Abstract—Geographic information systems belong to the group of applications that process spatial data. Some spatial analyses are performed repetitively in the same way for different areas or districts. Therefore, it is useful when the data processing can be run automatically by program extension designed especially for batch data processing. ArcGIS software offers the possibility to design the steps of data processing by data flow diagram in the graphic editor ModelBuilder. In some cases, the data flow diagram which is designed in ModelBuilder component is not sufficient for all require tasks. In such cases, it is possible to automatically convert data flow diagram into the Python language and supplement the program code with other program constructions and commands. Some practical examples are presented in this article. All Python scripts were developed as a part of several research projects at the Department of Geoinformatics at Palacký University in the Czech Republic. This article is a summary of experience with scripting in Python for ArcGIS geoprocessor.

Keywords - ArcGIS; cartography; dataflow; geoprocessor; GIS; scripting.

I. INTRODUCTION

Geographic information systems (GIS) are specific information systems that are oriented to process spatial data. Earth phenomena are represented by graphical data in a raster or vector format. Vector data are often joined with a lexical data in a relation database. The vector format uses three main forms of representation when describing reality: points, lines and polygon.

Nowadays, GIS are becoming more and more popular among municipal and territorial organizations, environmental experts for their ability to manage the spatial data [1]. Spatial decision making tasks, simulations of spatial evolution and tasks that involve comparing different data motivate growing interest in GIS. Common customers of GIS are usually familiar with interactive processing of data.

Overlay analyses are among the most often ones. These may involve clipping of a polygon or lines, creating buffers or subtraction of polygons (symmetrical difference). Operations such as "Union", "Identity" and "Identify" produce new spatial data with a combination of attribute data from source data (feature classes). GIS software are also helpful when creating cartographic output – a map. Functions for map creation are also automatically put together as sequence. These are for example operation for labeling point, line and polygon in a map and creating masking polygons for these labels. Another of the cartographic functions is a set of generalization operations. The generalization remakes the shapes of line and polygons for follow-up use in a map with smaller scale. Operation with raster data is called "map algebra". It is possible to add a raster data, reclassify a raster data etc. Raster formats are also used for representation of digital elevation model.

II. CURRENT STATE OF DATA FLOW PROCESSING IN ARCGIS

Many experienced users of GIS can construct data flow diagrams for repetitive processing of data in the graphic editor. Some GIS software have a program module extension for designing the batch data processing graphically. Comparing of visual languages in GIS is further described in article "Visual programming language in geographic information systems" [2]. Software AutoCAD Map 3D by Autodesk company offers a graphic editor called the Workflow Designer. In ArcGIS, from ESRI, the graphic editor is called ModelBuilder. Documentation explains that "Models are workflows that string together sequences of geoprocessing tools, feeding the output of one tool into another tool as input" [3]. The "geoprocessing tools" is a collective name for all spatial operations that can be practically used for processing spatial data. The main programming module that contains all functions is called "Geoprocessor".

The data flow model in ModelBuilder can be designed easily by "drag and dropping" any analytic spatial function and data from the interface of ArcToolBox. More than 500 tools can be utilised, depending on ArcGIS license level. Spatial functions are represented by yellow boxes while data is represented by blue or green ellipses (ovals). Ellipses represent the input and output data of spatial functions. An example of simple data flow model is on Fig. 1.

The basic function of the above mentioned example (Fig. 1) is to change the number and the shape of input polygons. Firstly, this cartographic model aggregates input polygons to one polygon. Input polygons represent the areas where infectious diseases were detected during last three years (Fig. 2.). Cartographer needs only one merged polygon for the whole period of time. Moreover, the edge of polygon for map expression must be smoother than original edges. These operations are called “generalization” in cartography. That task is realized by two polygon operation from Cartography toolset. Operation Simplify and Smooth were applied on one input aggregate polygon.

Programming Language Python for Data Processing

Ing. Zdena Dobesova, Ph.D.
Department of Geoinformatics, Faculty of Science
Palacký University in Olomouc
Olomouc, Czech Republic
zdena.dobesova@upol.cz
In general, any other input data can be processed by this model. The model process input data as a batch. User receives only one final polygon that is appropriate for a map. Fig. 2 shows the change of three input polygons into a resultant smooth polygon for final map.

A script in Python language for ArcGIS can be created in two ways. The first way is to export existing model from ModelBuilder to the Python script. Export is supported function of the graphical editor. The second way is to directly create a script in an integrated development environment. The first way is simpler than the second especially for beginners.

A. Model Exporting

Example of exported Python script of model from Fig. 1 is on Fig. 2. Automatic export creates a header with commentary lines about data, time etc. The export tool automatically imports module arcpy that contains library of all geoprocessing methods. Subsequently, automatic export creates commands for input parameters of scripts (name and location of input and output feature class). Finally, the geoprocessing method with parameters was added:

- arcpy.AggregatePolygons_cartography()
- arcpy.SimplifyPolygon_cartography() and
- arcpy.SmoothPolygon_cartography().

The advantage is that the script has full functionality without any change. It can now serve as a base for other extensions in Python.
B. Script creation in IDLE

IDLE Python is an integrated development environment for writing scripts. The advantages of IDLE are syntax highlighting, auto completion of commands, smart indent and integrated debugging functions (stepping, breakpoints). Partial results of scripting are displayed in Interactive window of IDLE. PythonWin is another example of IDLE for scripting. Editor window with the script is on Fig. 3.

Some methods from Geoprocessor are only accessible by scripting, e.g. enumeration methods. Enumeration methods return Python list of all file names or raster data sets in a directory on hard disk. This is one of the reasons for writing scripts directly in IDLE.

IV. IMPLEMENTING PYTHON SCRIPTS

Both a model from ModelBuilder and a Python script can be loaded as a custom toolbox beside the system toolbox in ArcToolBox. Custom toolbox has a unique name. Several models or scripts can be included together in one toolbox. The final model from the ModelBuilder is stored and encapsulated in a file with file extension TBX. The Python scripts, however, are stored separately in text files with PY extension. The Toolbox only references to them via a link.

Fig. 4 depicts ArcToolBox with system toolboxes. The last toolbox on the bottom is a custom toolbox WalkAbility Index. This toolbox is used for calculating four partial and one final index that described physical motion (walking) of people in a town district. This Python script was created under the IPEN scientific project [6].

A user interface is used when setting input parameters of a script. Help notes with instruction can be joined to every parameter (Fig. 5). The instructions explain the meaning of parameters or offer suitable values.

V. MODIFYING SCRIPTS BY PYTHON

ModelBuilder may not be sufficient in all required tasks. The program commands and function of Python can be utilized in scripts. During research about digital elevation models (DEM), over 480 testing datasets of DEMs were created automatically. It was necessary to read input parameters of various interpolation methods from text files [7]. Python has excellent basic function to open and read data from text files.

The number of letters is limited to 12 in the name of DEM in ArcGIS. Next task was to automatically create an abbreviation for the file name of DEM. String is maintained as Python list. String functions such as replace were used. The name of abbreviation was created by several program commands that put two letters together for interpolation method, four letters for numeric parameters of interpolation methods and two letters for earth area. Every two letters are separated by underscore character. The name of surface “ST_05_20_SR” means Spline with Tension method, weight 0.5 and number of points was 20 for search radius. SR is the name of explored area Sadska Rovina. Fragment of Python script is on Fig. 6.
ModelBuilder extends a base functionality of ArcGIS in batch processing. Final refining of exported model in Python can utilize a wide set of geoprocessing methods. The limitation is sometimes the time. The Python interprets every command in scripts. Therefore, the run of scripts is slower than run of compiled program. The basic functionality of ArcGIS can be also extended in another way by using four application interfaces (based on ArcObjects): COM, .NET, Java, and C++. Example of practical extension based on API and ArcObject was described in article "Program Extension for Diagram Maps" [8].

VI. CONCLUSION

Creation of own Python script for ArcGIS is notably useful for any advanced user. Basic functions of ArcGIS program can be extended by model designed in ModelBuilder. Models are aimed mainly to batch processing of a large amount of data. Additionally, a model exported to Python language can serve as a potent starting point for final sophisticated program. Scripting in Python has some requirement on user knowledge in programming skills, understanding of individual geoprocessor methods is also necessary. On the other hand, the Python has very clear syntax and offers an integrated development environment. The three presented practical examples from the field of cartography, physical moving assessment and interpolation of digital elevation model demonstrate the usability of scripting in ArcGIS software. Scripting in Python promises to be extremely helpful for future users of ArcGIS. Furthermore, scripts reduce manual work in processing a large amount of data.

ACKNOWLEDGMENT

The research was supported by a project of the Czech Science Foundation No. 205/09/1159 „Intelligent system for interactive support of thematic map creation“, which was awarded from 2009 to 2011.

REFERENCES