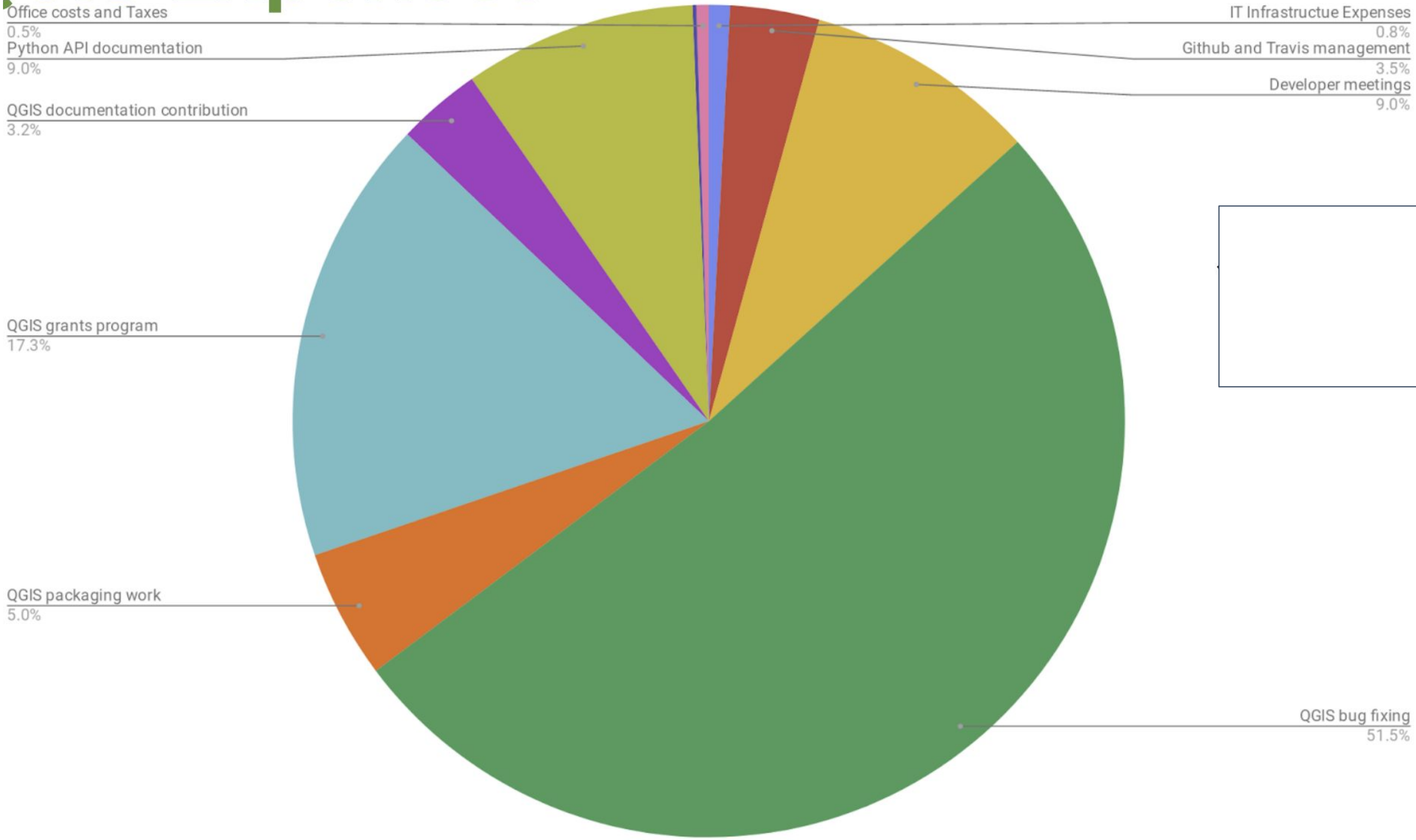
1 Representative per user group

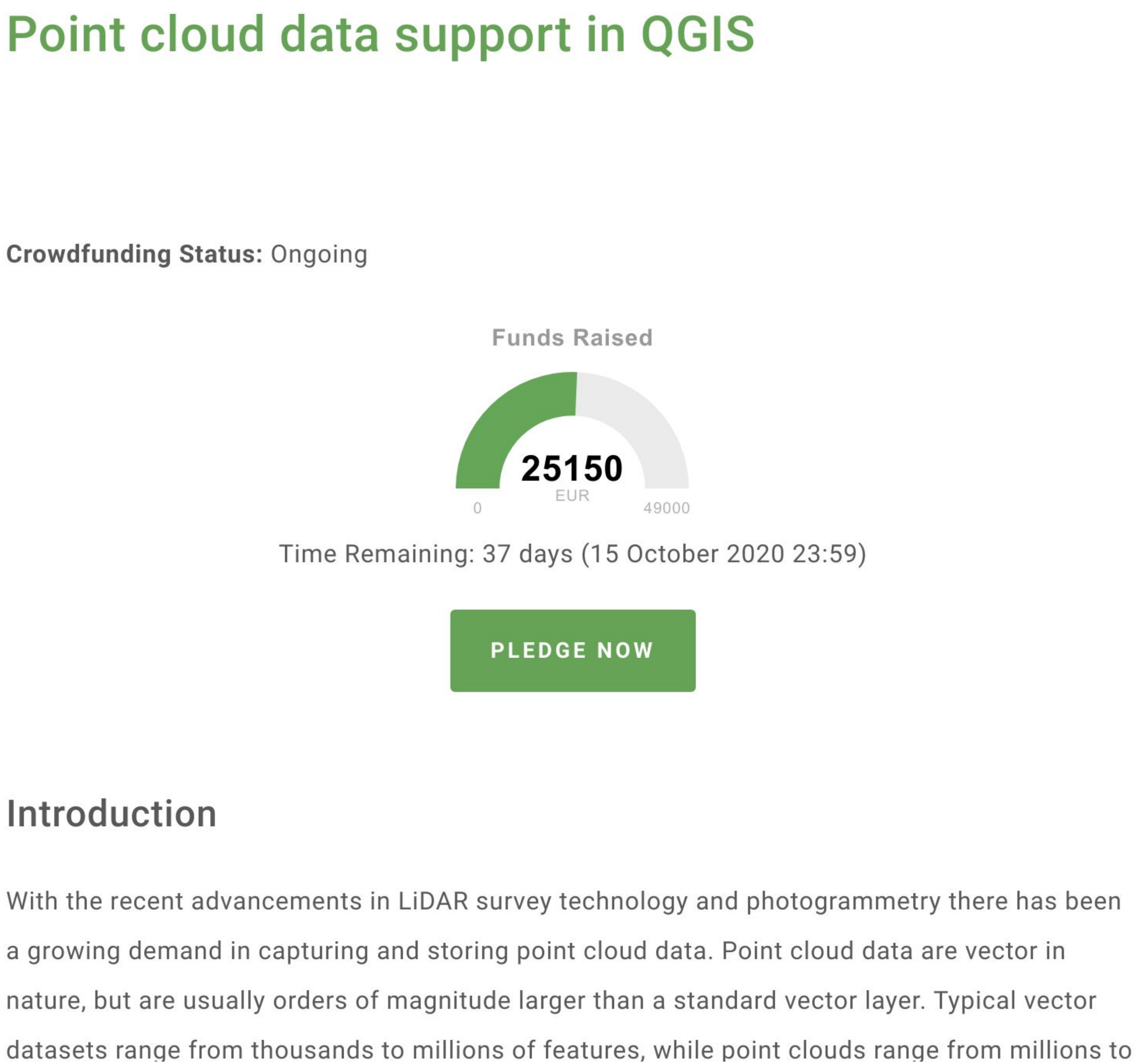
QGIS Community

QGIS User Groups


Contributions to dev meetings
1.3%
QGIS training certificates
2.8%
Contributions to bug fixing
4.3%
Donations
25.6%
Sponsorships
65.2%

200’000 €


Office costs and Taxes
0.5%
Python API documentation
9.0%
QGIS documentation contribution
3.2%
QGIS grants program
17.3%
QGIS packaging work
5.0%
IT Infrastructue Expenses
0.8%
Github and Travis management
3.5%
Developer meetings
9.0%
QGIS bug fixing
51.5%



Point cloud data support in QGIS

Crowdfunding Status: Ongoing

Funds Raised

25150

EUR

0

49000

Time Remaining: 37 days (15 October 2020 23:59)

PLEDGE NOW

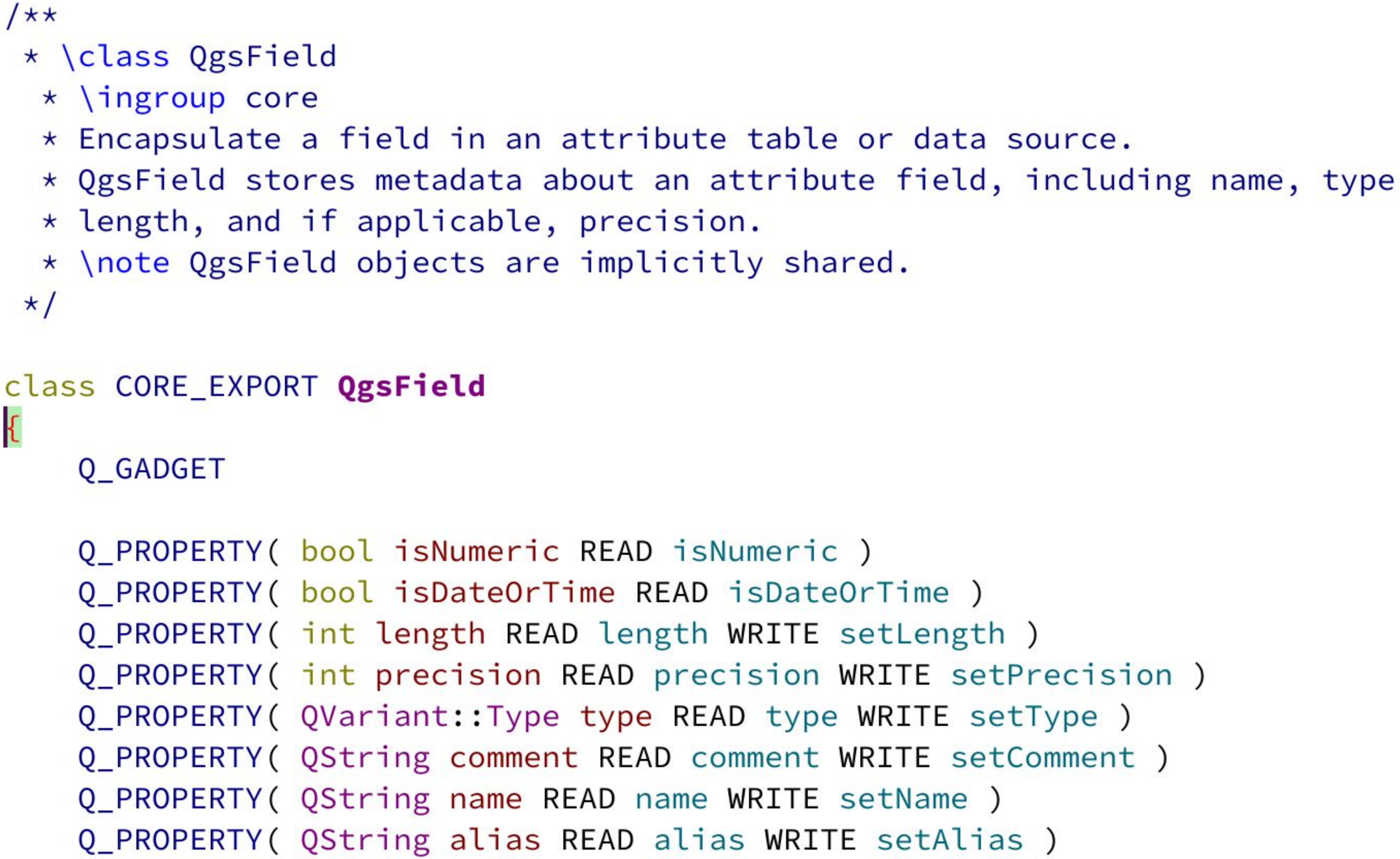
Introduction

With the recent advancements in LiDAR survey technology and photogrammetry there has been a growing demand in capturing and storing point cloud data. Point cloud data are vector in nature, but are usually orders of magnitude larger than a standard vector layer. Typical vector datasets range from thousands to millions of features, while point clouds range from millions to

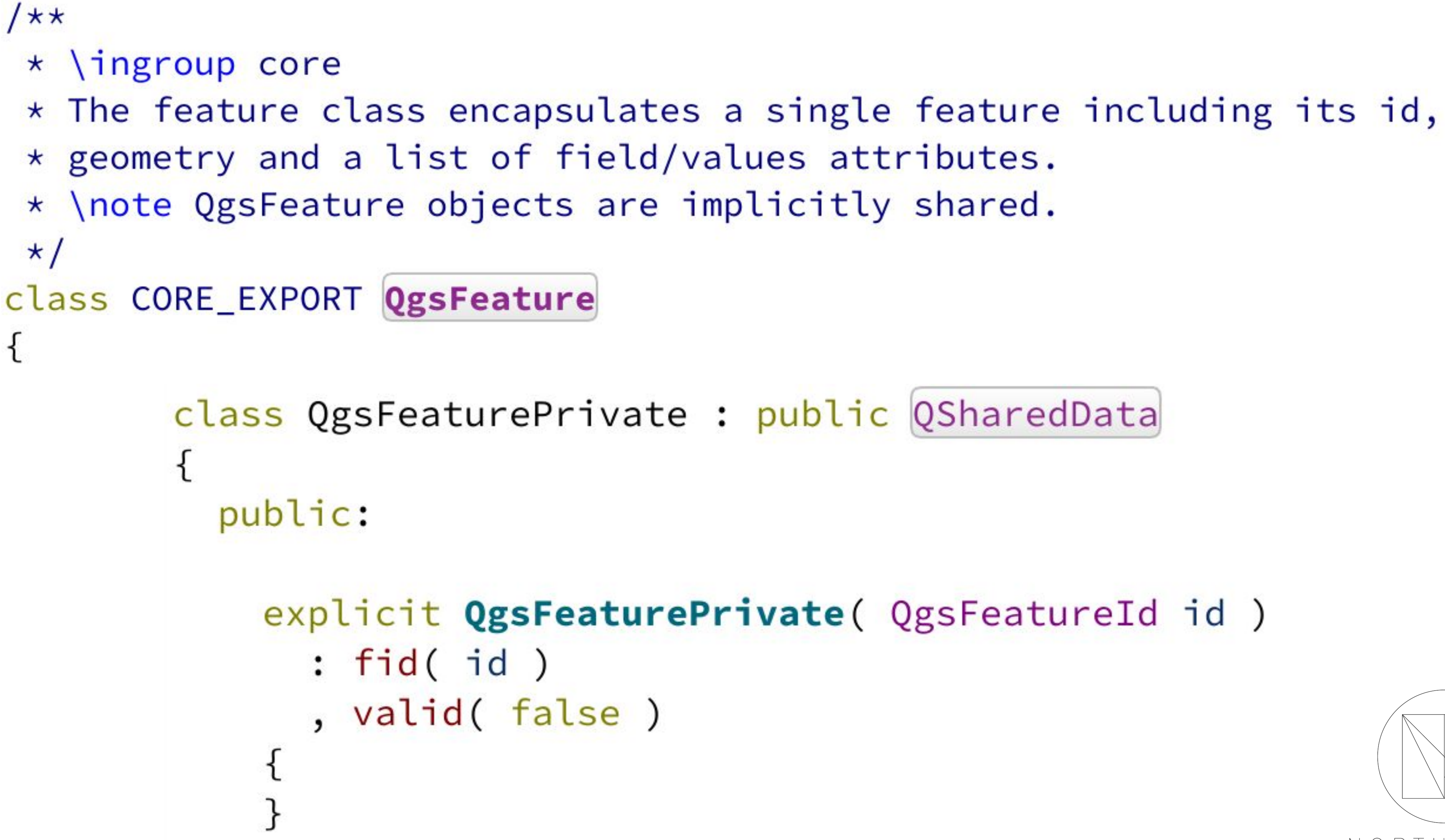
The code

Qt

```
/**
 * \ingroup core
 * \brief Point geometry type, with support for z-dimension and m-values.
 * \since QGIS 3.0, (previously QgsPointV2 since QGIS 2.10)
 */
class CORE_EXPORT QgsPoint: public QgsAbstractGeometry
{
    Q_GADGET

    Q_PROPERTY( double x READ x WRITE setX )
    Q_PROPERTY( double y READ y WRITE setY )
    Q_PROPERTY( double z READ z WRITE setZ )
    Q_PROPERTY( double m READ m WRITE setM )

  public:
```



```
/**
 * \class QgsField
  * \ingroup core
  * Encapsulate a field in an attribute table or data source.
  * QgsField stores metadata about an attribute field, including name, type
  * length, and if applicable, precision.
  * \note QgsField objects are implicitly shared.
 */

class CORE_EXPORT QgsField
{
    Q_GADGET

    Q_PROPERTY( bool isNumeric READ isNumeric )
    Q_PROPERTY( bool isDateOrTime READ isDateOrTime )
    Q_PROPERTY( int length READ length WRITE setLength )
    Q_PROPERTY( int precision READ precision WRITE setPrecision )
    Q_PROPERTY( QVariant::Type type READ type WRITE setType )
    Q_PROPERTY( QString comment READ comment WRITE setComment )
    Q_PROPERTY( QString name READ name WRITE setName )
    Q_PROPERTY( QString alias READ alias WRITE setAlias )
```



```
/**
 * \ingroup core
 * The feature class encapsulates a single feature including its id,
 * geometry and a list of field/values attributes.
 * \note QgsFeature objects are implicitly shared.
 */
class CORE_EXPORT QgsFeature
{

        class QgsFeaturePrivate : public QSharedData
        {
          public:

            explicit QgsFeaturePrivate( QgsFeatureId id )
              : fid( id )
              , valid( false )
            {
            }
```

Qt was QGIS’ early advantage

1. Ease of coding

2. Cross platform

3. QPainter

Qt

Very heavy use of

Qt Core

Qt GUI

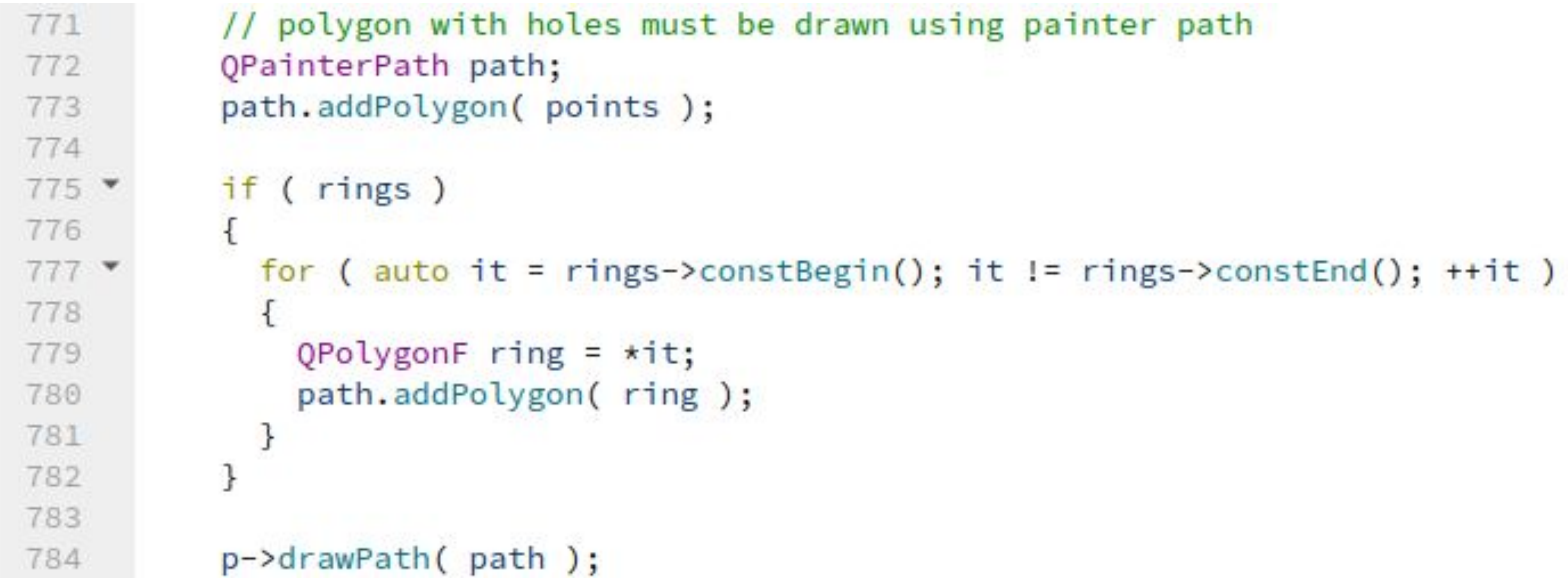
Qt Widgets

Some specifics

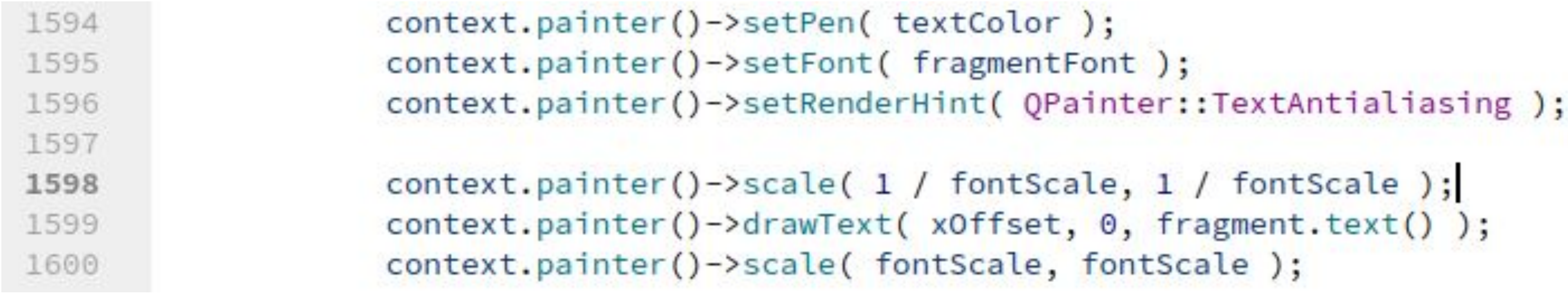
The Map Renderer

NORTH ROAD

QPainter?!

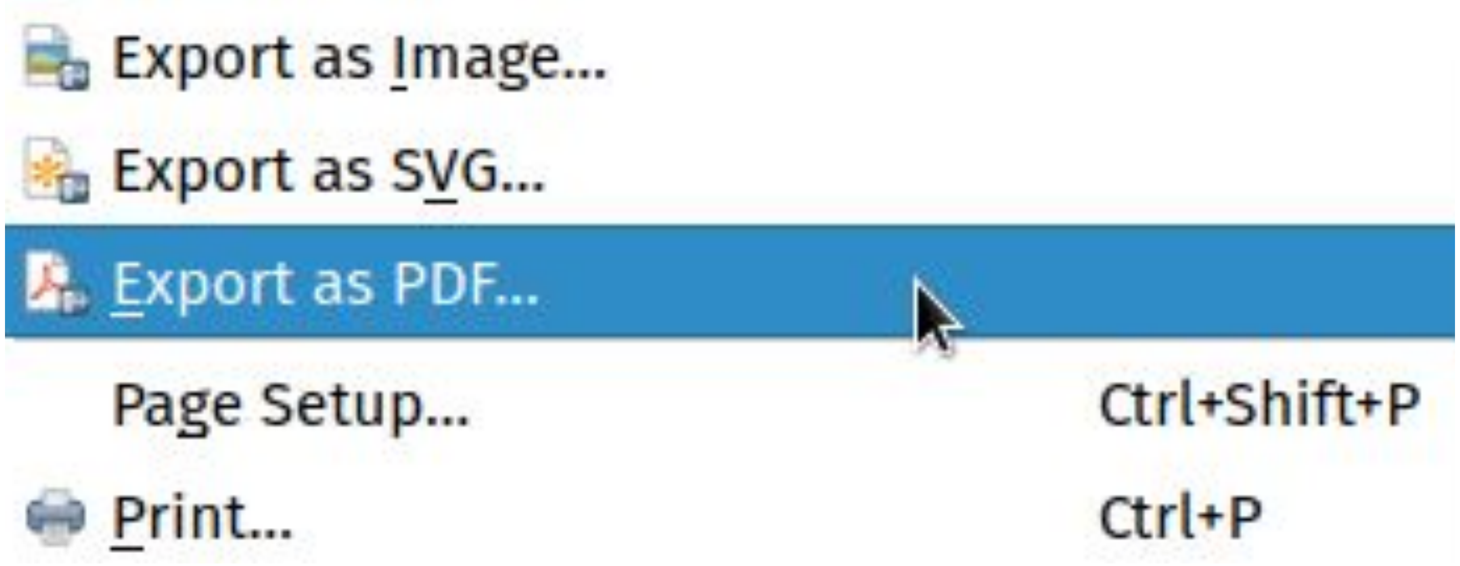


```
// polygon with holes must be drawn using painter path
QPainterPath path;
path.addPolygon( points );

if ( rings )
{
  for ( auto it = rings->constBegin(); it != rings->constEnd(); ++it )
  {
    QPolygonF ring = *it;
    path.addPolygon( ring );
  }
}

p->drawPath( path );
```



```
context.painter()->setPen( textColor );
context.painter()->setFont( fragmentFont );
context.painter()->setRenderHint( QPainter::TextAntialiasing );

context.painter()->scale( 1 / fontScale, 1 / fontScale );
context.painter()->drawText( xOffset, 0, fragment.text() );
context.painter()->scale( fontScale, fontScale );
```

heaps of ready to use drawing tools
easy export to PDF, images, SVG, printers...

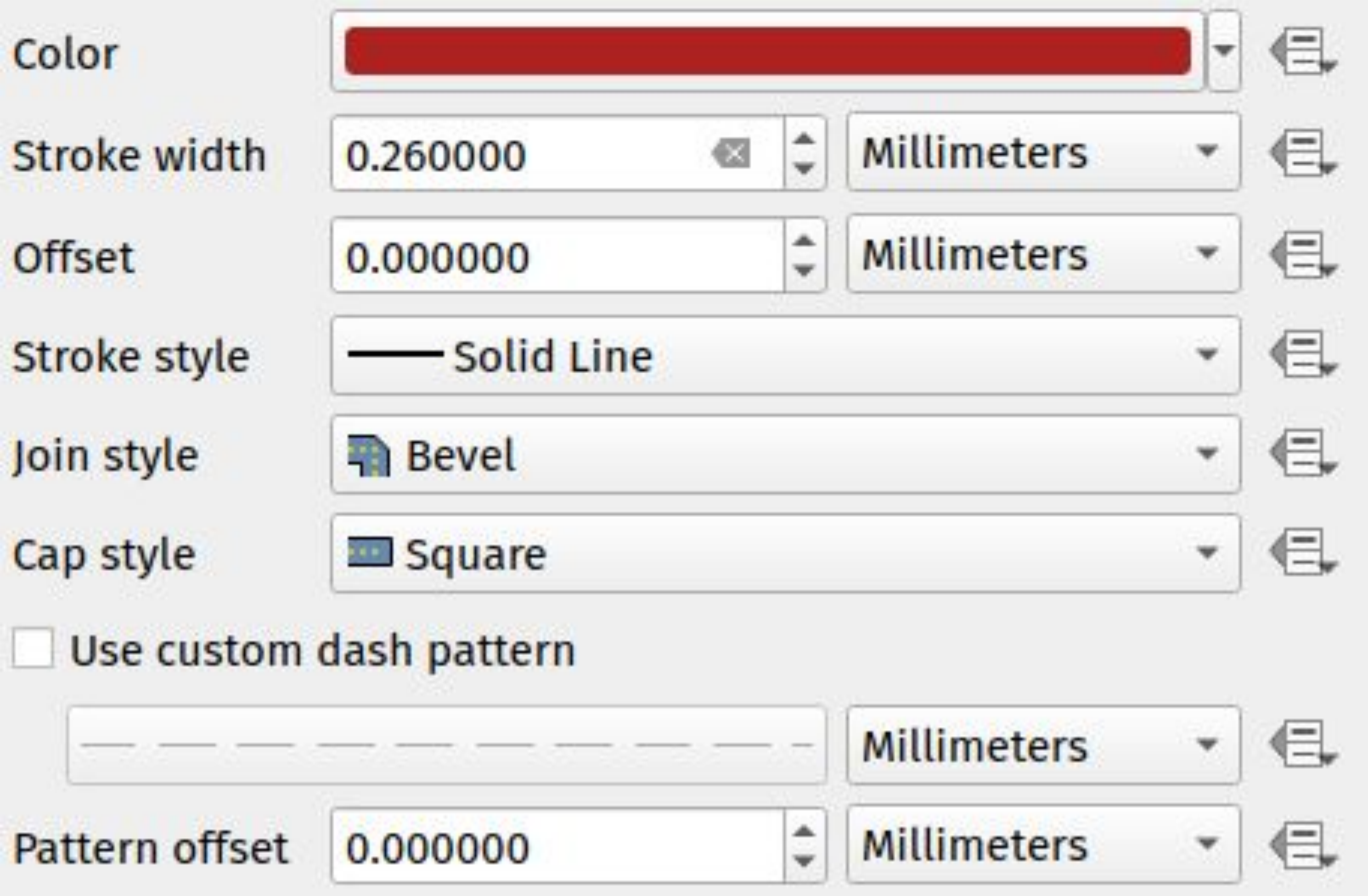

Export as Image...
Export as SVG...
Export as PDF...
Page Setup... Ctrl+Shift+P
Print... Ctrl+P

very high quality rendering (vector based)

NORTH ROAD

Ai

INKSCAPE



Color 🟥

Stroke width 0.260000 Millimeters

Offset 0.000000 Millimeters

Stroke style Solid Line

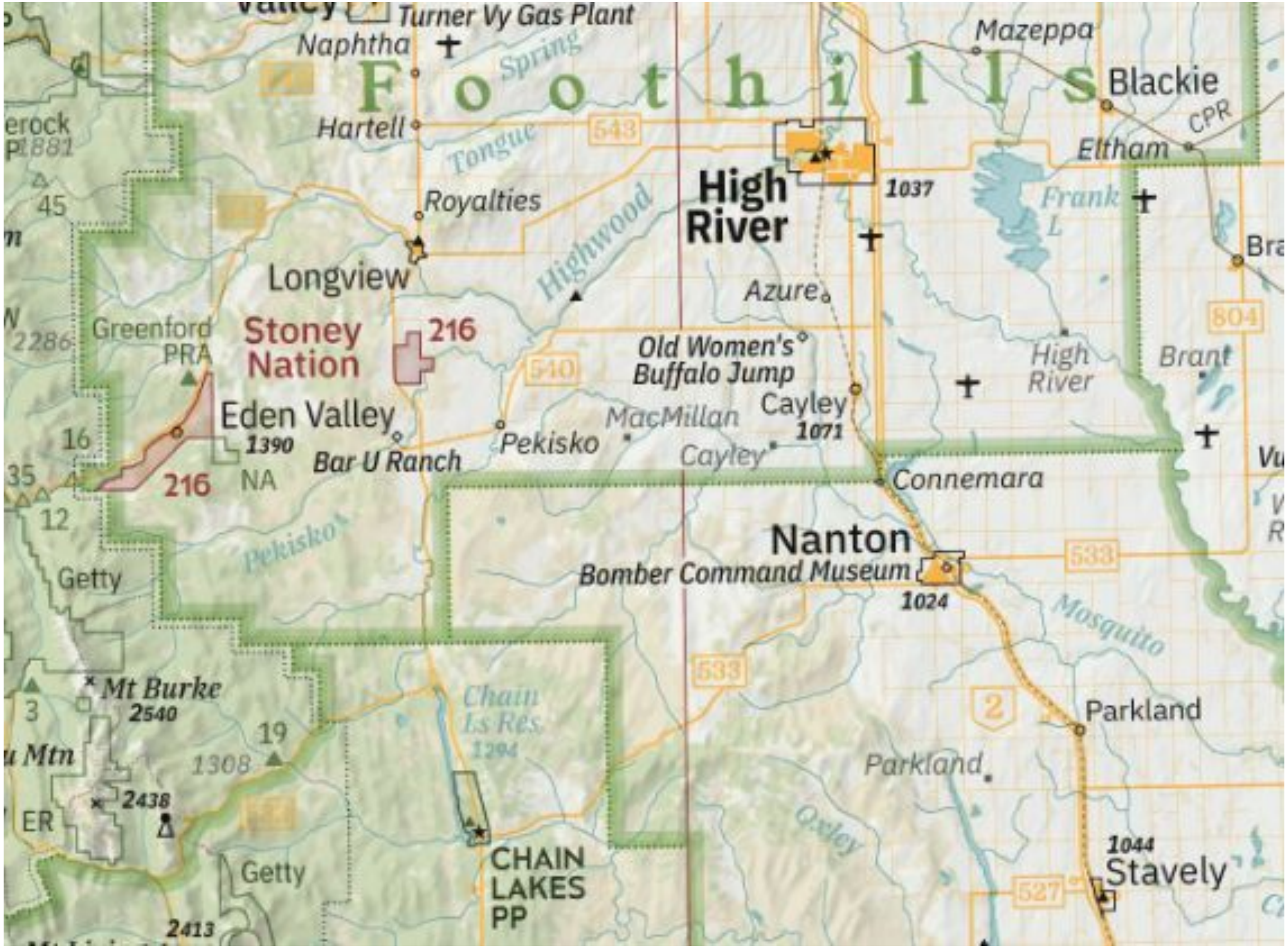
Join style Bevel

Cap style Square

□ Use custom dash pattern

Millimeters

Pattern offset 0.000000 Millimeters



Turner Vy Gas Plant
Naphtha
Foothills
Spring
Mazeppa
Blackie
CPR
Eltham
Hartell
543
Tongue
High River
1037
Frank L
Royalties
Highwood
Longview
Azure
Bra
804
Greenford PRA
Stoney Nation
216
Old Women's Buffalo Jump
High River
Brant
540
Cayley
1071
Eden Valley
1390
Pekisko
MacMillan
Cayley
Bar U Ranch
216
NA
Connemara
Nanton
Bomber Command Museum
1024
533
Mosquito
Pekisko
Getty
533
Mt Burke
2540
2
Parkland
Chain Ls Res.
1294
19
1308
Parkland
2438
Oxley
ER
Getty
CHAIN LAKES PP
1044
Stavely
527
2413
45
2286
16
35
12
3
Mtn
erock
1881

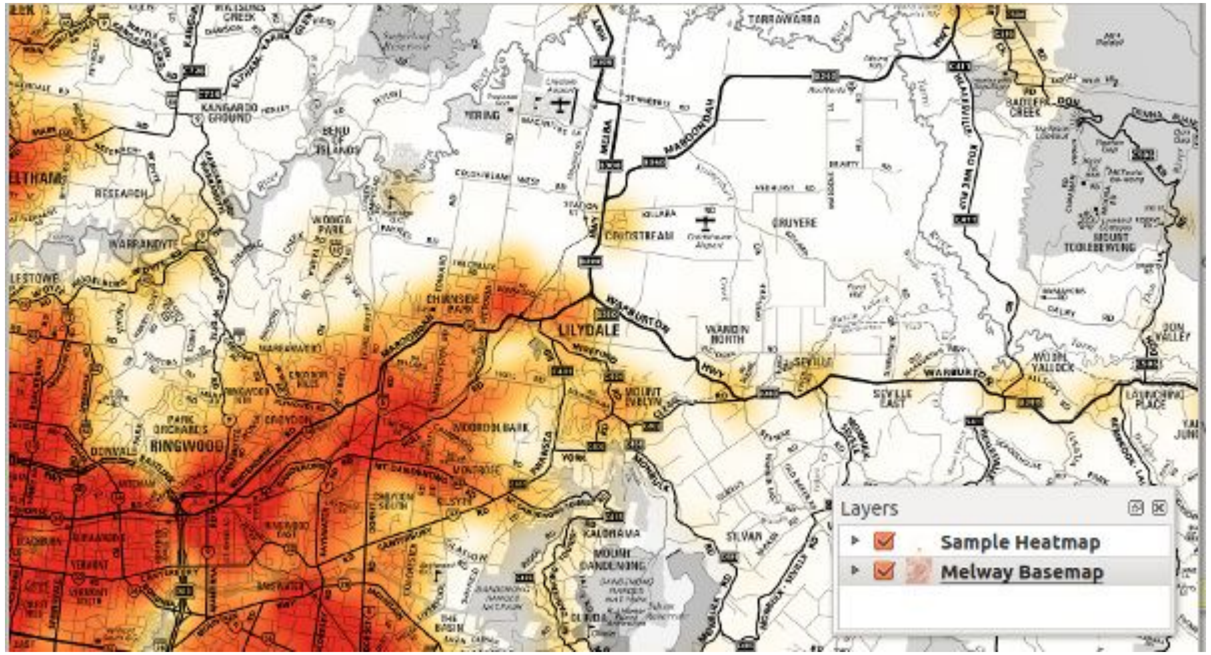
24-MARCH-2013

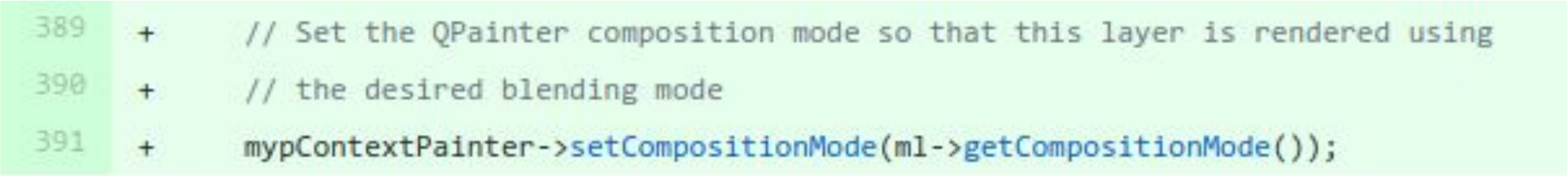
20 COMMENTS

QGIS

COMING SOON IN QGIS 2.0 – BLEND MODES FOR LAYERS

I've just pushed my first major contribution to QGIS — the ability to set the **compositing mode for a layer**. Compositing is a technique widely used by cartographers and graphic artists to fine tune how

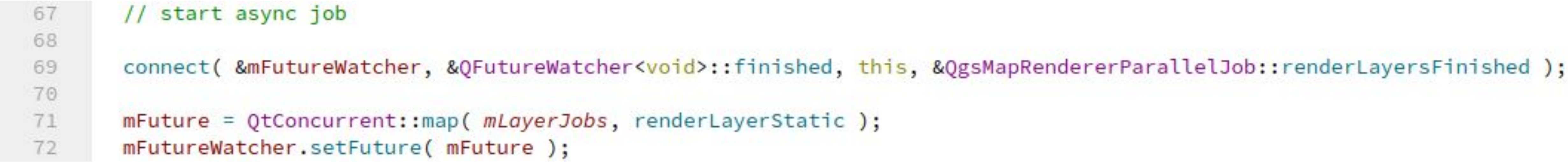




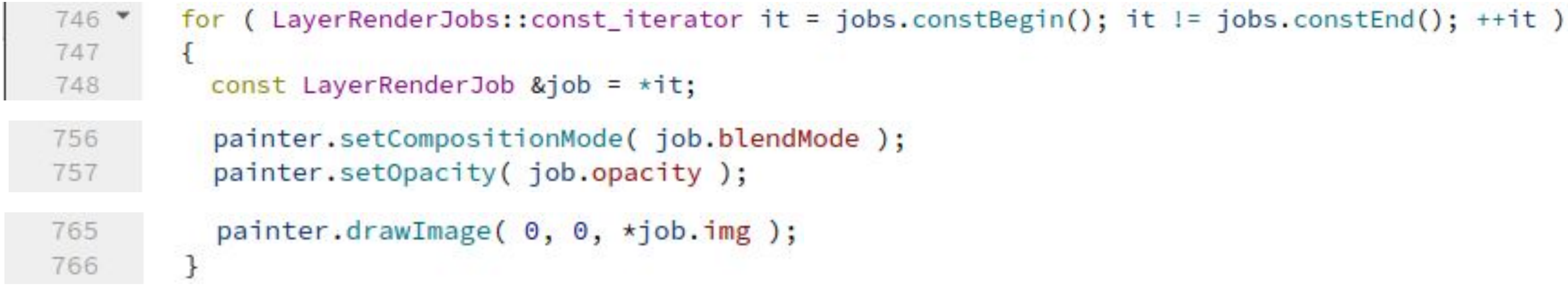
```
+     // Set the QPainter composition mode so that this layer is rendered using
+     // the desired blending mode
+     mypContextPainter->setCompositionMode(ml->getCompositionMode());
```

“multi-threaded renderer”

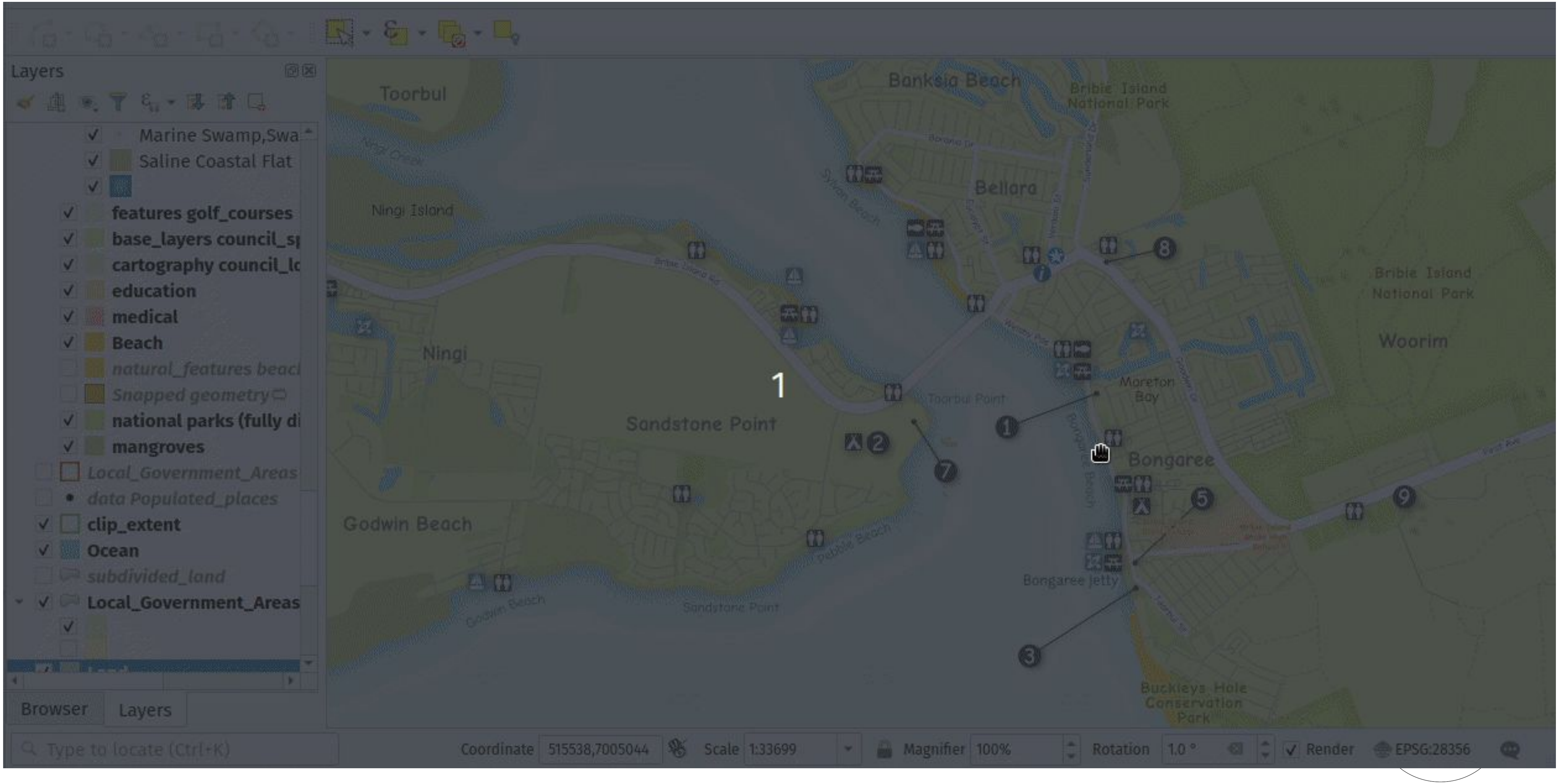
QtConcurrent



```
// start async job

connect( &mFutureWatcher, &QFutureWatcher<void>::finished, this, &QgsMapRendererParallelJob::renderLayersFinished );

mFuture = QtConcurrent::map( mLayerJobs, renderLayerStatic );
mFutureWatcher.setFuture( mFuture );
```



```
for ( LayerRenderJobs::const_iterator it = jobs.constBegin(); it != jobs.constEnd(); ++it )
{
  const LayerRenderJob &job = *it;

  painter.setCompositionMode( job.blendMode );
  painter.setOpacity( job.opacity );

  painter.drawImage( 0, 0, *job.img );
}
```



Layers

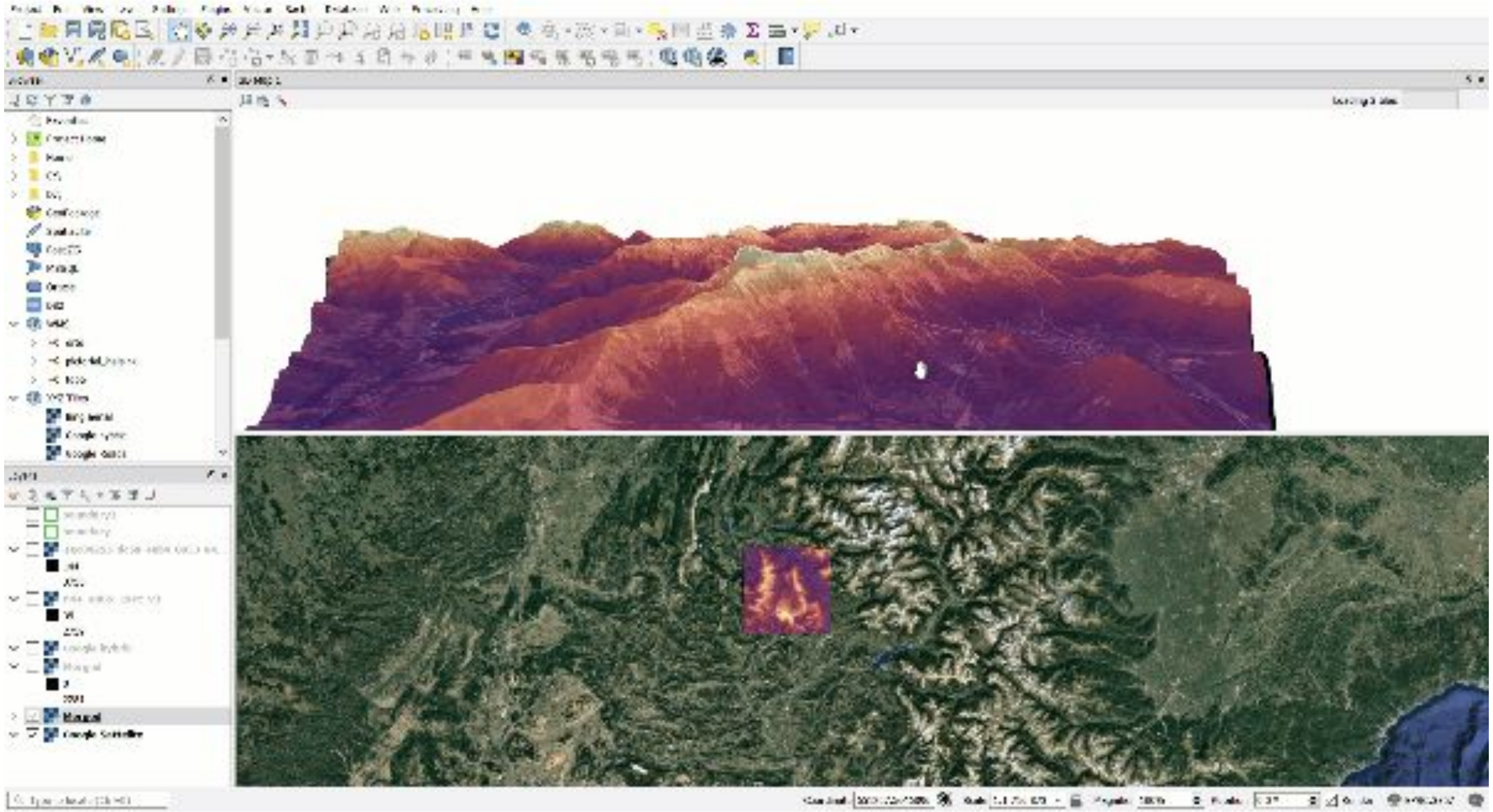
- ✓ Marine Swamp,Swa
- ✓ Saline Coastal Flat
- ✓
- ✓ features golf_courses
- ✓ base_layers council_sp
- ✓ cartography council_lc
- ✓ education
- ✓ medical
- ✓ Beach
- □ *natural_features beacl*
- □ *Snapped geometry*
- ✓ national parks (fully di
- ✓ mangroves
- □ *Local_Government_Areas*
- □ *data Populated_places*
- ✓ clip_extent
- ✓ Ocean
- □ *subdivided_land*
- ✓ Local_Government_Areas
- ✓

Browser | Layers

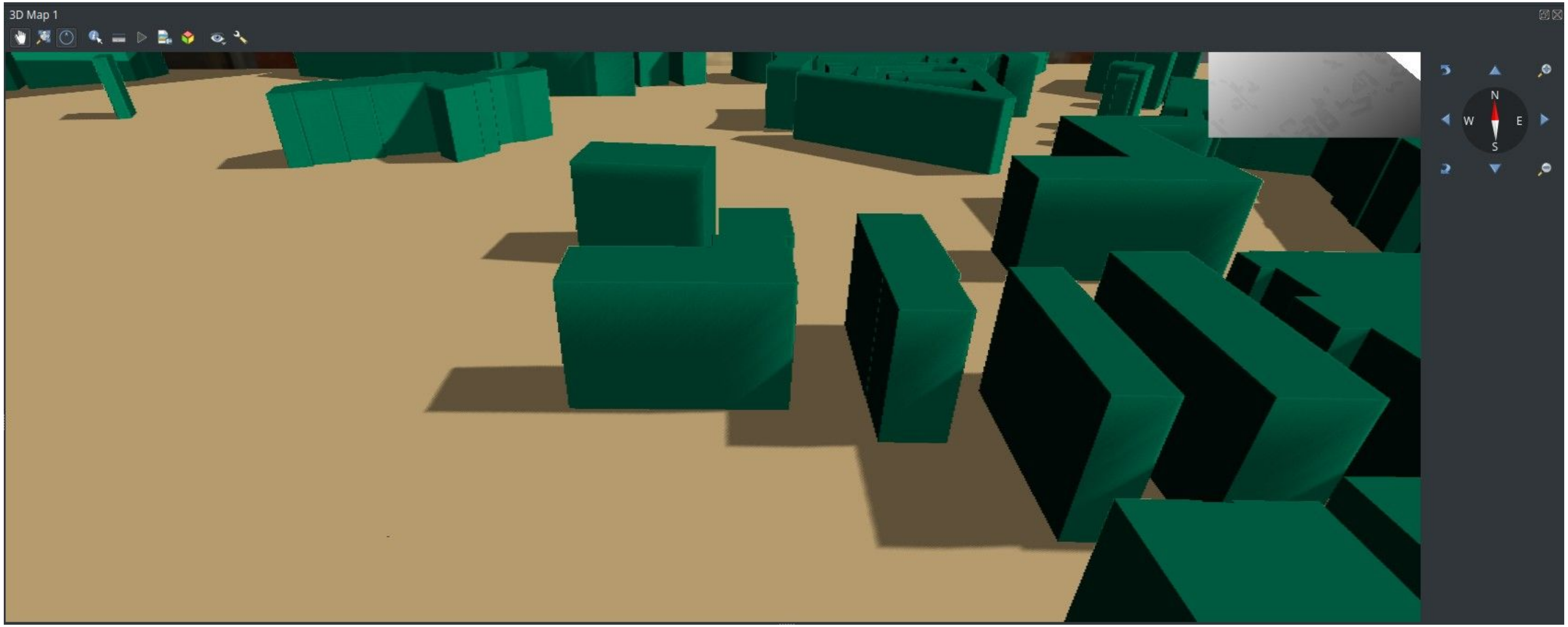
Type to locate (Ctrl+K)

Coordinate 515538,7005044 | Scale 1:33699 | Magnifier 100% | Rotation 1.0 ° | ✓ Render | EPSG:28356

Qt3D



NORTH ROAD



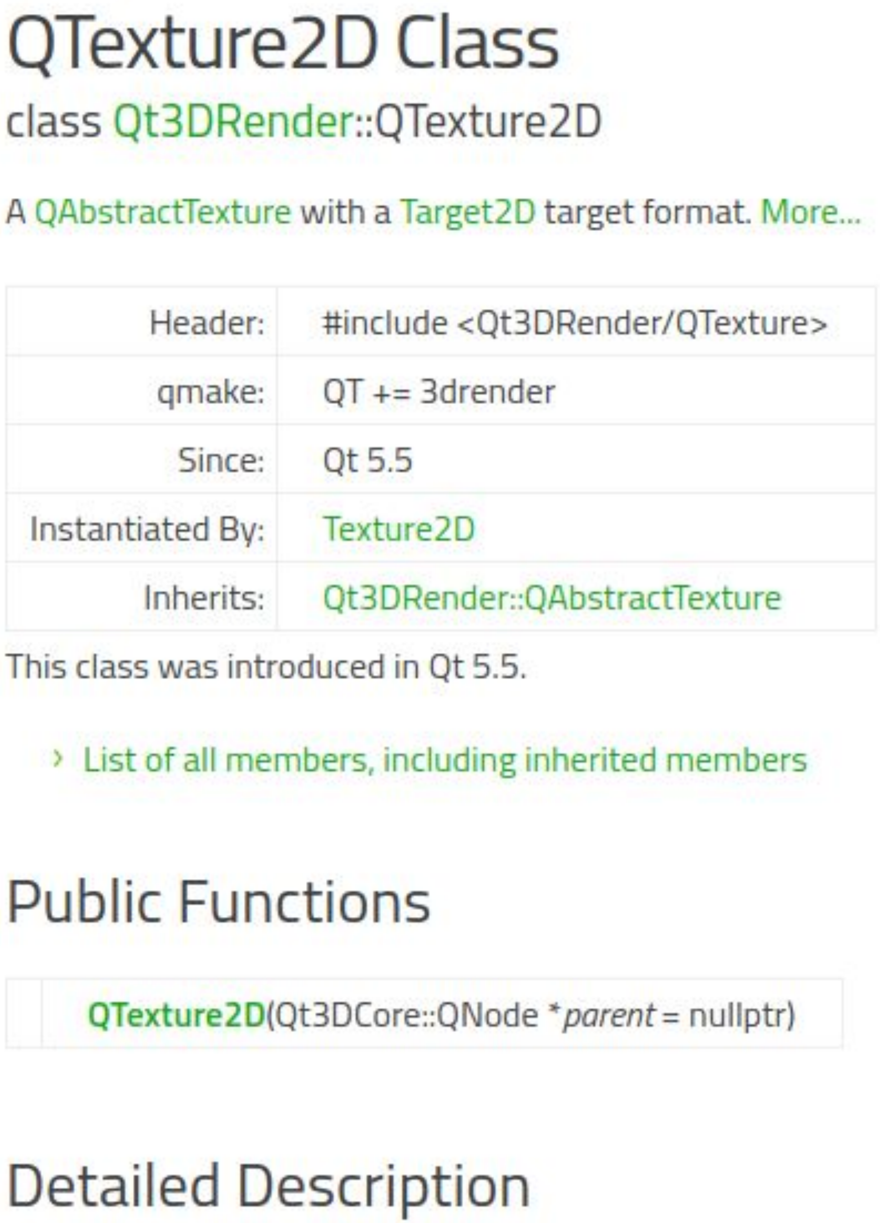
NORTH ROAD

Qt3D is great, but...

2 pain points...

1

NORTH ROAD



QTexture2D Class

class Qt3DRender::QTexture2D

A QAbstractTexture with a Target2D target format. More...

Header:	#include <Qt3DRender/QTexture>
qmake:	QT += 3drender
Since:	Qt 5.5
Instantiated By:	Texture2D
Inherits:	Qt3DRender::QAbstractTexture

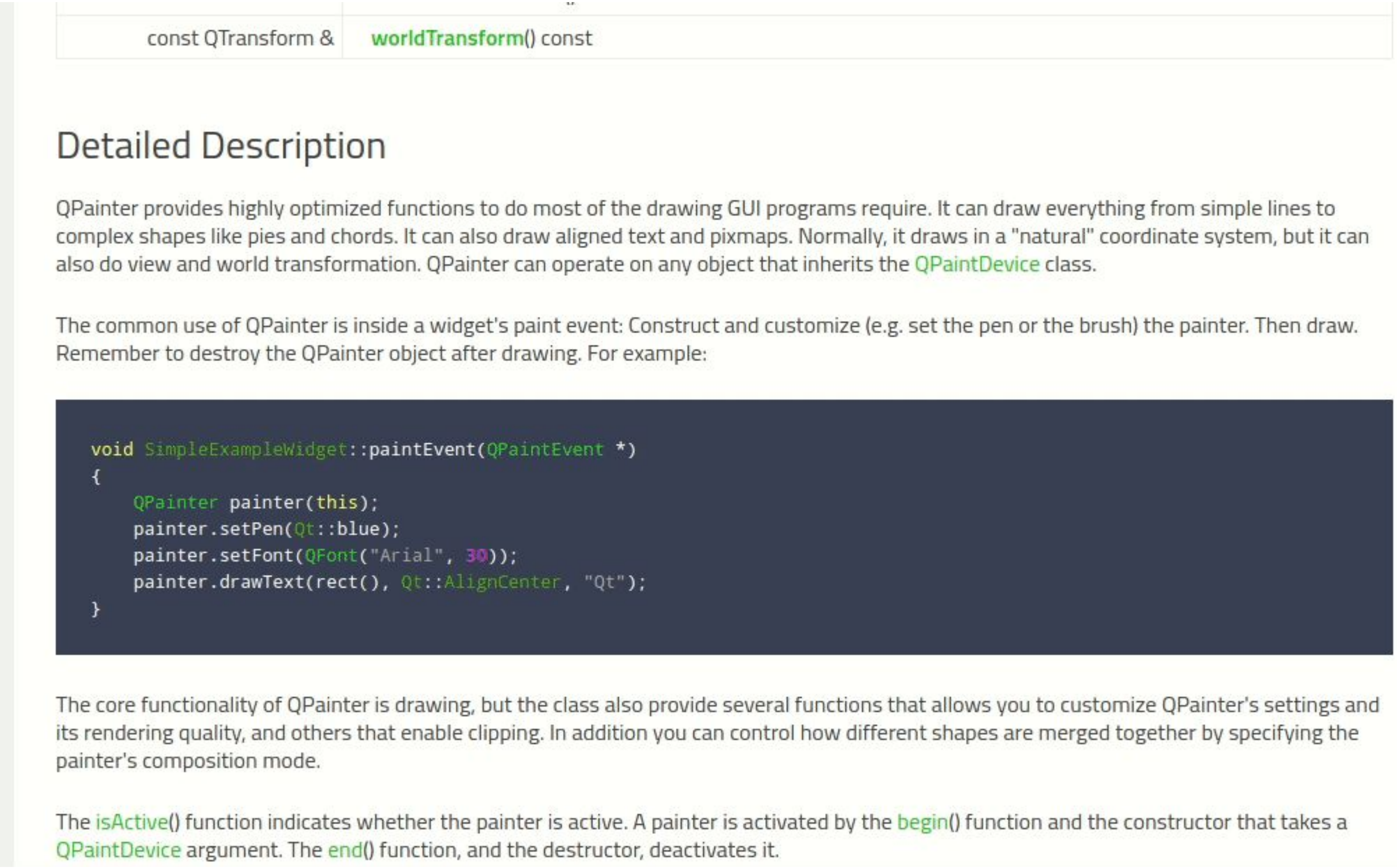
This class was introduced in Qt 5.5.

- List of all members, including inherited members

Public Functions

	QTexture2D(Qt3DCore::QNode **parent* = nullptr)

Detailed Description



const QTransform &	worldTransform() const

Detailed Description

QPainter provides highly optimized functions to do most of the drawing GUI programs require. It can draw everything from simple lines to complex shapes like pies and chords. It can also draw aligned text and pixmaps. Normally, it draws in a "natural" coordinate system, but it can also do view and world transformation. QPainter can operate on any object that inherits the QPaintDevice class.

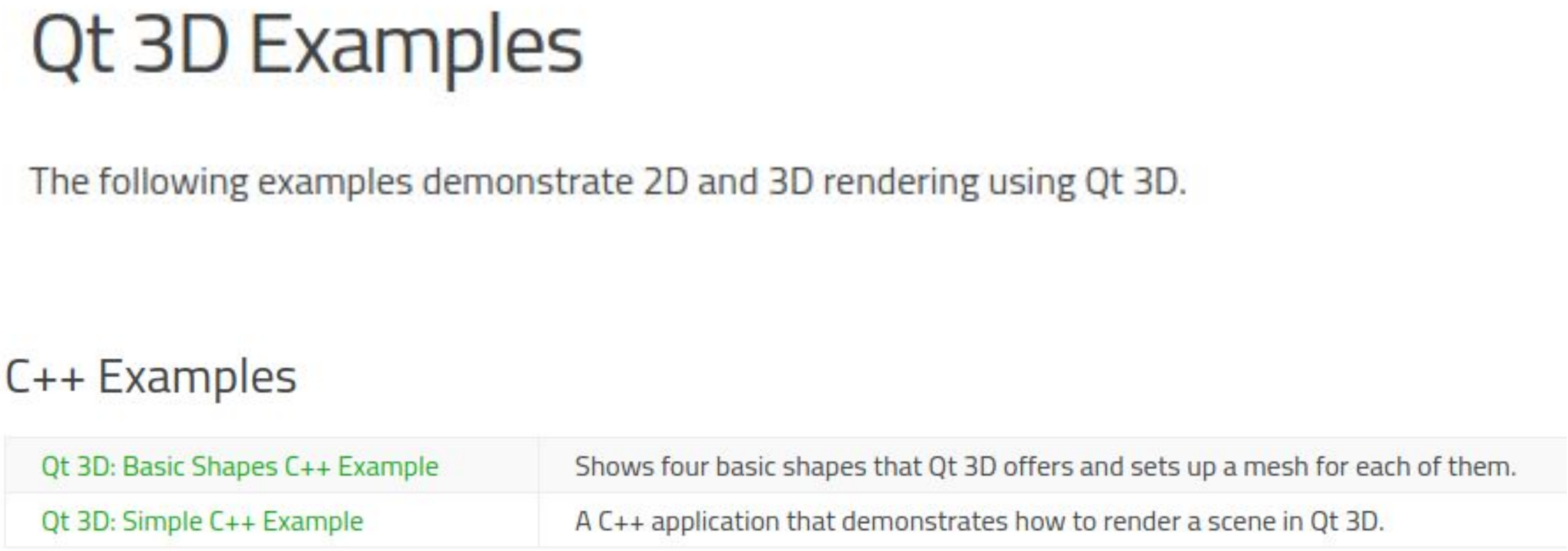
The common use of QPainter is inside a widget's paint event: Construct and customize (e.g. set the pen or the brush) the painter. Then draw. Remember to destroy the QPainter object after drawing. For example:

```
void SimpleExampleWidget::paintEvent(QPaintEvent *)
{
    QPainter painter(this);
    painter.setPen(Qt::blue);
    painter.setFont(QFont("Arial", 30));
    painter.drawText(rect(), Qt::AlignCenter, "Qt");
}
```

The core functionality of QPainter is drawing, but the class also provide several functions that allows you to customize QPainter's settings and its rendering quality, and others that enable clipping. In addition you can control how different shapes are merged together by specifying the painter's composition mode.

The isActive() function indicates whether the painter is active. A painter is activated by the begin() function and the constructor that takes a QPaintDevice argument. The end() function, and the destructor, deactivates it.

2

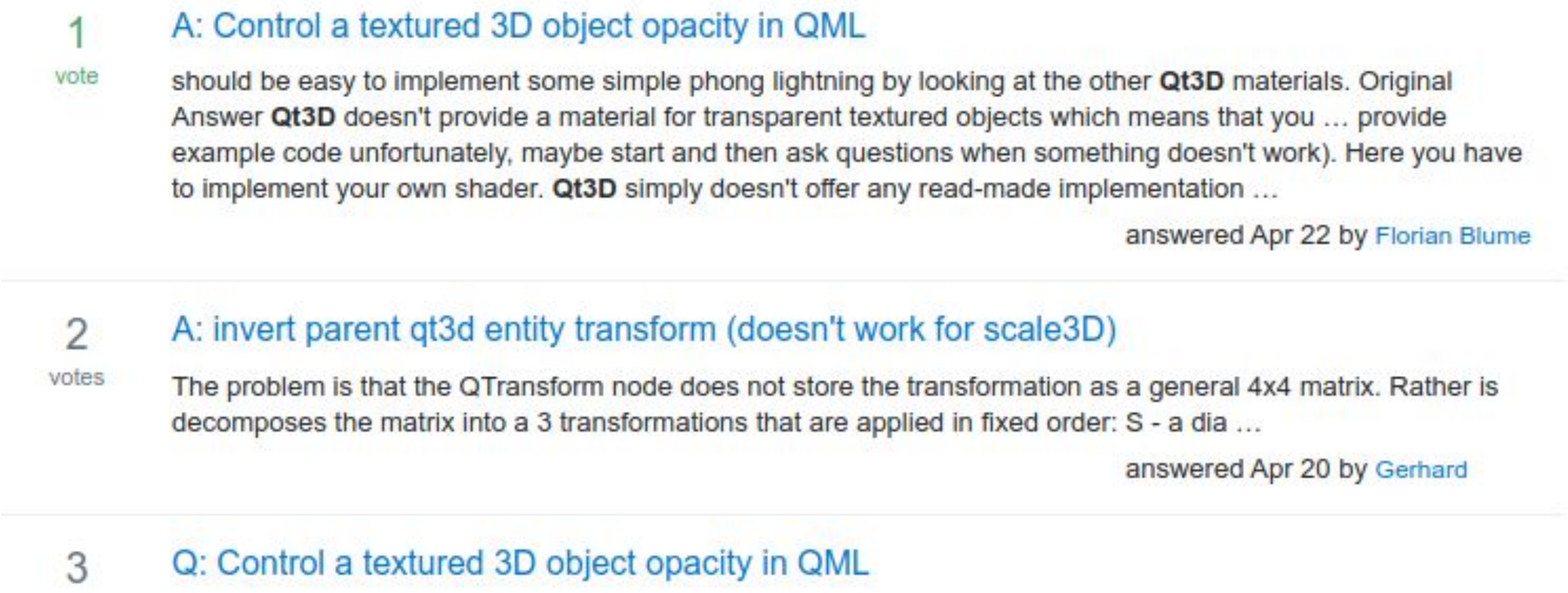


Qt 3D Examples

The following examples demonstrate 2D and 3D rendering using Qt 3D.

C++ Examples

Qt 3D: Basic Shapes C++ Example	Shows four basic shapes that Qt 3D offers and sets up a mesh for each of them.
Qt 3D: Simple C++ Example	A C++ application that demonstrates how to render a scene in Qt 3D.



1
vote

A: Control a textured 3D object opacity in QML

should be easy to implement some simple phong lightning by looking at the other **Qt3D** materials. Original Answer **Qt3D** doesn't provide a material for transparent textured objects which means that you … provide example code unfortunately, maybe start and then ask questions when something doesn't work). Here you have to implement your own shader. **Qt3D** simply doesn't offer any read-made implementation …

answered Apr 22 by Florian Blume

2
votes

A: invert parent qt3d entity transform (doesn't work for scale3D)

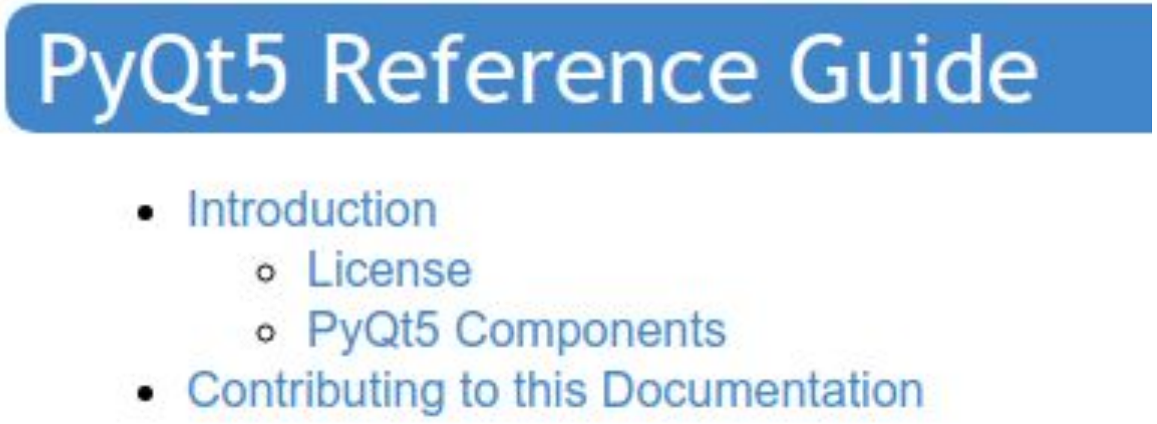
The problem is that the QTransform node does not store the transformation as a general 4x4 matrix. Rather is decomposes the matrix into a 3 transformations that are applied in fixed order: S - a dia …

answered Apr 20 by Gerhard

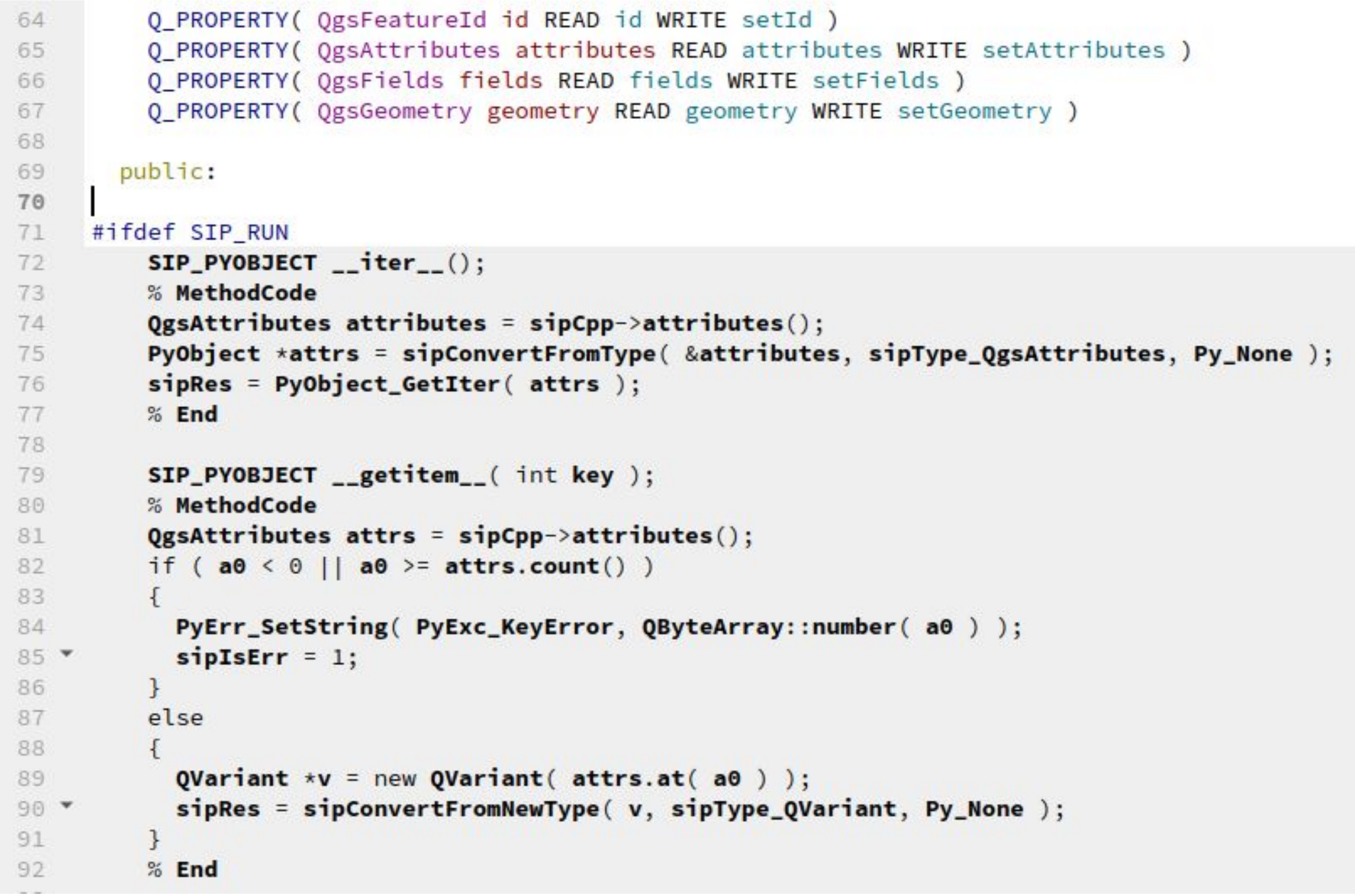
3

Q: Control a textured 3D object opacity in QML

python™

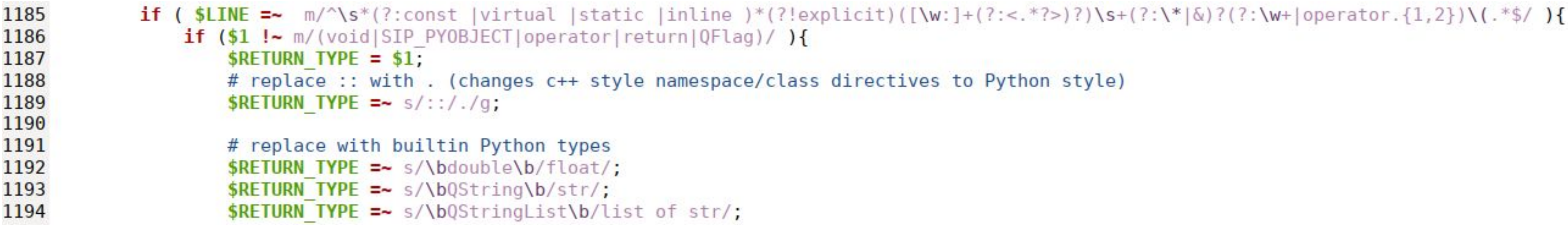


PyQt5 Reference Guide



```
    Q_PROPERTY( QgsFeatureId id READ id WRITE setId )
    Q_PROPERTY( QgsAttributes attributes READ attributes WRITE setAttributes )
    Q_PROPERTY( QgsFields fields READ fields WRITE setFields )
    Q_PROPERTY( QgsGeometry geometry READ geometry WRITE setGeometry )

  public:

#ifdef SIP_RUN
    SIP_PYOBJECT __iter__();
    % MethodCode
    QgsAttributes attributes = sipCpp->attributes();
    PyObject *attrs = sipConvertFromType( &attributes, sipType_QgsAttributes, Py_None );
    sipRes = PyObject_GetIter( attrs );
    % End

    SIP_PYOBJECT __getitem__( int key );
    % MethodCode
    QgsAttributes attrs = sipCpp->attributes();
    if ( a0 < 0 || a0 >= attrs.count() )
    {
      PyErr_SetString( PyExc_KeyError, QByteArray::number( a0 ) );
      sipIsErr = 1;
    }
    else
    {
      QVariant *v = new QVariant( attrs.at( a0 ) );
      sipRes = sipConvertFromNewType( v, sipType_QVariant, Py_None );
    }
    % End
```

NORTH ROAD



```
        if ( $LINE =~  m/^\s*(?:const |virtual |static |inline )*(?!explicit)([\w:]+(?:<.*?>)?)\s+(?:\*|&)?(?:\w+|operator.{1,2})\(.*$/ ){
            if ($1 !~ m/(void|SIP_PYOBJECT|operator|return|QFlag)/ ){
                $RETURN_TYPE = $1;
                # replace :: with . (changes c++ style namespace/class directives to Python style)
                $RETURN_TYPE =~ s/::/./g;

                # replace with builtin Python types
                $RETURN_TYPE =~ s/\bdouble\b/float/;
                $RETURN_TYPE =~ s/\bQString\b/str/;
                $RETURN_TYPE =~ s/\bQStringList\b/list of str/;
```

Qt for Python!!

QGIS and upstream

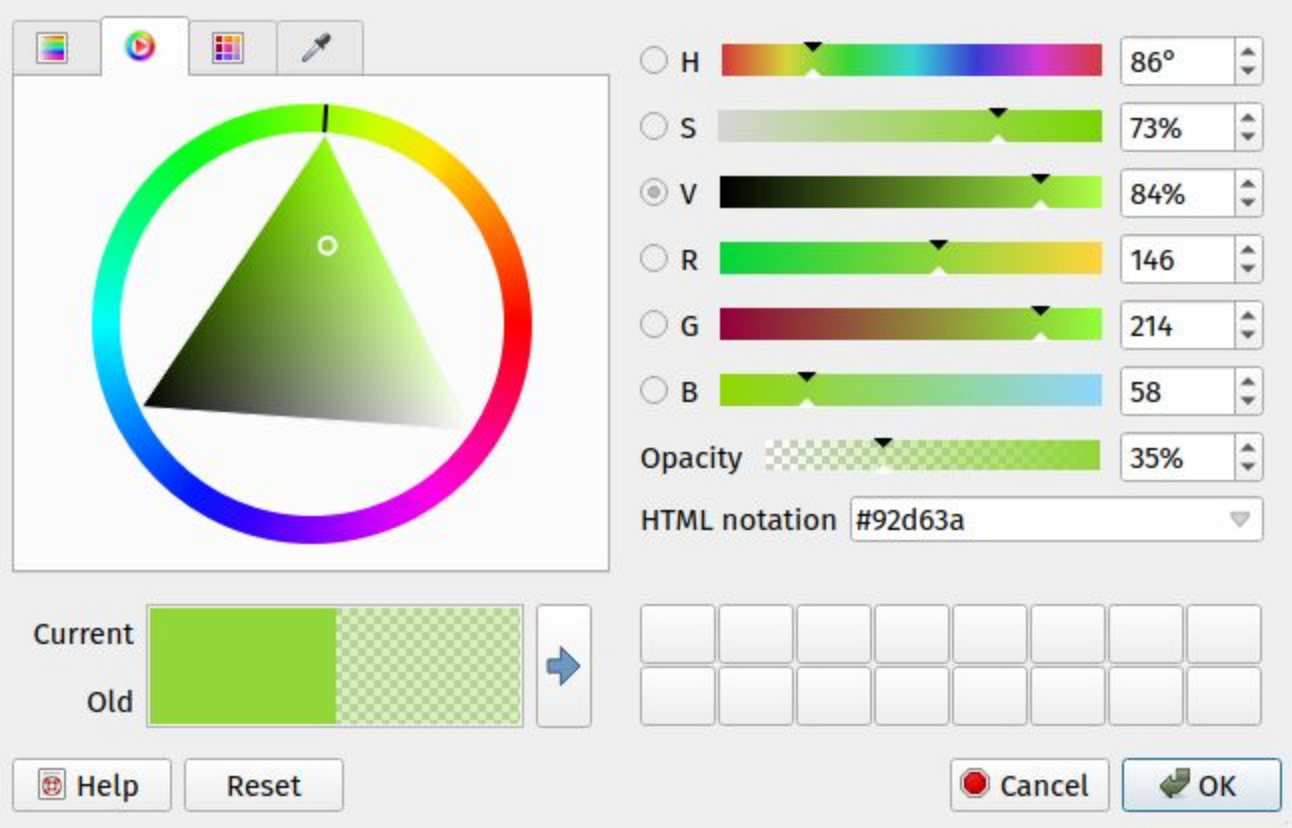
QGIS ~~and upstream~~

NORTH ROAD

1. Licensing issues

2. Packaging and distribution

3. Ambiguity vs effort



○ H 86°

○ S 73%

● V 84%

○ R 146

○ G 214

○ B 58

Opacity 35%

HTML notation #92d63a

NORTH ROAD

Questions?

@nyalldawson
@northroadgeo

https://qgis.org