GeoTIFF-3d

1.1

Generated by Doxygen 1.8.15

1 Hierarchical Index

1.1 Class Hierarchy

This inheritance list is sorted roughly, but not completely, alphabetically:

2 Class Index

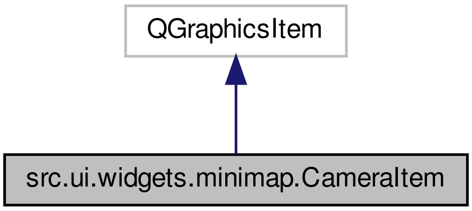
2.1 Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

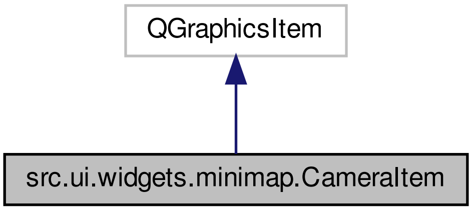
3 Class Documentation

3.1 src.ui.widgets.minimap.CameraItem Class Reference

Inheritance diagram for src.ui.widgets.minimap.CameraItem:



Collaboration diagram for src.ui.widgets.minimap.CameraItem:



Public Member Functions

- def **__init__** (self, processor, QVector3D pos, QVector3D rot, parent=None)
- def **updateCameraInfo** (self, pos=None, rot=None)
- def **paint** (self, QPainter painter, option, widget=None)
- def **boundingRect** (self)

Public Attributes

- **processor**
- **pos**
- **rot**
- **rect**

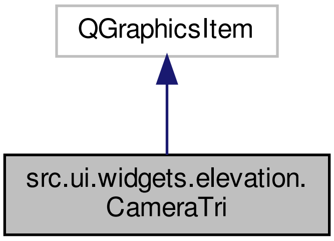
3.1.1 Detailed Description

```
An item to show a camera and it's angle
```

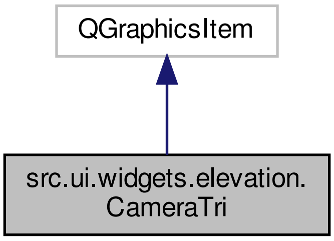
Definition at line 18 of file minimap.py.

3.2 src.ui.widgets.elevation.CameraTri Class Reference

Inheritance diagram for src.ui.widgets.elevation.CameraTri:



Collaboration diagram for src.ui.widgets.elevation.CameraTri:



Public Member Functions

- def **__init__** (self, start, end, pos, parent=None)
- def **updatePos** (self, pos)
- def **getPoint** (self)
- def **paint** (self, QPainter painter, option, widget=None)
- def **boundingRect** (self)

Public Attributes

- **end**
- **pos**
- **line**

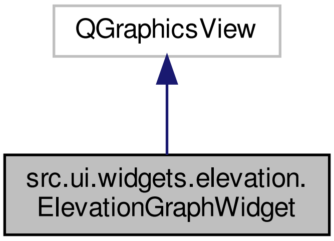
3.2.1 Detailed Description

```
A triangle showing a level of the camera
```

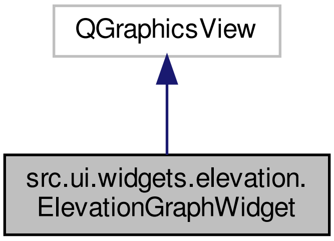
Definition at line 59 of file elevation.py.

3.3 src.ui.widgets.elevation.ElevationGraphWidget Class Reference

Inheritance diagram for src.ui.widgets.elevation.ElevationGraphWidget:



Collaboration diagram for src.ui.widgets.elevation.ElevationGraphWidget:



Public Member Functions

- def **__init__** (self, start, end, pos, width=240, height=240, levels=None, parent=None)
- def **initContent** (self)
- def updatePos (self, pos)

Public Attributes

- **end**
- **pos**
- **levels**
- **square**
- **tri**

3.3.1 Detailed Description

```
A widget to show an elevation of camera relative to the
given altitude
```

Definition at line 105 of file elevation.py.

3.3.2 Member Function Documentation

3.3.2.1 updatePos()

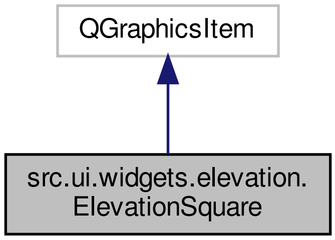
```
def src.ui.widgets.elevation.ElevationGraphWidget.updatePos (
              self,
              pos )
```

```
Update a position of the camera
```

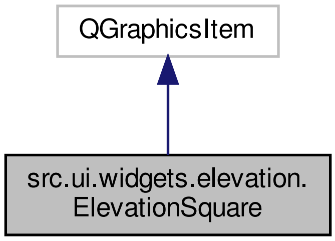
Definition at line 139 of file elevation.py.

3.4 src.ui.widgets.elevation.ElevationSquare Class Reference

Inheritance diagram for src.ui.widgets.elevation.ElevationSquare:



Collaboration diagram for src.ui.widgets.elevation.ElevationSquare:



Public Member Functions

- def **__init__** (self, start, end, levels, parent=None)
- def **paint** (self, QPainter painter, option, widget=None)
- def **boundingRect** (self)

Public Attributes

- **end**
- **levels**
- **rect**

3.4.1 Detailed Description

```
A red-green filled square with altitudes
```

Definition at line 22 of file elevation.py.

3.5 src.api.geotiff_processor.GeoTIFFProcessor Class Reference

Classes

- class PreprocessThread

Public Member Functions

- def __init__ (self, data=None)
- def data_loaded (self)
- def data_plotted (self)
- def **min_lon** (self)
- def **max_lon** (self)
- def **min_lat** (self)
- def **max_lat** (self)
- def get_borders (self, data_=None)
- def get_centered_borders (self, data, center)
- def get_dimensions (self, data=None)
- def get_value_limits (self, data=None)
- def get_real_scaling (self, data=None)
- def max_rad (self, lat, lon)
- def **center** (self)
- def **get_center** (self, data=None)
- def points_estimate (self, r=None, coef=1)
- def df_size_estimate (self, ∗args, ∗∗kwargs)
- def get_value (self, x, y, data=None)
- def normalizeLat (self, lat)
- def normalizeLon (self, lon)
- def normalizeValue (self, value)
- def denormalizeValue (self, value)
- def normalizePoint (self, point)
- def normalizePoints (self, polygon)
- def modify_data (self, lat=None, lon=None, r=None, coef=1)
- def get_contour (self, data=None, plot=False, ∗args, ∗∗kwargs)
- def extract_to_pandas (self, ∗args, ∗∗kwargs)
- def calculate_normals (self, df, normalize=False)
- def **get_polygon** (self, row, col, max_row, max_col, df)
- def polygon_generator (self, df)
- def **open_file** (self, name)
- def **save** (self, name, ∗args, ∗∗kwargs)
- def init_canvas (self)
- def draw_preview (self, data=None, ∗args, ∗∗kwargs)

Public Attributes

- **data**
- **borders**
- **min_val**
- **max_val**
- **ax**
- **canvas**

Private Member Functions

- def _resize_data (self, data, coef)
- def _contour_cmap (self)

Private Attributes

- **_points_num**
- **_center**

3.5.1 Detailed Description

```
Main class to manage GeoTIFF file
```

Definition at line 17 of file geotiff_processor.py.

3.5.2 Constructor & Destructor Documentation

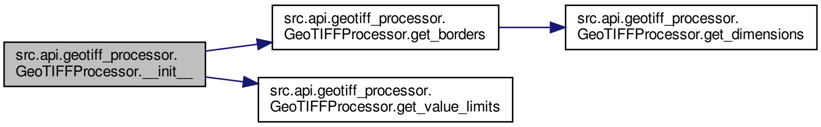
3.5.2.1 __init__()

```
def src.api.geotiff_processor.GeoTIFFProcessor.__init__ (
              self,
              data = None )
```

```
:param data: an instance of GeoRaster
```

Definition at line 90 of file geotiff_processor.py.

Here is the call graph for this function:


src.api.geotiff_processor.
GeoTIFFProcessor.__init__
src.api.geotiff_processor.
GeoTIFFProcessor.get_borders
src.api.geotiff_processor.
GeoTIFFProcessor.get_dimensions
src.api.geotiff_processor.
GeoTIFFProcessor.get_value_limits

3.5.3 Member Function Documentation

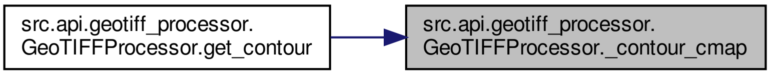
3.5.3.1 _contour_cmap()

```
def src.api.geotiff_processor.GeoTIFFProcessor._contour_cmap (
              self )  [private]
```

```
A red-green colormap for matplotlib
```

Definition at line 379 of file geotiff_processor.py.

Here is the caller graph for this function:


src.api.geotiff_processor.
GeoTIFFProcessor.get_contour
src.api.geotiff_processor.
GeoTIFFProcessor._contour_cmap

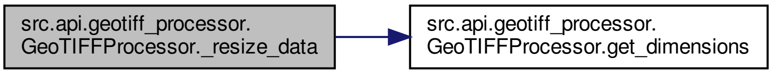
3.5.3.2 _resize_data()

```
def src.api.geotiff_processor.GeoTIFFProcessor._resize_data (
              self,
              data,
              coef )  [private]
```

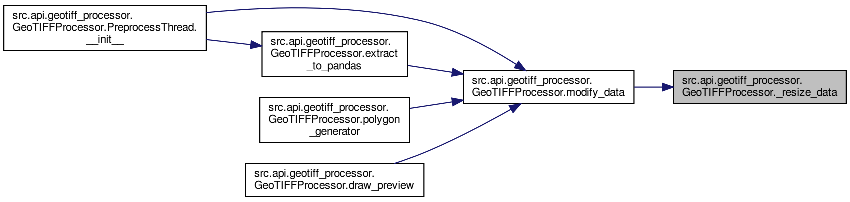
```
Fix of georasters.resize for correct management of incomplete
GeoTIFF-s

:param data: GeoRaster
:param coef: coefficient > 0 for resizing
```

Definition at line 350 of file geotiff_processor.py.

Here is the call graph for this function:


src.api.geotiff_processor.
GeoTIFFProcessor._resize_data
src.api.geotiff_processor.
GeoTIFFProcessor.get_dimensions

Here is the caller graph for this function:


src.api.geotiff_processor.
GeoTIFFProcessor.PreprocessThread.
__init__
src.api.geotiff_processor.
GeoTIFFProcessor.extract
_to_pandas
src.api.geotiff_processor.
GeoTIFFProcessor.polygon
_generator
src.api.geotiff_processor.
GeoTIFFProcessor.draw_preview
src.api.geotiff_processor.
GeoTIFFProcessor.modify_data
src.api.geotiff_processor.
GeoTIFFProcessor._resize_data

3.5.3.3 calculate_normals()

```
def src.api.geotiff_processor.GeoTIFFProcessor.calculate_normals (
              self,
              df,
              normalize = False )
```

```
Run GeoTIFFProcessor.PreprocessThread for given df

:param df: DataFrame from GeoTIFFProcessor.extract_to_pandas
```

Definition at line 488 of file geotiff_processor.py.

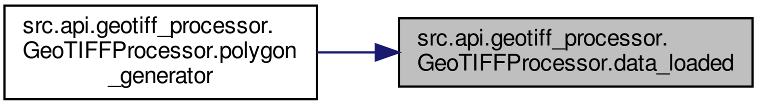
3.5.3.4 data_loaded()

```
def src.api.geotiff_processor.GeoTIFFProcessor.data_loaded (
              self )
```

```
Is data loaded
```

Definition at line 108 of file geotiff_processor.py.

Here is the caller graph for this function:


src.api.geotiff_processor.
GeoTIFFProcessor.polygon
_generator
src.api.geotiff_processor.
GeoTIFFProcessor.data_loaded

3.5.3.5 data_plotted()

```
def src.api.geotiff_processor.GeoTIFFProcessor.data_plotted (
              self )
```

```
Was data plotted
```

Definition at line 113 of file geotiff_processor.py.

3.5.3.6 denormalizeValue()

```
def src.api.geotiff_processor.GeoTIFFProcessor.denormalizeValue (
              self,
              value )
```

```
Get an actual altitude by a normalized value

:param value:
```

Definition at line 309 of file geotiff_processor.py.

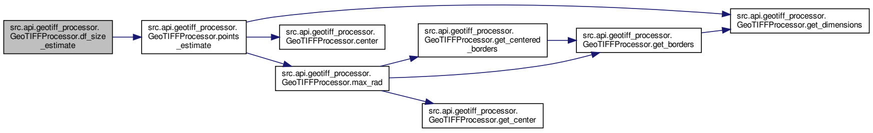
3.5.3.7 df_size_estimate()

```
def src.api.geotiff_processor.GeoTIFFProcessor.df_size_estimate (
              self,
             * args,
             ** kwargs )
```

```
Estimate size of a DataFrame with calculated normals after
an extraction

:param *args: Passed to GeoTIFFProcessor.points_estimate
:param **kwargs: Passed to GeoTIFFProcessor.points_estimate
```

Definition at line 242 of file geotiff_processor.py.

Here is the call graph for this function:


src.api.geotiff_processor.
GeoTIFFProcessor.df_size
_estimate
src.api.geotiff_processor.
GeoTIFFProcessor.points
_estimate
src.api.geotiff_processor.
GeoTIFFProcessor.center
src.api.geotiff_processor.
GeoTIFFProcessor.max_rad
src.api.geotiff_processor.
GeoTIFFProcessor.get_centered
_borders
src.api.geotiff_processor.
GeoTIFFProcessor.get_center
src.api.geotiff_processor.
GeoTIFFProcessor.get_borders
src.api.geotiff_processor.
GeoTIFFProcessor.get_dimensions

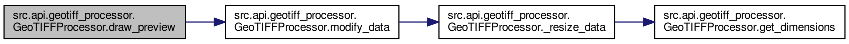
3.5.3.8 draw_preview()

```
def src.api.geotiff_processor.GeoTIFFProcessor.draw_preview (
              self,
              data = None,
             * args,
             ** kwargs )
```

```
Draw a preview of a data on GeoTIFFProcessor.ax

:param data: GeoRaster
:param *args: passed to GeoTIFFProcessor.modify_data
:param **kwargs: passed to GeoTIFFProcessor.modify_data
```

Definition at line 559 of file geotiff_processor.py.

Here is the call graph for this function:


src.api.geotiff_processor.
GeoTIFFProcessor.draw_preview
src.api.geotiff_processor.
GeoTIFFProcessor.modify_data
src.api.geotiff_processor.
GeoTIFFProcessor._resize_data
src.api.geotiff_processor.
GeoTIFFProcessor.get_dimensions

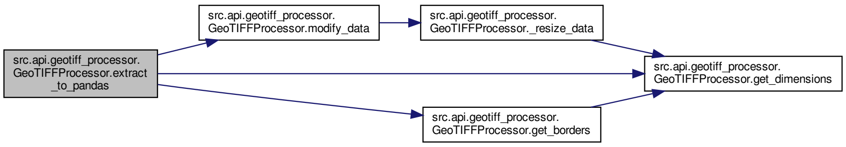
3.5.3.9 extract_to_pandas()

```
def src.api.geotiff_processor.GeoTIFFProcessor.extract_to_pandas (
              self,
             * args,
             ** kwargs )
```

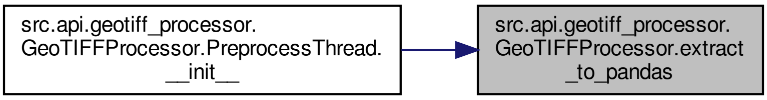
```
Extraction of a GeoRaster to pandas DataFrame with columns like
[row, col, value, x, y]

Unlike georasters.to_pandas, this method fills masked values with zeros

:param *args: passed to GeoTIFFProcessor.modify_data
:param **kwargs: GeoTIFFProcessor.modify_data
```

Definition at line 446 of file geotiff_processor.py.

Here is the call graph for this function:


src.api.geotiff_processor.
GeoTIFFProcessor.extract
_to_pandas
src.api.geotiff_processor.
GeoTIFFProcessor.modify_data
src.api.geotiff_processor.
GeoTIFFProcessor._resize_data
src.api.geotiff_processor.
GeoTIFFProcessor.get_dimensions
src.api.geotiff_processor.
GeoTIFFProcessor.get_borders

Here is the caller graph for this function:


src.api.geotiff_processor.
GeoTIFFProcessor.PreprocessThread.
__init__
src.api.geotiff_processor.
GeoTIFFProcessor.extract
_to_pandas

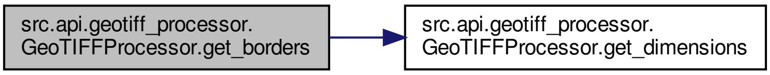
3.5.3.10 get_borders()

```
def src.api.geotiff_processor.GeoTIFFProcessor.get_borders (
              self,
              data_ = None )
```

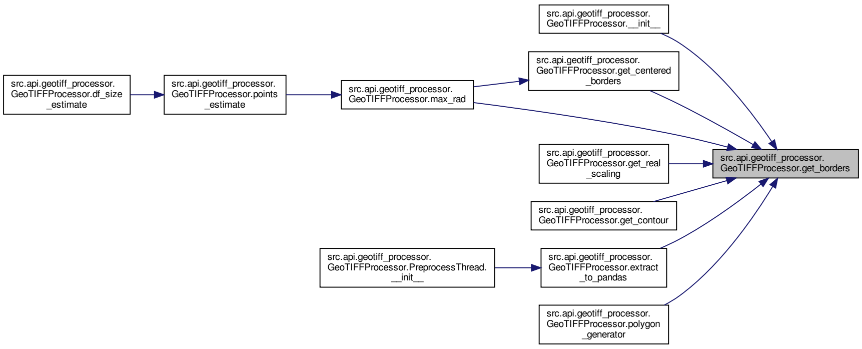
```
Calculate Georaster's borders

WARNING: This works incorrently with exracted data
:param data: GeoRaster
```

Definition at line 134 of file geotiff_processor.py.

Here is the call graph for this function:


src.api.geotiff_processor.
GeoTIFFProcessor.get_borders
src.api.geotiff_processor.
GeoTIFFProcessor.get_dimensions

Here is the caller graph for this function:


src.api.geotiff_processor.
GeoTIFFProcessor.df_size
_estimate
src.api.geotiff_processor.
GeoTIFFProcessor.points
_estimate
src.api.geotiff_processor.
GeoTIFFProcessor.max_rad
src.api.geotiff_processor.
GeoTIFFProcessor.__init__
src.api.geotiff_processor.
GeoTIFFProcessor.get_centered
_borders
src.api.geotiff_processor.
GeoTIFFProcessor.get_real
_scaling
src.api.geotiff_processor.
GeoTIFFProcessor.get_contour
src.api.geotiff_processor.
GeoTIFFProcessor.PreprocessThread.
__init__
src.api.geotiff_processor.
GeoTIFFProcessor.extract
_to_pandas
src.api.geotiff_processor.
GeoTIFFProcessor.polygon
_generator
src.api.geotiff_processor.
GeoTIFFProcessor.get_borders

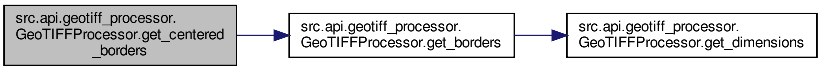
3.5.3.11 get_centered_borders()

```
def src.api.geotiff_processor.GeoTIFFProcessor.get_centered_borders (
              self,
              data,
              center )
```

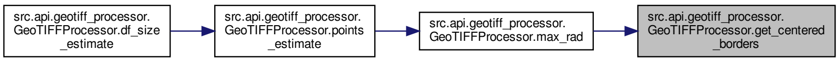
```
Calculate Georaster's borders and align them to a center.
It is a workaround for bug in GeoTIFFProcessor.get_borders

:param data: GeoRaster
:param center: (lon, lat)
```

Definition at line 149 of file geotiff_processor.py.

Here is the call graph for this function:


src.api.geotiff_processor.
GeoTIFFProcessor.get_centered
_borders
src.api.geotiff_processor.
GeoTIFFProcessor.get_borders
src.api.geotiff_processor.
GeoTIFFProcessor.get_dimensions

Here is the caller graph for this function:


src.api.geotiff_processor.
GeoTIFFProcessor.df_size
_estimate
src.api.geotiff_processor.
GeoTIFFProcessor.points
_estimate
src.api.geotiff_processor.
GeoTIFFProcessor.max_rad
src.api.geotiff_processor.
GeoTIFFProcessor.get_centered
_borders

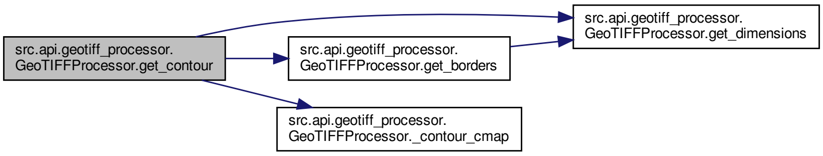
3.5.3.12 get_contour()

```
def src.api.geotiff_processor.GeoTIFFProcessor.get_contour (
              self,
              data = None,
              plot = False,
             * args,
             ** kwargs )
```

```
Calculate a list of contour lines for GeoRaster

:param data: GeoRaster
:param plot: if True, a plot on GeoRaster.ax will be performed
:param *args: passed to GeoTIFFProcessor.modify_data
:param **kwargs: passed to GeoTIFFProcessor.modify_data

:return: a list like [altitude, [(x1, y1), ...]]
```

Definition at line 388 of file geotiff_processor.py.

Here is the call graph for this function:


src.api.geotiff_processor.
GeoTIFFProcessor.get_contour
src.api.geotiff_processor.
GeoTIFFProcessor.get_borders
src.api.geotiff_processor.
GeoTIFFProcessor.get_dimensions
src.api.geotiff_processor.
GeoTIFFProcessor._contour_cmap

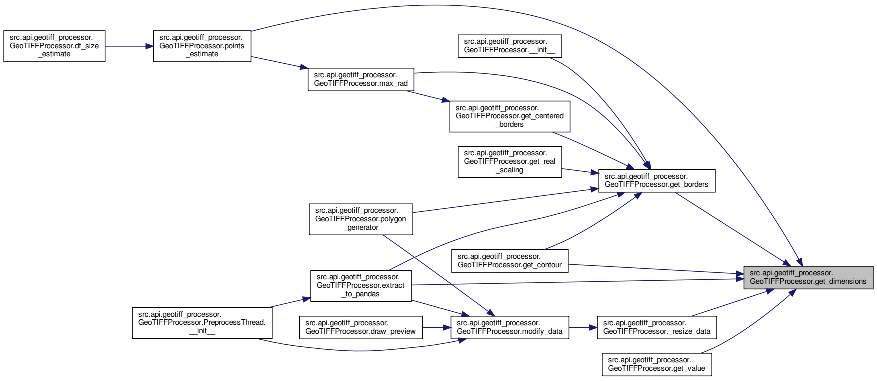
3.5.3.13 get_dimensions()

```
def src.api.geotiff_processor.GeoTIFFProcessor.get_dimensions (
              self,
              data = None )
```

```
Get number of elements for each axis

:param data: GeoRaster
```

Definition at line 163 of file geotiff_processor.py.

Here is the caller graph for this function:


src.api.geotiff_processor.
GeoTIFFProcessor.df_size
_estimate
src.api.geotiff_processor.
GeoTIFFProcessor.points
_estimate
src.api.geotiff_processor.
GeoTIFFProcessor.__init__
src.api.geotiff_processor.
GeoTIFFProcessor.max_rad
src.api.geotiff_processor.
GeoTIFFProcessor.get_centered
_borders
src.api.geotiff_processor.
GeoTIFFProcessor.get_real
_scaling
src.api.geotiff_processor.
GeoTIFFProcessor.get_borders
src.api.geotiff_processor.
GeoTIFFProcessor.polygon
_generator
src.api.geotiff_processor.
GeoTIFFProcessor.get_contour
src.api.geotiff_processor.
GeoTIFFProcessor.get_dimensions
src.api.geotiff_processor.
GeoTIFFProcessor.extract
_to_pandas
src.api.geotiff_processor.
GeoTIFFProcessor.PreprocessThread.
__init__
src.api.geotiff_processor.
GeoTIFFProcessor.draw_preview
src.api.geotiff_processor.
GeoTIFFProcessor.modify_data
src.api.geotiff_processor.
GeoTIFFProcessor._resize_data
src.api.geotiff_processor.
GeoTIFFProcessor.get_value

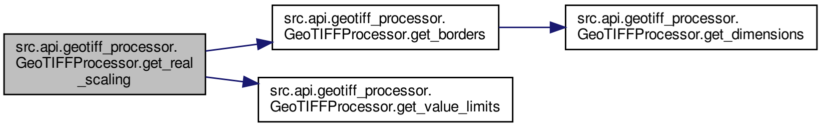
3.5.3.14 get_real_scaling()

```
def src.api.geotiff_processor.GeoTIFFProcessor.get_real_scaling (
              self,
              data = None )
```

```
Get a real scale of altitude to longitude

:param data: GeoRaster
```

Definition at line 183 of file geotiff_processor.py.

Here is the call graph for this function:


src.api.geotiff_processor.
GeoTIFFProcessor.get_real
_scaling
src.api.geotiff_processor.
GeoTIFFProcessor.get_borders
src.api.geotiff_processor.
GeoTIFFProcessor.get_dimensions
src.api.geotiff_processor.
GeoTIFFProcessor.get_value_limits

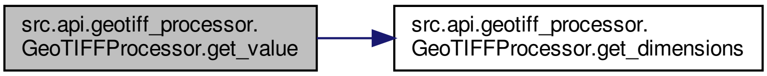
3.5.3.15 get_value()

```
def src.api.geotiff_processor.GeoTIFFProcessor.get_value (
              self,
              x,
              y,
              data = None )
```

```
Bilinear interpolation of a value at given coordinates

:param x: longitude
:param y: latitude
:param data: GeoRaster
```

Definition at line 251 of file geotiff_processor.py.

Here is the call graph for this function:


src.api.geotiff_processor.
GeoTIFFProcessor.get_value
src.api.geotiff_processor.
GeoTIFFProcessor.get_dimensions

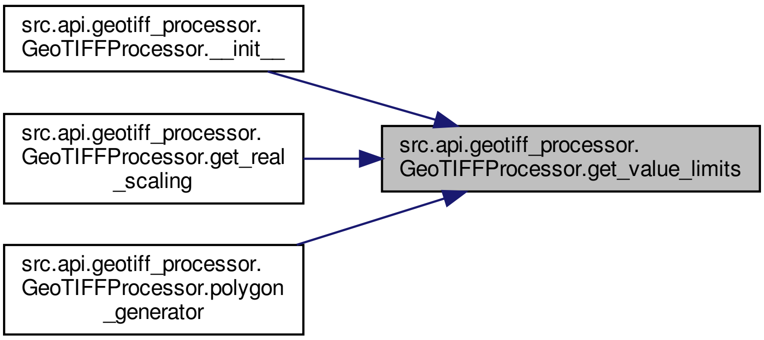
3.5.3.16 get_value_limits()

```
def src.api.geotiff_processor.GeoTIFFProcessor.get_value_limits (
              self,
              data = None )
```

```
Get minumum and maximum value (altitude)

:param data: GeoRaster
```

Definition at line 175 of file geotiff_processor.py.

Here is the caller graph for this function:


src.api.geotiff_processor.
GeoTIFFProcessor.__init__
src.api.geotiff_processor.
GeoTIFFProcessor.get_real
_scaling
src.api.geotiff_processor.
GeoTIFFProcessor.polygon
_generator
src.api.geotiff_processor.
GeoTIFFProcessor.get_value_limits

3.5.3.17 init_canvas()

```
def src.api.geotiff_processor.GeoTIFFProcessor.init_canvas (
              self )
```

```
Init matplotlib FigureCanvas
This can be used to embed stuff in PyQt
```

Definition at line 552 of file geotiff_processor.py.

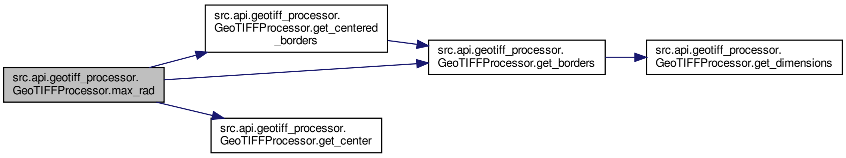
3.5.3.18 max_rad()

```
def src.api.geotiff_processor.GeoTIFFProcessor.max_rad (
              self,
              lat,
              lon )
```

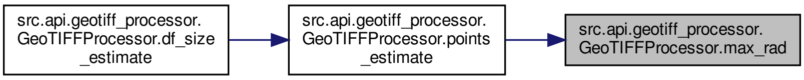
```
Get maximum radius to which can be extracted at given coordinates

:param lat:
:param lon:
```

Definition at line 199 of file geotiff_processor.py.

Here is the call graph for this function:


src.api.geotiff_processor.
GeoTIFFProcessor.max_rad
src.api.geotiff_processor.
GeoTIFFProcessor.get_centered
_borders
src.api.geotiff_processor.
GeoTIFFProcessor.get_borders
src.api.geotiff_processor.
GeoTIFFProcessor.get_dimensions
src.api.geotiff_processor.
GeoTIFFProcessor.get_center

Here is the caller graph for this function:



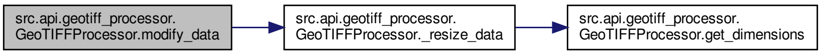
3.5.3.19 modify_data()

```
def src.api.geotiff_processor.GeoTIFFProcessor.modify_data (
              self,
              lat = None,
              lon = None,
              r = None,
              coef = 1 )
```

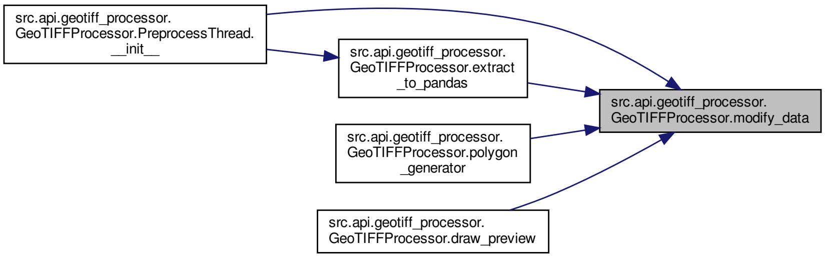
```
Extract a GeoRaster

:param lat: latitude of center
:param lon: longitude of center
:param r: radius for extraction
:param coef: if <1, data resolution will decrease
```

Definition at line 335 of file geotiff_processor.py.

Here is the call graph for this function:


src.api.geotiff_processor.
GeoTIFFProcessor.modify_data
src.api.geotiff_processor.
GeoTIFFProcessor._resize_data
src.api.geotiff_processor.
GeoTIFFProcessor.get_dimensions

Here is the caller graph for this function:


src.api.geotiff_processor.
GeoTIFFProcessor.PreprocessThread.
__init__
src.api.geotiff_processor.
GeoTIFFProcessor.extract
_to_pandas
src.api.geotiff_processor.
GeoTIFFProcessor.modify_data
src.api.geotiff_processor.
GeoTIFFProcessor.polygon
_generator
src.api.geotiff_processor.
GeoTIFFProcessor.draw_preview

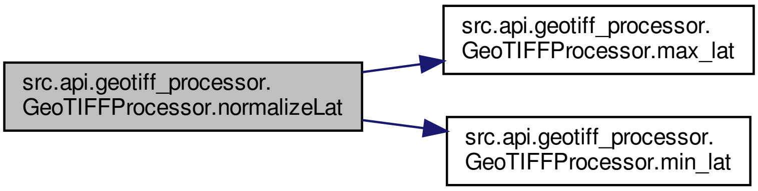
3.5.3.20 normalizeLat()

```
def src.api.geotiff_processor.GeoTIFFProcessor.normalizeLat (
              self,
              lat )
```

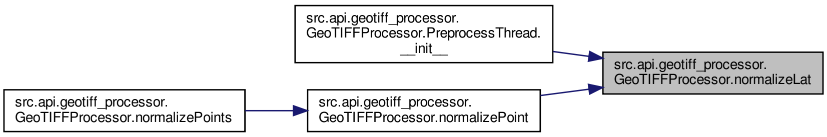
```
Returns a normalized (0 <= return <= 1) value of the latitude

:param lat: latitude
```

Definition at line 282 of file geotiff_processor.py.

Here is the call graph for this function:


src.api.geotiff_processor.
GeoTIFFProcessor.normalizeLat
src.api.geotiff_processor.
GeoTIFFProcessor.max_lat
src.api.geotiff_processor.
GeoTIFFProcessor.min_lat

Here is the caller graph for this function:


src.api.geotiff_processor.
GeoTIFFProcessor.PreprocessThread.
__init__
src.api.geotiff_processor.
GeoTIFFProcessor.normalizePoints
src.api.geotiff_processor.
GeoTIFFProcessor.normalizePoint
src.api.geotiff_processor.
GeoTIFFProcessor.normalizeLat

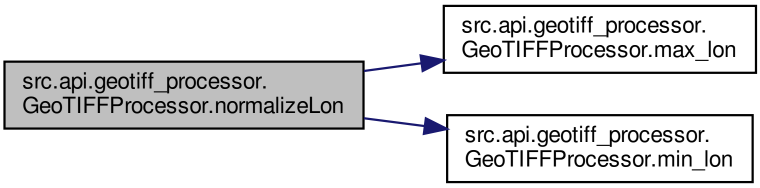
3.5.3.21 normalizeLon()

```
def src.api.geotiff_processor.GeoTIFFProcessor.normalizeLon (
              self,
              lon )
```

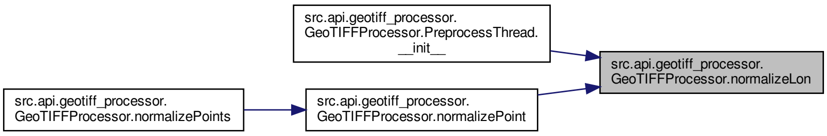
```
The same as GeoTIFFProcessor.normalizeLat

:param lon: longitude
```

Definition at line 292 of file geotiff_processor.py.

Here is the call graph for this function:


src.api.geotiff_processor.
GeoTIFFProcessor.normalizeLon
src.api.geotiff_processor.
GeoTIFFProcessor.max_lon
src.api.geotiff_processor.
GeoTIFFProcessor.min_lon

Here is the caller graph for this function:


src.api.geotiff_processor.
GeoTIFFProcessor.PreprocessThread.
__init__
src.api.geotiff_processor.
GeoTIFFProcessor.normalizeLon
src.api.geotiff_processor.
GeoTIFFProcessor.normalizePoints
src.api.geotiff_processor.
GeoTIFFProcessor.normalizePoint

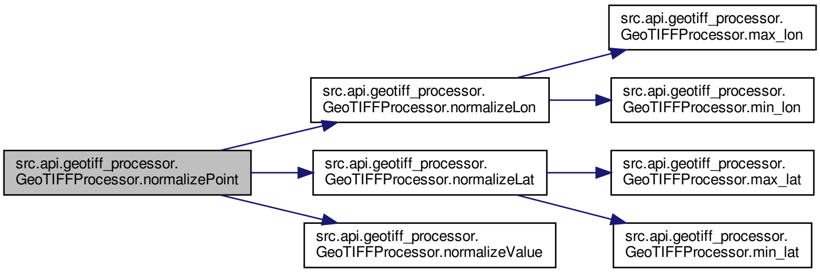
3.5.3.22 normalizePoint()

```
def src.api.geotiff_processor.GeoTIFFProcessor.normalizePoint (
              self,
              point )
```

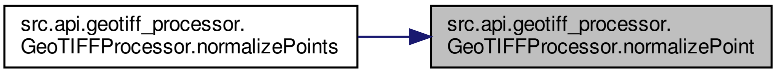
```
Normalize (lon, lat, value)

:param point:
```

Definition at line 316 of file geotiff_processor.py.

Here is the call graph for this function:


src.api.geotiff_processor.
GeoTIFFProcessor.max_lon
src.api.geotiff_processor.
GeoTIFFProcessor.normalizeLon
src.api.geotiff_processor.
GeoTIFFProcessor.min_lon
src.api.geotiff_processor.
GeoTIFFProcessor.normalizePoint
src.api.geotiff_processor.
GeoTIFFProcessor.normalizeLat
src.api.geotiff_processor.
GeoTIFFProcessor.max_lat
src.api.geotiff_processor.
GeoTIFFProcessor.normalizeValue
src.api.geotiff_processor.
GeoTIFFProcessor.min_lat

Here is the caller graph for this function:


src.api.geotiff_processor.
GeoTIFFProcessor.normalizePoints
src.api.geotiff_processor.
GeoTIFFProcessor.normalizePoint

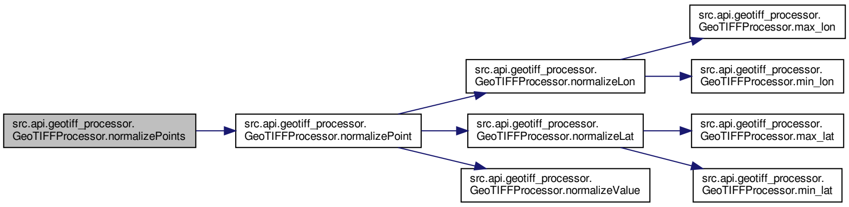
3.5.3.23 normalizePoints()

```
def src.api.geotiff_processor.GeoTIFFProcessor.normalizePoints (
              self,
              polygon )
```

```
Apply GeoTIFFProcessor.normalizePoints for each element

:param polygon: iterable of (lon, lat, value)
```

Definition at line 327 of file geotiff_processor.py.

Here is the call graph for this function:


src.api.geotiff_processor.
GeoTIFFProcessor.normalizePoints
src.api.geotiff_processor.
GeoTIFFProcessor.normalizePoint
src.api.geotiff_processor.
GeoTIFFProcessor.normalizeLon
src.api.geotiff_processor.
GeoTIFFProcessor.normalizeLat
src.api.geotiff_processor.
GeoTIFFProcessor.normalizeValue
src.api.geotiff_processor.
GeoTIFFProcessor.max_lon
src.api.geotiff_processor.
GeoTIFFProcessor.min_lon
src.api.geotiff_processor.
GeoTIFFProcessor.max_lat
src.api.geotiff_processor.
GeoTIFFProcessor.min_lat

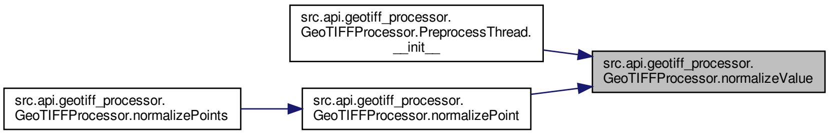
3.5.3.24 normalizeValue()

```
def src.api.geotiff_processor.GeoTIFFProcessor.normalizeValue (
              self,
              value )
```

```
See GeoTIFFProcessor.normalizeLat

:param value: altitude
```

Definition at line 302 of file geotiff_processor.py.

Here is the caller graph for this function:


src.api.geotiff_processor.
GeoTIFFProcessor.PreprocessThread.
__init__
src.api.geotiff_processor.
GeoTIFFProcessor.normalizeValue
src.api.geotiff_processor.
GeoTIFFProcessor.normalizePoints
src.api.geotiff_processor.
GeoTIFFProcessor.normalizePoint

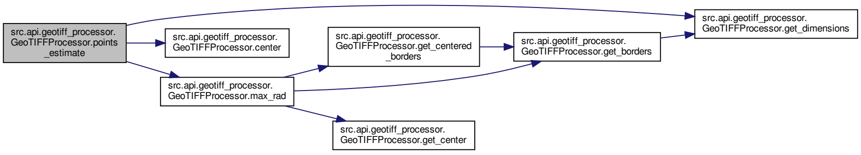
3.5.3.25 points_estimate()

```
def src.api.geotiff_processor.GeoTIFFProcessor.points_estimate (
              self,
              r = None,
              coef = 1 )
```

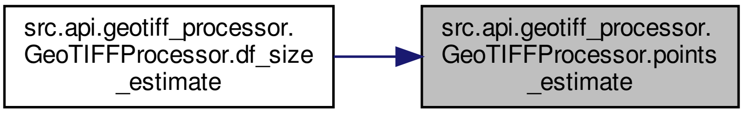
```
Roughly estimate a number of points after extraction with given
radius

:param r: radius (must be in the same scale as lat and lon)
:param coef: scale coefficient
```

Definition at line 221 of file geotiff_processor.py.

Here is the call graph for this function:


src.api.geotiff_processor.
GeoTIFFProcessor.points
_estimate
src.api.geotiff_processor.
GeoTIFFProcessor.center
src.api.geotiff_processor.
GeoTIFFProcessor.max_rad
src.api.geotiff_processor.
GeoTIFFProcessor.get_centered
_borders
src.api.geotiff_processor.
GeoTIFFProcessor.get_borders
src.api.geotiff_processor.
GeoTIFFProcessor.get_dimensions
src.api.geotiff_processor.
GeoTIFFProcessor.get_center

Here is the caller graph for this function:



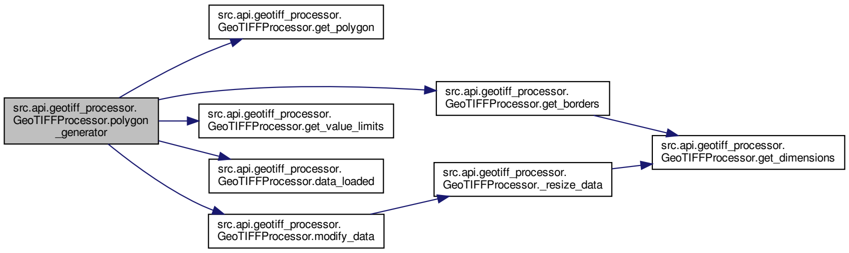
3.5.3.26 polygon_generator()

```
def src.api.geotiff_processor.GeoTIFFProcessor.polygon_generator (
              self,
              df )
```

```
Iterate through polygons in DataFrame

:param df: df from GeoTIFFProcessor.extract_to_pandas
```

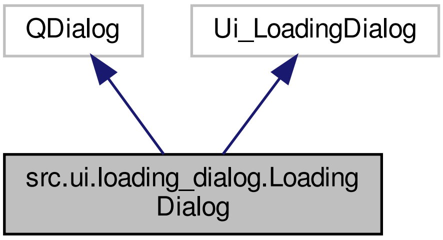
Definition at line 525 of file geotiff_processor.py.

Here is the call graph for this function:

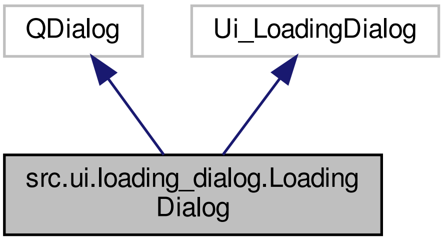

src.api.geotiff_processor.
GeoTIFFProcessor.polygon
_generator
src.api.geotiff_processor.
GeoTIFFProcessor.get_polygon
src.api.geotiff_processor.
GeoTIFFProcessor.get_borders
src.api.geotiff_processor.
GeoTIFFProcessor.get_value_limits
src.api.geotiff_processor.
GeoTIFFProcessor.data_loaded
src.api.geotiff_processor.
GeoTIFFProcessor.modify_data
src.api.geotiff_processor.
GeoTIFFProcessor._resize_data
src.api.geotiff_processor.
GeoTIFFProcessor.get_dimensions

3.6 src.ui.loading_dialog.LoadingDialog Class Reference

Inheritance diagram for src.ui.loading_dialog.LoadingDialog:



Collaboration diagram for src.ui.loading_dialog.LoadingDialog:



Public Member Functions

- def **__init__** (self, str operation, total=0, parent=None)
- def **show** (self)
- def **set_status** (self, status)
- def **set_operation** (self, operation)
- def **set_done** (self, int value)
- def **ready** (self)

Public Attributes

- **total**
- **start_time**

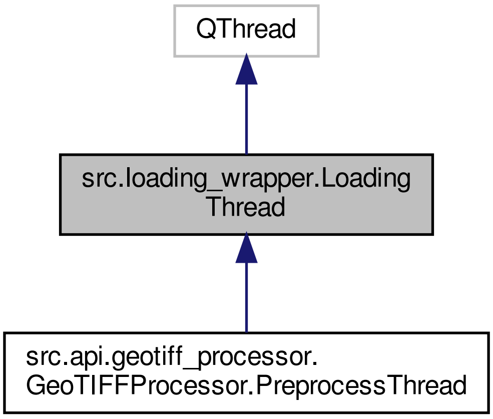
3.6.1 Detailed Description

```
A class to show nice loading dialog with ETA estimation and stuff
```

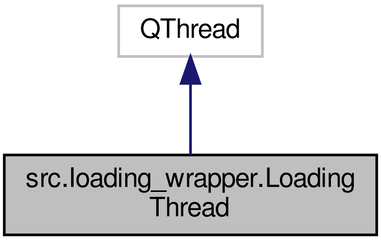
Definition at line 23 of file loading_dialog.py.

3.7 src.loading_wrapper.LoadingThread Class Reference

Inheritance diagram for src.loading_wrapper.LoadingThread:


QThread
src.loading_wrapper.Loading Thread
src.api.geotiff_processor. GeoTIFFProcessor.PreprocessThread

Collaboration diagram for src.loading_wrapper.LoadingThread:



Public Member Functions

- def **__init__** (self, parent=None)
- def set_interval (self, iter_num)
- def check_percent (self, iter_)
- def **run** (self, ∗args, ∗∗kwargs)

Public Attributes

- **operation**
- **i**
- **interval**
- **total**

Static Public Attributes

- **updateStatus** = pyqtSignal(str)
- **updatePercent** = pyqtSignal(int)
- **updateMaxPercent** = pyqtSignal(int)
- **loadingDone** = pyqtSignal()

3.7.1 Detailed Description

```
This is base class of thread for using with LoadingWrapper
The idea is to move some heavy operations to a special thread and show
progress on the LoadingDialog.
This actually decreases perfomance a bit because of GIL, but improves user
experience
```

Definition at line 7 of file loading_wrapper.py.

3.7.2 Member Function Documentation

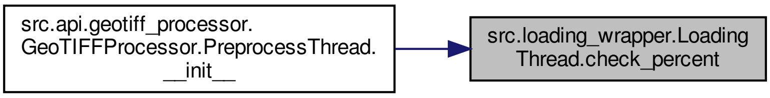
3.7.2.1 check_percent()

```
def src.loading_wrapper.LoadingThread.check_percent (
              self,
              iter_ )
```

```
Update percent for current operation number.
Intended to be used after LoadingThread.set_interval

:param iter_: 0 <= iter_ <= iter_num
```

Definition at line 36 of file loading_wrapper.py.

Here is the caller graph for this function:


src.api.geotiff_processor.
GeoTIFFProcessor.PreprocessThread.
__init__
src.loading_wrapper.Loading
Thread.check_percent

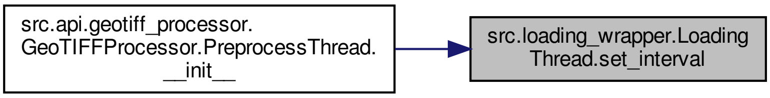
3.7.2.2 set_interval()

```
def src.loading_wrapper.LoadingThread.set_interval (
              self,
              iter_num )
```

```
Set maximum number of operations

:param iter_num: Number of operations
```

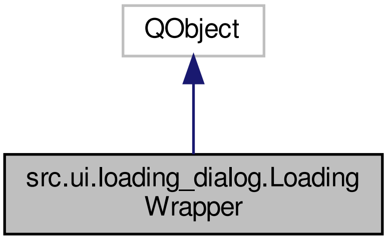
Definition at line 24 of file loading_wrapper.py.

Here is the caller graph for this function:

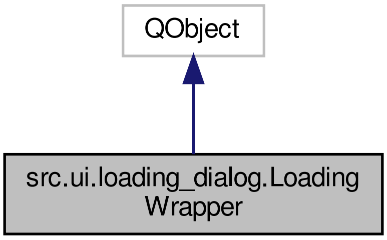

src.api.geotiff_processor.
GeoTIFFProcessor.PreprocessThread.
__init__
src.loading_wrapper.Loading
Thread.set_interval

3.8 src.ui.loading_dialog.LoadingWrapper Class Reference

Inheritance diagram for src.ui.loading_dialog.LoadingWrapper:



Collaboration diagram for src.ui.loading_dialog.LoadingWrapper:



Public Member Functions

- def **__init__** (self, LoadingThread thread, parent=None)
- def **start** (self)

Public Attributes

- **thread**
- **dialog**

Static Public Attributes

- **loadingDone** = pyqtSignal()

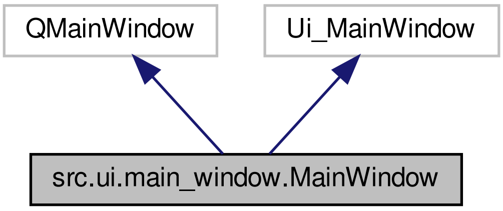
3.8.1 Detailed Description

```
This class runs a LoadingThread and shows the process
in a LoadingDialog
```

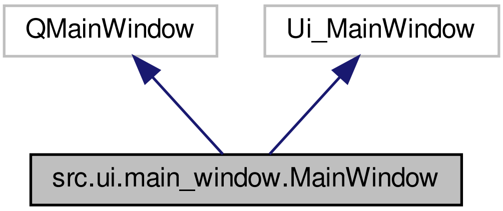
Definition at line 61 of file loading_dialog.py.

3.9 src.ui.main_window.MainWindow Class Reference

Inheritance diagram for src.ui.main_window.MainWindow:



Collaboration diagram for src.ui.main_window.MainWindow:



Public Member Functions

- def **__init__** (self, processor, df, parent=None)
- def updateUi (self)
- def onOpenAnother (self)
- def mapDockWidgetControls (self)
- def prepareScene (self)
- def **getLightSourceCoords** (self)
- def **getMapPolygons** (self)
- def getColorByValue (self, value)

- def **swapPoint** (self, lon, lat, value)
- def **swapPoints** (self, polygon)
- def prepareNormalLines (self, polygons, normals, colors)
- def preparePolygons (self, polygons, normals, colors, start_index=None)
- def **getGrid** (self)
- def **getContour** (self)
- def **getLightLines** (self)
- def prepareLines (self, lines, line_colors, start_index=None)
- def **prepareLine** (self, line, colors, start_index=None)
- def **initializeGL** (self)
- def **setUpShaders** (self)
- def initVertexArrays (self)
- def **updateGL** (func)
- def updateCameraInfo (func)
- def updateLightData (self)
- def updateBuffer (self)
- def **paintGL** (self)
- def **loadScene** (self)
- def updateMatrices (self)
- def updateParams (self)
- def drawScene (self)
- def **drawPreparedPolygons** (self, start, end)
- def **drawPreparedLines** (self, start, end)
- def **drawPreparedLineStrips** (self, arr)
- def setupControls (self)
- def **keyPressed** (self, event)
- def **mouseMoved** (self, event)
- def **getRotVec** (self)
- def **moveCamera** (self, az=0, pol=0, x=0, y=0, z=0)
- def **moveCameraAroundCenter** (self, az=0, pol=0)
- def **scaleView** (self, x=None, y=None, z=None)
- def **setLight** (self, x=None, y=None, z=None, ambient=None, diffuse=None)
- def **setDisplay** (self, alpha=None, invisible=None)
- def **setGrid** (self, show)
- def **setContour** (self, show=None, levels=None)
- def **toggleGrabKeyboard** (self, bool grab)
- def **toggleGrabMouse** (self, bool grab)
- def **getMouseCenter** (self)

Public Attributes

- **parent**
- **processor**
- **df**
- **keyPressEvent**
- **mouseMoveEvent**
- **mouse_grabbed**
- **camera_pos**
- **center**
- **rot_center**
- **camera_rot**
- **scale_vec**
- **real_prop**
- **light_pos**

- **ambient**
- **diffuse**
- **alpha**
- **normals**
- **colors**
- **coords_array**
- **update_light**
- **update_buffer**
- **show_grid**
- **show_contour**
- **contour_levels**
- **show_light_lines**
- **grid_freq**
- **grid_color**
- **contour_color**
- **light_line_color**
- **shaders**
- **elevationWidget**
- **minimapWidget**
- **dock_widgets**
- **dock_actions**
- **map_data**
- **light_data**
- **light_lines_data**
- **grid_data**
- **contour_data**
- **invisible**
- **mouse_center**

3.9.1 Detailed Description

Definition at line 28 of file main_window.py.

3.9.2 Member Function Documentation

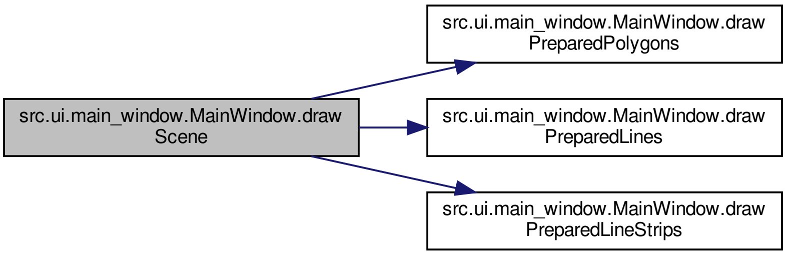
3.9.2.1 drawScene()

```
def src.ui.main_window.MainWindow.drawScene (
              self )
```

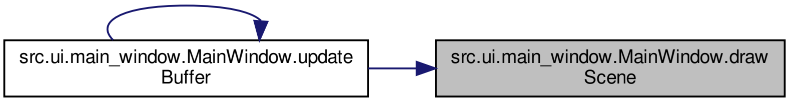
```
Paint all the prepared data
```

Definition at line 481 of file main_window.py.

Here is the call graph for this function:


src.ui.main_window.MainWindow.draw
Scene
src.ui.main_window.MainWindow.draw
PreparedPolygons
src.ui.main_window.MainWindow.draw
PreparedLines
src.ui.main_window.MainWindow.draw
PreparedLineStrips

Here is the caller graph for this function:


src.ui.main_window.MainWindow.update
Buffer
src.ui.main_window.MainWindow.draw
Scene

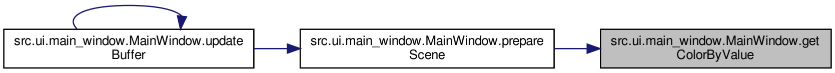
3.9.2.2 getColorByValue()

```
def src.ui.main_window.MainWindow.getColorByValue (
              self,
              value )
```

```
Get QVector4D-color from green (value == 0) to red (value == 1)
:param value: 0 - 1
```

Definition at line 211 of file main_window.py.

Here is the caller graph for this function:


src.ui.main_window.MainWindow.update
Buffer
src.ui.main_window.MainWindow.prepare
Scene
src.ui.main_window.MainWindow.get
ColorByValue

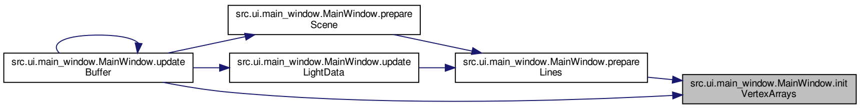
3.9.2.3 initVertexArrays()

```
def src.ui.main_window.MainWindow.initVertexArrays (
              self )
```

```
Init buffers
```

Definition at line 381 of file main_window.py.

Here is the caller graph for this function:


src.ui.main_window.MainWindow.prepare Scene
src.ui.main_window.MainWindow.update Buffer
src.ui.main_window.MainWindow.update LightData
src.ui.main_window.MainWindow.prepare Lines
src.ui.main_window.MainWindow.init VertexArrays

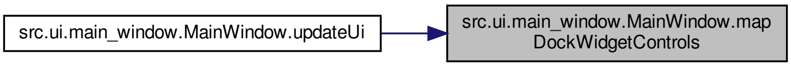
3.9.2.4 mapDockWidgetControls()

```
def src.ui.main_window.MainWindow.mapDockWidgetControls (
              self )
```

```
Show/hide dockwidgets via corresponding actions
```

Definition at line 123 of file main_window.py.

Here is the caller graph for this function:


src.ui.main_window.MainWindow.updateUi
src.ui.main_window.MainWindow.map DockWidgetControls

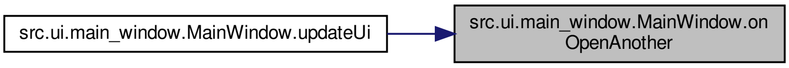
3.9.2.5 onOpenAnother()

```
def src.ui.main_window.MainWindow.onOpenAnother (
              self )
```

```
Open another GeoTIFF file
```

Definition at line 117 of file main_window.py.

Here is the caller graph for this function:


src.ui.main_window.MainWindow.updateUi
src.ui.main_window.MainWindow.on OpenAnother

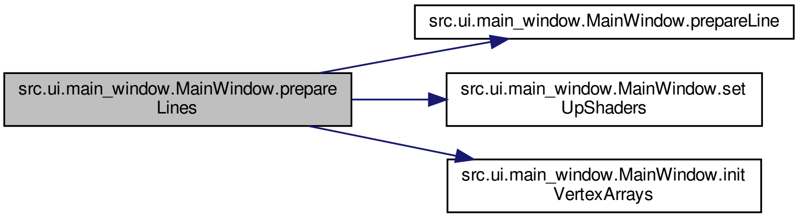
3.9.2.6 prepareLines()

```
def src.ui.main_window.MainWindow.prepareLines (
              self,
              lines,
              line_colors,
              start_index = None )
```

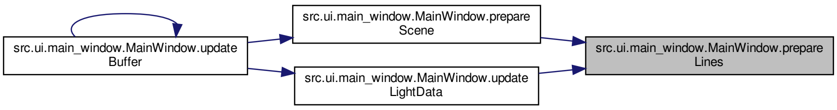
```
Prepare lines for bufferization

:param lines: list like one for polygons
:param line_colors: list of colors of each line
:param start_index: start index
```

Definition at line 327 of file main_window.py.

Here is the call graph for this function:


src.ui.main_window.MainWindow.prepareLine
src.ui.main_window.MainWindow.prepare Lines
src.ui.main_window.MainWindow.set UpShaders
src.ui.main_window.MainWindow.init VertexArrays

Here is the caller graph for this function:


src.ui.main_window.MainWindow.update Buffer
src.ui.main_window.MainWindow.prepare Scene
src.ui.main_window.MainWindow.update LightData
src.ui.main_window.MainWindow.prepare Lines

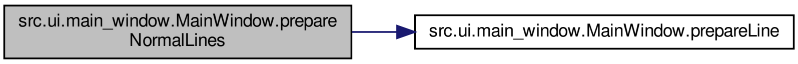
3.9.2.7 prepareNormalLines()

```
def src.ui.main_window.MainWindow.prepareNormalLines (
              self,
              polygons,
              normals,
              colors )
```

```
Normal lines for each polygon.
Debug only
```

Definition at line 226 of file main_window.py.

Here is the call graph for this function:


src.ui.main_window.MainWindow.prepare NormalLines
src.ui.main_window.MainWindow.prepareLine

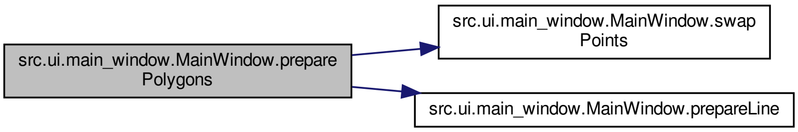
3.9.2.8 preparePolygons()

```
def src.ui.main_window.MainWindow.preparePolygons (
              self,
              polygons,
              normals,
              colors,
              start_index = None )
```

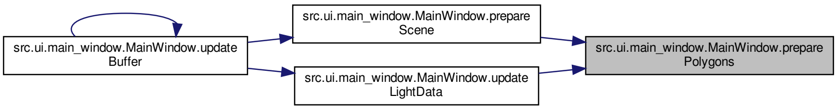
```
Prepare polygons for bufferization

:param polygons: a list like [(p11, p12, ...), (p21, p22, ...)]
:param normals: normals with number of normals equal to points
:param colors: colors for each point
:param start_index: if None, start at the end of each array.
Otherwise, overwrite
```

Definition at line 244 of file main_window.py.

Here is the call graph for this function:


src.ui.main_window.MainWindow.prepare Polygons
src.ui.main_window.MainWindow.swap Points
src.ui.main_window.MainWindow.prepareLine

Here is the caller graph for this function:


src.ui.main_window.MainWindow.update Buffer
src.ui.main_window.MainWindow.prepare Scene
src.ui.main_window.MainWindow.update LightData
src.ui.main_window.MainWindow.prepare Polygons

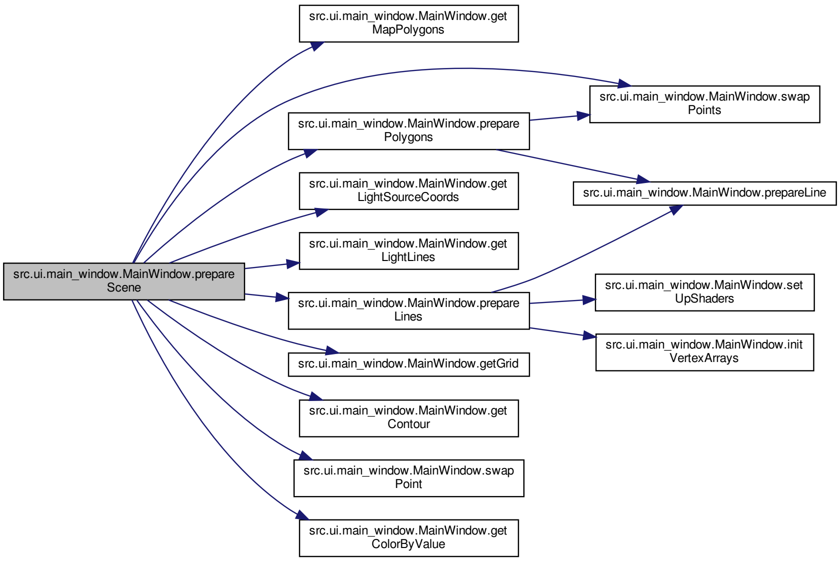
3.9.2.9 prepareScene()

```
def src.ui.main_window.MainWindow.prepareScene (
              self )
```

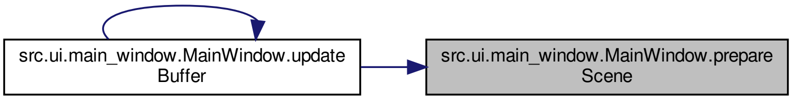
```
Prepare the data for bufferization
```

Definition at line 150 of file main_window.py.

Here is the call graph for this function:


src.ui.main_window.MainWindow.prepare Scene
src.ui.main_window.MainWindow.get MapPolygons
src.ui.main_window.MainWindow.swap Points
src.ui.main_window.MainWindow.prepare Polygons
src.ui.main_window.MainWindow.get LightSourceCoords
src.ui.main_window.MainWindow.prepareLine
src.ui.main_window.MainWindow.get LightLines
src.ui.main_window.MainWindow.set UpShaders
src.ui.main_window.MainWindow.prepare Lines
src.ui.main_window.MainWindow.init VertexArrays
src.ui.main_window.MainWindow.getGrid
src.ui.main_window.MainWindow.get Contour
src.ui.main_window.MainWindow.swap Point
src.ui.main_window.MainWindow.get ColorByValue

Here is the caller graph for this function:


src.ui.main_window.MainWindow.update Buffer
src.ui.main_window.MainWindow.prepare Scene

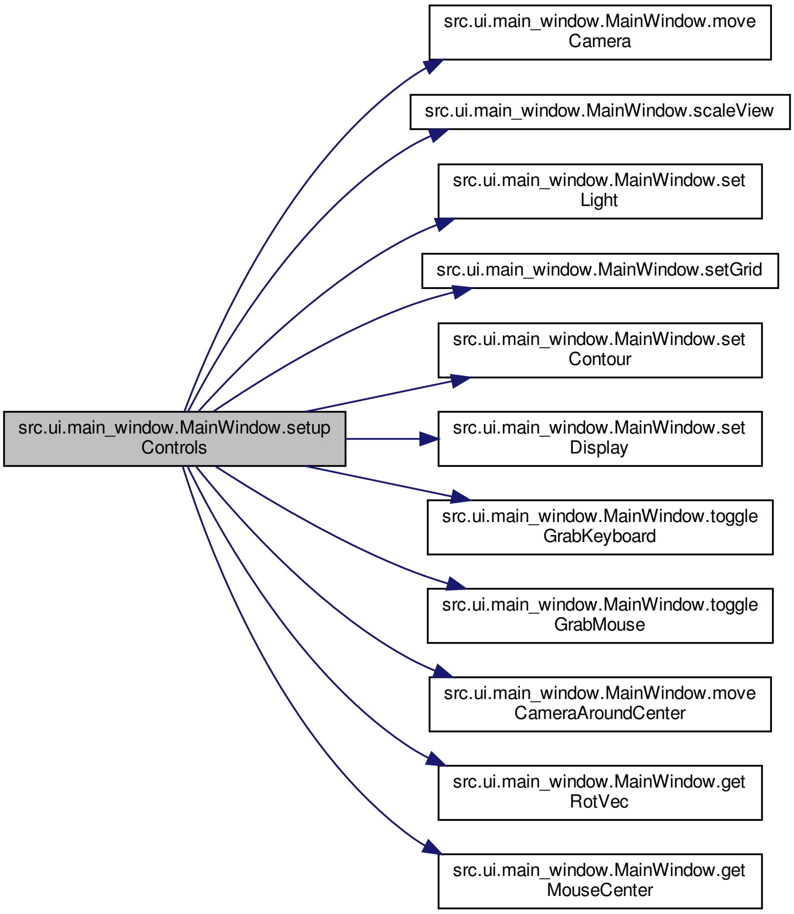
3.9.2.10 setupControls()

```
def src.ui.main_window.MainWindow.setupControls (
              self )
```

```
Connect controls to this class' methods
```

Definition at line 514 of file main_window.py.

Here is the call graph for this function:


src.ui.main_window.MainWindow.setup Controls
src.ui.main_window.MainWindow.move Camera
src.ui.main_window.MainWindow.scaleView
src.ui.main_window.MainWindow.set Light
src.ui.main_window.MainWindow.setGrid
src.ui.main_window.MainWindow.set Contour
src.ui.main_window.MainWindow.set Display
src.ui.main_window.MainWindow.toggle GrabKeyboard
src.ui.main_window.MainWindow.toggle GrabMouse
src.ui.main_window.MainWindow.move CameraAroundCenter
src.ui.main_window.MainWindow.get RotVec
src.ui.main_window.MainWindow.get MouseCenter

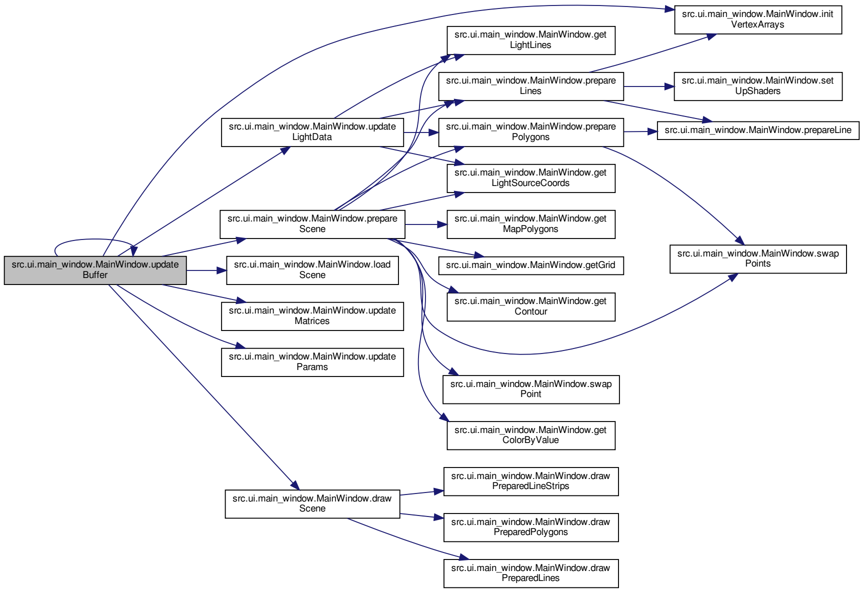
3.9.2.11 updateBuffer()

```
def src.ui.main_window.MainWindow.updateBuffer (
              self )
```

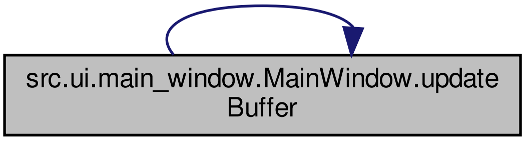
```
Update the whole buffer
```

Definition at line 425 of file main_window.py.

Here is the call graph for this function:


src.ui.main_window.MainWindow.update Buffer
src.ui.main_window.MainWindow.update LightData
src.ui.main_window.MainWindow.prepare Scene
src.ui.main_window.MainWindow.load Scene
src.ui.main_window.MainWindow.update Matrices
src.ui.main_window.MainWindow.update Params
src.ui.main_window.MainWindow.draw Scene
src.ui.main_window.MainWindow.get LightLines
src.ui.main_window.MainWindow.prepare Lines
src.ui.main_window.MainWindow.prepare Polygons
src.ui.main_window.MainWindow.get LightSourceCoords
src.ui.main_window.MainWindow.get MapPolygons
src.ui.main_window.MainWindow.getGrid
src.ui.main_window.MainWindow.get Contour
src.ui.main_window.MainWindow.swap Point
src.ui.main_window.MainWindow.get ColorByValue
src.ui.main_window.MainWindow.draw PreparedLineStrips
src.ui.main_window.MainWindow.draw PreparedPolygons
src.ui.main_window.MainWindow.draw PreparedLines
src.ui.main_window.MainWindow.init VertexArrays
src.ui.main_window.MainWindow.set UpShaders
src.ui.main_window.MainWindow.prepareLine
src.ui.main_window.MainWindow.swap Points

Here is the caller graph for this function:



3.9.2.12 updateCameraInfo()

```
def src.ui.main_window.MainWindow.updateCameraInfo (
              func )
```

```
A decorator to update camera widgets after excution
```

Definition at line 399 of file main_window.py.

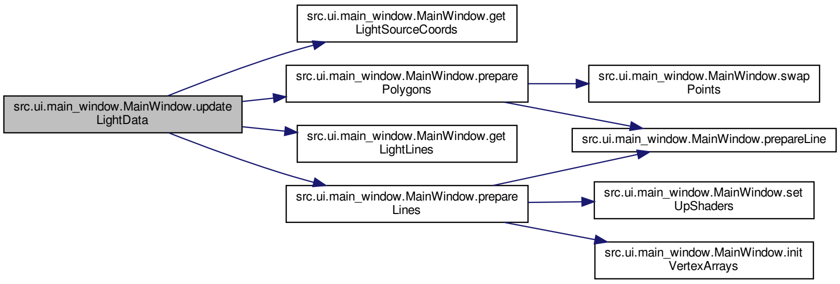
3.9.2.13 updateLightData()

```
def src.ui.main_window.MainWindow.updateLightData (
              self )
```

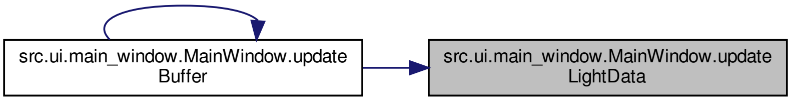
```
Update parts of buffer used to display the light source
```

Definition at line 416 of file main_window.py.

Here is the call graph for this function:


src.ui.main_window.MainWindow.update LightData
src.ui.main_window.MainWindow.get LightSourceCoords
src.ui.main_window.MainWindow.prepare Polygons
src.ui.main_window.MainWindow.get LightLines
src.ui.main_window.MainWindow.prepare Lines
src.ui.main_window.MainWindow.swap Points
src.ui.main_window.MainWindow.prepareLine
src.ui.main_window.MainWindow.set UpShaders
src.ui.main_window.MainWindow.init VertexArrays

Here is the caller graph for this function:


src.ui.main_window.MainWindow.update Buffer
src.ui.main_window.MainWindow.update LightData

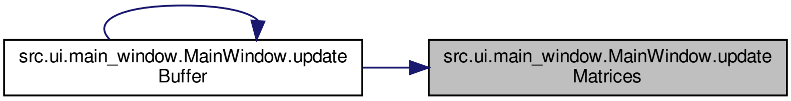
3.9.2.14 updateMatrices()

```
def src.ui.main_window.MainWindow.updateMatrices (
              self )
```

```
Update projection matrices
```

Definition at line 453 of file main_window.py.

Here is the caller graph for this function:


src.ui.main_window.MainWindow.update Buffer
src.ui.main_window.MainWindow.update Matrices

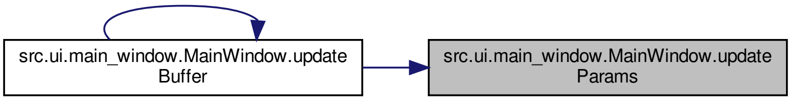
3.9.2.15 updateParams()

```
def src.ui.main_window.MainWindow.updateParams (
              self )
```

```
Update light, alpha and scaling parameters
```

Definition at line 472 of file main_window.py.

Here is the caller graph for this function:


src.ui.main_window.MainWindow.update
Buffer
src.ui.main_window.MainWindow.update
Params

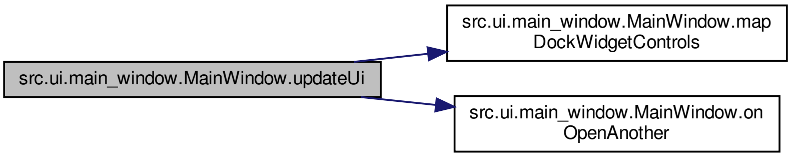
3.9.2.16 updateUi()

```
def src.ui.main_window.MainWindow.updateUi (
              self )
```

```
Set up custom widgets & positions
```

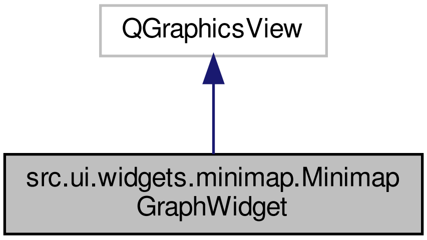
Definition at line 85 of file main_window.py.

Here is the call graph for this function:

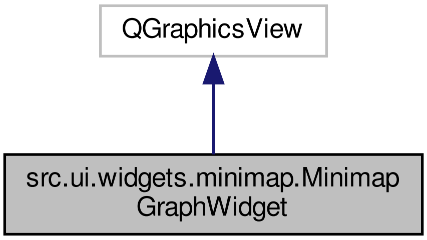

src.ui.main_window.MainWindow.map
DockWidgetControls
src.ui.main_window.MainWindow.updateUi
src.ui.main_window.MainWindow.on
OpenAnother

3.10 src.ui.widgets.minimap.MinimapGraphWidget Class Reference

Inheritance diagram for src.ui.widgets.minimap.MinimapGraphWidget:



Collaboration diagram for src.ui.widgets.minimap.MinimapGraphWidget:



Public Member Functions

- def **__init__** (self, processor, QVector3D position, QVector3D rotation, width=240, height=240, parent=None)
- def **updateCameraInfo** (self, pos, rot)
- def **getThresholds** (self)
- def **onResize** (self, event)
- def **initContent** (self)

Public Attributes

- **processor**
- **pos**
- **rot**
- **resizeEvent**
- **cnv_x_thresholds**
- **cnv_y_thresholds**

- **contour_kwargs**
- **x_threshold**
- **y_threshold**
- **big_canvas**
- **camera**
- **canvas**

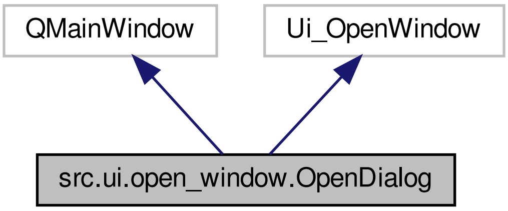
3.10.1 Detailed Description

```
The widget to show a minimap with a camera position
```

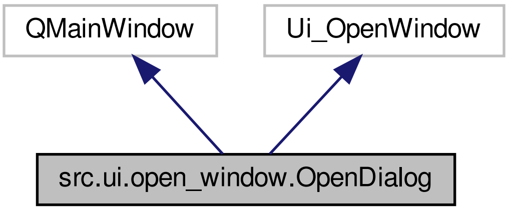
Definition at line 84 of file minimap.py.

3.11 src.ui.open_window.OpenDialog Class Reference

Inheritance diagram for src.ui.open_window.OpenDialog:



Collaboration diagram for src.ui.open_window.OpenDialog:



Public Member Functions

- def **__init__** (self, parent=None)
- def connect_controls (self)
- def **set_rad_control** (self)
- def **proc_params** (self)
- def on_open (self)
- def set_controls (self)
- def update_ranges (self, data=None)
- def **on_preview** (self)
- def **init_matplotlib** (self)
- def on_params_changed (self)
- def **on_ok_pressed** (self)
- def **on_normals_ready** (self, df)
- def **on_save** (self)

Public Attributes

- **processor**
- **r**
- **coef**
- **thread**
- **activate**
- **canvas**
- **nav_toolbar**
- **loading**
- **window**

Static Public Attributes

- **params_changed** = pyqtSignal()

3.11.1 Detailed Description

Definition at line 22 of file open_window.py.

3.11.2 Member Function Documentation

3.11.2.1 connect_controls()

```
def src.ui.open_window.OpenDialog.connect_controls (
              self )
```

```
Connect all the controls in OpenDialog
```

Definition at line 43 of file open_window.py.

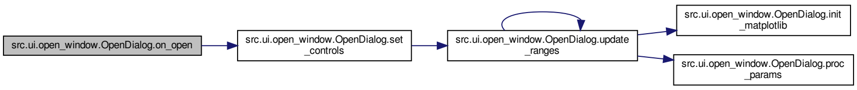
3.11.2.2 on_open()

```
def src.ui.open_window.OpenDialog.on_open (
              self )
```

```
Execute on open file
```

Definition at line 81 of file open_window.py.

Here is the call graph for this function:


src.ui.open_window.OpenDialog.on_open
src.ui.open_window.OpenDialog.set _controls
src.ui.open_window.OpenDialog.update _ranges
src.ui.open_window.OpenDialog.init _matplotlib
src.ui.open_window.OpenDialog.proc _params

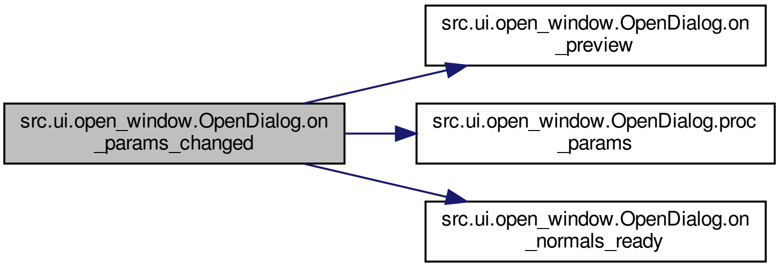
3.11.2.3 on_params_changed()

```
def src.ui.open_window.OpenDialog.on_params_changed (
              self )
```

```
Executed whenever a parameter is changed
```

Definition at line 137 of file open_window.py.

Here is the call graph for this function:


src.ui.open_window.OpenDialog.on _params_changed
src.ui.open_window.OpenDialog.on _preview
src.ui.open_window.OpenDialog.proc _params
src.ui.open_window.OpenDialog.on _normals_ready

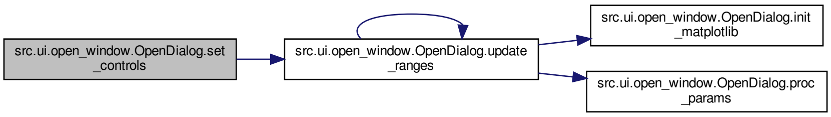
3.11.2.4 set_controls()

```
def src.ui.open_window.OpenDialog.set_controls (
              self )
```

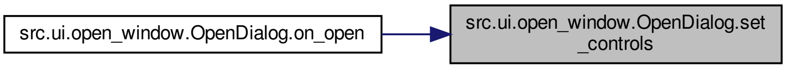
```
Update controls in accordance to a loaded file
```

Definition at line 91 of file open_window.py.

Here is the call graph for this function:


src.ui.open_window.OpenDialog.set
_controls
src.ui.open_window.OpenDialog.update
_ranges
src.ui.open_window.OpenDialog.init
_matplotlib
src.ui.open_window.OpenDialog.proc
_params

Here is the caller graph for this function:


src.ui.open_window.OpenDialog.on_open
src.ui.open_window.OpenDialog.set
_controls

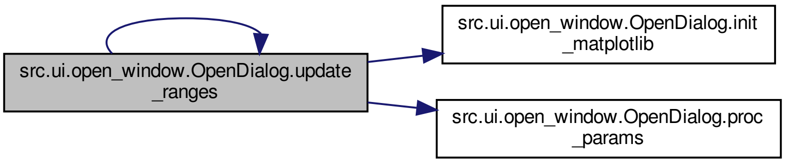
3.11.2.5 update_ranges()

```
def src.ui.open_window.OpenDialog.update_ranges (
              self,
              data = None )
```

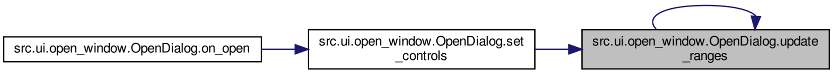
```
Update ranges for an extracted data

:param data: GeoRaster
```

Definition at line 105 of file open_window.py.

Here is the call graph for this function:

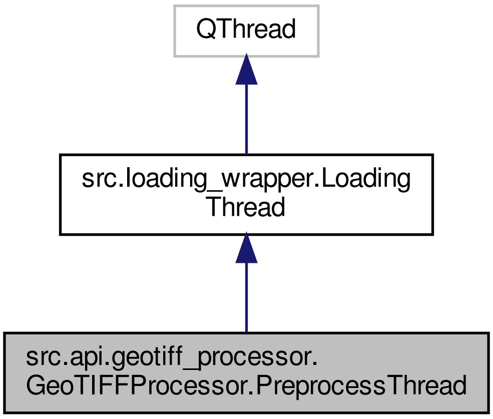

src.ui.open_window.OpenDialog.update
_ranges
src.ui.open_window.OpenDialog.init
_matplotlib
src.ui.open_window.OpenDialog.proc
_params

Here is the caller graph for this function:

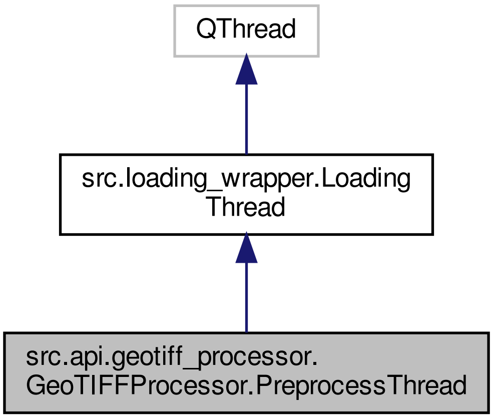

src.ui.open_window.OpenDialog.on_open
src.ui.open_window.OpenDialog.set_controls
src.ui.open_window.OpenDialog.update_ranges

3.12 src.api.geotiff_processor.GeoTIFFProcessor.PreprocessThread Class Reference

Inheritance diagram for src.api.geotiff_processor.GeoTIFFProcessor.PreprocessThread:


QThread
src.loading_wrapper.LoadingThread
src.api.geotiff_processor.GeoTIFFProcessor.PreprocessThread

Collaboration diagram for src.api.geotiff_processor.GeoTIFFProcessor.PreprocessThread:


QThread
src.loading_wrapper.LoadingThread
src.api.geotiff_processor.GeoTIFFProcessor.PreprocessThread

Public Member Functions

- def __init__ (self, proc, normalize=False, parent=None, ∗args, ∗∗kwargs)
- def **run** (self, df=None)

Public Attributes

- **operation**
- **proc**
- **args**
- **kwargs**
- **df**
- **normalize**

Static Public Attributes

- **df_ready** = pyqtSignal(object)

3.12.1 Detailed Description

```
A thread to do
- extraction
- normalization
- normals calculation
```

Definition at line 19 of file geotiff_processor.py.

3.12.2 Constructor & Destructor Documentation

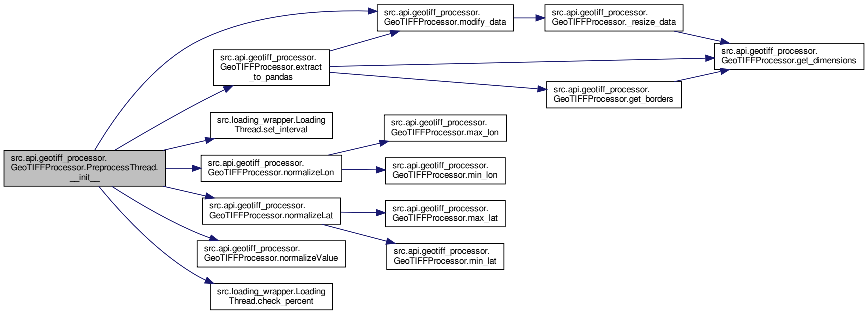
3.12.2.1 __init__()

```
def src.api.geotiff_processor.GeoTIFFProcessor.PreprocessThread.__init__ (
              self,
              proc,
              normalize = False,
              parent = None,
             * args,
             ** kwargs )
```

```
:param proc: an instance of GeoTIFFProcessor
:param normalize: toggle normalization
:param parent: a parent for QThread
:param *args: Passed to GeoTIFFProcessor.modify_data
:param **kwargs: Passed to GeoTIFFProcessor.modify_data
```

Definition at line 27 of file geotiff_processor.py.

Here is the call graph for this function:


src.api.geotiff_processor.
GeoTIFFProcessor.PreprocessThread.
__init__
src.api.geotiff_processor.
GeoTIFFProcessor.modify_data
src.api.geotiff_processor.
GeoTIFFProcessor._resize_data
src.api.geotiff_processor.
GeoTIFFProcessor.extract
_to_pandas
src.api.geotiff_processor.
GeoTIFFProcessor.get_dimensions
src.api.geotiff_processor.
GeoTIFFProcessor.get_borders
src.loading_wrapper.Loading
Thread.set_interval
src.api.geotiff_processor.
GeoTIFFProcessor.max_lon
src.api.geotiff_processor.
GeoTIFFProcessor.normalizeLon
src.api.geotiff_processor.
GeoTIFFProcessor.min_lon
src.api.geotiff_processor.
GeoTIFFProcessor.normalizeLat
src.api.geotiff_processor.
GeoTIFFProcessor.max_lat
src.api.geotiff_processor.
GeoTIFFProcessor.normalizeValue
src.api.geotiff_processor.
GeoTIFFProcessor.min_lat
src.loading_wrapper.Loading
Thread.check_percent

Index