Chapter 1

Camera

A Roassal view and therefore a Trachel canvas is seen through a camera. The direction of the camera is always perpendicular to the view, else Roassal would be a 3D engine and not a 2D engine. Camera may (i) go up and down, which is perceived by zomming out and in and (ii) be horizontally translated, which is perceived as scrolling the whole scene.

1.1 Operations

A camera is accessible from the canvas and is described by the class TRCamera. This class defines the following methods:

- translateTo: to translate the camera to a given point. This method takes a point as argument.
- translateBy: for a step-wise translation. This method takes also a point as argument.
- scale: for zooming in and out. This method takes a float number as argument. An argument greater than 1.0 is perceived as a zoom in, and less than 1.0 as a zoom in.

1.2 Initialization

When a view is opened, the camera is positioned at the center of it. This is the defaut behavior of a camera. This behavior may be changed in case you need to position the camera at a different location. Consider the following example:

```
v := RTView new.
map := RTOSM new.
e := map element.

v add: e.

paris := 48.8567 @ 2.3508.

v @ RTDraggableView.

v canvas camera translateTo: (map latLonToRoassal: paris).
v canvas camera noInitializationWhenOpen.
v
```

This short script use the OpenStreeMap integration to move the camera above Paris. Coordinate of Paris are here specified in latitude and longitude. The call to latLonToRoassal: converts the point argument, specified as latitude @ longitude, into the Roassal space. The camera is then moved to the computed point.