

Readme

Advanced Clip, Cursor Clip and Symbology Toolbox for ArcGIS Pro

##SYNOPSIS

The Advanced Clip, Cursor Clip and Symbology Toolbox is a set of three tools designed to provide extended and at the same time easy-to-use opportunities for user specific data selection and clipping as well as assigning predefined symbology styles to the results (or other feature layers).

Tool Advanced Clip:

The **Advanced Clip** tool extracts input features that overlay clip features. For user specific queries, optional SQL expressions are available that can be used to add, update, or remove a selection based on an attribute query. The result is then automatically added to the Contents Pane and displayed in the Map.

Tool AdvancedMaxCursorClip:

The **AdvancedMaxCursorClip** again is used for clipping but features a divergent functionality: While the selection of features to clip and be used for clipping is the same, the user then selects the feature attribute which furtheron is used by the tool to automatically return and clip to the attributes entity disposing the maximum value, eg. the district with the largest area. The results is automatically added to the Contents Pane and displayed in the Map.

Tool Apply Predefined Symbology:

The **Apply Predefined Symbology** tool provides automated, predefined symbology styles for an input point, polyline and/or polygon feature layer respectively. Although using only one input field, the tool automatically evaluates the shape type of the input and determines the particular symbology to be assigned. The tool is implemented in a way that the original symbology is replaced by the predefined style.

##MOTIVATION

The combination of these tools in one toolset simply derives from experienced shortcomings of the provided preinstalled tools when selections and clippings are required for multiple analysis steps and the results shall be presented in a corporate design.

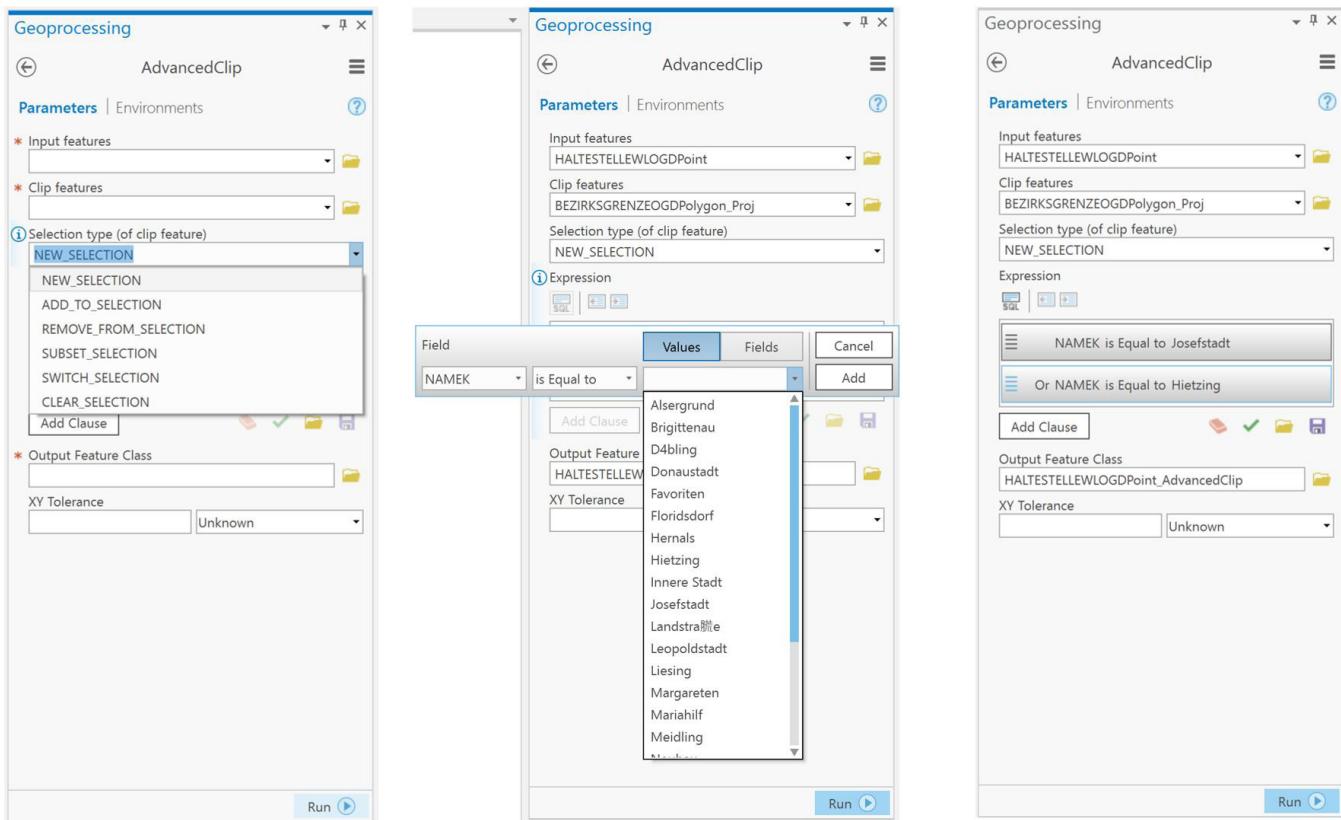
The toolbox contains 3 tools. The first two follow a similar aim, but are code differently to experiment with search cursor.

The **Apply Predefined Symbology** complements these two, to apply a uniform style to the results. It aims to automatically assign a preset style to input feature layers.

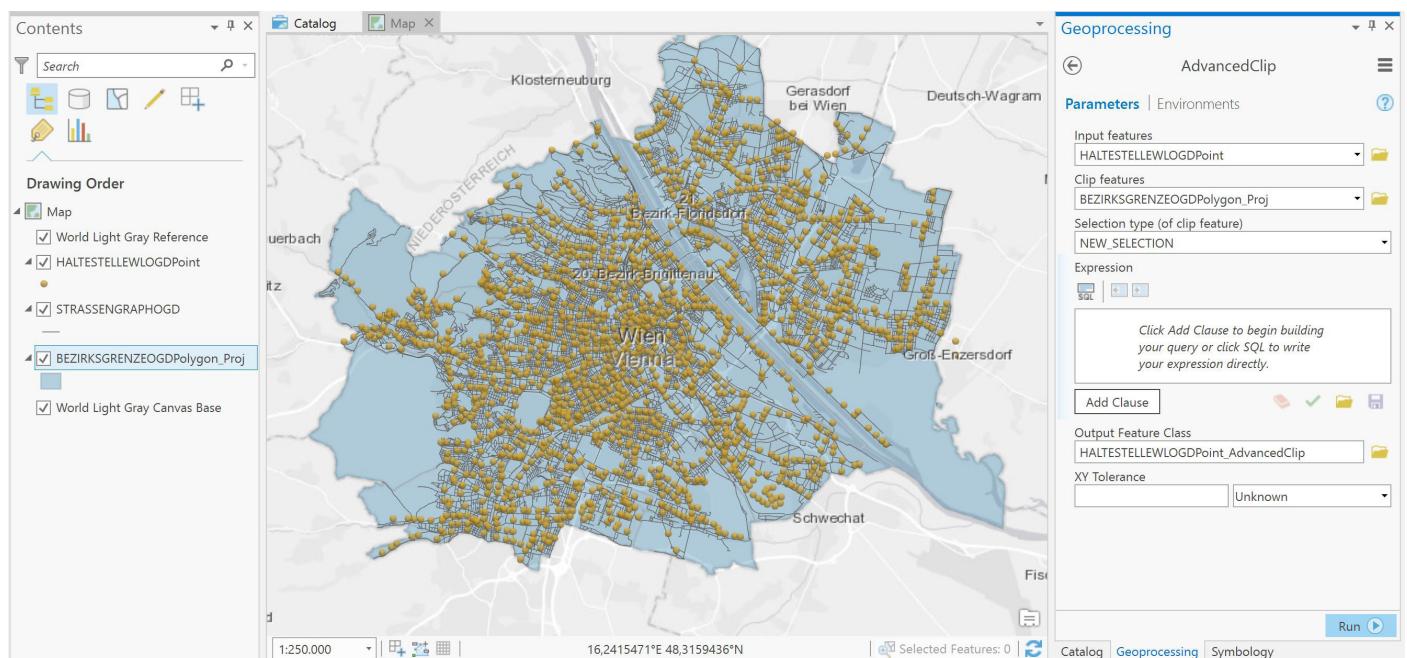
APPLICATION AND CODE EXAMPLES

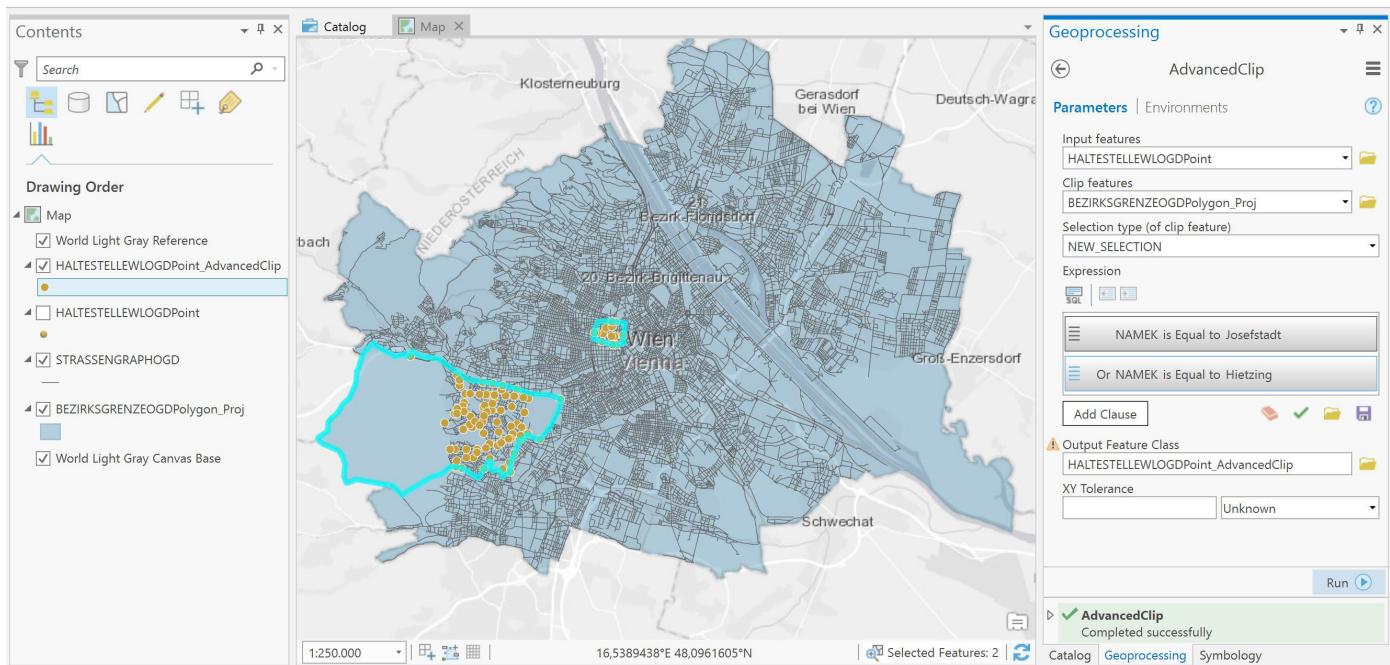
AdvancedClip

The AdvancedClip is a combination of the Select Layer By Attribute and the Clip tool. Therefore, all functions of both tools can be used. The selection affects the clip feature only.



The input feature and clip feature are generalized and works on all feature types. As for the clip feature, a different selection type can be defined. In the following example, only those stops of public transit will be clipped which are found within the two selected districts Josefstadt or Hietzing of the capital city Vienna.





As can be seen, only the district "Josefstadt" and „Hietzing“ are selected and the stops of public transit within this districts are clipped and create a new output feature class which is automatically added to the Contents Pane.

Code for the AdvancedClip tool:

```
# Import system modules
import arcpy

# Set local variables
in_features = arcpy.GetParameterAsText(0) #features to be clipped
clip_features = arcpy.GetParameterAsText(1) #The features used to clip the input features
selection_type = arcpy.GetParameterAsText(2) #Determines the selection type
sql_expression = arcpy.GetParameterAsText(3) #SQL expression, to select specific features by attributes
out_feature_class = arcpy.GetParameterAsText(4) #defines name and path where the output feature class should be created
tol_value = arcpy.GetParameterAsText(5) #defines the xy tolerance

arcpy.SelectLayerByAttribute_management(clip_features, selection_type, sql_expression) #implementation of the selection
xy_tolerance = tol_value

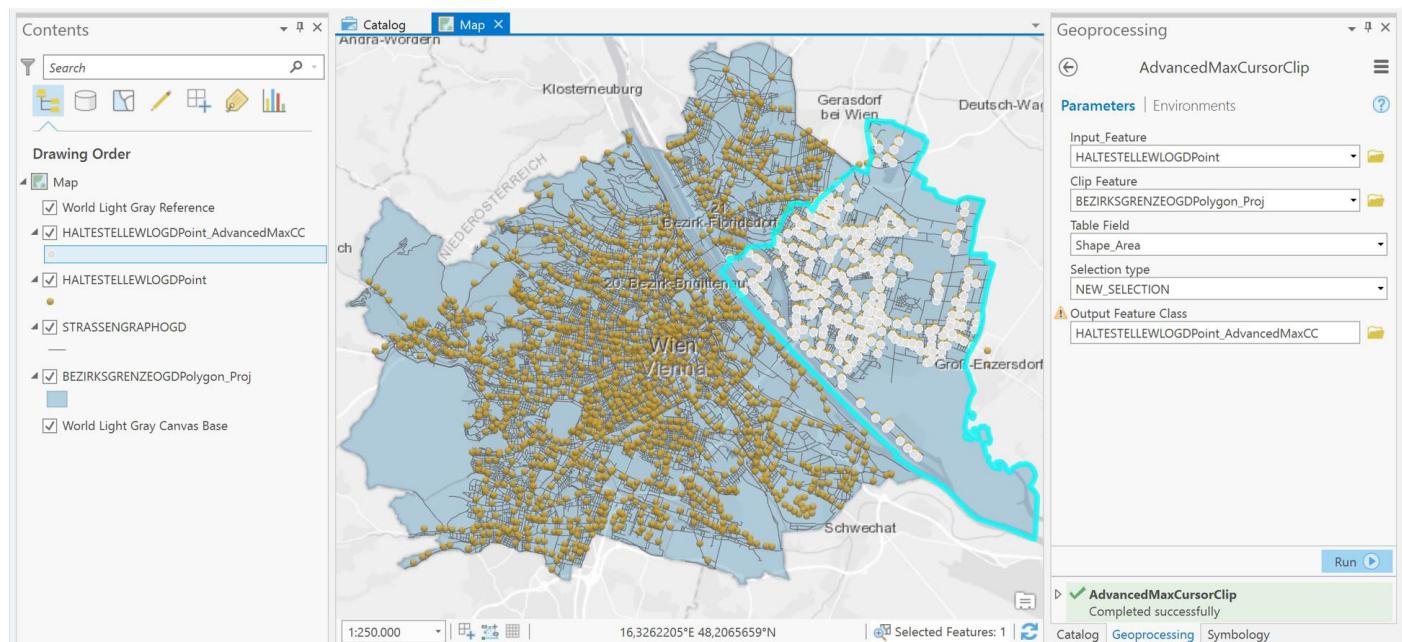
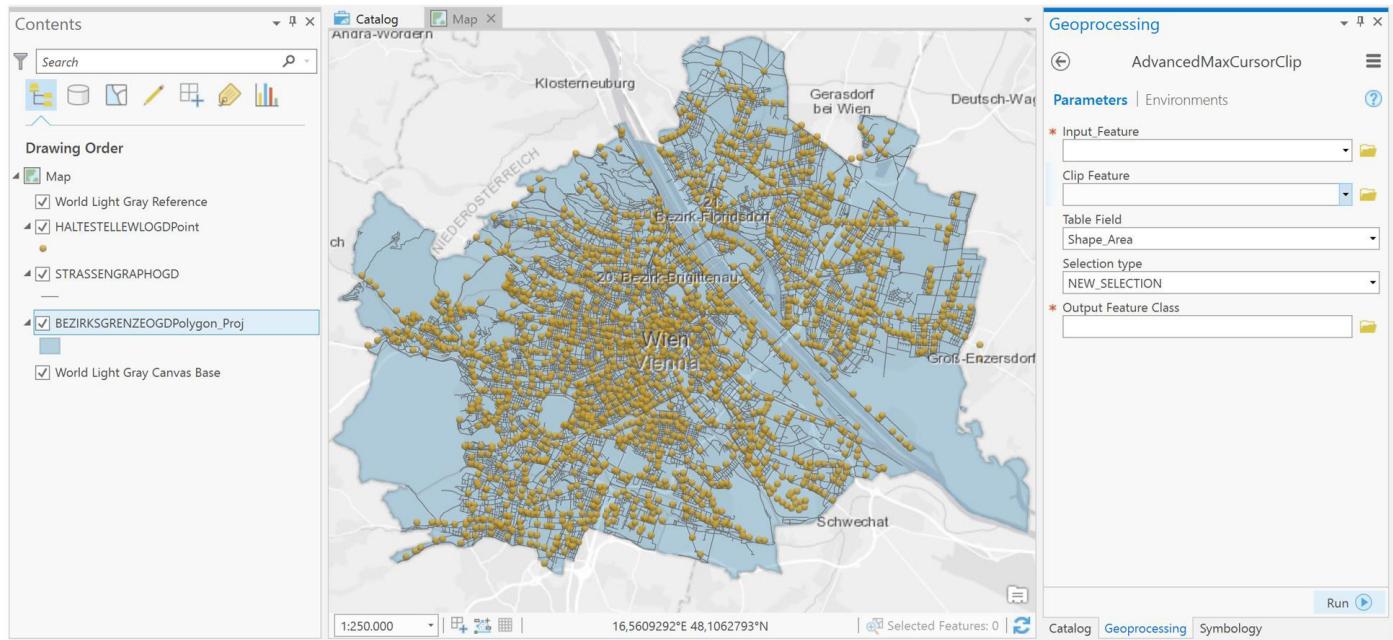
arcpy.Clip_analysis(in_features, clip_features, out_feature_class, xy_tolerance) #implementation of the clipping operation

#Prints the expression as message in ArcGIS Pro, when running the tool
if sql_expression == "":
    arcpy.AddMessage("Finished clipping without using an expression")

else:
    arcpy.AddMessage("Finished clipping using the expression(s): " + sql_expression)
```

AdvancedMaxCursorClip

When using the **AdvancedMaxCursorClip**, the user again defines the feature to be clipped and the feature that is used for clipping (setting the extent). But in contrast to the **AdvancedClip** tool, there is no SQL expression to select attributes. Instead, a Table Field offers a selection of attributes to set for the clipping and a drop-down menu offers the selection types. The tool then automatically clips to the entity disposing the maximum value and clips on basis of the selection of feature attribute.



Code for the AdvancedMaxCursorClip tool:

```
import arcpy

#define inputs: feature layer and the field in the table that should be selected
#set variables
input = arcpy.GetParameterAsText (0) #features to be clipped
clip_feature = arcpy.GetParameterAsText (1) #The features used to clip the input features
table_field = arcpy.GetParameterAsText (2) #table field of the clip feature can be chosen
selection_type = arcpy.GetParameterAsText (3) #Determines the selection type
output_feature_class = arcpy.GetParameterAsText (4) #output feature class to be created

# search cursor that looks for the maximum value in the field of the table of the feature layer
with arcpy.da.SearchCursor(clip_feature, table_field) as cursor:
    max(cursor)

maxvalue = max(cursor)

maxvalue_str = ''.join(map(str, maxvalue)) #converts the list value into string
```

```

#get the derived maximum value as message in Arcgis Pro
arcpy.AddMessage("Max value of " + table_field + " = " + maxvalue_str)

# Create field name with the proper delimiters
whereclause = """{} = {}""".format(arcpy.AddFieldDelimiters(clip_feature, table_field), maxvalue_str)

#implementation of the selection with error handling
try:
    arcpy.SelectLayerByAttribute_management(clip_feature, selection_type, whereclause)

except arcpy.ExecuteError:
    arcpy.AddWarning("Executed without selection! The table field you have chosen is empty.")

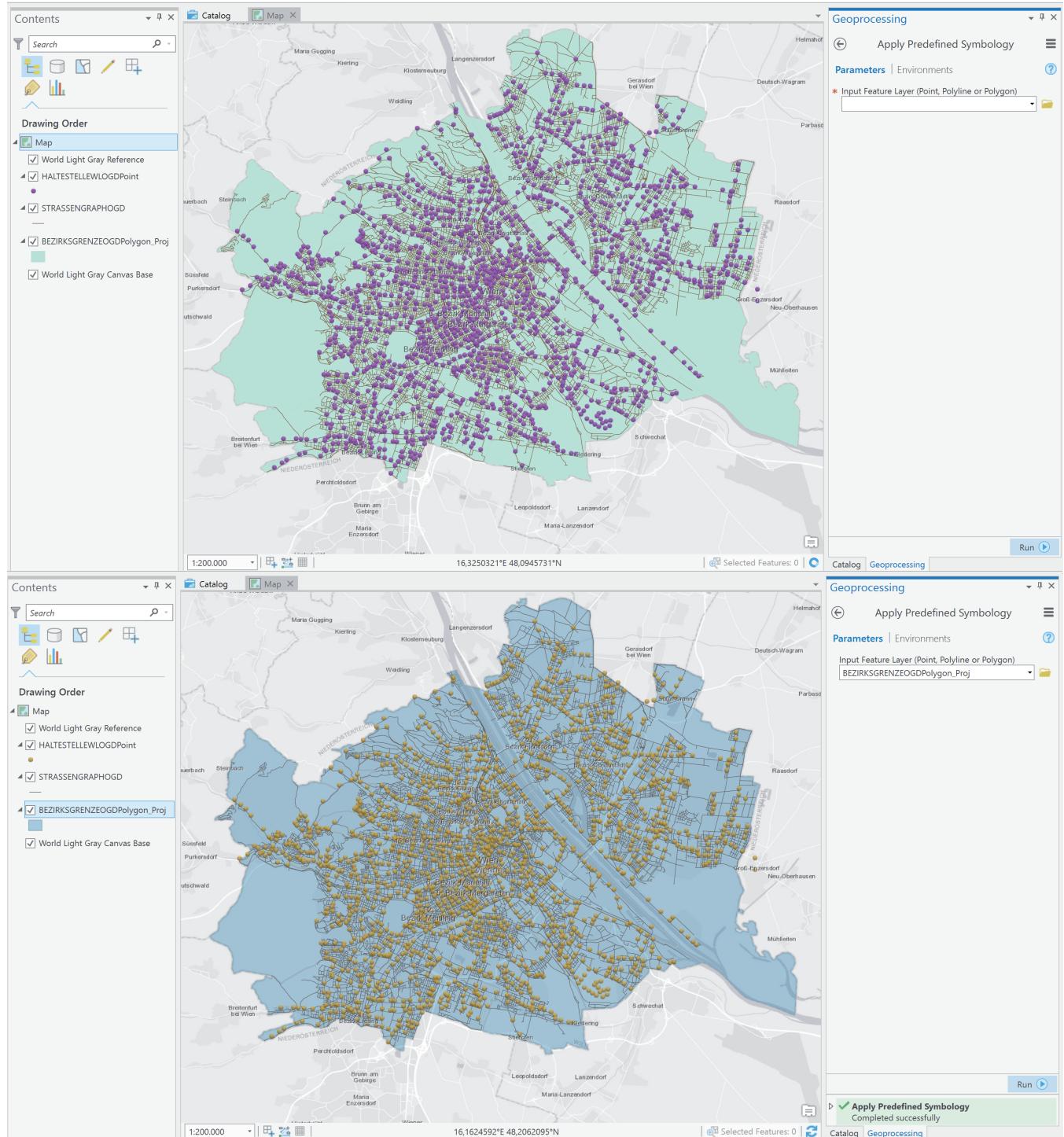
xy_tolerance = ""

#implementation of the clipping
arcpy.Clip_analysis(input, clip_feature, output_feature_class, xy_tolerance)

```

Apply Predefined Symbology

Running the **Apply Predefined Symbology** tool for point, polyline and/or polygon feature layers automatically assigns the preset styles to the respective shape type by overwriting its original style parameters.



Code example for assigning the predefined style to a point feature layer:

```
# import system modules
import arcpy

shape = arcpy.GetParameterAsText(0)

# Describe evaluates the shape type (polyline, polygon, point) of the users input data and determines which symbology parameters are appropriate
desc = arcpy.Describe(shape)

#####
# If user input is a Point Feature Layer, the subsequent symbology settings will be applied:
if desc.shapeType == "Point":
    p = arcpy.mp.ArcGISProject('current')           # goes to mp. library and accesses the current project
    m = p.listMaps('Map')[0]                        # gets the first map

    lyr = m.listLayers(shape)[0]                     # gets the first layer
    sym = lyr.symbology

    sym.renderer.symbol.applySymbolFromGallery("Circle 3") # applies the symbol "Circle 3" from the Format Point Symbol Gallery
    sym.renderer.symbol.color = {'RGB' : [200, 161, 60, 80]} # accesses the symbology of the above selected layer: golden color for circles (opacity 80 %)
    sym.renderer.symbol.size = 5                      # symbol size is set to 5pt
    lyr.symbology = sym

    sym.renderer.symbol                           # gets renderer when symbol variable is set
```

Hints for user-specific adaptions of the tool:

Within the frame of this tool, other shape types than point, polyline and polygon are not supported (yet), but extensions or adoptions can easily be implemented. Within the provided python code, the shape-type related sections are quite self-explaining and therefore open to effortless modifications: alterations in the style settings or supplements of additional feature types.

Within the „Parameters“ tab of the tool properties in ArcGIS Pro, the desired additional feature types can be set in „Filter“.

INSTALLATION

Software environment for the toolbox creation was ArcGIS Pro version 2.0.0 with Python version 3.5.3, using Pyzo for coding.

After downloading the Toolbox it recommended to establish a folder connection in Arcgis Pro to the path where the Toolbox is saved. In the catalog window browse to the folder that contains the Toolbox. Alternatively add the Toolbox directly to the current project under: Insert --> Toolbox --> add Toolbox in the menu bar on the top. Again browse to the folder and the toolbox appears in the catalog. Now the tools inside the toolbox can be accessed and used.

REFERENCES

Apart from the documents provided in the course, the following sources were used for code generation and settings in ArcGIS Pro in the frame of this project:

ArcGIS Pro: Tool Reference - Clip. Available under: <http://pro.arcgis.com/en/pro-app/tool-reference/analysis/clip.htm>

ArcGIS Pro: Python - Describing data. Available under: <http://pro.arcgis.com/en/pro-app/arcpy/get-started/describing-data.htm>

ArcGIS Pro: Python - Specifying a query in Python. Available under: <https://pro.arcgis.com/en/pro-app/arcpy/get-started/specifying-a-query.htm>

ArcGIS Pro: Tool Reference - Select Layer By Attribute. Available under: <http://pro.arcgis.com/en/pro-app/tool-reference/data-management/select-layer-by-attribute.htm>

Esri Events: Map Automation in ArcGIS Pro. Available under: <https://www.youtube.com/watch?v=oOcckhgd3Sk>

Esri GeoNet: Acpy „if“, „elif“, and „else“ statement (ArcGIS Pro). Available under: <https://community.esri.com/thread/206158-acpy-if-elif-and-else-statement-arcgis-pro>

Stackoverflow: How to convert list to string [duplicate]. Available under: <https://stackoverflow.com/questions/5618878/how-to-convert-list-to-string>

CONTRIBUTORS

The toolbox and its components were developed by Simon Ecke, Eva Missoni and Philipp Straßer