

# 3D-IMAGE REPORT

## GEOAPP

Tutorial

GeoDict release 2024 SP1

Published: May 3, 2024



GEO DICT

© Math2Market GmbH 2024

Citation:

Jonas Schabernack, Anne Blumer, Barbara Planas. Tutorials for GeoDict 2024. GeoApp: 3D-Image report. Math2Market GmbH, Germany.

All rights reserved. It is not permitted to reproduce the book or parts thereof in any form by photocopy, microfilm or other methods or to transfer it into a language suitable for machines, in particular data processing systems, without the express permission of the publisher. The same applies to the right of public reproduction.

The tutorials for GeoDict from Math2Market GmbH can be obtained from:

Math2Market GmbH  
Richard-Wagner-Strasse 1  
67655 Kaiserslautern  
Germany

Phone: +49 631 205 605 0  
Fax: +49 631 205 605 99  
Email: [info@math2market.de](mailto:info@math2market.de)  
Web: [www.math2market.de](http://www.math2market.de)

# 3D-IMAGE REPORT GEOAPP: CHECKING THE QUALITY OF 3D GRAY VALUE IMAGES

1

## INTRODUCTION

1

### MOTIVATION

1

### BACKGROUND OF THE BENTHEIMER SANDSTONE SAMPLE

1

### HOW TO USE THIS EXAMPLE TUTORIAL

1

## 3D-IMAGE CHECK

3

### 3D-IMAGE REPORT GEOAPP

7

### CUSTOMIZE THE GEOAPP

10

### CUSTOMIZE THE POWERPOINT® TEMPLATE FILE

10

### CUSTOMIZE THE GEOPY SCRIPTS

13

## CONCLUSION

17

## REFERENCES

17

# 3D-IMAGE REPORT GEOAPP: CHECKING THE QUALITY OF 3D GRAY VALUE IMAGES

## INTRODUCTION

### MOTIVATION

Good insights into the micro-CT images on hand are crucial when processing the data for further property predictions via **GeoDict** simulations.

For this, we supply a GeoApp, a script in the form of a Python macro to produce PowerPoint® presentation slides from 3D gray-value image data. The automatically created PowerPoint® presentation is a quick and easy way to check the quality of the images and discover possible artifacts.

### BACKGROUND OF THE BENTHEIMER SANDSTONE SAMPLE

The shallow-marine Bentheimer Sandstone was deposited in one of the NW-SE trending basins north of the London-Brabant and Rhenish massifs during the Valanginian (Early Cretaceous). The Bentheimer Sandstone forms an important reservoir rock for petroleum and is closely related to the Gildehauser Sandstone. Both their names relate to locations in the northwest of Germany, where outcrops are exposed ([Dubelaar and Nijland, 2015](#)). The Bentheimer sample was kindly provided by Matthew Andrew of Carl Zeiss X-ray Microscopy.

For this tutorial, a cut-out of  $128^3$  voxels was prepared from the larger micro-CT image, in order to reduce computational times. The sample is used to illustrate the capabilities of the **GeoDict** software and its **ImportGeo** module.

### HOW TO USE THIS EXAMPLE TUTORIAL

The tutorial illustrates how to produce a PowerPoint® presentation out of 3D gray value images, explains the underlying **GeoPy** scripts and presents how to customize the PowerPoint® template and the **GeoPy** scripts.

This tutorial is divided into three parts:

- Create a PowerPoint® presentation using the 3D-image report **GeoApp**.
- Understanding the workflow of the **GeoApp** scripts.
- Usage of a customized PowerPoint® template and modification of the **GeoPy** scripts.

---

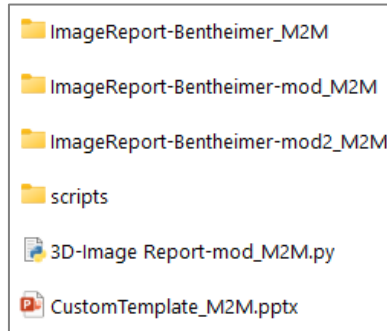
*Modules needed to follow this tutorial: **ImportGeo**-Vol*

---

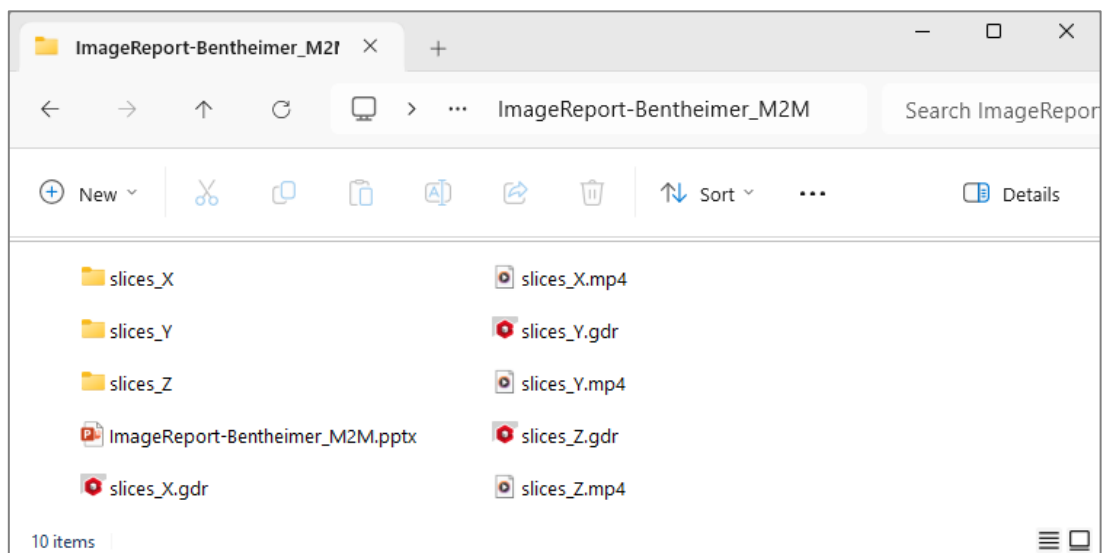
The **GeoApp-3D-image-report** folder contains 3 folders and this PDF file.

The folder **Input-Data** contains the folder **Bentheimer\_ImageStack** which contains pictures of the individual slices seen from Z-direction.

The **Results-M2M** folder contains the results that M2M obtained by following this tutorial, which are 4 folders, a customized PowerPoint® template file and a modified GeoPy script of the GeoApp. The **scripts** folder contains 2 GeoPy scripts that are used by the main GeoApp script.



The three image report folders each contain 3 subfolders and 7 files:




For each of the three spatial directions, the Z-,Y- and X-direction, a GeoDict result file (\*.gdr), a movie and a corresponding folder with the same name are saved from GeoDict. They provide all necessary data for the PowerPoint® presentation, which consists of some 2D pictures from the 3D data as well as movies for the different directions.

The **Results-User** folder is empty and will be filled with your results during this tutorial.

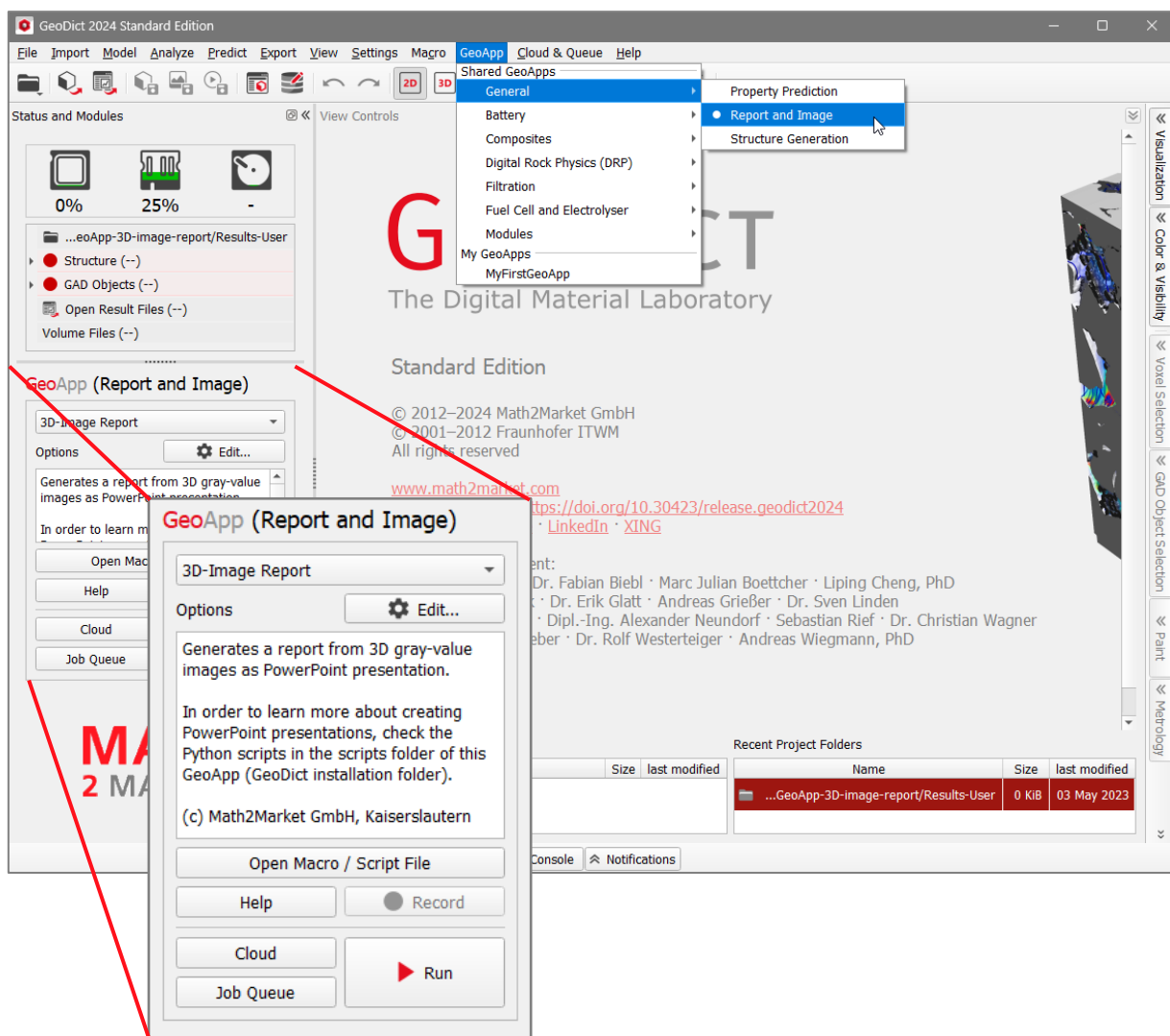
## 3D-IMAGE CHECK

To produce a PowerPoint® presentation containing data from the Bentheimer Sandstone proceed with the following workflow:

1. Start **GeoDict**.
2. Set the project folder where you want to store the data by selecting **File → Choose Project Folder → Select Project Folder...** in the menu bar, or by clicking the  icon in the toolbar and choose **Select Project Folder**.

For this tutorial, we recommend selecting **GeoApp-3D-image-report/Results-User**, which is the folder for this tutorial.

3. In the menu bar, select **GeoApp → General → Report and Image**.
4. In the **GeoApp (Report and Image)** section (to the bottom left of the **GeoDict** GUI), select **3D-Image Report** from the pull-down menu and click **Edit...** to change the parameters.

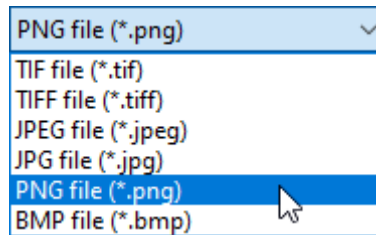


5. In the **3D-Image Report Parameters** dialog, set the following:
  - a. Adjust the **Result Name** for the PowerPoint® presentation to **ImageReport-Bentheimer**. The resulting files, i.e. images and movies for all directions and the PowerPoint® presentation, are saved in a folder with the same name. If the

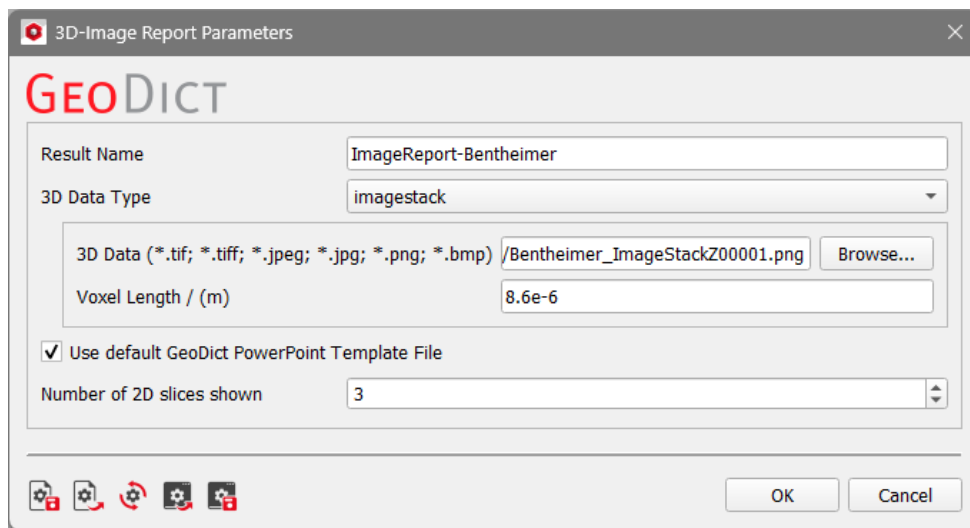
folder does not yet already exist, **GeoDict** will automatically create a new folder with this name.

- b. Choose the corresponding **3D Data Type**. Possible are imagestack, 3D tiff files, **GeoDict** raw files (\*.grw), and raw files (\*.raw). For this tutorial, we select **imagestack**.
- c. Click **Browse...** to search for the **3D data** file. For this tutorial, navigate to the **Bentheimer\_ImageStack** folder, which was provided with this tutorial in the **Input-Data** folder, and select **Bentheimer\_ImageStackZ00001.png**.

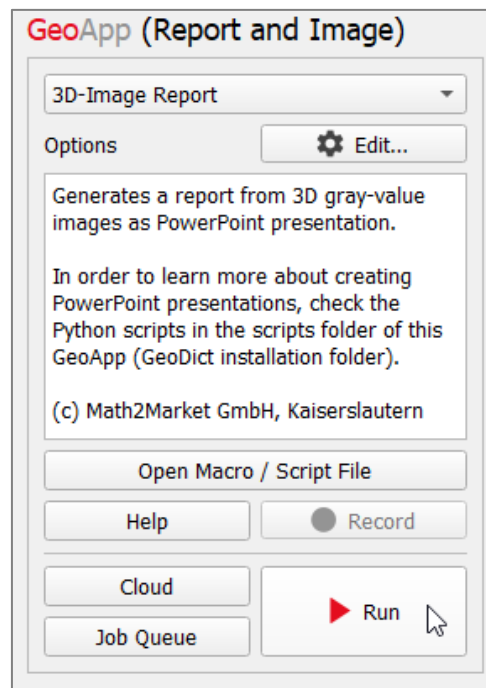
**Note that**, in case of image stacks, you can select any one of the 2D image files. Please take care that you choose the according file extension in the **Select File** dialog, which is \*.png for this tutorial.



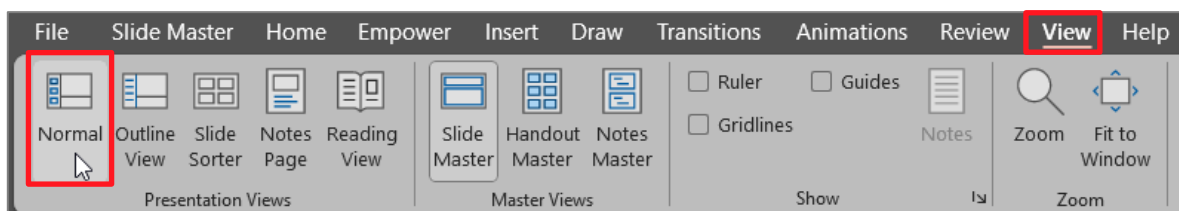
- d. Enter the **Voxel Length** in meters for the chosen image data. For this tutorial, enter **8.6e-6**. This option will not appear when using **GeoDict** raw files (.grw).



- e. In this tutorial step, we use the default **GeoDict** PowerPoint® template file, so make sure this box is checked. Alternatively, see below [Customize the PowerPoint Template File](#) for instructions on how to create and use your own **GeoDict** PowerPoint® template file.
- f. Choose the number of 2D slices to be shown per axis in the resulting PowerPoint® presentation. For this tutorial we choose 3.
- g. Click **OK** to save the parameter settings and click **Run** in the **GeoApp (Report and Image)** section to start the script execution.



6. An **ImageReport-Bentheimer** folder is now saved in the **Result-User** folder and contains the **ImageReport-Bentheimer.pptx** PowerPoint® file as well as the files described [above](#). If PowerPoint® opens the **ImageReport-Bentheimer.pptx** file in **Slide Master** view, please navigate to **View** → **Normal** in PowerPoint® to see the generated presentation.



The presentation has a title slide and for each of the Z-, Y-, and X-directions a 2D movie and the above specified number of 2D image slices is included. Each movie shows the whole image data viewed from the corresponding axis (see figures below).


---


*A careful examination of the PowerPoint® presentation slides running as a loop may reveal previously unknown image artifacts.*

---



Click icon to add picture

  
The Digital Material Laboratory




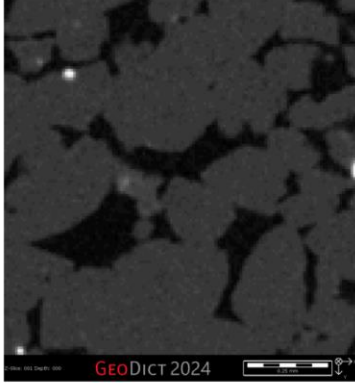
**IMAGEREPORT-BENTHEIMER**


3D-Image Rep

Image Data:  
Contains 2D-images

**FRAME Z - 2D MOVIE**




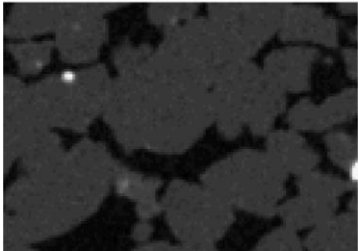


2000x 1000 Depth: 1000 **GeoDict 2024** 


**MATH**  
2 MARKET

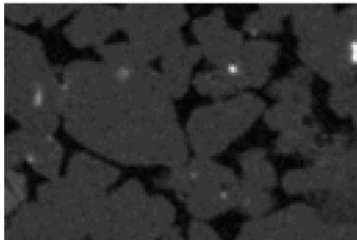
**FRAME Z - SLICE 1**






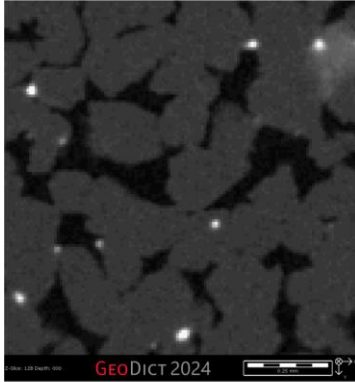
**FRAME Z - SLICE 63**






**FRAME Z - SLICE 128**





2000x 1000 Depth: 1000 **GeoDict 2024** 

**MATH**  
2 MARKET

## 3D-IMAGE REPORT GEOAPP

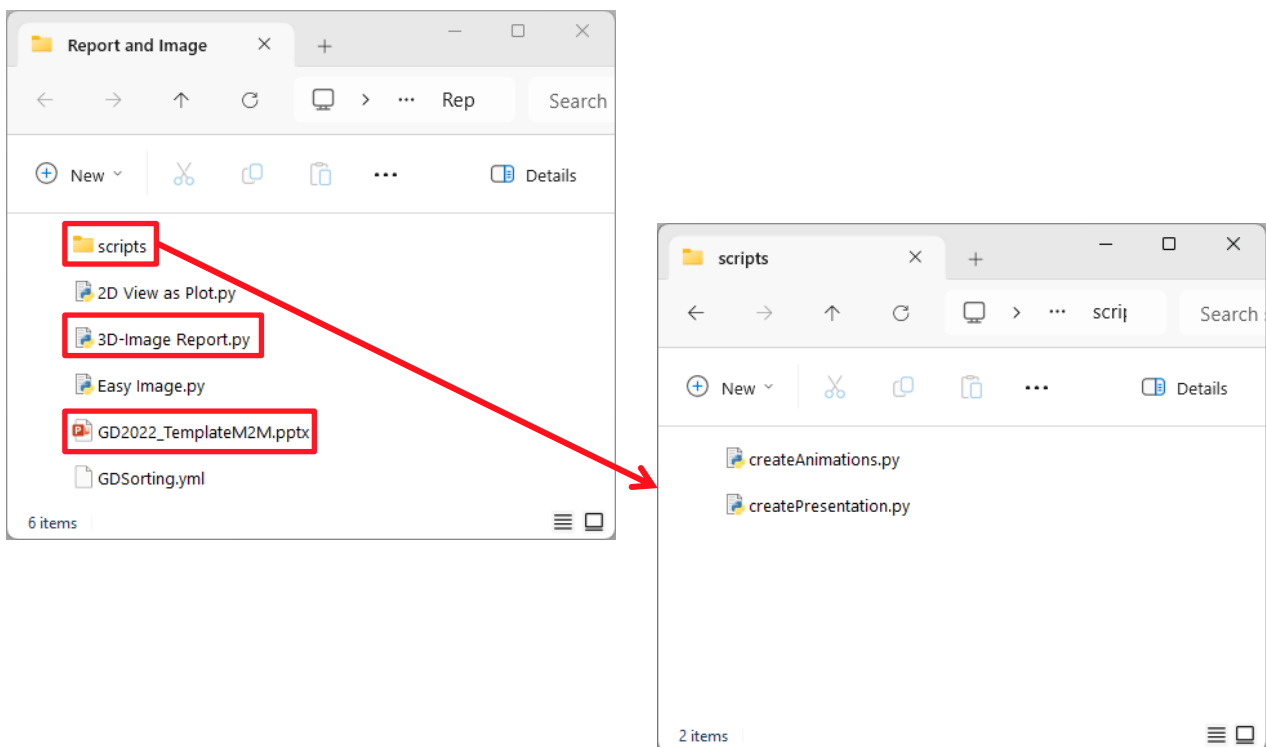
In order to customize the GeoApp, it is necessary to understand the underlying GeoPy Python scripts that are used when executing the GeoApp. In summary, the **3D-Image Report GeoApp** performs the following tasks:

1. Create 2D image slices in Z-, Y- and X-direction of the selected dataset.
2. Create movies for all directions using the generated image slices.
3. Create a PowerPoint® presentation using the selected template.
4. Insert image slices depending on the chosen number of 2D slices and movies into the respective PowerPoint® presentation slides.

The GeoPy scripts are located under

« OS (C:) » Program Files » Math2Market GmbH » GeoDict 2024 » GeoApps » General » Report and Image »

for a default GeoDict installation on Windows. In this folder, the \*.py files for all **Report and Image GeoApps**, as well as the default GeoDict PowerPoint® template file and a **scripts** subfolder are stored.



A detailed description of GeoPy scripts and their application can be found in the [GeoPy scripting to automate GeoDict simulations](#) handbook. To view and modify the GeoPy scripts, a text editor is required. For Windows **Notepad++** is recommended. Open the **3D-Image Report.py** file with a text editor.

In general, GeoPy scripts run a sequence of commands and consist of (at least) a **header**, **description**, **variable** and **command** block.

```

1  # GeoDict -- The Virtual Material Laboratory
2  # Copyright (C) 2019-2024, Math2Market GmbH
3  #
4  # GeoDict executable & support materials
5  # are sold exclusively by Math2Market GmbH.
6  # This code may not be used or modified
7  # without written consent by Math2Market GmbH.
8  #
9  # Creation: 2019
10 # Author: Christian Hinz
11 #
12 # $Id: 3D-Image Report.py 70958 2024-01-04 10:05:07Z griesser $
13 #
14
15 import os
16
17 Header = {
18
19 Description = '''Generates a report from 3D gray-value images as PowerPoint presentation.
20
21 RequiredLicenses = ["ImportGeoVol"]
22
23 Variables = {
24
25     macroFileFolder = gd.getMacroFileFolder()
26
27     if useDefaultPPTX:
28     else:
29
30         presentationFileName = projectFolder + '.pptx'
31         macroFileFolder = gd.getMacroFileFolder()
32
33         assert os.path.isfile(templatePPTX), f'PPTX Template file not found: {templatePPTX}, working dir is: {os.getcwd()}'
34
35         OriginalStructure = gd.getStructure()
36         viewOld = gd.getViewStatus()
37         cm = gd.getConstituentMaterials(Header['Release'])
38
39 makeMP4 args = {
40 makePPTX args = {
41
42     gd.runCmd "GeoDict:ExecuteMacro", makeMP4_args, Header['Release']
43     gd.runCmd "GeoDict:ExecuteMacro", makePPTX_args, Header['Release']
44
45     #undo steps for rendering
46     viewOld['dataView']['Structure']['Shown'] = True
47     gd.runCmd "GeoDict:SetViewStatus", viewOld, '2024')
48     gd.runCmd "ConstituentMaterials", cm, Header['Release'])
49
50 if OriginalStructure is not None:
51 else:

```

The variables in the **3D-Image Report.py** file contain all the parameters that can be set in the **3D-Image Report Parameters** dialog in **GeoDict**.

```

Variables = {
    'NumberOfVariables' : 15,
    'Variable1' : {
        'Name' : 'projectFolder',
        'Label' : 'Result Name',
        'Type' : 'string',
        'ToolTip' : 'Define the name for the results folder containing the created image and animation.\n'
        'Also creates the PowerPoint presentation according to that name.',
        'BuiltinDefault' : 'ImageReport' },

```

The main function of **3D-Image Report.py** script is to collect all required parameters and execute two other **GeoPy** scripts to create the movies and the PowerPoint® presentation. These scripts are located in the **scripts** subfolder.

The **createAnimations.py** script loads the 3D gray-value image data via **ImportGeo-Vol**, sets a defined visualization and creates a series of 2D images and a movie for all three directions.

The **createPresentation.py** script uses the created 2D images and movies and inserts them into placeholders in a given PowerPoint® template.

For the next tutorial step, copy the three GeoPy scripts and the default PowerPoint® template to your project folder (e.g., **Results-User**). It is important to keep the **createAnimations.py** and **createPresentation.py** files in the **scripts** subfolder. Please rename the **3D-Image Report.py** file to **3D-Image Report-mod.py** and the PowerPoint® template to **CustomTemplate.pptx**. Your own PowerPoint® template file may be used for the resulting PowerPoint® slides. In this case, ensure that the placeholders are named exactly as described in the next step of this tutorial.

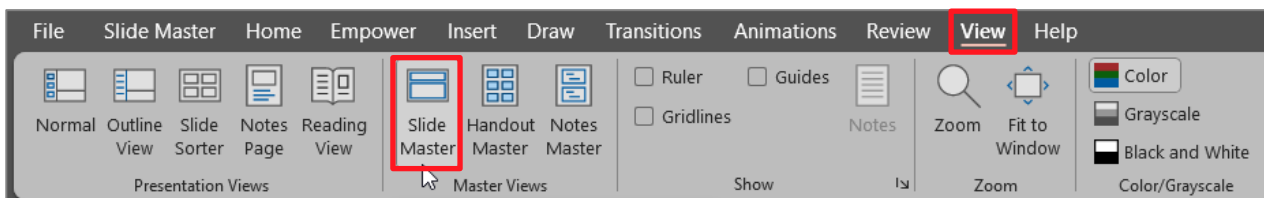
## CUSTOMIZE THE GEOAPP

### CUSTOMIZE THE POWERPOINT® TEMPLATE FILE

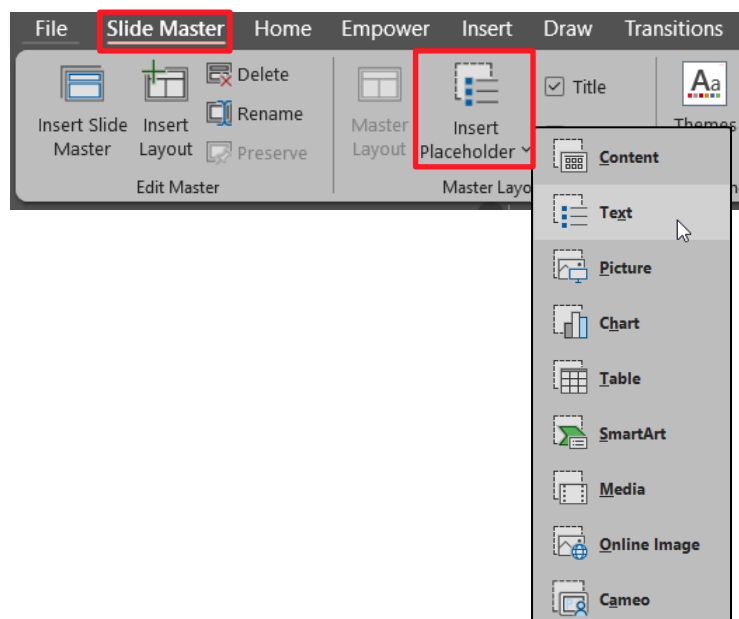
For this tutorial, we customize the default GeoDict PowerPoint® template file that was saved as **CustomTemplate.pptx** to the project folder in the previous tutorial step. Open this \*.pptx file with PowerPoint®. The default PowerPoint® template consists of four different slide types:

1. A title slide (slide 0) with three placeholders: *title*, *subtitle* and *info*
2. An image slide (slide 1) with two placeholders: *title* and *image*
3. A movie slide (slide 2) with two placeholders: *title* and *movie*
4. A combined image and movie slide (slide 3) with three placeholders: *title*, *movie* and *image*. This slide is not used by the GeoApp.

To customize the template file, select the Slide Master view through **View → Slide Master** in the menu bar of PowerPoint®.

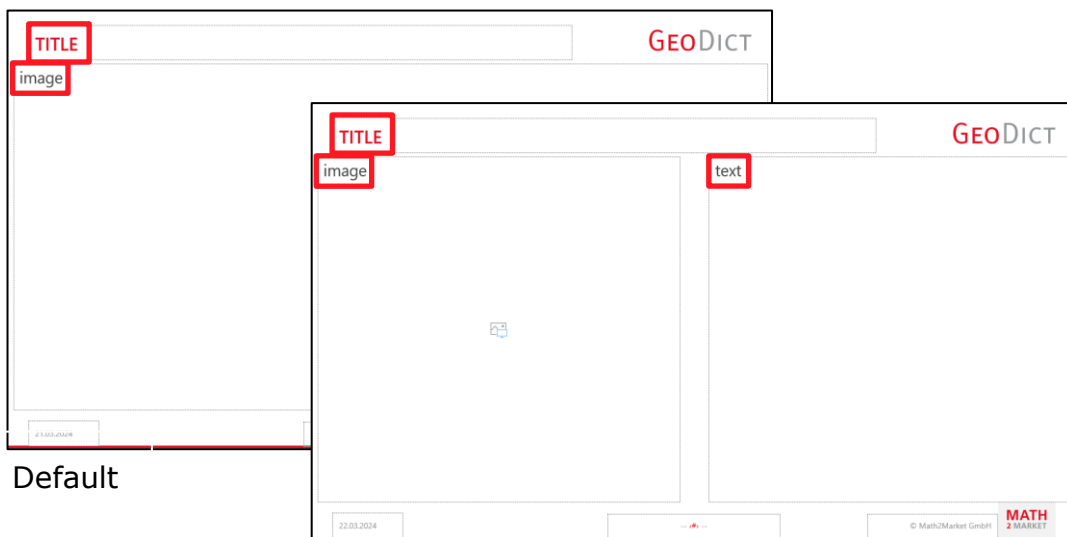
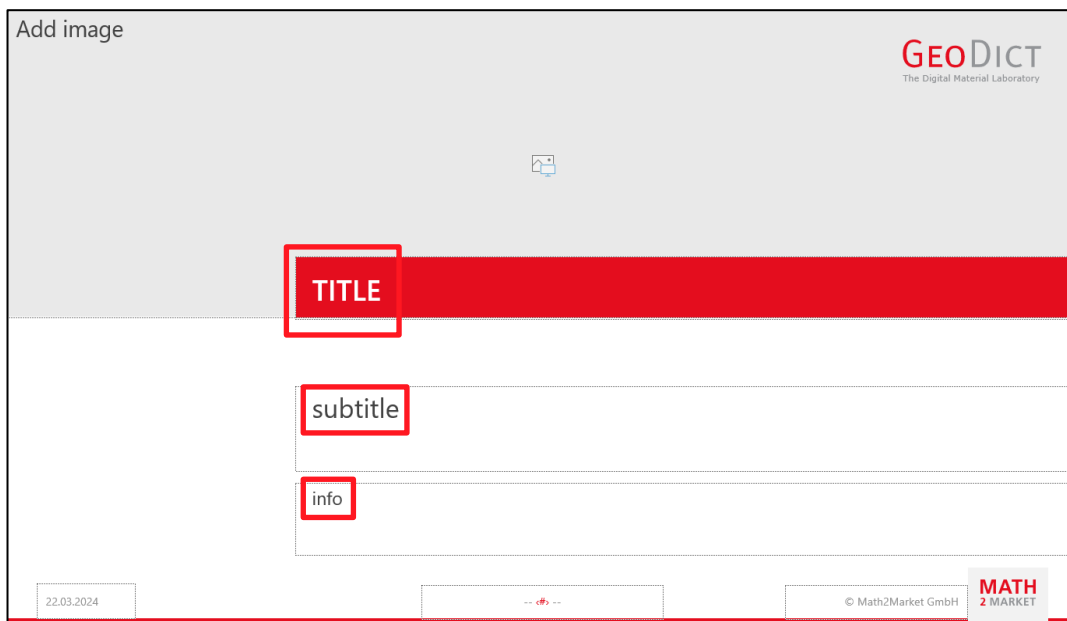


The title slide is not changed and remains the same in this tutorial. In the image slide, decrease the size of the *image* placeholder to the left half of the slide. Insert a text placeholder by clicking on the **Slide Master** tab and then **Insert Placeholder → Text**.



Replace the default text with the word **text** in the placeholder. Repeat this step for the movie slide and save the presentation. The placement of the placeholder on the slide is not important and you can choose a different layout. However, please use the word **text** as text content in any case as it will be used to identify the placeholder in

a following tutorial step. A comparison between the default and our newly created custom PowerPoint® template is shown below:



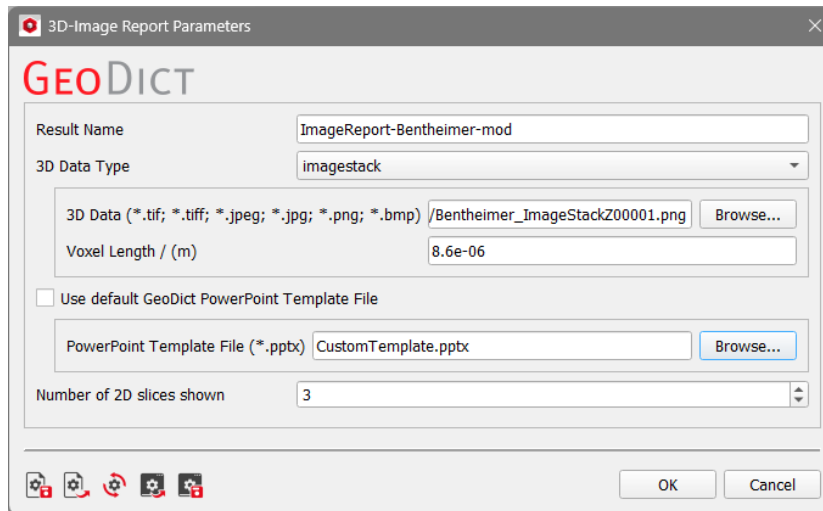
Custom



Custom

After customizing the PowerPoint® template file, return to the **3D-Image Report Parameters** dialog (see step [5](#) in 3D-Image Check section) and do the following:

1. Change the **Result Name** to **ImageReport-Bentheimer-mod**.
2. Uncheck **Use default GeoDict PowerPoint Template File**.
3. Browse for the file **CustomTemplate.pptx** or enter the full file path into the **PowerPoint Template File** box.



4. If the settings were changed, continue with step [5f](#) in the 3D-Image Check section.
5. Click **OK** and click **Run**.
6. A new **ImageReport-Bentheimer-mod** folder is saved in the **Result-User** folder. In the resulting presentation, an empty textblock is located next to each image and movie. This textblock can be edited afterwards. Alternatively, the **GeoApp** can be modified to automatically fill text inside the placeholder, which is presented in the next tutorial step.

## CUSTOMIZE THE GEOPY SCRIPTS

To customize the content that is placed into the PowerPoint® template placeholders, the **GeoPy** scripts need to be modified. Navigate to the **createPresentation.py** file in the copied **scripts** folder and open it with a text editor. The variable for the PowerPoint® slides in the script is always named **sl**.

1. The commands to write content to the title slide are located in lines **77** to **80**. Three **sl.add\_text** commands write text to the three placeholders *title*, *subtitle* and *info*.

```

77  sl = rep.add_slide(0) # title slide
78  sl.add_text("title", projectFolder.split('/')[-1].split('\\')[-1])
79  sl.add_text("subtitle", "3D-Image Report")
80  sl.add_text("info", f"Image Data: {imageName}\nContains 2D-images and animations for each of Z-, Y- and X-direction.")

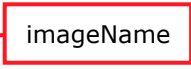
```

- a. In line **78**, the *title* placeholder is filled with the name of the project folder that is set in the **3D-Image Report Parameters** dialog under **Result Name**.
- b. In line **79**, the *subtitle* placeholder is filled with the text **3D-Image Report**.
- c. In line **80**, the *info* placeholder is filled with `f"Image Data: {imageName}\nContains 2D-images and animations for each of Z-, Y- and X-direction."`. The preceding **f** turns the string into a "formatted string" in which you can enter the values of variables, here the variable `{imageName}`. The **\n** starts a new line in the PowerPoint® text box. **imageName** is a variable that is currently empty and thus no text is written in the upper line after "Image Data:".
  - d. To add information to this variable, navigate to the **3D-Image Report-mod.py** file and open it with a text editor. In line **186**, we can see the **imageName** variable that is provided to **createPresentation.py** and is currently empty.
  - e. Write **imageName** instead of `' '` behind the colon.

```

180  makePPTX_args = {
181      'FileName'      : f'{macroFileFolder}/scripts/createPresentation.py',
182      'Variables'     : {
183          'projectFolder' : projectFolder,
184          'OutputFile'    : presentationFileName,
185          'templatePPTX'  : templatePPTX,
186          'imageName'     : ' ',
187          'numberOfSlicesUser' : numberOfSlicesUser, }, }

```



- f. Now a variable with the name **imageName** needs to be defined in the variables section of the **GeoPy** script. Navigate to line **31** and change the number of variables from **15** to **16**.
- g. Copy the text for variable 1 (lines **32-38**). Go to line **151**, click after the comma and press enter to create a new line (**152**) below. Paste the copied text.
- h. Change **'Variable1'** to **'Variable16'** and make sure to apply the same indentation as for the other variables, which is 2 spaces here.
- i. Change the name to **'imageName'**, the label to **'Image Name'** and leave the type as **string**. Reduce the tooltip to one line and write **'Define the name of the image dataset.'**. Empty the built-in default by writing `''`. Then save the script.



```

152     'Variable16' : {
153         'Name'      : 'imageName',
154         'Label'     : 'Image Name',
155         'Type'      : 'string',
156         'ToolTip'   : 'Define the name of the image dataset.',
157         'BuiltinDefault' : '' },

```

- j. Now the new variable is added to the **3D-Image Report Parameters** dialog when executing the script and can be specified by the user. Close the **3D-Image Report-mod.py** file and return to the **createPresentation.py** file.
2. The commands to write content to the movie slide are located in lines **86** to **88**. Three slides are generated, one for each direction (Z, Y, X).
  - a. In line **87**, the *title* placeholder is filled with `f"Frame {D} - 2D Movie"`, where **D** is the direction.
  - b. In line **88**, the *movie* placeholder is filled by using the command `sl.add_movie` to insert the movie file with the corresponding direction.
  - c. To add content to the created *text* placeholder in the previous tutorial step, add a new line **89** below line **88**.
  - d. Write a new command to fill the *text* placeholder: `sl.add_text("text",f"Movie showing the {D}-slices of the micro-CT dataset.")`. This command will write the given text together with the corresponding direction. Make sure the indentation is identical to the lines above.

```

85     ### movie slide ###
86     sl = rep.add_slide(2) #add 2D-movie slide
87     sl.add_text(         "title",f"Frame {D} - 2D Movie")
88     sl.add_movie(        "movie",f"slices_{D}.mp4")
89     sl.add_text(         "text", f"Movie showing the {D}-slices of the micro-CT dataset.")

```

3. The commands to write content to the image slide are located in lines **99** to **102**. For each direction, the number of generated PowerPoint® slides is equal to the slice number set by the user in the **3D-Image Report Parameters** dialog (see [5f](#)). Therefore, the value 3 used in this tutorial results in 9 slides with images.
  - a. In line **100**, the *title* placeholder is filled with `f"Frame {D} - Slice {slice}"`, where **D** is the direction and **slice** is the number of the shown slice.
  - b. In line **102**, the *image* placeholder is filled by using the command `sl.add_picture` to insert the image file with the corresponding direction and slice number.
  - c. To add content to the created *text* placeholder in the previous tutorial step, add a new line **103** below line **102**.
  - d. Write a new command to fill the *text* placeholder: `sl.add_text("text",f"Image showing the {D}-slice with the number {slice}.")`. This command will write the given text together with the corresponding direction and slice number. Make sure the indentation is identical to the lines above.

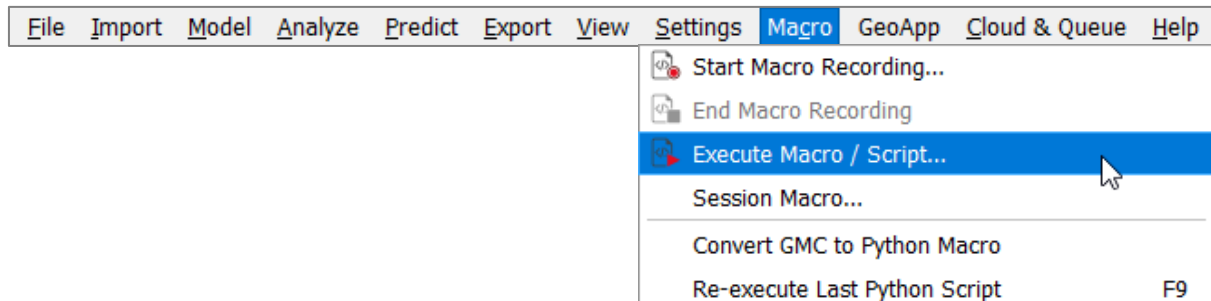
```

99     sl = rep.add_slide(1) #add 2D-image slide
100    sl.add_text(         "title",f"Frame {D} - Slice {slice}")
101    slicestring = '%0*d' % (len(str(end)), slice)
102    sl.add_picture("image",f"slices_{D}/images/Frame{slicestring}.png")
103    sl.add_text(         "text", f"Image showing the {D}-slice with the number {slice}.")

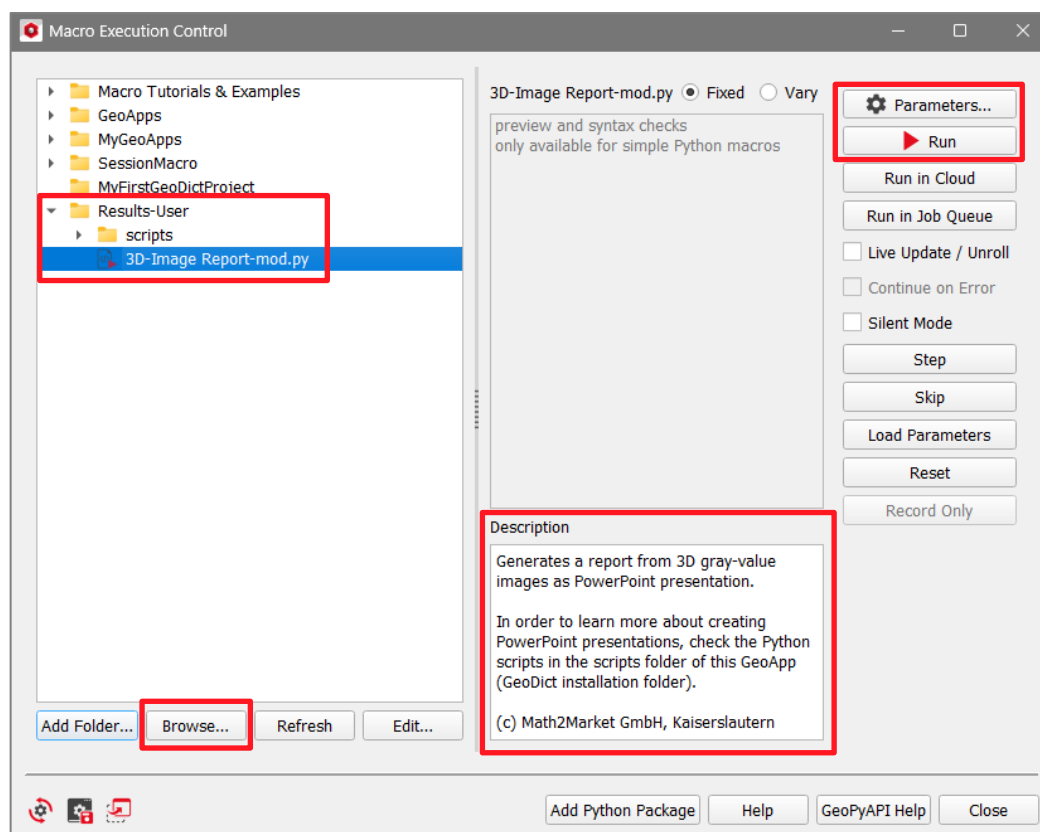
```

- e. Save the customized **createPresentation.py** script and close it.

4. The customized GeoPy script cannot be executed from the GeoApp menu as before since the original GeoApp scripts were not modified. If GeoDict is installed in Program Files under Windows, the original GeoApp scripts can only be changed with administrator rights. Then the files are changed for all users.
5. The customized copy of the GeoApp can be executed like any other GeoPy script. In the menu bar, select **Macro → Execute Macro / Script**.



6. In the **Macro Execution Control** dialog the **Results-User** folder can be found in the top left section and the contained Python scripts are selectable when expanding the folder. If you do not see the folder click on **Browse** below and navigate to the **3D-Image Report-mod.py** file. Select the file and click open.



7. When the Python script is selected the description text of the GeoApp is visible in the **Description** box.
8. Click the **Parameters...** button in the upper right corner and use the same settings as before (see step 5 in 3D-Image Check section)
  - a. Change the **Result Name** to **ImageReport-Bentheimer-mod2**.
  - b. **Uncheck Use default GeoDict PowerPoint Template File** and navigate to and select the **CustomTemplate.pptx**.

- c. The last parameter in the **Parameters** dialog is the **Image Name** that was added to the variables in the **GeoPy** script. Enter **Bentheimer Sandstone** here.
- d. Click **OK** to save the parameter settings.

9. Click **Run** below the **Parameter** button.

The results are saved to the new folder **ImageReport-Bentheimer-mod2**. Within is a new PowerPoint® presentation with all changes of the customized **GeoPy** scripts.

## CONCLUSION

This tutorial showed how to produce a PowerPoint® presentation out of 3D image data using the **3D-Image Report GeoApp**. For this purpose, we used a sample of the Bentheimer Sandstone. For each viewing direction, the presentation contains the chosen number of 2D slices. Moreover, since we have a sample of  $128^3$  voxels, for each viewing direction all 128 slices of 2D-images are saved and the slicing through all of them is captured in the movie.

Additionally, the tutorial explained how to incorporate a self-designed PowerPoint® template to display the results and how to modify the underlying GeoPy scripts to change the automated content of the PowerPoint® presentation.

## REFERENCES

[1] Dubelaar C.W., Nijland T.G. (2015). The Bentheim sandstone: geology, petrophysics, varieties and its use as dimension stone. Engineering Geology for Society and Territory 8, 557-563

Technical  
documentation:

**Jonas Schabernack**

**Anne Blumer**

**Barbara Planas**



Math2Market GmbH

Richard-Wagner-Str. 1, 67655 Kaiserslautern, Germany  
[www.geodict.com](http://www.geodict.com)

© Fraunhofer Institut Techno- und Wirtschaftsmathematik ITWM, 2003-2011.

© Math2Market GmbH, 2011-2024. All rights reserved.