Module Integration Test (30.05 - 13.06)

Oliver Hennhöfer

[01.06.2018]

Adding self-sufficient creation of a table in the connected Database:

After connecting to a local database and exporting data from the software tool into this database; a new table named *toolobjects_db* will be created automatically (if not existing). So far you had to manually create a table in the database with the exact same name to make the exportation work.

createTable()

[02.06.2018]

Enhancing the live-display of geometries that are currently drawn:

The way, the geometries - that are currently drawn - were displayed, leaded to the problem that the last object (of each type of object) drawn, appeared twice in the table of the .csv file and the database after exporting. The problem appeared because the process of displaying used the lists of the ObjectManager to store the different states of the objects currently drawn, which were deleted after moving the mouse and wrote again into the list for the current position of the mouse. After clicking and therefore confirming the geometry (for example a line) the geometry was stored in the ObjectManager without iterating once again through the *MouseMoved* command for deleting the last state of displaying. The problem was solved by creating separate lists (for the sake of simplicity directly in the DrawPanel) whose contents were also affected by the *repaint()*-method and therefore shown on the canvas - and adding the elements of the geometry to this list. After confirming a geometry, the list will be reset and the geometry will be drawn by the coordinates stored in the corresponding list in the ObjectManager.

storeDrawingLineElements(ToolLine line)

storeDrawingTriangleElements(ToolTriangle triangle)

storeDrawingRectangleElements(ToolRectangle rectangle)

clearDrawingElements()

[05.06.2018]

Adding live display for the 'change'- and 'movement'-Tool

So far, the result of using one of these tools was first displayed after clicking again on the desired position. Now you can watch the current state of change/move operation in real-time.