26

IMPLEMENTATION OF A KAOLIN DEPOSIT MODEL IN VISUAL STUDIO 2015

26.1 INTRODUCTION

The article describes a part of the project TA CR TE02000029 - Competence Centre for Effective and Ecological Mining of Mineral Resources (CEEMIR), whose main objective is to review selected non-energy resources, which are among the critical commodity EU [1]. One part of the project's Work Package WP 4 - Spatial modeling of mineral deposits, which deals with the transfer of selected deposits into digital models with appropriate mathematical methods based on the study and a review of data from archival materials. One of the selected deposits is a kaolin deposit in the surroundings of the municipality of Jimlíkov it the Karlovy Vary region.

This article is based on article Staněk, F., Jarošová, M., Staňková, J.: Dynamic model of a kaolin deposit, exactly on methodological procedure creation and visualization of 3D model of the deposit kaolin (subchapters):

2.6 3D visualization of the input data for the kaolin deposit in the Voxler environment, creation of 3D grids of the content of technological parameters, and export of the 2D grids in individual horizons in the *grd* Surfer format (program *Kaolin_A*).

2.7 Categorization of the blocks of reserves in 2D grids (in individual horizons) based of both the grids of technological parameters (exported using the program *Kaolin_A*) and predefined parameters for the categories of reserves, transformation categories of the blocks of reserves into a 3D grid and estimation of the reserves of the deposit (program *Kaolin_Viz*).

2.8 2D visualization of horizontal sections in the Surfer software environment (program *Kaolin_Viz*).

2.9 2D visualization of the series of vertical sections in the Surfer software environment (program *Kaolin_Viz*).

2.10 3D visualization of categories of blocks of reserves in the Voxler software environment (program *Kaolin_Viz*).

The examples of program realization *Kaolin_A* and *Kaolin_Viz* in Visual Studio 2015 [2] are in the article, mainly using Voxler automation object model [3] and Surfer automation object model [4] made by the Golden Software Company.

26.2 WORKING WITH OBJECTS VOXLER A SURFER IN VISUAL STUDIA 2015

Voxler and Surfer can be called from any automation-compatible programming languages such as VB.NET. It is this approach which was taken for the implementation of programs *Kaolin_A* and *Kaolin_Viz* in Visual Studio 2015. For the usage of applications Voxler and Surfer in this environment, it is necessary for the project program to add a reference to that application.

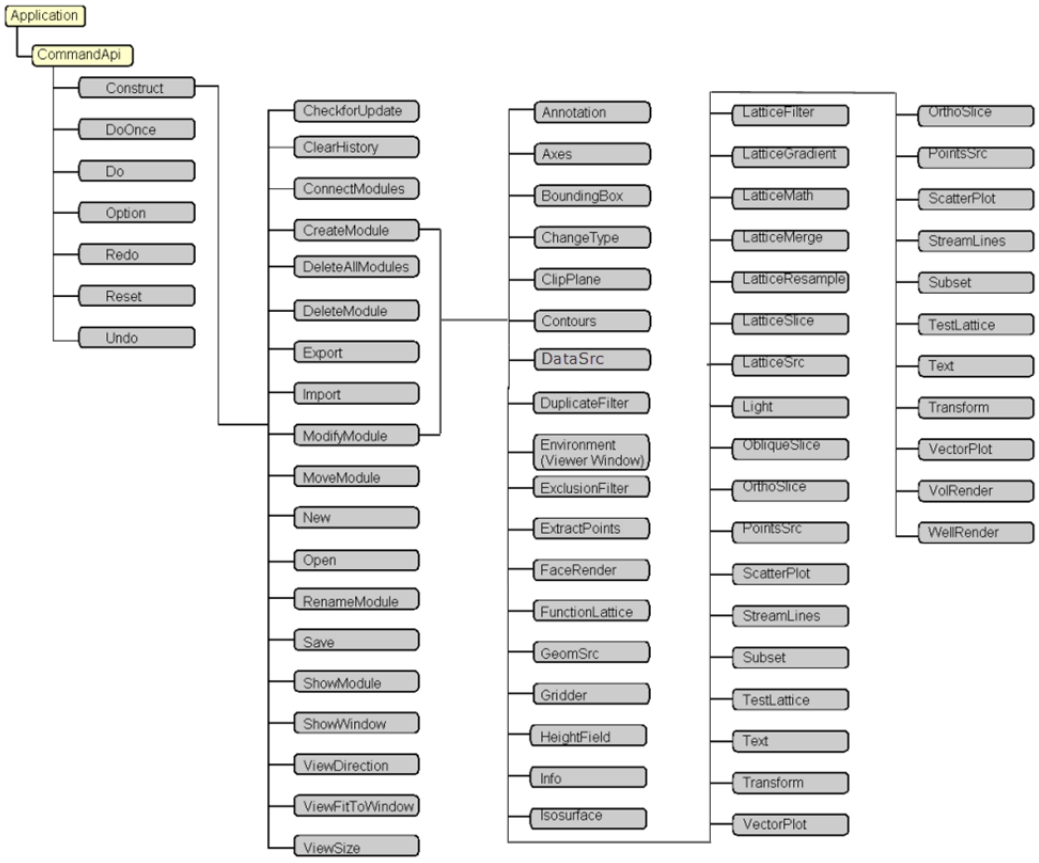

Fig. 26.1 Voxler automation model [3], objects (the yellow boxes), methods and properties (the gray boxes)

In the Fig. 26.1 is automation model of hierarchy Voxler objects (Application and CommandApi), methods and properties. The model displays a flow-path to create the type of module desired using automation and shows you which objects provide access to other objects in the hierarchy. The Application object is at the top of the hierarchy and all objects are directly accessible from the Application object. To access many objects you must traverse from the Application object through one or more layers of sub-objects. The CommandApi object contains all of the properties of the various modules in the Voxler program. CommandApi refers to the accessing the commands from the Application programming

interface. Using the CommandApi object requires accessing the property with the Construct method, specifying any settings with the Option method, and making the action with the Do or DoOnce method.

In the Fig. 26.2 is Surfer automation object model. This chart shows objects that provide access to other objects. Surfer groups most objects in collections. Collection objects are containers for groups of related objects. Although these collections contain different types of data, they can be processed using similar techniques. Non-container objects represent a specific part of Surfer. Several objects shown in the Fig. 26.2 share common features (for example PlotDocument provide SaveAs, Activate, and Close methods). The online Surfer help is the complete reference for all of the Surfer automation objects, their properties, and their methods.

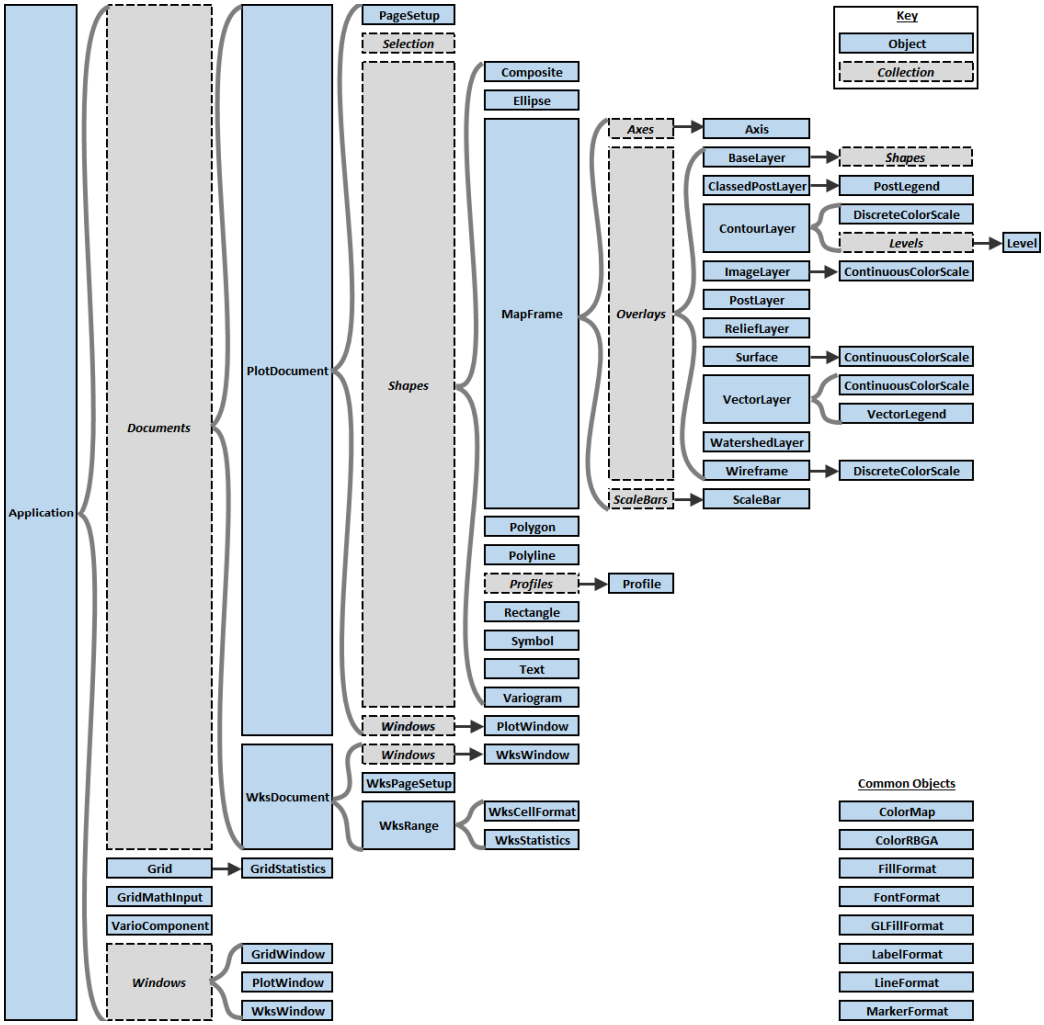

Fig. 26.2 Surfer automation object model [4], collection objects (the gray boxes) and objects (the blue boxes)

26.3 PROGRAM KAOLIN_A

After starting the program *Kaolin_A* it is necessary to enter an initialization file that sets the appropriate input parameters necessary to run the program (Fig. 26.3). These input parameters can possibly be modified by the user after the program starts.

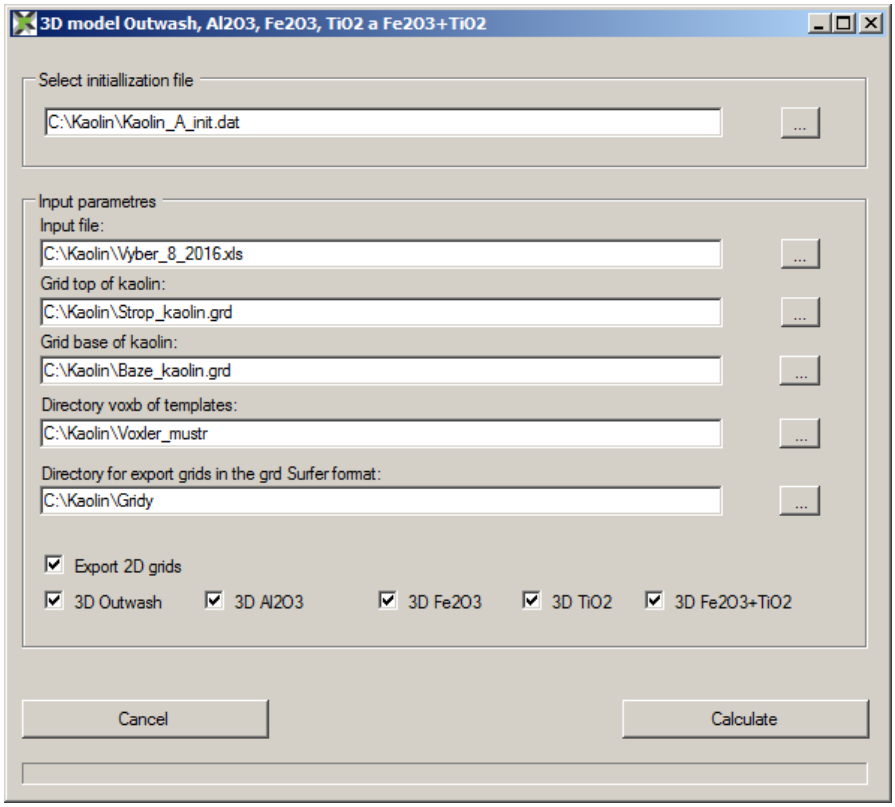

Fig. 26.3 Window program of Kaolin_A for setting the calculate parameters

The next example from step 2.6 of methodology (see Introduction) contents a part of code language VB.NET (Fig. 26.4) using Voxler automation model for gradual creation 3D models of five technological parameters kaolin whose are crucial for determining categories of blocks of reserves: Contents of the Outwash, Al_2O_3, Fe_2O_3, TiO_2 and Fe_2O_3+TiO_2 (the order of parameters presents a variable Cislo_Par in the code). This example implements import of input data for WellData modul (Fig. 26.5), setting their parameters and creating of connections between input data and WellData modul.

The example of one of the outputs of the program *Kaolin_A* is in the Fig. 26.5 – 3D visualization of the kaolin outwash content. In the left panel of this figure a structure of individual Voxler modules generated by *Kaolin_A* is shown.

26. 4 PROGRAM KAOLIN_VIZ

The program *Kaolin_Viz* implements steps 2.7 to 2.10 of the methodological procedure (see Introduction). After its launch, it is necessary to specify an initialization file that sets the appropriate input parameters necessary to run the program (Fig. 26.6). These input parameters can possibly be user modified after the program starts.

```
Dim VoxlerApp(1 To 5) As Object, CommandApi As Object
…
For Cislo_Par = 1 To 5
…
  VoxlerApp(Cislo_Par) = CreateObject("Voxler.Application")
   'Make Voxler visible
   VoxlerApp(Cislo_Par).Visible = True
   'Access CommandApi
   CommandApi = VoxlerApp(Cislo_Par).CommandApi
   'Open an existing file
   CommandApi.Construct("Open")
   Select Case Cislo_Par
   Case 1
     CommandApi.Option("Path", Mustr_vyplav)
   Case 2
     CommandApi.Option("Path", Mustr_Al2O3)
   Case 3
     CommandApi.Option("Path", Mustr_Fe2O3)
   Case 4
     CommandApi.Option("Path", Mustr_TiO2)
   Case 5
     CommandApi.Option("Path", Mustr_FeTi)
   End Select
   CommandApi.DoOnce
   'Import the Collars table
   CommandApi.Construct("Import")
   CommandApi.Option("Path", Vst_soubor)
   CommandApi.Option("Options", "Defaults = 1;Sheet=Collars;Filter=xls")
   CommandApi.Do
   CommandApi.Construct("ModifyModule")
   CommandApi.Option("Module", Vst_nazev + " - Collars")
   CommandApi.Option("GuiEnabled", "True")
   CommandApi.Option("OutputType", "1")
   CommandApi.Do
   CommandApi.Option("WellSheetType", "1")
   CommandApi.Do
   CommandApi.Option("WellColID", "1")
   CommandApi.Do
   CommandApi.Option("WellColTopX", "1")
   CommandApi.Do
   CommandApi.Option("WellColTopY", "2")
   CommandApi.Do
   CommandApi.Option("WellColTopZ", "3")
   CommandApi.Do
   CommandApi.Option("WellColAz", "4")
   CommandApi.Do
   CommandApi.Option("WellColVertType", "0")
   CommandApi.Do
   CommandApi.Option("WellColVertDip", "5")
   CommandApi.Do
   CommandApi.Option("WellColTotalDepth", "6")
   CommandApi.Do
   'Import the Samples
   CommandApi.Construct("Import")
   CommandApi.Option("Path", Vst_soubor)
   CommandApi.Option("Options",
"Defaults=1;Sheet=Samples;Filter=xls")
   CommandApi.Do
…
   'Connect the Weldata and data modules
   CommandApi.Construct("ConnectModules")
   CommandApi.Option("SourceModule",  Vst_nazev + " - Collars")
   CommandApi.Option("TargetModule", "WellData")
   CommandApi.Do
   CommandApi.Construct("ConnectModules")
   CommandApi.Option("SourceModule", Vst_nazev + "  - Samples")
   CommandApi.Option("TargetModule", "WellData")
   CommandApi.Option("TargetPort", "2")
   CommandApi.Do
…
Next Cislo_Par
```

Fig. 26.4 The example of program code using Voxler automation model

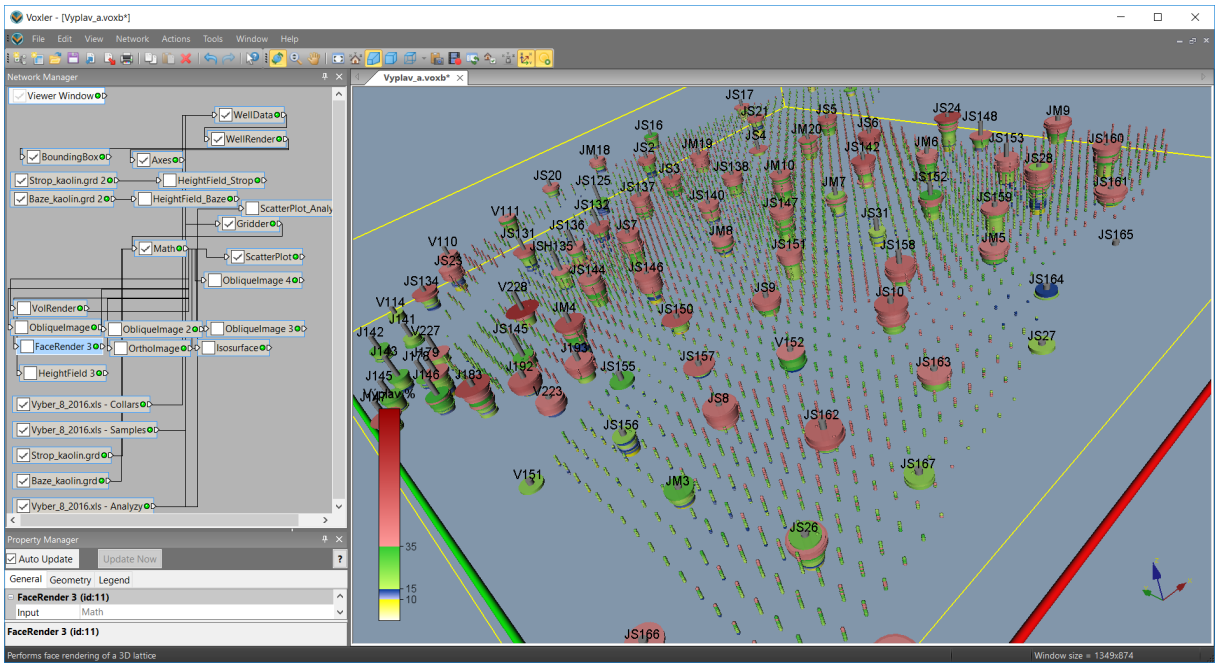

Fig. 26.5 3D visualization of the kaolin outwash content– proportion depicting of outwash content in the drill holes using WellRender and 3D grid using ScatterPlot

The next example from step 2.9 of methodology (see Introduction) contents a part of code language VB.NET (Fig. 26.7) using Surfer automation model for visualization of the series of vertical sections (Fig. 26.8). This example implements drawing Post map of data about individual blocks of reserves (object Bloky) and Base map of segments of selected drill holes for the section with samples of the K1, K2, …, NEG categories (objects Base_K1, Base_K2, …, Base_NEG) (see Fig. 26.8).

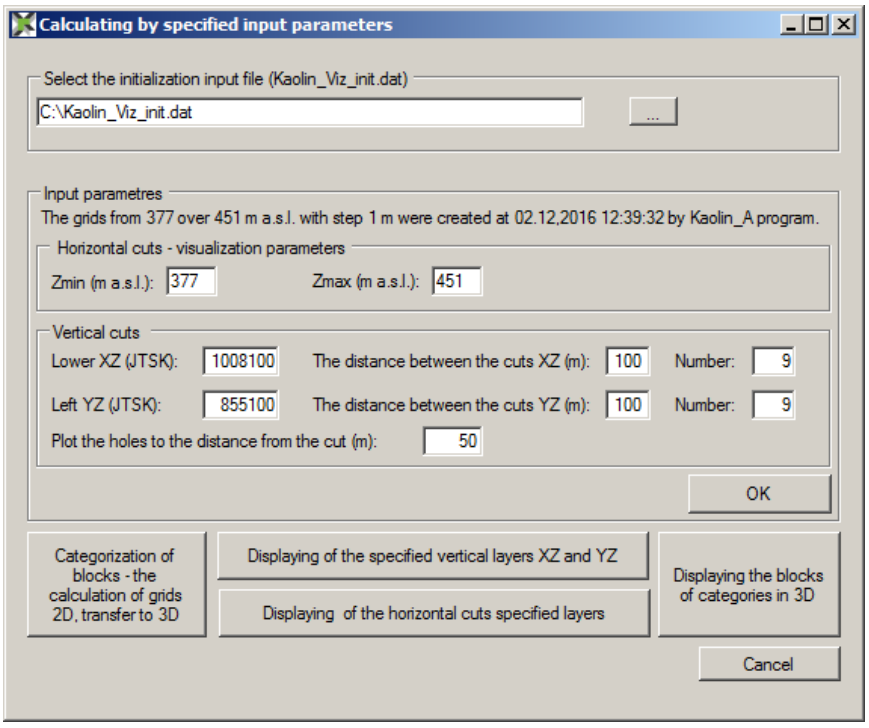

Fig. 26.6 Window program of Kaolin_Viz for setting the calculate parameters

```
..Dim SurferApp As Object
..SurferApp = CreateObject("Surfer.Application")
..Dim Plot As Object
..Plot = SurferApp.Documents.Add(srfDocPlot)
..Dim Shapes As Object
..Shapes = Plot.Shapes
..Dim MapFrame As Object
..Dim PageSetup As Object
..'Assigns the plot document page setup to the variable named
"PageSetup"
..PageSetup = Plot.PageSetup
..PageSetup.Orientation = srfLandscape
..MapFrame = Shapes.AddPostMap(DataFileName:=Dir_Gridy + Pom + -
..Sou_Rezu + ".dat")
..Dim PostMap As Object
..PostMap = MapFrame.Overlays(1)
..PostMap.xCol = 1
..PostMap.yCol = 2
..PostMap.LabCol = 3 ' Cislo bloku
..PostMap.LabelFont.Size = 1
..PostMap.LabelPos = srfPostPosCenter
..PostMap.LabelFont.Face = "Arial CE"
..PostMap.Symbol.Size = 0.1
..PostMap.Symbol.Index = 120
..PostMap.Name = "Bloky"
..PostMap.Visible = False
```

```
'   Nastavení měřítka
..MapFrame.xMapPerPU = Vert_X_Units_per_cm
..MapFrame.yMapPerPU = Vert_Y_Units_per_cm
…
..'vykreslení bln souborů jednotlivých kategorií
..Dim MapFrame_Kat_rez(1 To 11) As Object
..Dim BaseMap_Kat_rez(1 To 11) As Object
..For i = 1 To 11
...If Exist_kat(i) Then
....MapFrame_Kat_rez(i) = Shapes.AddBaseMap(ImportFileName:=CurDir
+ "\Kat_rez_" + Kategorie(i) + ".bln")
....BaseMap_Kat_rez(i) = MapFrame_Kat_rez(i).Overlays(1)
....BaseMap_Kat_rez(i).Line.ForeColorRGBA.Red = Kat_rgb(i, 1)
....BaseMap_Kat_rez(i).Line.ForeColorRGBA.Green = Kat_rgb(i, 2)
....BaseMap_Kat_rez(i).Line.ForeColorRGBA.Blue = Kat_rgb(i, 3)
....BaseMap_Kat_rez(i).Line.Style = "solid"
....BaseMap_Kat_rez(i).Line.Width = 0.1
....BaseMap_Kat_rez(i).Name = "Base_" + Kategorie(i)
....BaseMap_Kat_rez(i).Visible = True
...End If
..Next i
...
```

Fig. 26.7 The example of program code using Surfer automation model

CONCLUSION

For the example programs *Kaolin_A* and *Kaolin_Viz* and it is clear that the use of Voxler and Surfer automation object models in Visual Studio 2015 enables effective automatic model creation of kaolin bearings and generates various types of 2D and 3D visualization according to steps 2.6 to 2.10 from the methodological procedure (see Introduction). This leads to the rapid creation of alternative models of kaolin bearings including rapid updating of these models when adding or changing the input data.

Since January 2017 is this dynamic model utilized in practice by company Sedlecký kaolin for the kaolin deposit Jimlíkov – east.

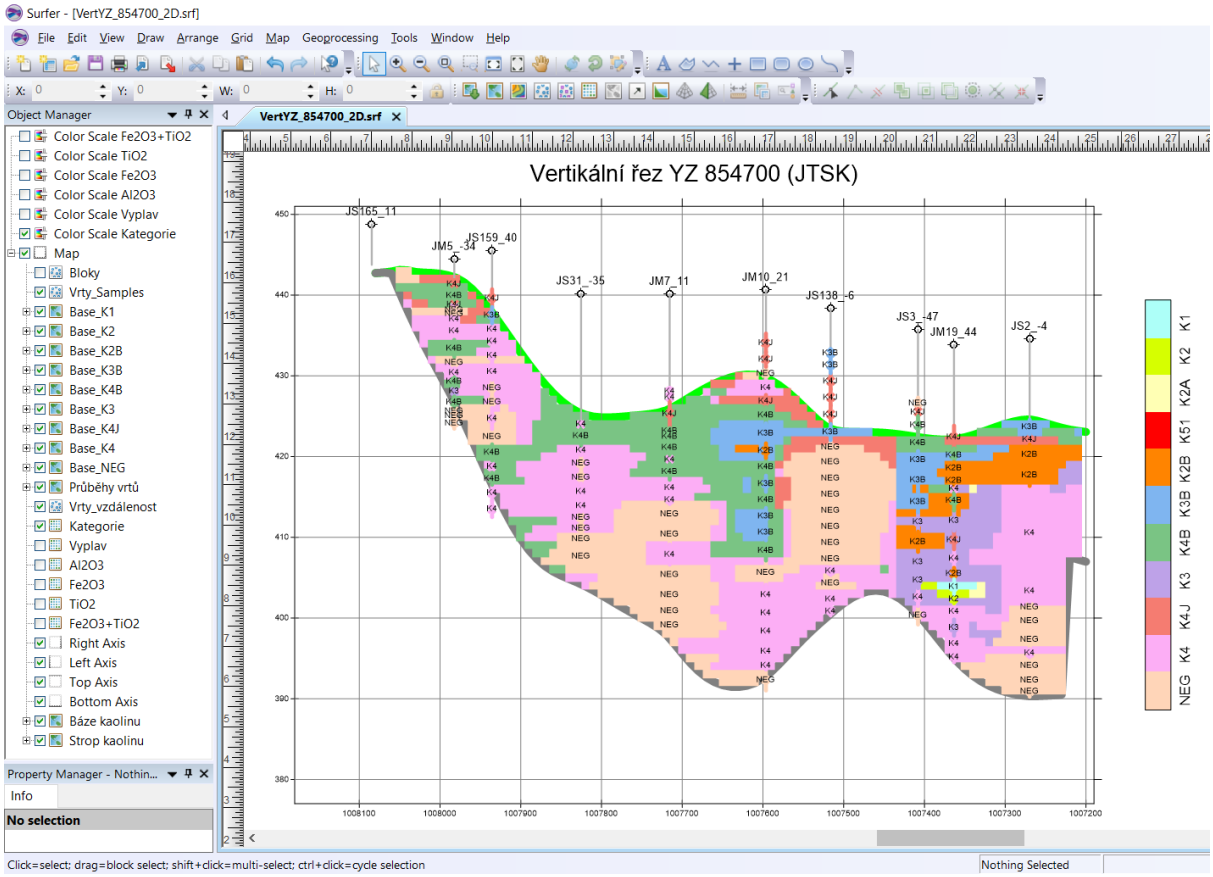

Fig. 26.8 Visualization of the vertical section YZ 854700 in the Surfer environment

ACKNOWLEDGEMENTS

This work was supported by the project TAČR TE02000029 – CEEMIR (Competence Centre for Effective and Ecological Mining of Mineral Resources).

LITERATURE, REFERENCES

1. "Competence Centre for Effective and Ecological Mining of Mineral Resources" Internet: https://www.hgf.vsb.cz/ceemir/cs/, [Feb. 20, 2017].
2. "Microsoft MSDN Library." Internet: https://msdn.microsoft.com/cs-cz/library/dd831853.aspx, [Feb. 20, 2017].
3. "POWER FORWARD INTO 3D VISUALIZATION - VOXLER v4." Internet: http://www.goldensoftware.com/products/voxler/features, [Feb. 20, 2017].
4. "EXPLORE THE DEPTHS OF YOUR DATA - SURFER v14." Internet: http://www.goldensoftware.com/products/surfer/features, [Feb. 20, 2017].

IMPLEMENTATION OF A KAOLIN DEPOSIT MODEL IN VISUAL STUDIO 2015

Abstract: *This paper focuses on research within the project TE02000029 Competence Centre for Effective and Ecological Mining of Mineral Resources, granted by The Technology Agency of the Czech Republic, and, more specifically, on the research within its work package WP4 - Spatial modelling of mineral deposits. The focus of this work package is digital modelling of selected non-energetic raw materials, which belong to the critical commodities, as defined by the European Union. For modelling these deposits, suitable mathematical procedures, based on study and reevaluation of archived data, are needed. One of the selected deposits is a kaolin deposit near the village Jimlíkov near the city Karlovy Vary. The article describes the implementation of the methodology used for processing deposits of kaolin in Visual Studio 2015 by means of objects of Surfer and Voxler created by Golden Software. This software solution is installed at the company Sedlecky kaolin a.s. and thus is fully exploited in practice.*

Keywords*: kaolin deposit, Visual Studio 2015, spatial modeling, Surfer, Voxler*

IMPLEMENTACE MODELU LOŽISKA KAOLINU V PROSTŘEDÍ VISUAL STUDIO 2015

Abstrak*t: Článek popisuje část řešení projektu TE02000029 - Centrum kompetence efektivní a ekologické těžby nerostných surovin (CEEMIR) financovaného Technologickou agenturou ČR, přesněji výsledky řešení Work Package WP4 - Prostorové modelování ložisek nerostných surovin. Hlavním cílem WP4 je digitální modelování vybraných neenergetických surovin, které se řadí mezi kritické komodity EU. Pro modelování ložisek se využívají vhodné matematické postupy na základě studia a přehodnocení dat z archivních materiálů. Jedním z vybraných ložisek je ložisko kaolínu v okolí obce Jimlíkov na Karlovarsku. V článku je popis implementace metodiky zpracování ložiska kaolínu v prostředí Visual Studia 2015 s využitím objektů Surfer a Voxler firmy Golden Software. Vytvořené softwarové řešení je instalováno u firmy Sedlecký kaolín a.s. a je tak plně využíváno v praxi.*

Klíčová slova*: ložisko kaolinu, Visual Studio 2015, prostorové modelování, Surfer, Voxler.*

Date of submission of the article to the Editor: 04.2017
Date of acceptance of the article by the Editor: 05.2017

Mgr. Marcela Jarošová
VŠB-TU Ostrava
Katedra matematiky a deskriptivní geometrie, Institut čistých technologií těžby a užití energetických surovin
17. listopadu, 708 33 Ostrava-Poruba
tel.: +420 597 323 827, e-mail: marcela.jarosova@vsb.cz

Doc. RNDr. František Staněk, Ph.D.
VŠB-TU Ostrava
Katedra matematiky a deskriptivní geometrie, Institut čistých technologií těžby a užití energetických surovin
17. listopadu, 708 33 Ostrava-Poruba
tel.: +420 597 325 484, e-mail: frantisek.stanek@vsb.cz

RNDr. Jana Staňková, Ph.D.
VŠB-TU Ostrava
Katedra matematiky a deskriptivní geometrie
17. listopadu, 708 33 Ostrava-Poruba
tel.: +420 597 325 484, e-mail: jana.stankova@vsb.cz