GenISys
Advancing the Standard

BEAMER

Solving complex application cases
using the **Python** module

Parameter based flow control with Python

Outline

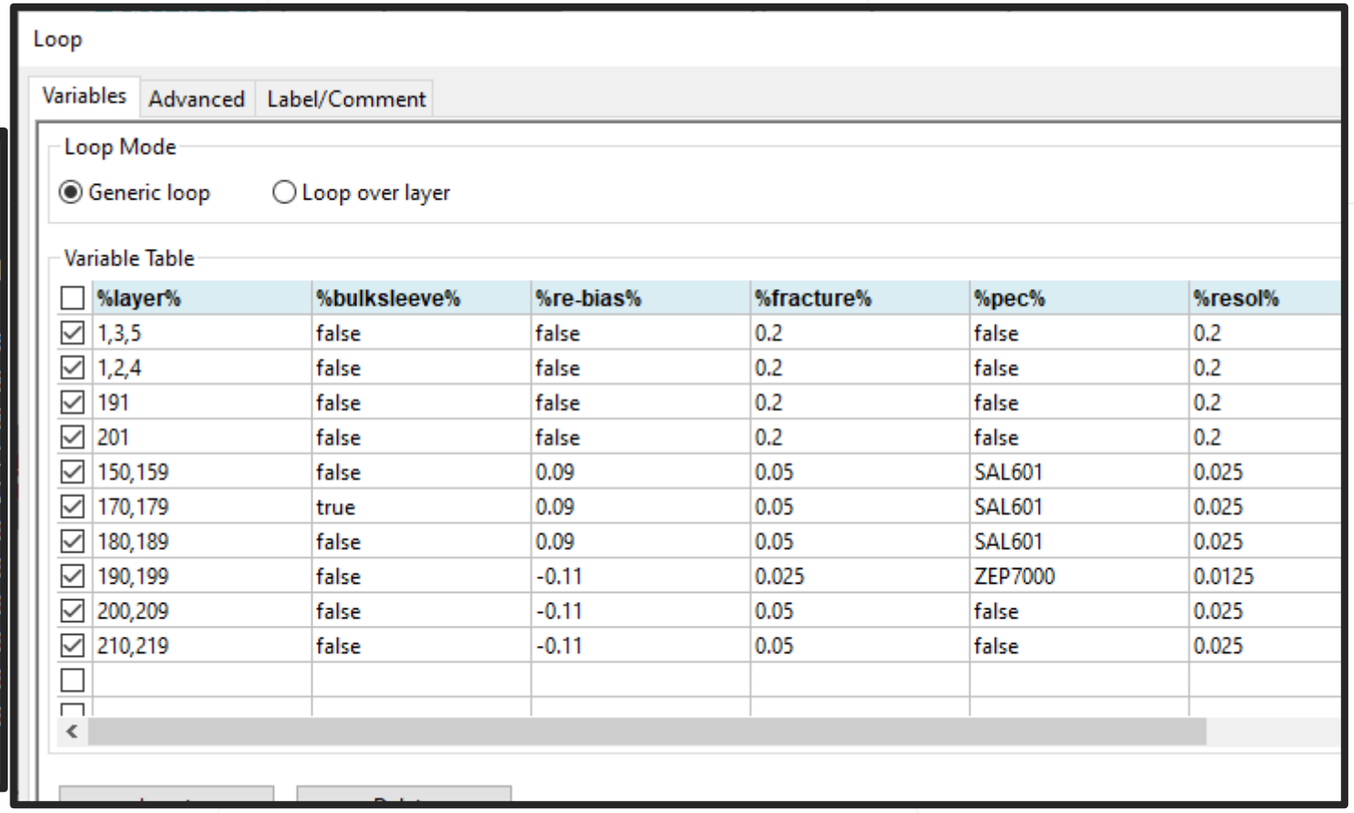
- Flow processing with loops – control file
  - Flow branching based on loop parameters using IF
- How to solve this with python

GenISys
Advancing the Standard

Automation using Control File Processing

Use EXCEL or any other text editor to create your own controls.

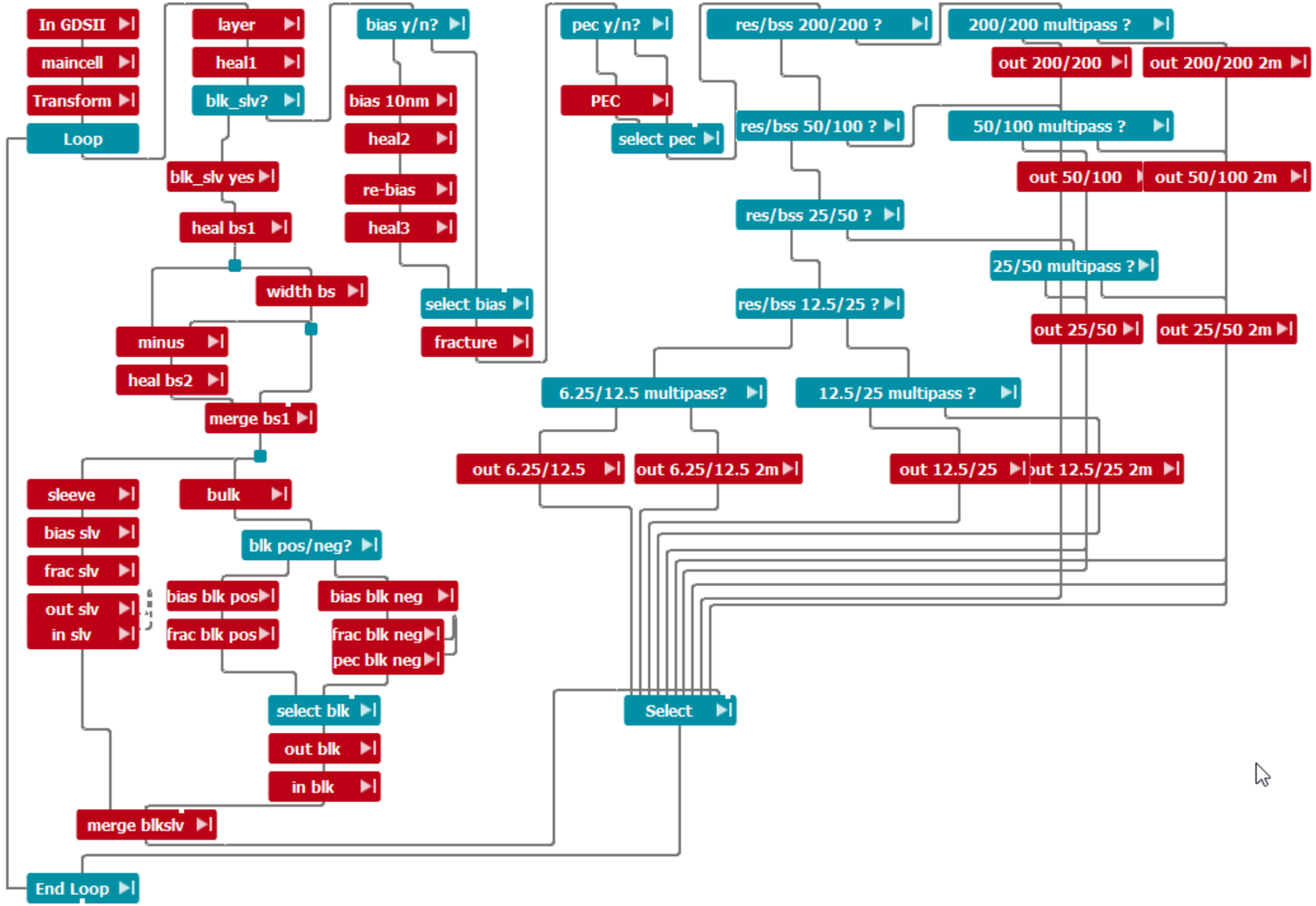
```
# LOOP VARIABLES TABLE
# FIRST LINE contains all VARIABLE NAMES (tab-separated)
# FIRST COLUMN contains the BOOL value of the loop row
# Each following COLUMN contains VALUES for corresponding VARIABL

        %layer% %bulksleeve%    %re-bias%   %fracture%  %pec%   %
true    1,3,5   false   false   0.2 false   0.2 0.2 900 PD_201908
true    1,2,4   false   false   0.2 false   0.2 0.2 900 PD_201908
true    191 false   false   0.2 false   0.2 0.2 900 PD_20190807_1
true    201 false   false   0.2 false   0.2 0.2 900 PD_20190807_1
true    150,159 false   0.09    0.05    SAL601  0.025   0.05    0
true    170,179 true    0.09    0.05    SAL601  0.025   0.05    0
true    180,189 false   0.09    0.05    SAL601  0.025   0.05    0
true    190,199 false   -0.11   0.025   ZEP7000 0.0125  0.025   0
true    200,209 false   -0.11   0.05    false   0.025   0.05    0
true    210,219 false   -0.11   0.05    false   0.025   0.05    0
```



	%layer%	%bulksleeve%	%re-bias%	%fracture%	%pec%	%resol%
☑	1,3,5	false	false	0.2	false	0.2
☑	1,2,4	false	false	0.2	false	0.2
☑	191	false	false	0.2	false	0.2
☑	201	false	false	0.2	false	0.2
☑	150,159	false	0.09	0.05	SAL601	0.025
☑	170,179	true	0.09	0.05	SAL601	0.025
☑	180,189	false	0.09	0.05	SAL601	0.025
☑	190,199	false	-0.11	0.025	ZEP7000	0.0125
☑	200,209	false	-0.11	0.05	false	0.025
☑	210,219	false	-0.11	0.05	false	0.025
☐						

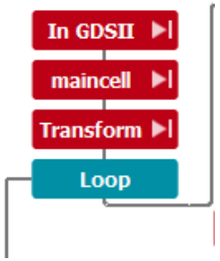
GenISys
Advancing the Standard

Combined sample



- one GDS pattern
- Bulk & Sleeve
- Fracture Resolution
- PEC Process
- Writing Resolution
- Beam Step Size
- Field Size
- Multipass Parameters

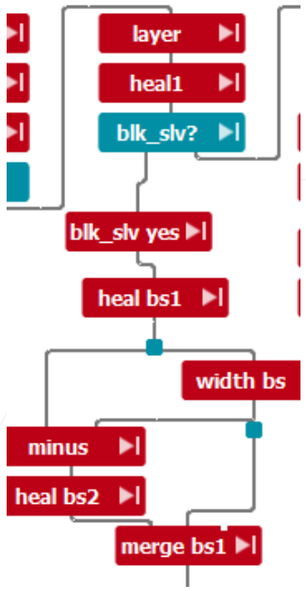
Combined sample



Loading a single GDS file
Selecting the cell of interest
Reflect Y as Mask

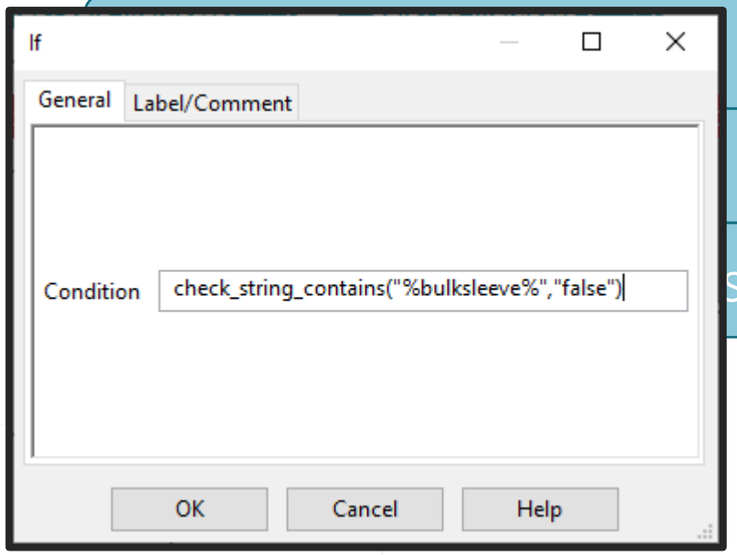
- one GDS pattern
- Bulk & Sleeve
- Fracture Resolution
- PEC Process
- Writing Resolution
- Beam Step Size
- Multipass Parameters

Combined sample

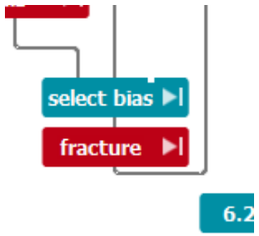


Extract the layers of interest
HEAL to layer 1
Check for Bulk and Sleeve condition

- one GDS pattern
- Bulk & Sleeve
- Fracture Resolution
- PEC Process



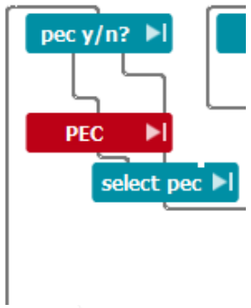
Combined sample



SELECT brings the branched flow back together
Result is passed over to the FRACTURE with the defined resolution

- one GDS pattern
- Bulk & Sleeve
- Fracture Resolution
- PEC Process
- Writing Resolution
- Beam Step Size
- Multipass Parameters

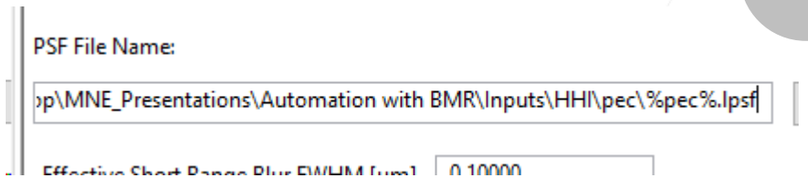
Combined sample



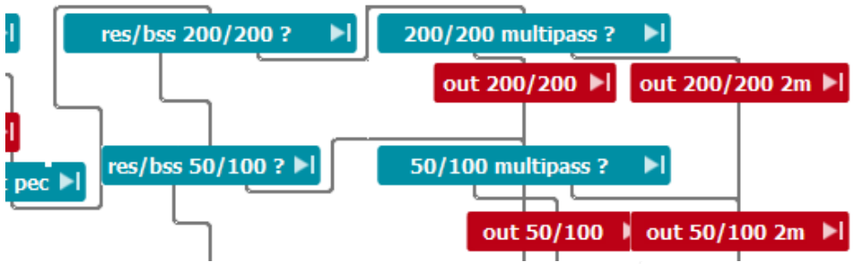
- one GDS pattern
- Bulk & Sleeve
- Fracture Resolution
- **PEC Process**
- Writing Resolution
- Beam Step Size
- Multipass Parameters

Check for the PEC process to be used

Reference to the PSF by using lpsf files



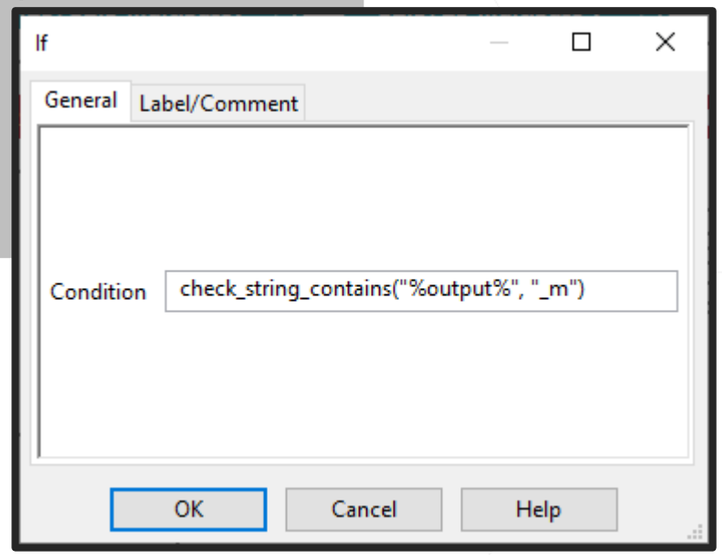
Combined sample



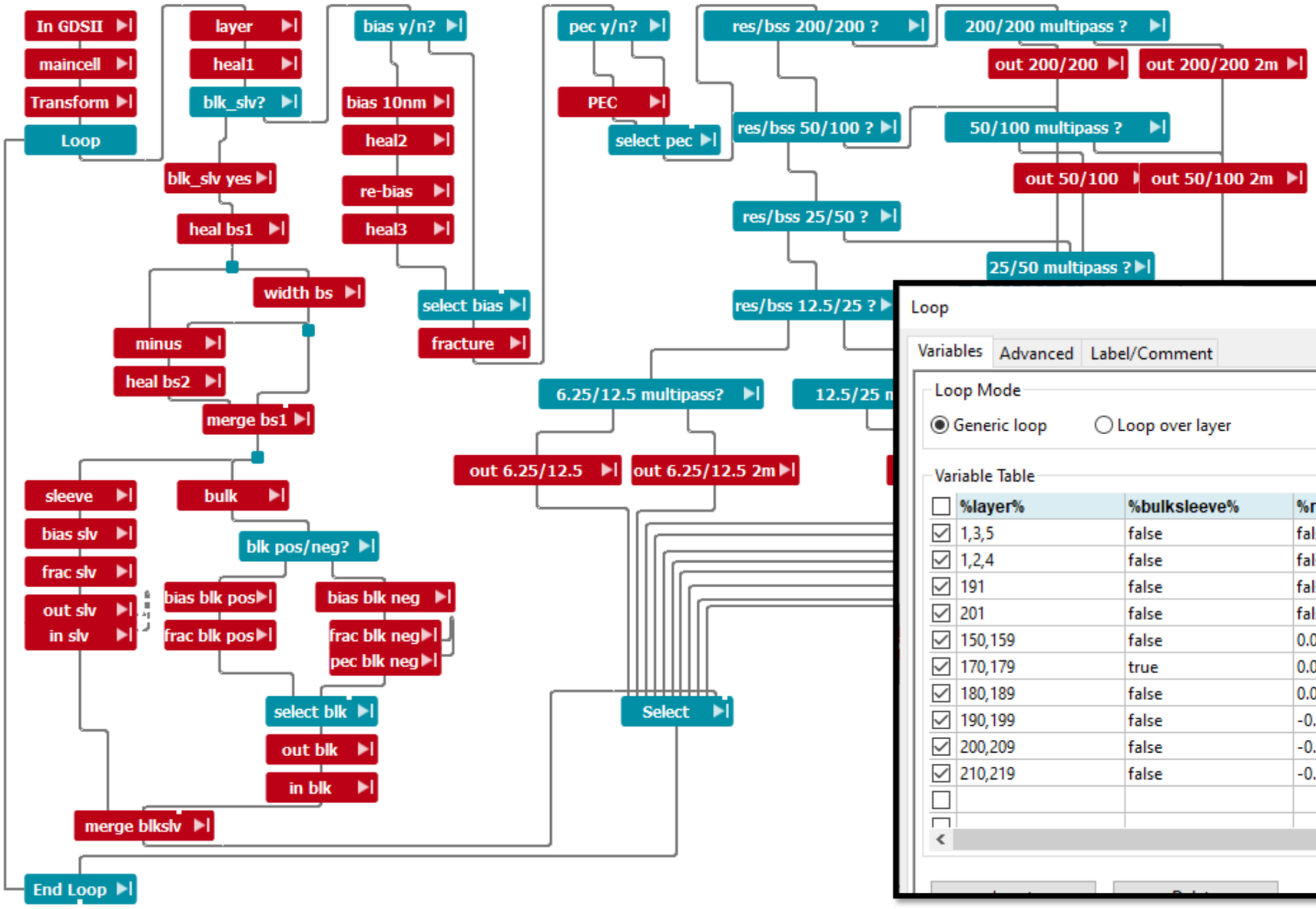
Cascade for resolution/bss checks

Move through the possible resolutions
(Fixed because of other export parameters)

Check for Multipass exposure
(part of the export file name)



- one GDS pattern
- Bulk & Sleeve
- Fracture Resolution
- PEC Process
- Writing Resolution
- Beam Step Size
- Multipass Parameters



Loop

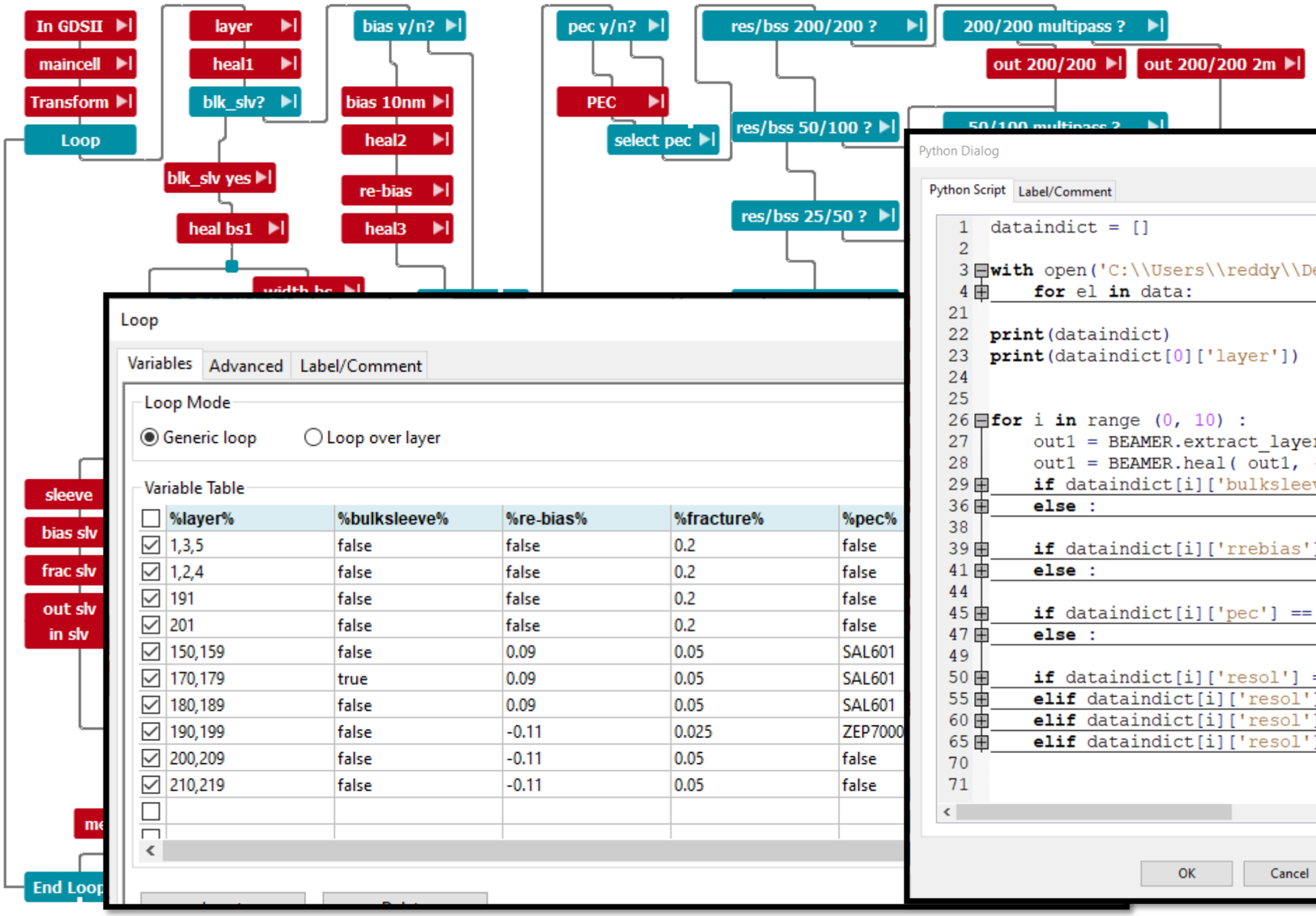
Variables | Advanced | Label/Comment

Loop Mode

(•) Generic loop ( ) Loop over layer

Variable Table

	%layer%	%bulksleeve%	%re-bias%	%fracture%	%pec%	%resol%
☑	1,3,5	false	false	0.2	false	0.2
☑	1,2,4	false	false	0.2	false	0.2
☑	191	false	false	0.2	false	0.2
☑	201	false	false	0.2	false	0.2
☑	150,159	false	0.09	0.05	SAL601	0.025
☑	170,179	true	0.09	0.05	SAL601	0.025
☑	180,189	false	0.09	0.05	SAL601	0.025
☑	190,199	false	-0.11	0.025	ZEP7000	0.0125
☑	200,209	false	-0.11	0.05	false	0.025
☑	210,219	false	-0.11	0.05	false	0.025
☐						



Loop

Variables | Advanced | Label/Comment

Loop Mode

(•) Generic loop ( ) Loop over layer

Variable Table

	%layer%	%bulksleeve%	%re-bias%	%fracture%	%pec%
☑	1,3,5	false	false	0.2	false
☑	1,2,4	false	false	0.2	false
☑	191	false	false	0.2	false
☑	201	false	false	0.2	false
☑	150,159	false	0.09	0.05	SAL601
☑	170,179	true	0.09	0.05	SAL601
☑	180,189	false	0.09	0.05	SAL601
☑	190,199	false	-0.11	0.025	ZEP7000
☑	200,209	false	-0.11	0.05	false
☑	210,219	false	-0.11	0.05	false
☐					

Python Dialog

Python Script | Label/Comment

```
1  dataindict = []
2
3  with open('C:\\Users\\reddy\\Desktop\\Code\\control.txt', 'r') as data:
4      for el in data:
21
22 print(dataindict)
23 print(dataindict[0]['layer'])
24
25
26 for i in range (0, 10) :
27     out1 = BEAMER.extract_layer( in1, { 'ExtentMode' : 'Default', 'Cell
28     out1 = BEAMER.heal( out1, { 'TargetLayer' : '1(0)', 'SoftFrame' : 0
29     if dataindict[i]['bulksleeve'] == 1 :
36     else :
38
39     if dataindict[i]['rrebias'] == 0 :
41     else :
44
45     if dataindict[i]['pec'] == 0 :
47     else :
49
50     if dataindict[i]['resol'] == 0.2 :
55     elif dataindict[i]['resol'] == 0.05 :
60     elif dataindict[i]['resol'] == 0.025 :
65     elif dataindict[i]['resol'] == 0.0125 :
70
71
```

- Modules
- Evaluations
- LayoutPy
- Math
- Examples
- Snippets

OK | Cancel | Help | Open Viewer | Preview

Python for algorithmic pattern generation

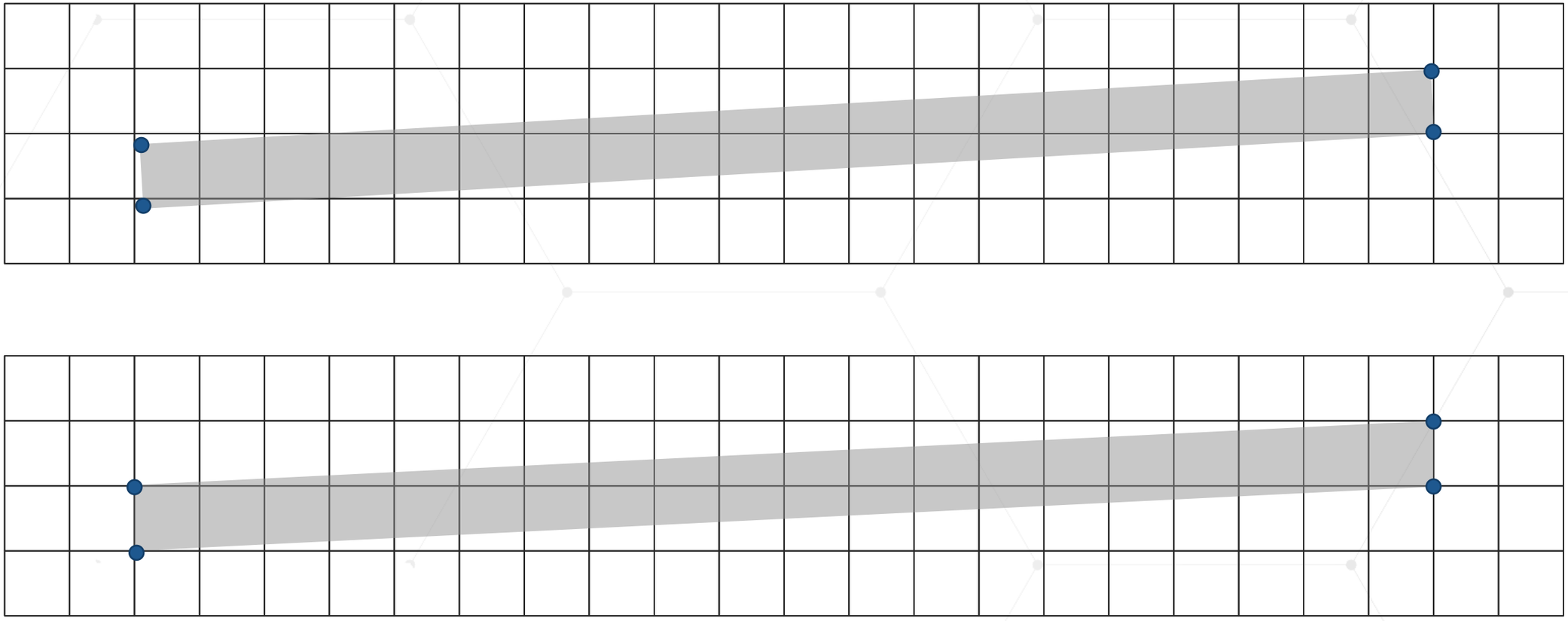
Introduction

- Augmented and Virtual Reality devices are a big topic these days and manufacturing these with high precision can be a challenge due to the nature of the used gratings and our ebeam world working on cartesian coordinates.
- This talk will plant some small ideas to make life easier with such tasks.
  - Problem
  - Proposal
  - Presentation

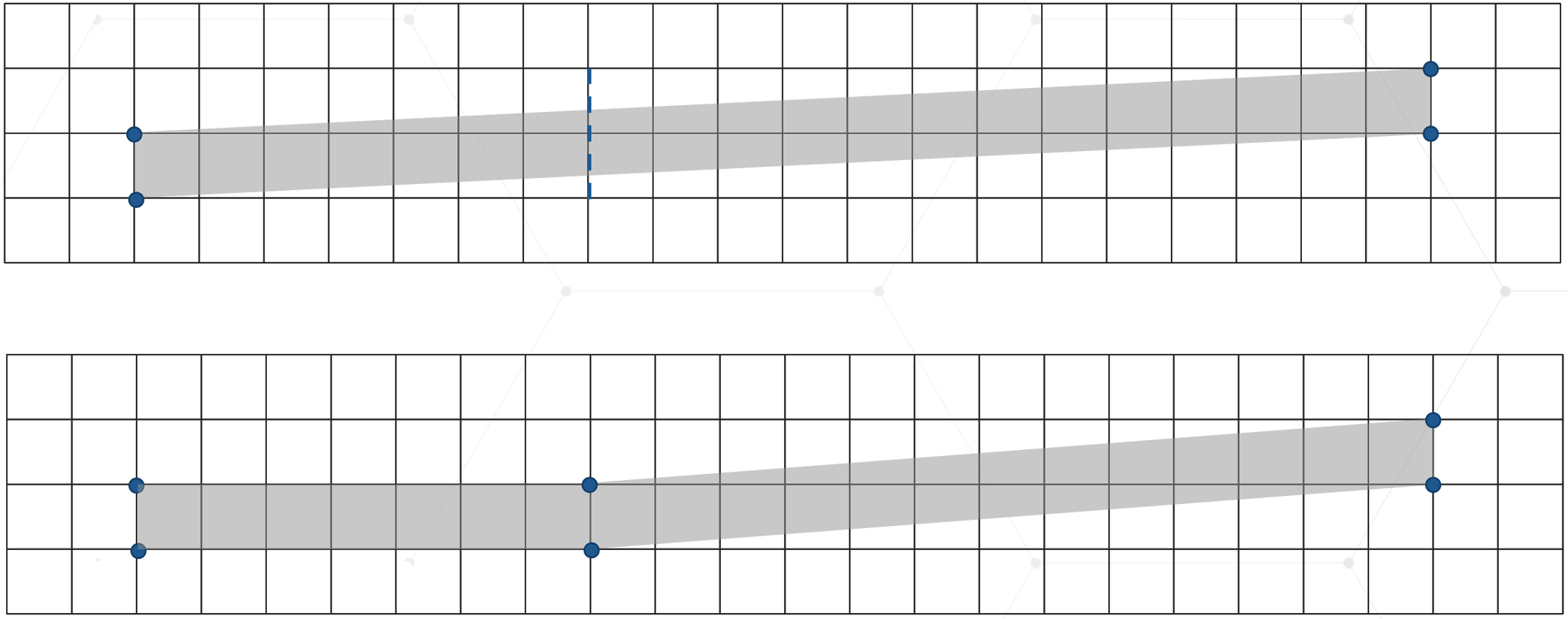
Problem

- A common concept for tilted gratings is using a designed grating with the desires line width and pitch and apply a rotation on these.
- These gratings will then be cut into shapes desired and being exposed by the tools.
- In these steps already some tripstone can surface:
  - When tilting the grating the verticies of the lines snap to the cartesian coordinates and can change line width and pitch of the grating introducing a multitude of pitches destroying the design intent
  - Trimming the grating into shape will also cause grid snapping adding to the challenges
  - Export to the tools will introduce more cuts causing more potential issues

Rectangle snapping to database grid



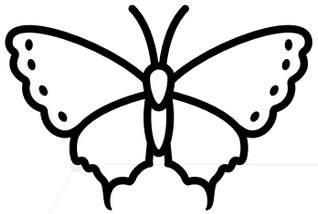
Polygon fractured and snapping to database grid



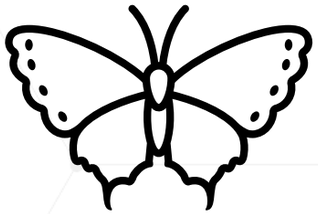
Proposal

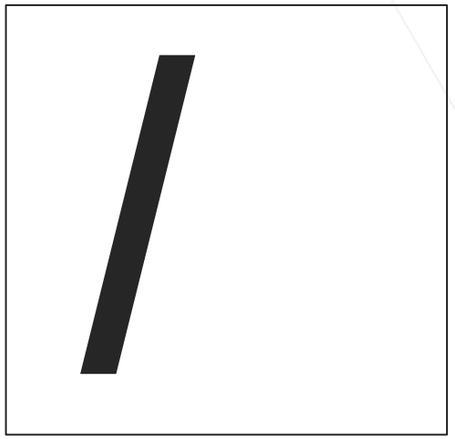
- Knowing capabilities of the tools the pattern can be generated algorithmically ensuring that best possible outcome on this situation.
- Field sizes and subfield sized will introduce fractures to the pattern. Taking these into account the data an be designed such to have these cuts at desired position.

- Take information from target design

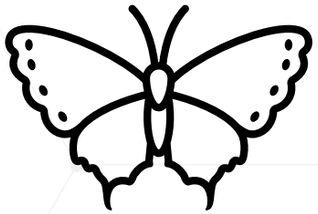


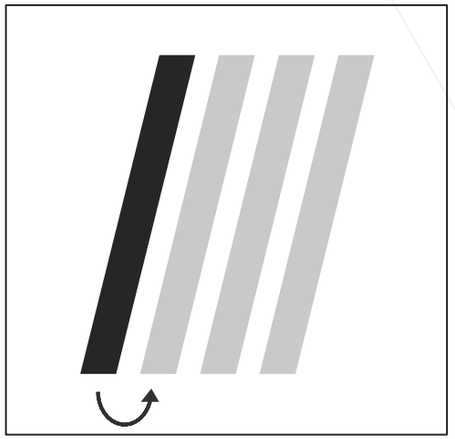
- Take information from target design
- Compute ideal single line segment



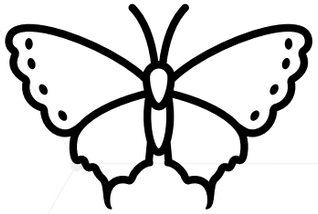


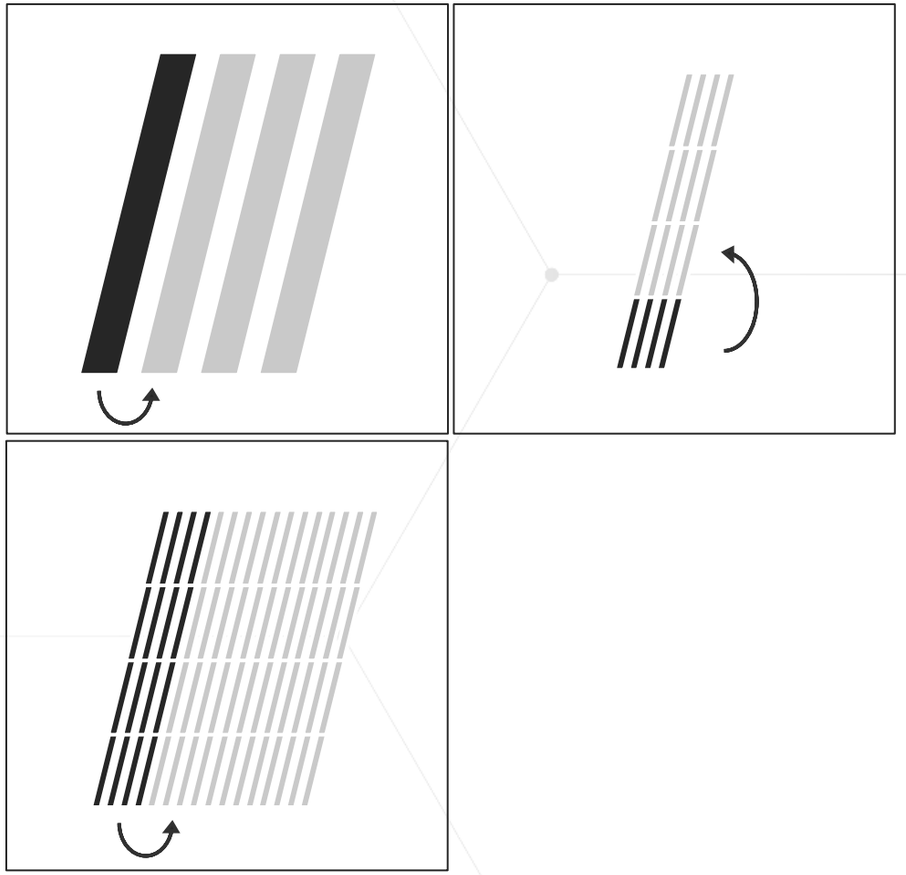
- Take information from target design
- Compute ideal single line segment
- Repeat until subfield is formed



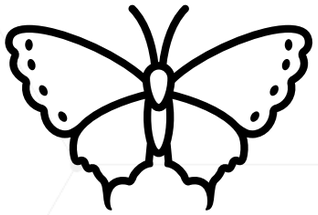


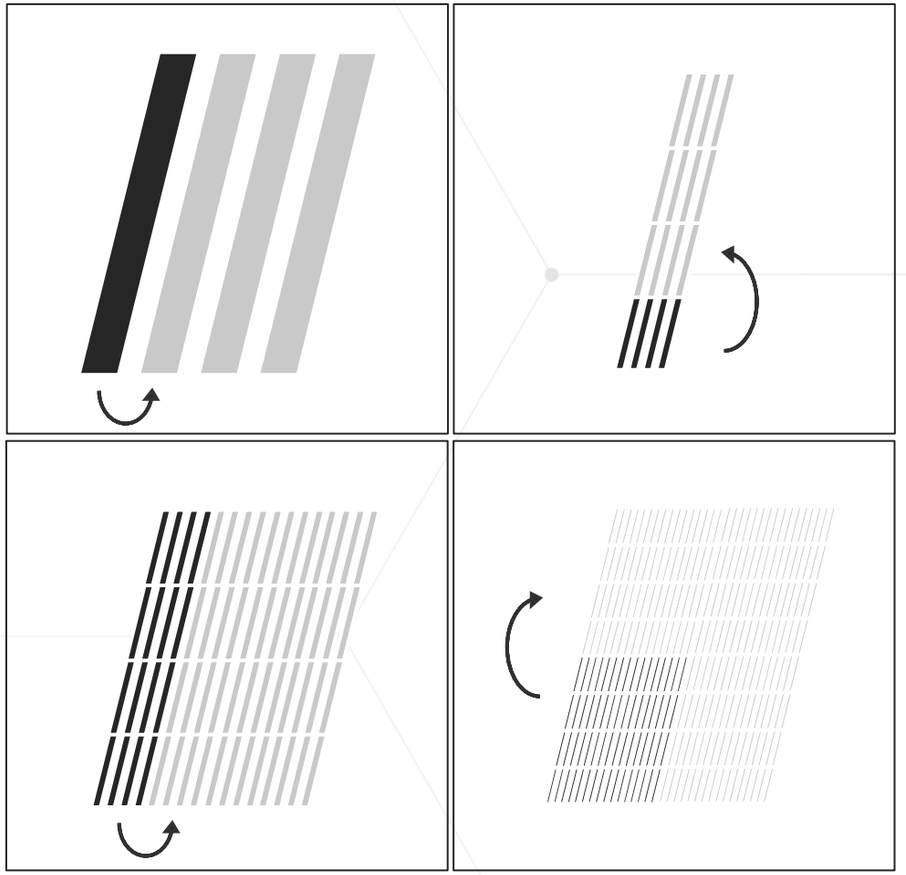
- Take information from target design
- Compute ideal single line segment
- Repeat until subfield is formed
- Repeat until mainfield is formed



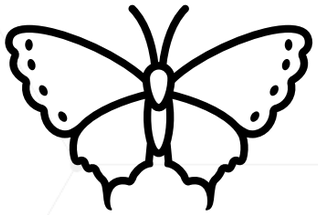


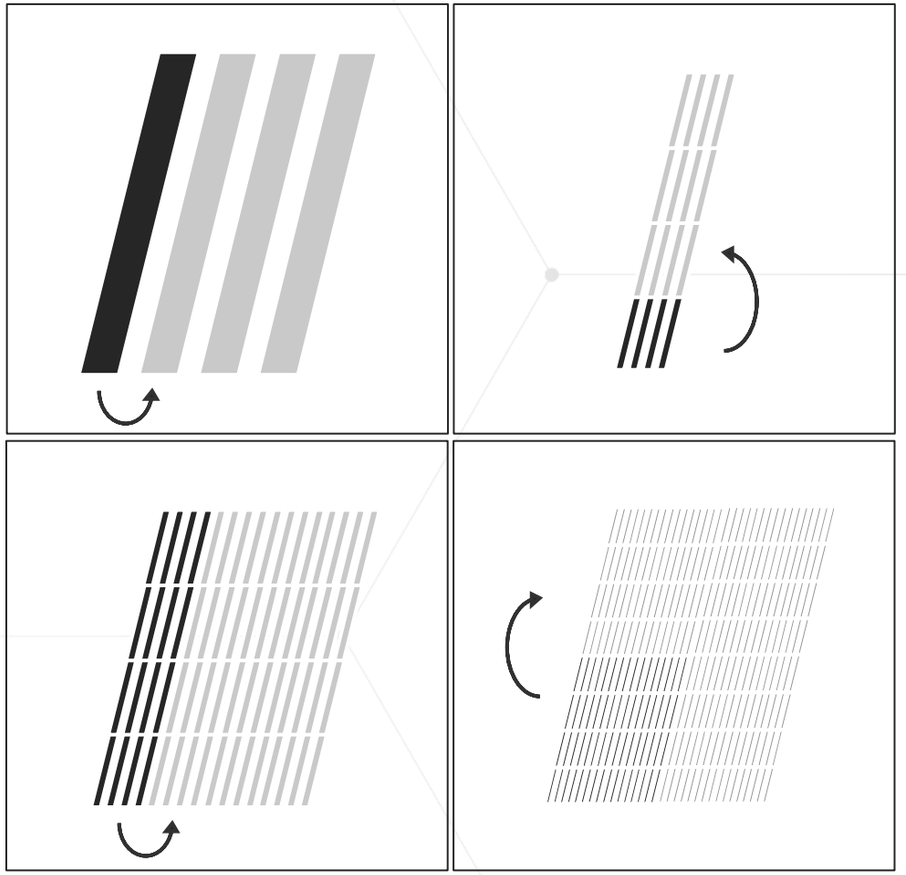
- Take information from target design
- Compute ideal single line segment
- Repeat until subfield is formed
- Repeat until mainfield is formed
- Repeat until entire pattern area is filled

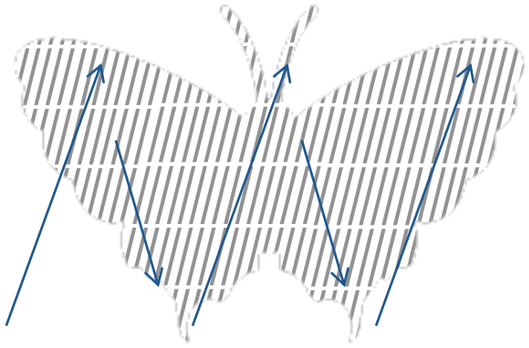




- Take information from target design
- Compute ideal single line segment
- Repeat until subfield is formed
- Repeat until mainfield is formed
- Repeat until entire pattern area is filled
- Extract desired target from the gratings…





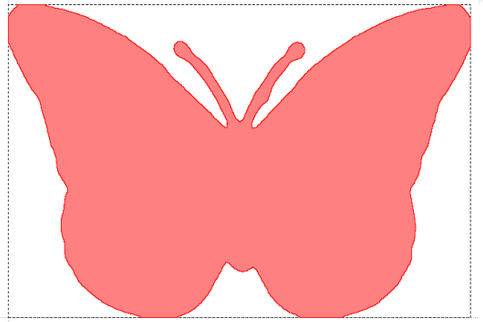


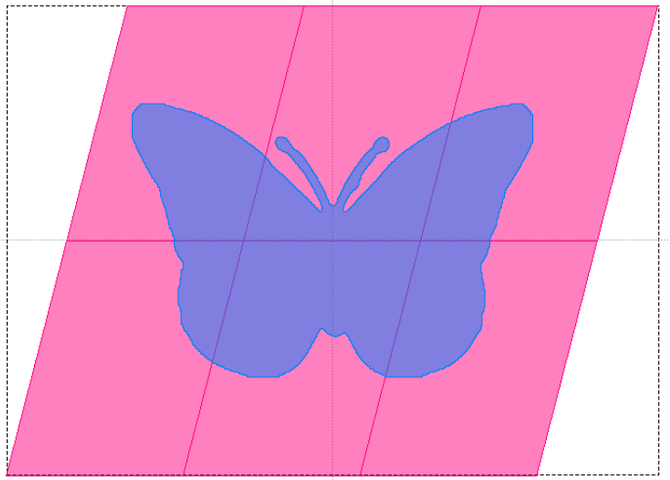
Presentation

From parameters to

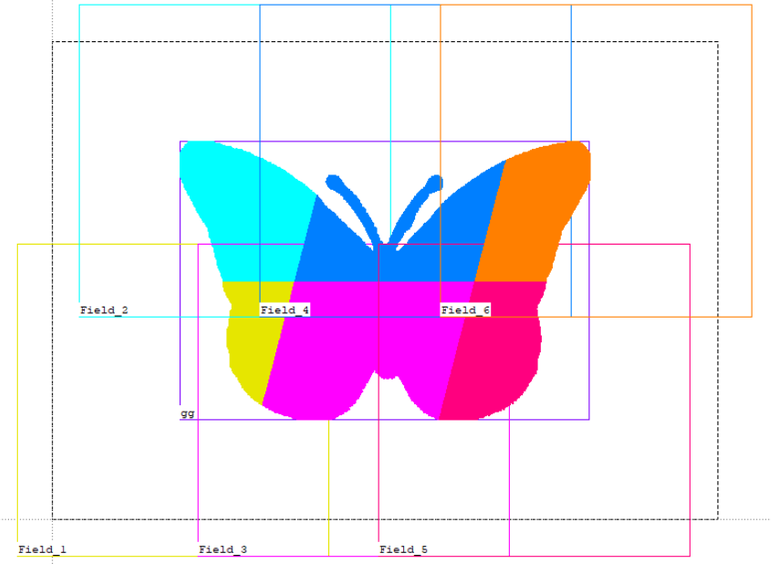
…

final exposure data



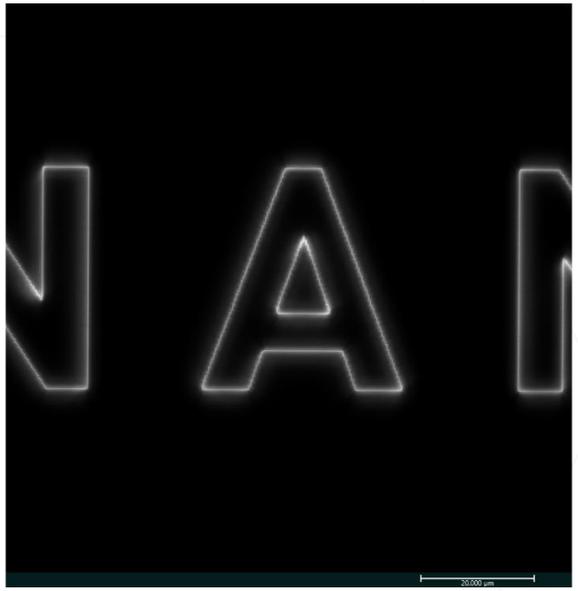


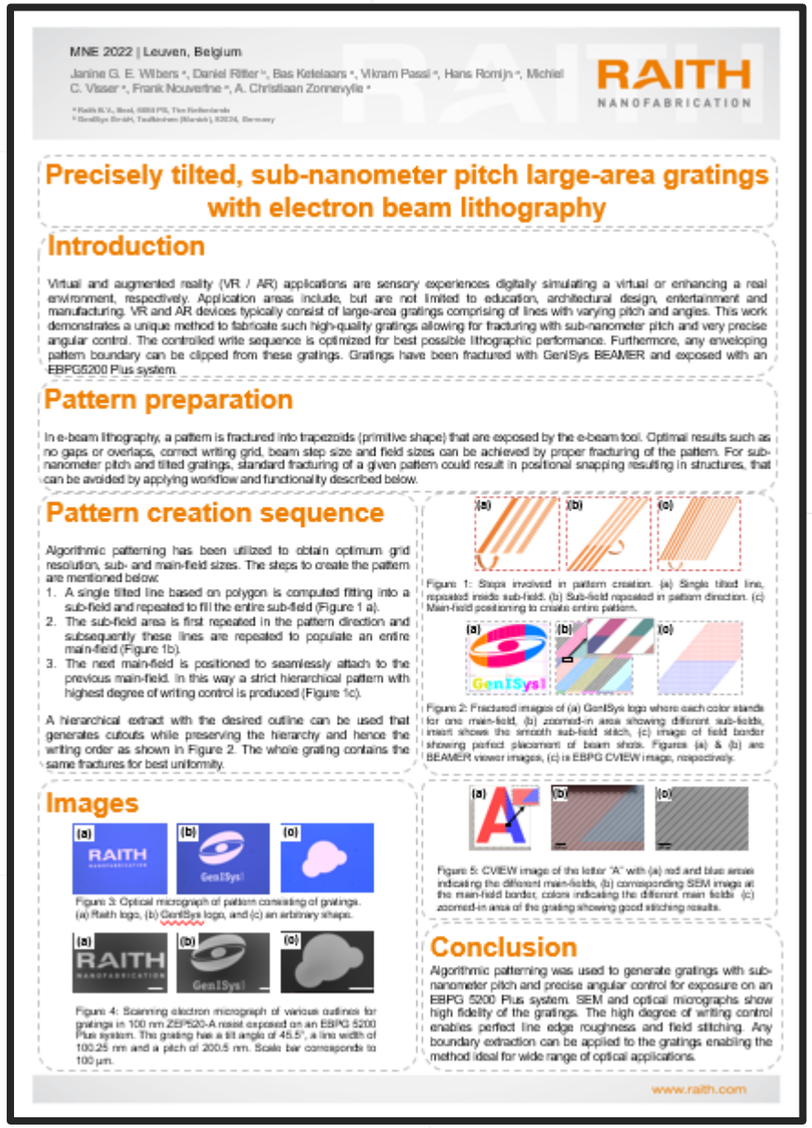
```
from LAYOUTpy import *
import math

#
# GRATING PARAMETERS
#

angle =75.5                     # Angle of grating in [°]
perpendicular_width = 0.070     # Line Width perpendicular [um]
perpendicular_pitch = 0.19321   # Pitch perpendicular [um]

print("\n#################################")
print ("perpendicular linewidth :: ", perpendicular_width)
print ("perpendicular pitch :: ", perpendicular_pitch)


#
# TOOL SETTINGS
#
subfieldsize = 4        # Subfield Size in [ um ]
subfield_usage = 0.9    # factor of subfield usage
mainfieldsize = 800    # Mainfield Size in [ um ]
mainfield_usage = 0.9   # factor of subfield usage
res = 0.001

```



Let’s go live …

- Visit also a poster where this approach for pattern generation was used in a similar form
- Darkfield Images show good results on field borders





MNE 2022 | Leuven, Belgium

Janine G. E. Wilbers [a], Daniel Ritter [a], Bas Ketelaars [a], Vikram Passi [a], Hans Romijn [a], Michiel C. Visser [a], Frank Nouwertne [a], A. Christiaan Zonnevylle [a]

RAITH NANOFABRICATION

Precisely tilted, sub-nanometer pitch large-area gratings with electron beam lithography

Introduction

Virtual and augmented reality (VR / AR) applications are sensory experiences digitally simulating a virtual or enhancing a real environment, respectively. Application areas include, but are not limited to education, architectural design, entertainment and manufacturing. VR and AR devices typically consist of large-area gratings comprising of lines with varying pitch and angles. This work demonstrates a unique method to fabricate such high-quality gratings allowing for fracturing with sub-nanometer pitch and very precise angular control. The controlled write sequence is optimized for best possible lithographic performance. Furthermore, any enveloping pattern boundary can be clipped from these gratings. Gratings have been fractured with GenISys BEAMER and exposed with an EBPG5200 Plus system.

Pattern preparation

In e-beam lithography, a pattern is fractured into trapezoids (primitive shape) that are exposed by the e-beam tool. Optimal results such as no gaps or overlaps, correct writing grid, beam step size and field sizes can be achieved by proper fracturing of the pattern. For sub-nanometer pitch and tilted gratings, standard fracturing of a given pattern could result in positional snapping resulting in structures, that can be avoided by applying workflow and functionality described below.

Pattern creation sequence

Algorithmic patterning has been utilized to obtain optimum grid resolution, sub- and main-field sizes. The steps to create the pattern are mentioned below:

1. A single tilted line based on polygon is computed fitting into a sub-field and repeated to fill the entire sub-field (Figure 1 a).
2. The sub-field area is first repeated in the pattern direction and subsequently these lines are repeated to populate an entire main-field (Figure 1b).
3. The next main-field is positioned to seamlessly attach to the previous main-field. In this way a strict hierarchical pattern with highest degree of writing control is produced (Figure 1c).

A hierarchical extract with the desired outline can be used that generates cutouts while preserving the hierarchy and hence the writing order as shown in Figure 2. The whole grating contains the same fractures for best uniformity.

Figure 1: Steps involved in pattern creation. (a) Single tilted line, repeated inside sub-field, (b) Sub-field repeated in pattern direction. (c) Main-field positioning to create entire pattern.

Figure 2: Fractured images of (a) GenISys logo where each color stands for one main-field, (b) zoomed-in area showing different sub-fields, insert shows the smooth sub-field stitch, (c) image of field border showing perfect placement of beam shots. Figures (a) & (b) are BEAMER viewer images, (c) is EBPG CVIEW image, respectively.

Images

Figure 3: Optical micrograph of pattern consisting of gratings. (a) Raith logo, (b) GenISys logo, and (c) an arbitrary shape.

Figure 4: Scanning electron micrograph of various outlines for gratings in 100 nm ZEP520-A resist exposed on an EBPG 5200 Plus system. The grating has a tilt angle of 45.5°, a line width of 100.25 nm and a pitch of 200.5 nm. Scale bar corresponds to 100 µm.

Figure 5: CVIEW image of the letter "A" with (a) red and blue areas indicating the different main-fields, (b) corresponding SEM image at the main-field border, colors indicating the different main fields (c) zoomed-in area of the grating showing good stitching results.

Conclusion

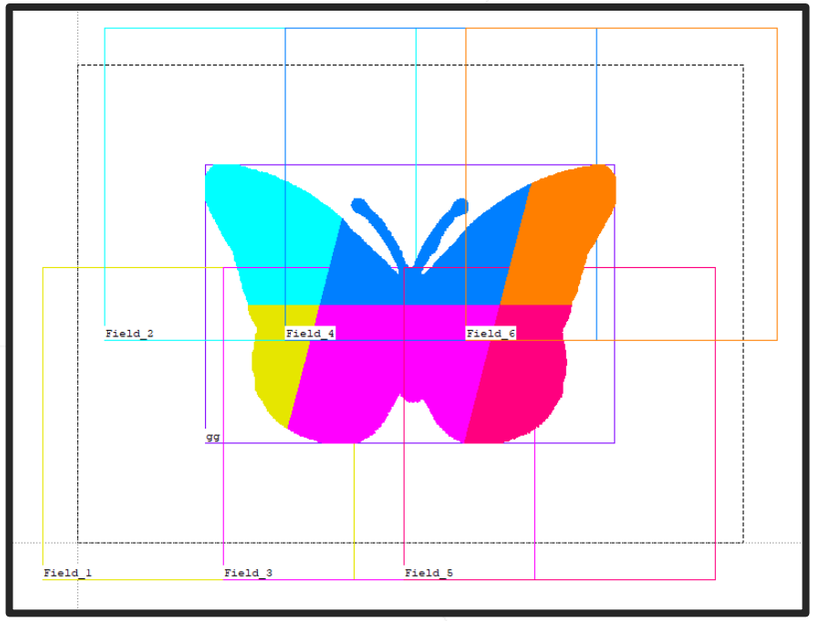
Algorithmic patterning was used to generate gratings with sub-nanometer pitch and precise angular control for exposure on an EBPG 5200 Plus system. SEM and optical micrographs show high fidelity of the gratings. The high degree of writing control enables perfect line edge roughness and field stitching. Any boundary extraction can be applied to the gratings enabling the method ideal for wide range of optical applications.

www.raith.com

Summary

- Reviewing pattern design from a perspective of tool capabilities for application of gratings for AR and VR devices
- Showcased a simple approach in design and usage for grating applications

```
from LAYOUTpy import *
import math

#
# GRATING PARAMETERS
#

angle =75.5                     # Angle of grating in [°]
perpendicular_width = 0.070     # Line Width perpendicular [um]
perpendicular_pitch = 0.19321   # Pitch perpendicular [um]

print("\n###################################")
print ("perpendicular linewidth :: ", perpendicular_width)
print ("perpendicular pitch :: ", perpendicular_pitch)


#
# TOOL SETTINGS
#
subfieldsize = 4        # Subfield Size in [ um ]
subfield_usage = 0.9    # factor of subfield usage
mainfieldsize = 800    # Mainfield Size in [ um ]
mainfield_usage = 0.9   # factor of subfield usage
res = 0.001

```



GenISys
Advancing the Standard

Thank You!

support@genisys-gmbh.com

Headquarters
GenISys GmbH
Eschenstr. 66
D-82024 Taufkirchen (Munich)
GERMANY
+49-(0)89-3309197-60
+49-(0)89-3309197-61
info@genisys-gmbh.com

USA Office
GenISys Inc.
P.O. Box 410956
San Francisco, CA
94141-0956
USA
+1 (408) 353-3951
usa@genisys-gmbh.com

Japan / Asia Pacific Office
GenISys K.K.
German Industry Park
1-18-2 Hakusan Midori-ku
Yokohama 226-0006
JAPAN
+81 (0)45-530-3306
+81 (0)45-532-6933
apsales@genisys-gmbh.com