PAUL SCHERRER INSTITUT

PSI

Esteban Marin

TXLWizard: Generate and Convert TXL Files for E-Beam Masks with Python

2016-05-25

PAUL SCHERRER INSTITUT

Outline

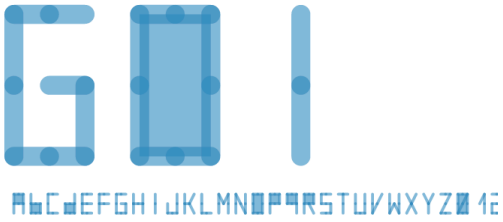
- Introduction
- Example
- TXLConverter
- Documentation

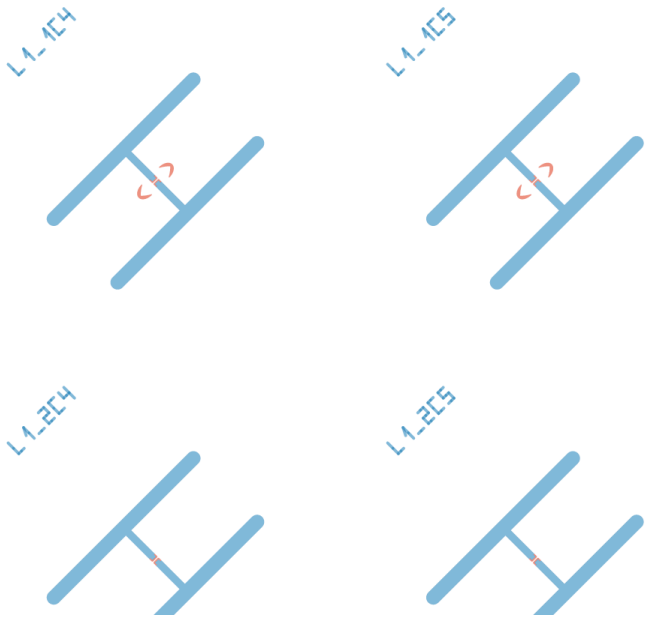
Introduction: What does it do?

- The „TXLWizard“ provides **routines for generating TXL files (.txl) with python code**
  - **Text files** for preparation of **E-Beam lithography masks**
  - TXL files can be processed with BEAMER

- The generated TXL files are also converted to **HTML / SVG**
  - Presentation in any modern browser or graphics application
  - Rapid mask development
  - Don‘t block LayoutBEAMER

- Moreover, a command line interface „TXLConverter“ provides **conversion of existing** TXL files to **HTML / SVG**





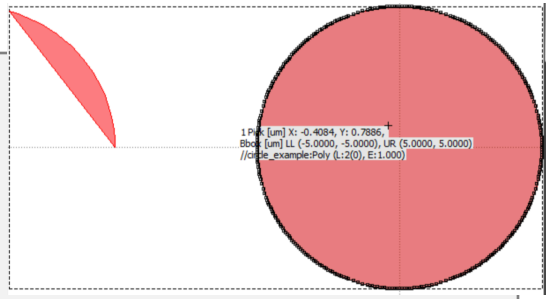
Introduction: Motivation

- Why TXL File Format?
  - Text-based file format
  - Can be generated with scripts
  - Easy to use
  - Optimized E-Beam Performance due to „References“ to objects and array of replicated objects („SREF“, „AREF“)
- Why TXLWizard?
  - Create masks with well-structured scripts
  - Flexible Python Scripting
    - Easy and Powerful
  - Code easy to read and reusable
  - Automated label generation
  - Save your masks as Image for documentation purposes
  - Minimize time blocking Layout Beamer for viewing masks

Introduction: Structure / Pattern / Attribute

- Structure
  - Object containing one or more „Pattern“ objects.
    TXL command: „STRUCT“
- Pattern
  - Pattern object such as a circle, a polygon, an ellipse, a path, etc.
    - Circle („C“)
    - Ellipse („ELP“)
    - Polygon („B“)
    - Polyline („P“)
    - Reference („SREF“)
    - Array („AREF“)

- Attribute
  - Visual Property of a „Pattern“
    - Layer („LAYER“)
    - DataType („DATATYPE“)
    - RotationAngle („ANGLE“)
    - StrokeWidth („WIDTH“)
    - ScaleFactor („MAG“)



```
LETXTLIB 1.0.0
UNIT MICRON
RESOLVE 0.001
BEGLIB

     STRUCT circle_example
     LAYER 2
     DATATYPE 0
     C 5.000 -15.000,0.000 (0.000 75.000 100) ENDC
     C 5.000   0.000,0.000 (0.000 360.000 360) ENDC
     ENDSTRUCT

ENDLIB
```

Introduction: Installation

- Download „TXLWizard“ from http://cad035.psi.ch/TXL_Viewer_DownloadPage.html

- Runs in Python Version 2.7+ and 3.1+

- Copy the „TXLWizard“ folder to a location you like
- Prepend the following command to your python script:

```
import sys
sys.path.append('path_to_the_folder_containing_TXLWizard')
```

- That‘s it, so let‘s use it!

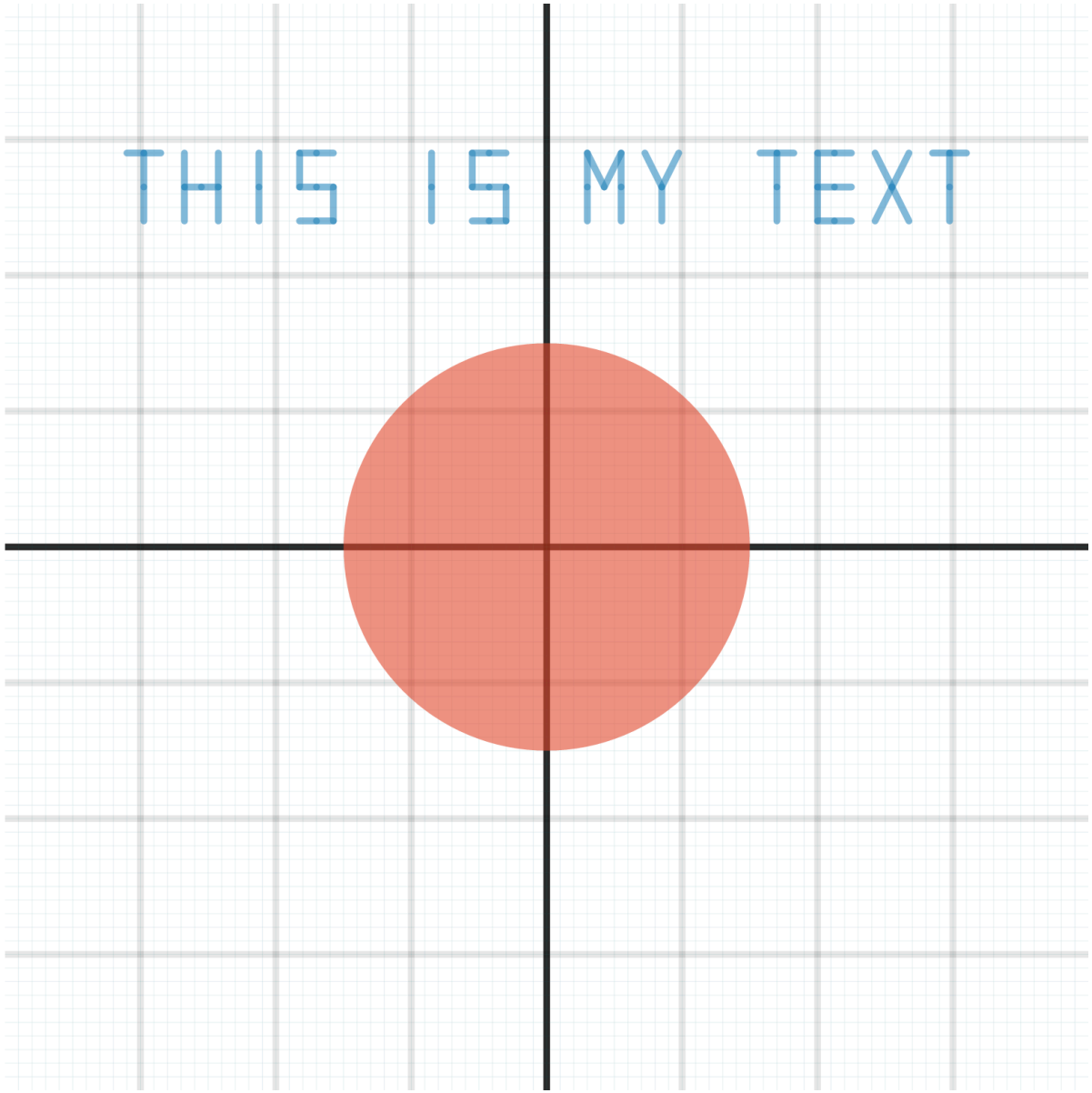
PAUL SCHERRER INSTITUT

Example: Code

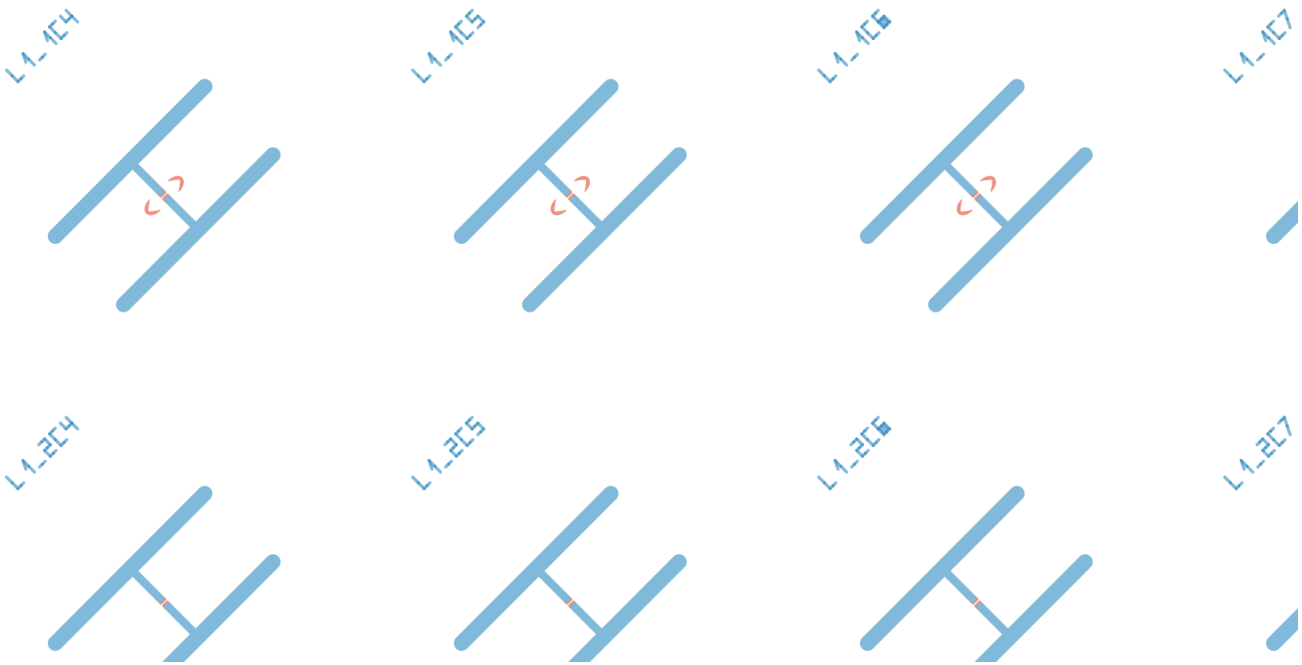
```
import sys
sys.path.append('path_to_the_folder_containing_TXLWizard')
```

```
# Import TXLWriter, the main class for generating TXL Output
import TXLWizard.TXLWriter

# Import Pre-Defined Shapes / Structures wrapped in functions
import TXLWizard.ShapeLibrary.Label

# Initialize TXLWriter
TXLWriter = TXLWizard.TXLWriter.TXLWriter()

# Give the sample a nice label
SampleLabelObject = TXLWizard.ShapeLibrary.Label.GetLabel(
    TXLWriter,
    Text='This is my text',
    OriginPoint=[-310, 240],
    FontSize=50,
    StrokeWidth=5,
    RoundCaps=True,  # Set to False to improve e-Beam performance
    Layer=1
)

# Create Content Structure for Circle with ID `MyCircle`
CircleStructure = TXLWriter.AddContentStructure('MyCircle')

# Add a `Pattern` of type `Circle`
CircleStructure.AddPattern(
    'Circle',
    Center=[0, 0],
    Radius=150,
    Layer=2
)

# Generate Output Files
# Note: The suffix (.txl, .html, .svg) will be appended automatically
TXLWriter.GenerateFiles('Masks/Example_Introduction')
```

Example: Output



Example: Advanced



TXLConverter

- Existing TXL files can be converted to SVG / HTML files with command line script

```
python TXLWizard/Tools/TXLConverterCLI.py
```

```
### TXL Converter v1.6 ###
Converts TXL Files to SVG/HTML
written by Esteban Marin (estebanmarin@gmx.ch)


Full TXL File / Folder Path
If the path is a folder, you can enter the filename separately.
[/home/john.mega/masks]: /Users/esteban/Desktop/masks2/test2.txl

SampleWidth in um
used to draw coordinate system
[1500]:

SampleHeight in um
used to draw coordinate system
[1500]:

Layers to process
comma-separated, e.g. 1,4,5. Type -1 for all layers.
[-1]:

Do Conversion (y/n)? [y]

Files written:
/Users/esteban/Desktop/masks2/test2.html
/Users/esteban/Desktop/masks2/test2.svg

Done
```

Documentation

- The documentation can be found here (PDF and HTML):
  http://cad035.psi.ch/TXL_Viewer_DownloadPage.html

- Contents
  - Introduction
  - Examples
  - Python Module Reference
    - Describes all Patterns, including Parameters and Attributes

- For further help, contact
  - Vitaliy Guzenko (vitaliy.guzenko@psi.ch)
  - Esteban Marin (estebanmarin@gmx.ch)

Wir schaffen Wissen – heute für morgen

My thanks go to

- Hans Sigg
- Thomas Zabel
- Vitaliy Guzenko