Scale

A few important steps must be done after executing the scale transformation:

- the pivot must change itself according to the scale factor
- the object must resize itself according to the scale factor
- the scaled object will have the same base point (not the same pivot)
- the scale factor will remain 1: other scale operations will refer to the scaled shape (not to the original shape)

How is is done:

- the fist step is to extract the rotation matrix from the transformation matrix. The scale values are saved in this matrix. Since the base values must remain the same, it is important to extract only the rotation values.

- moving on, we will extract the pivot coordinates. In order to adjust the pivot coordinates, we will add a vector to the base point. This vector represents the scaled difference between the pivot coordinates and the base point coordinates.

- in order to scale the difference and the rotation matrix, a scale vector will be used. By using a scale vector, we can even scale the shape with different values on each dimension. However, the scale made from the grid view is uniform (same scale factor in all dimensions). This is why the scale vector has the same value for all dimensions.

- the next step is pretty easy: apply the scale transformation for the rotation matrix and difference vector.

- the last thing to do is to change the transformation matrix with the calculated values.

A general scale method was made, but some problems appear when the scale operation doesn't have the same value in all dimensions: the object transformation matrix is calculated correctly, but the shape is drawn wrong. It's most probably an OpenCascade bug, but it's not worth to investigate too much because the scale operation with different factors is not needed at the moment. In case this operation will be of greater interest for the users, we can perform a hack to make it work.