GenISys
Advancing the Standard

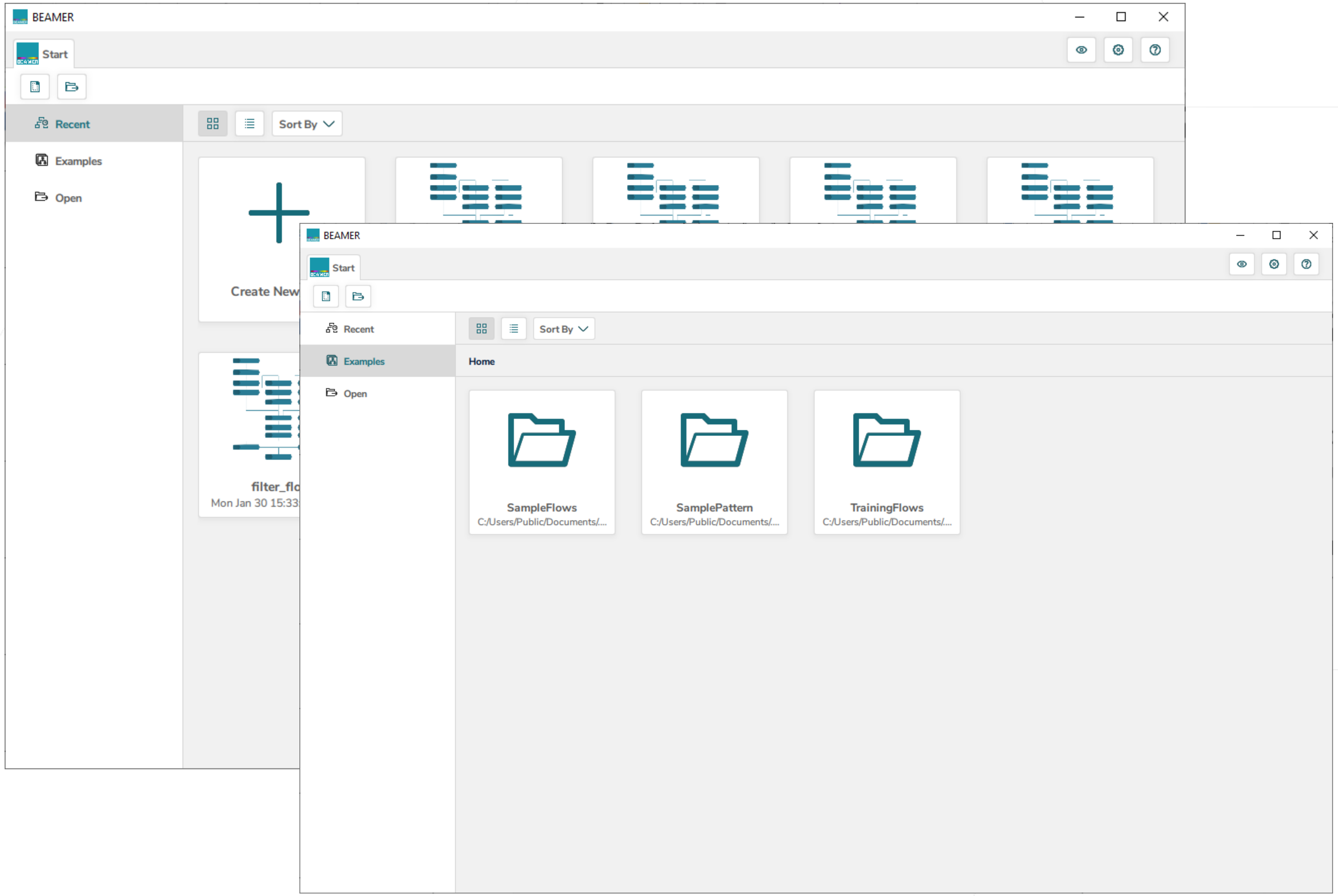
BEAMER

What‘s new

GenISys
Advancing the Standard

User interface

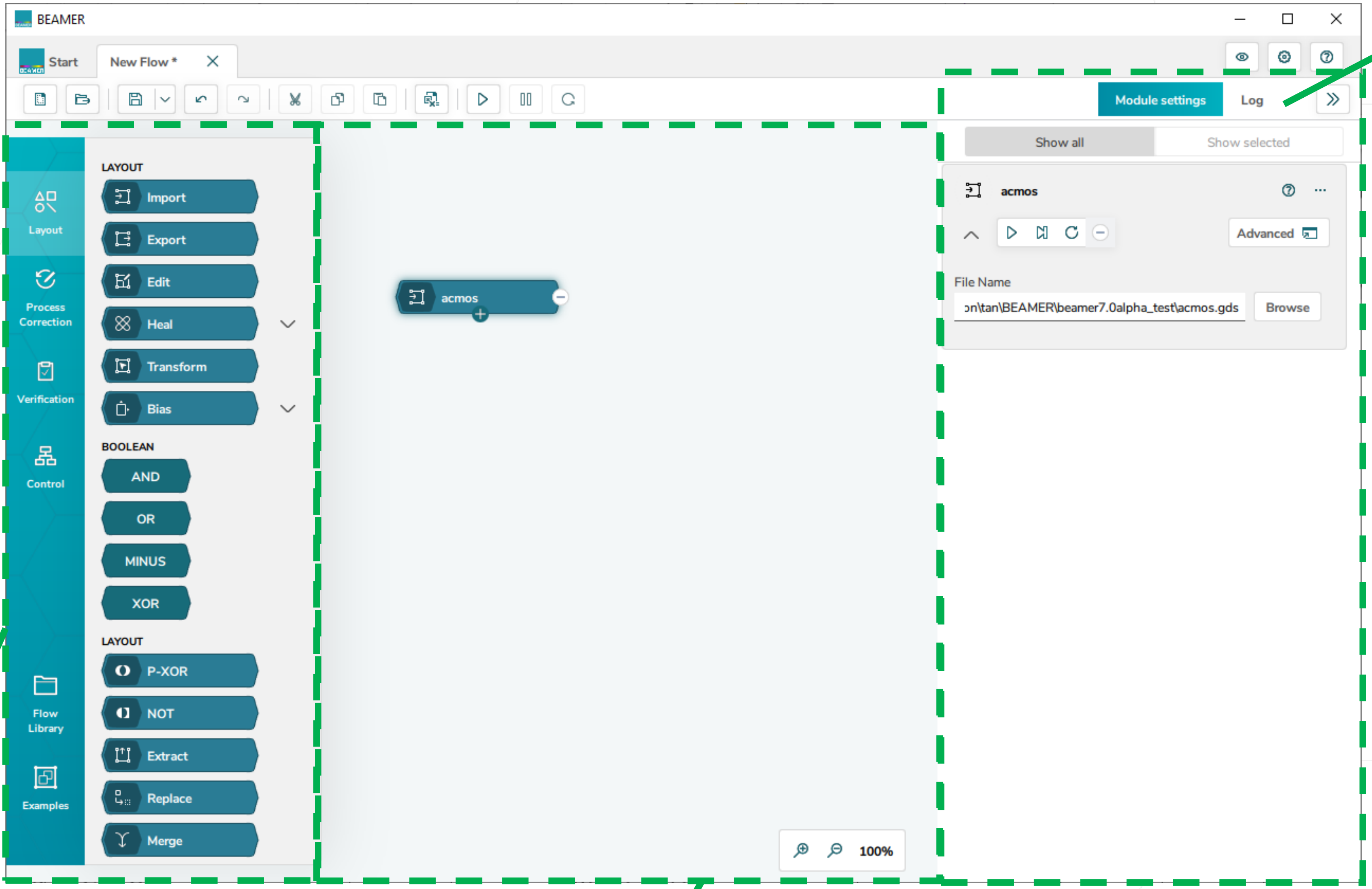
Start up



The start up dialog provides you a selection of recently used flow and the example flows provided with every version.

GenISys
Advancing the Standard

Main window



Log information

Quick access

Module selection

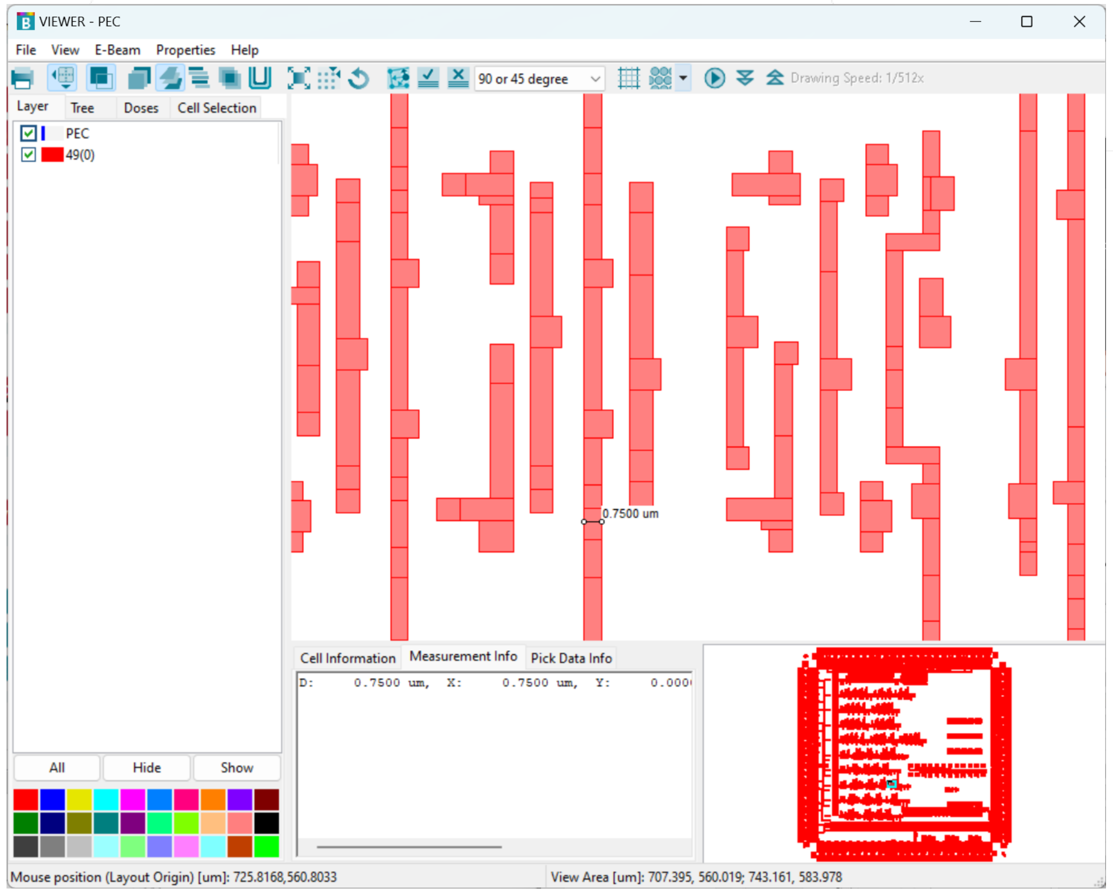
Flow area

GenISys
Advancing the Standard

VIEWER

The viewer UI has been upgrade for a cleaner and more functional design.

BEAMER 6.4

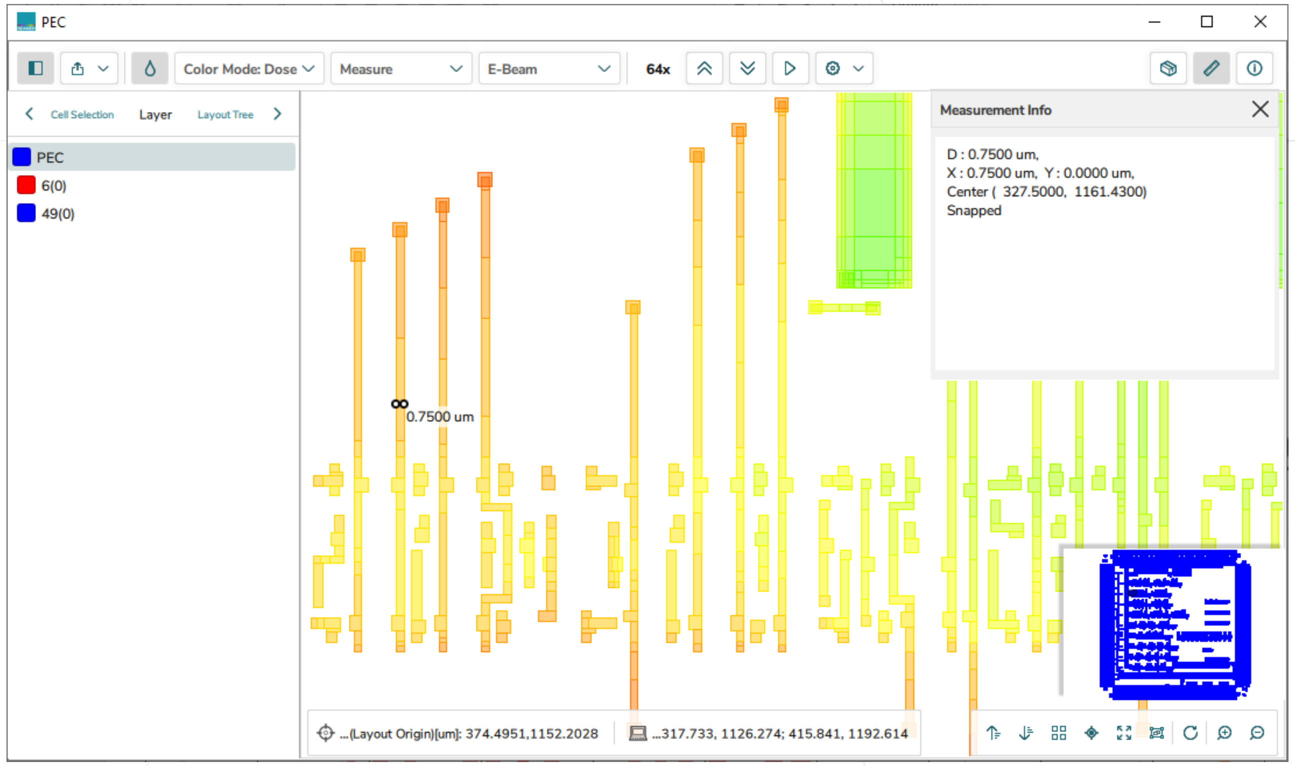


D: 0.7500 um, X: 0.7500 um, Y: 0.000

Mouse position (Layout Origin) [um]: 725.8168,560.8033

View Area [um]: 707.395, 560.019; 743.161, 583.978

BEAMER 7.0



D : 0.7500 um,
X : 0.7500 um, Y : 0.0000 um,
Center (327.5000, 1161.4300)
Snapped

...(Layout Origin)[um]: 374.4951,1152.2028

...317.733, 1126.274; 415.841, 1192.614

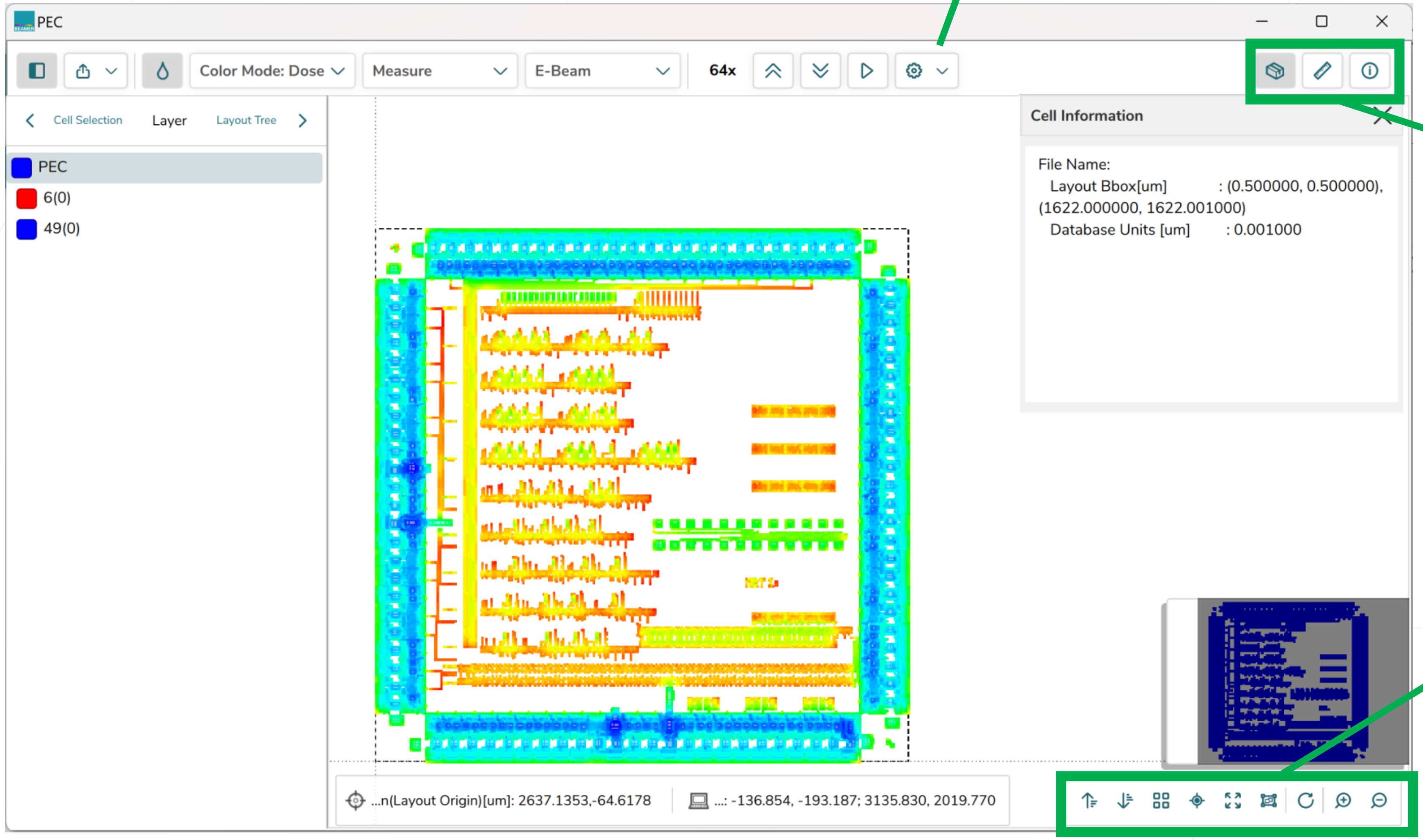
GenISys
Advancing the Standard

VIEWER

Settings menu contains all VIEWER setting

Cell info / Measurement info / Pick info

Quick navigation menu (Hierarchy control +- / Grid / Go-to / Fit all / Minimap / Undo view change / Zoom)

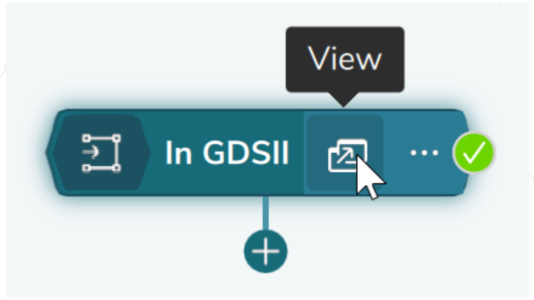


File Name:
Layout Bbox[um] : (0.500000, 0.500000), (1622.000000, 1622.001000)
Database Units [um] : 0.001000

...n(Layout Origin)[um]: 2637.1353,-64.6178
...: -136.854, -193.187; 3135.830, 2019.770

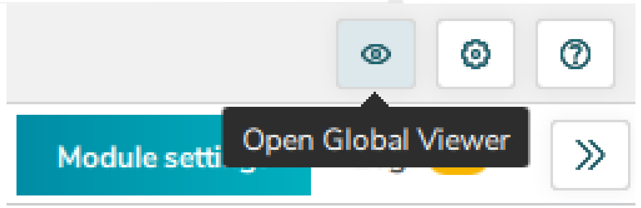
VIEWER

Local VIEWER



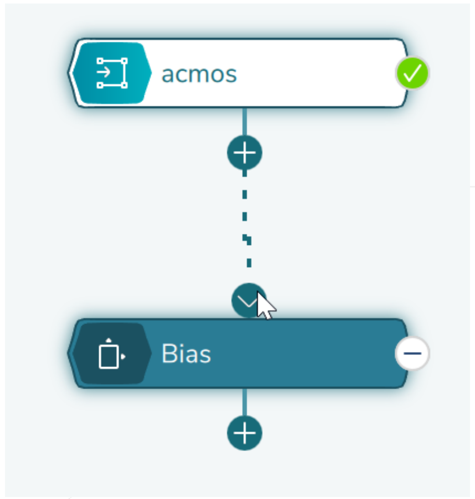
- This VIEWER shows the results of the module
- The View doesn’t depend on the module selection

Global/Multi VIEWER

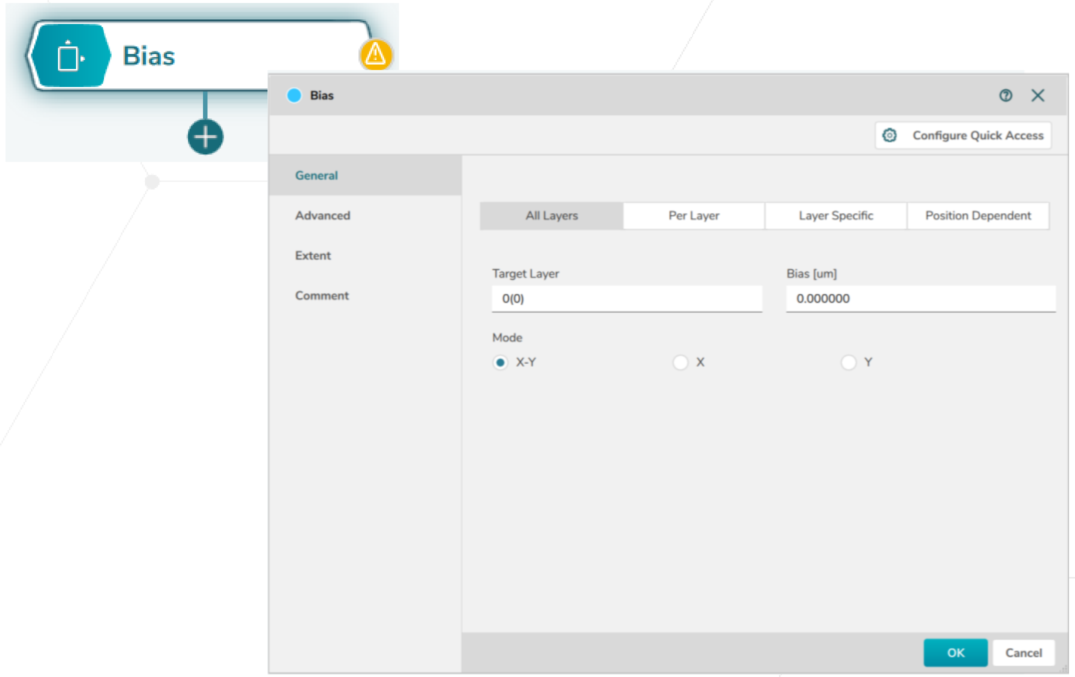


- This VIEWER shows the results of the selected module
- The view changes with module selection
- Multiple modules can be selected to be viewed at the same time.

Module states and interactions

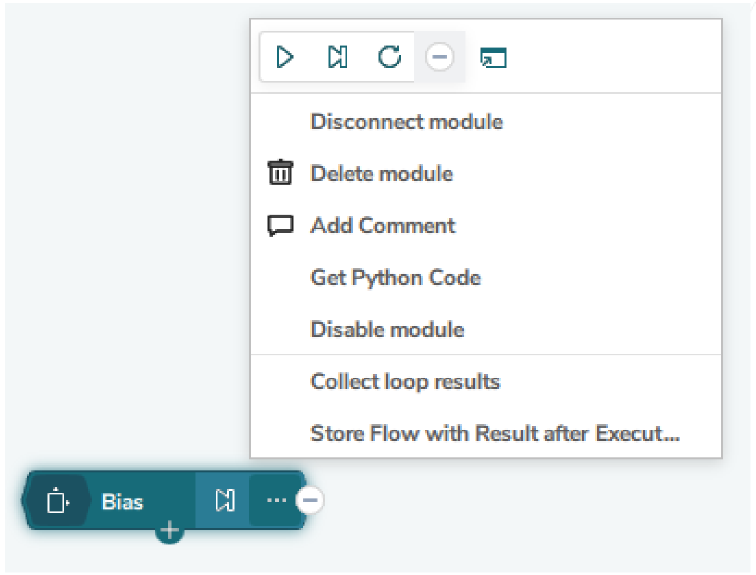


Connect modules by dragging the "+" on the following module. Disconnect the module by moving the connector off the module.



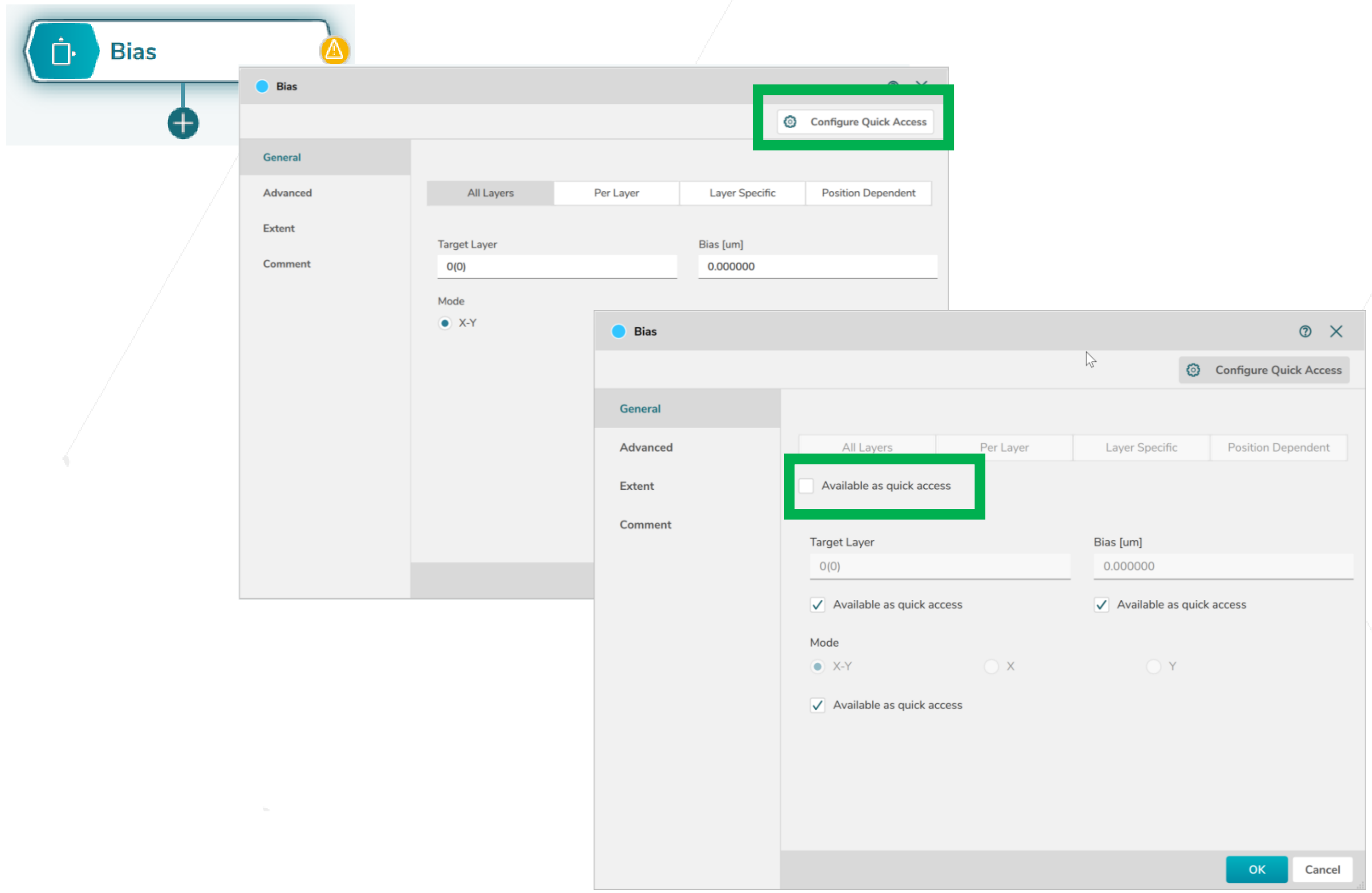
Target Layer: 0(0)
Bias [um]: 0.000000
Mode: [✓ X-Y] X Y

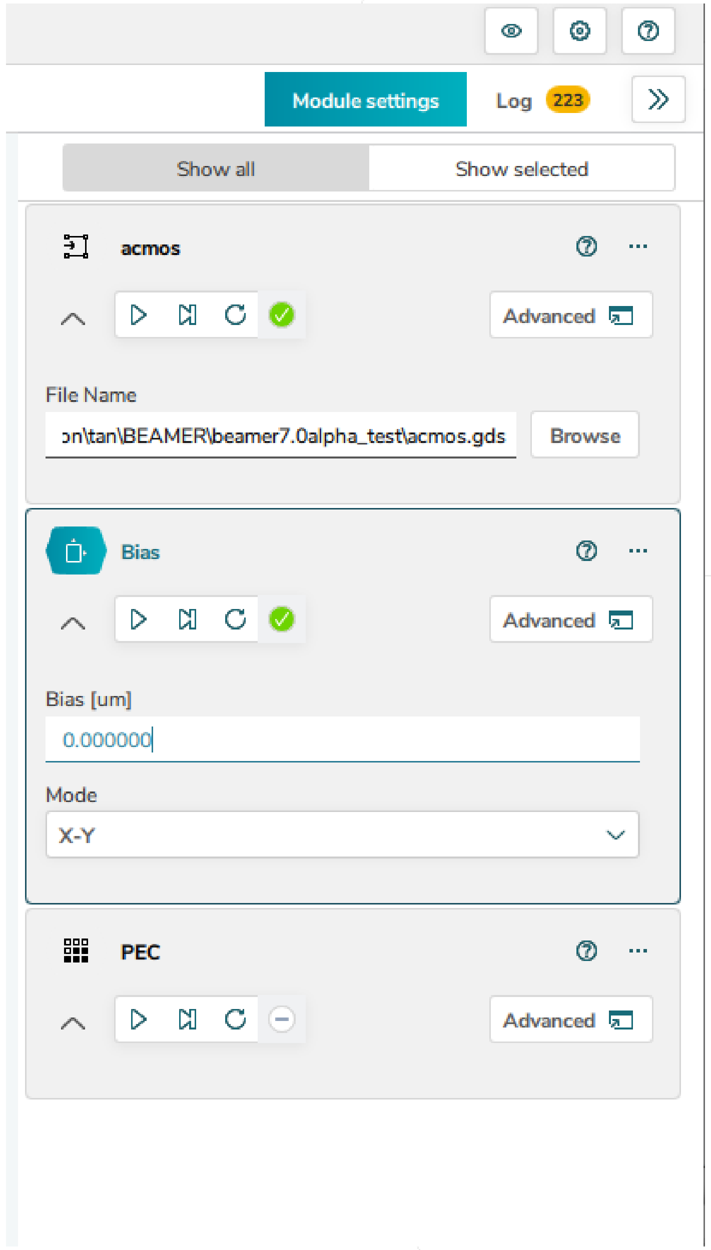
Double click on module name opens advanced settings



Context menu of the module

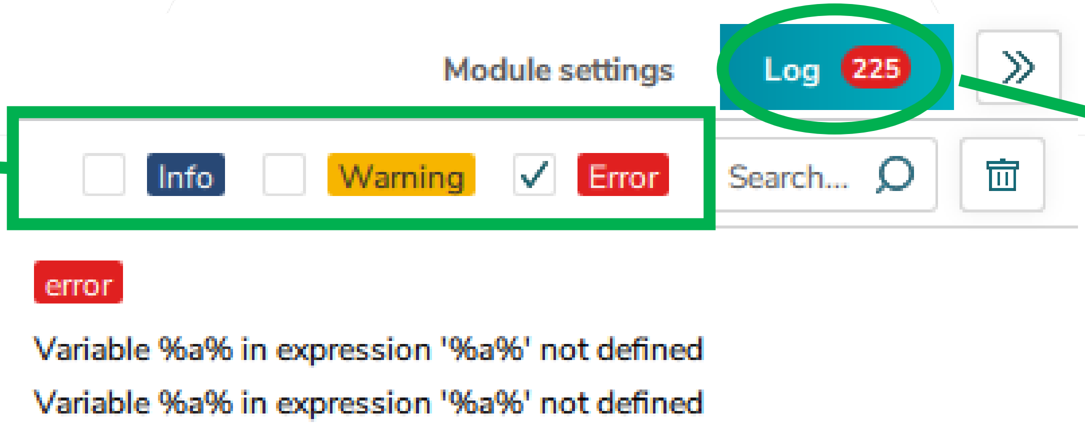
Quick access definition





Module states and interactions

Filter options for log information



```
Variable %a% in expression '%a%' not defined
Variable %a% in expression '%a%' not defined
```

Notification for log info

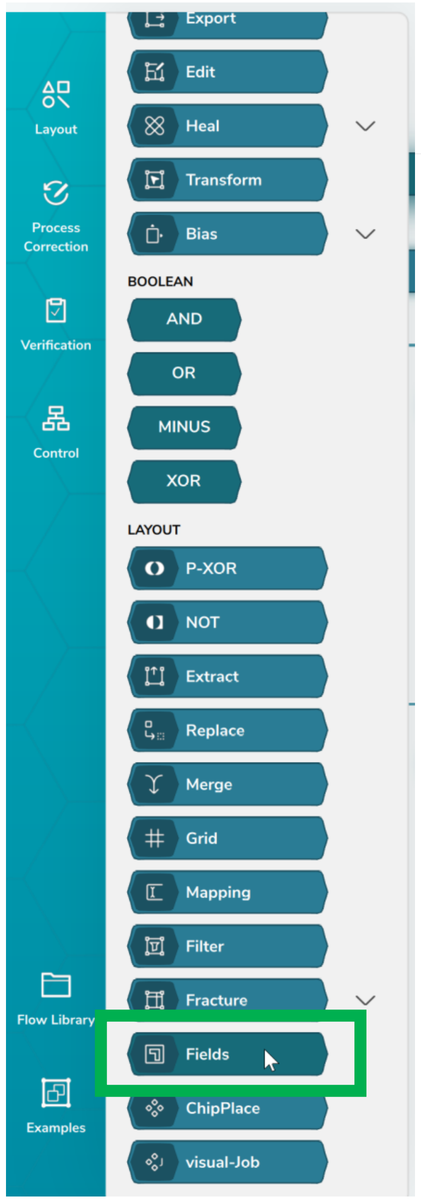
- red = error
- yellow = warning
- blue = info

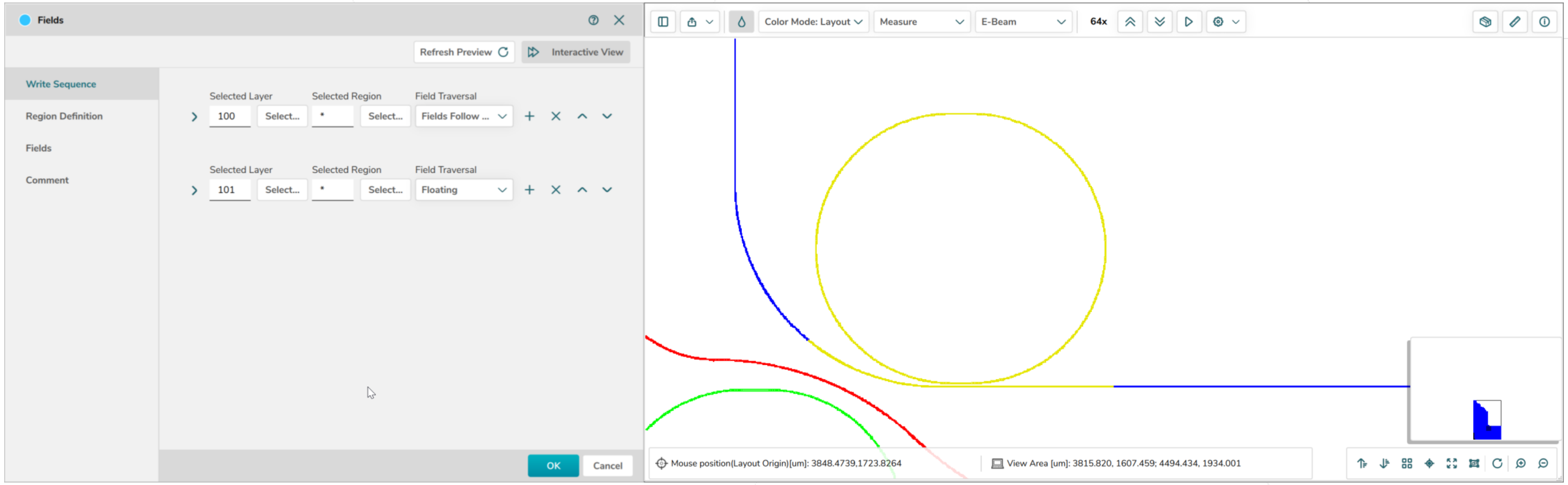
Improvements

GenISys
Advancing the Standard

Fields module

New field sorting module – for advanced Region and field sorting applications





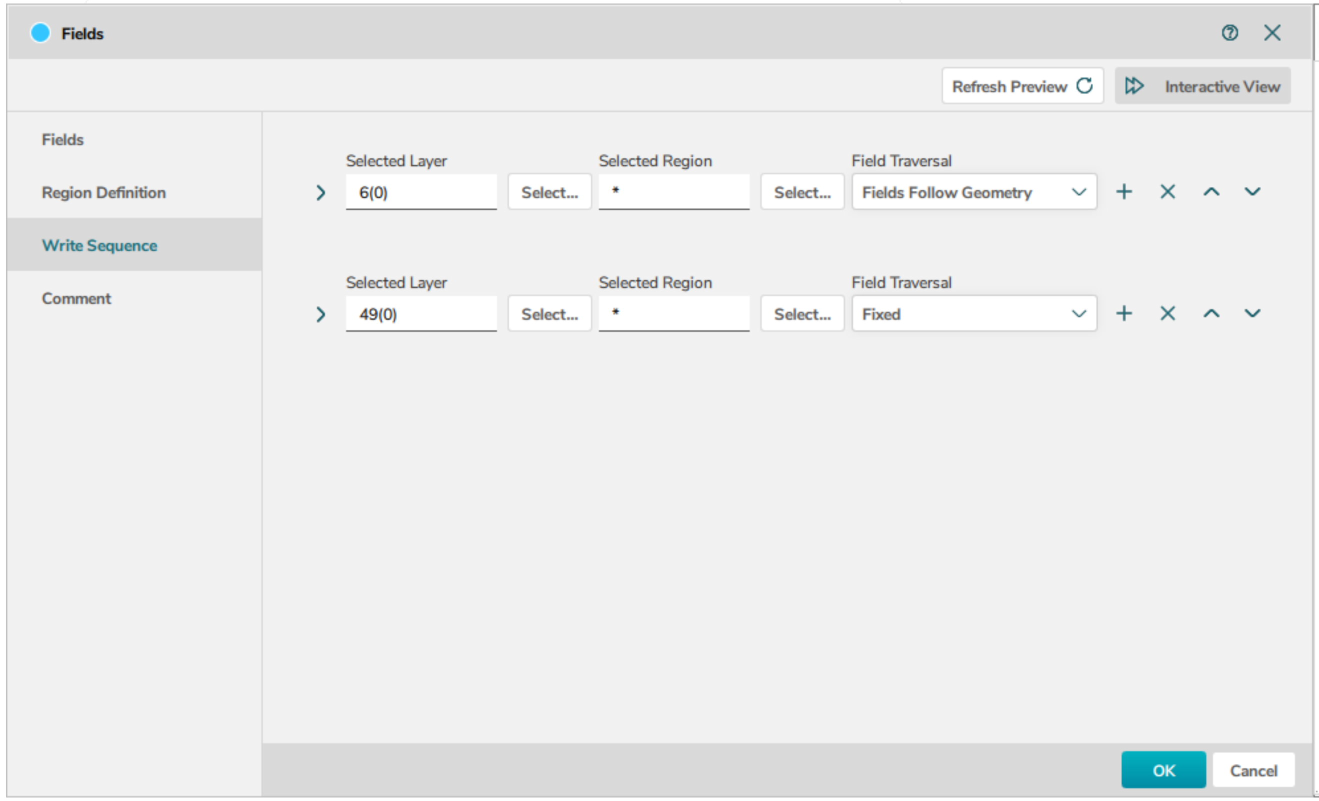
Selected Layer		Selected Region		Field Traversal
100	Select...	*	Select...	Fields Follow ...
101	Select...	*	Select...	Floating

Mouse position(Layout Origin)[um]: 3848.4739,1723.8264
View Area [um]: 3815.820, 1607.459; 4494.434, 1934.001

Fields module

New field sorting module – for advanced Region and field sorting applications

- The fields module can create field and region/ sub fields within a layout
- During the system specific export, the user makes use of this structure via cell to field / cell to region/SF (depending on the ebeam system format)



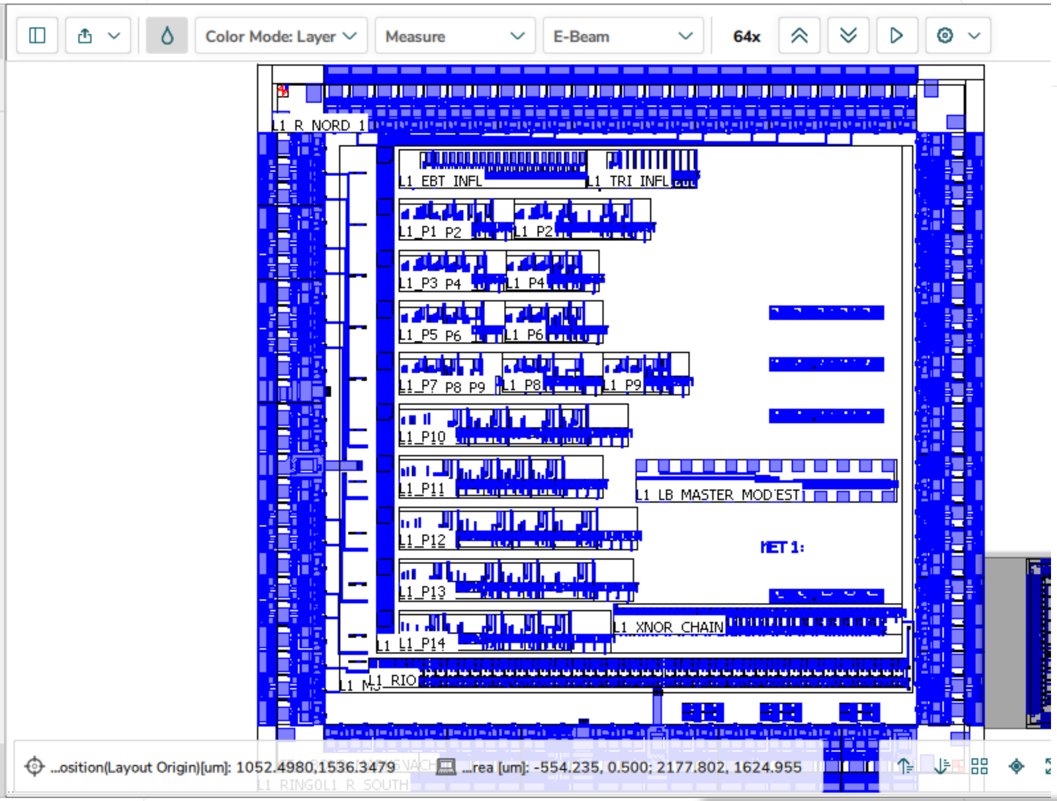
Fields

Refresh Preview | Interactive View

Fields
Region Definition
Write Sequence
Comment

Selected Layer		Selected Region		Field Traversal
6(0)	Select...	*	Select...	Fields Follow Geometry
49(0)	Select...	*	Select...	Fixed

OK | Cancel



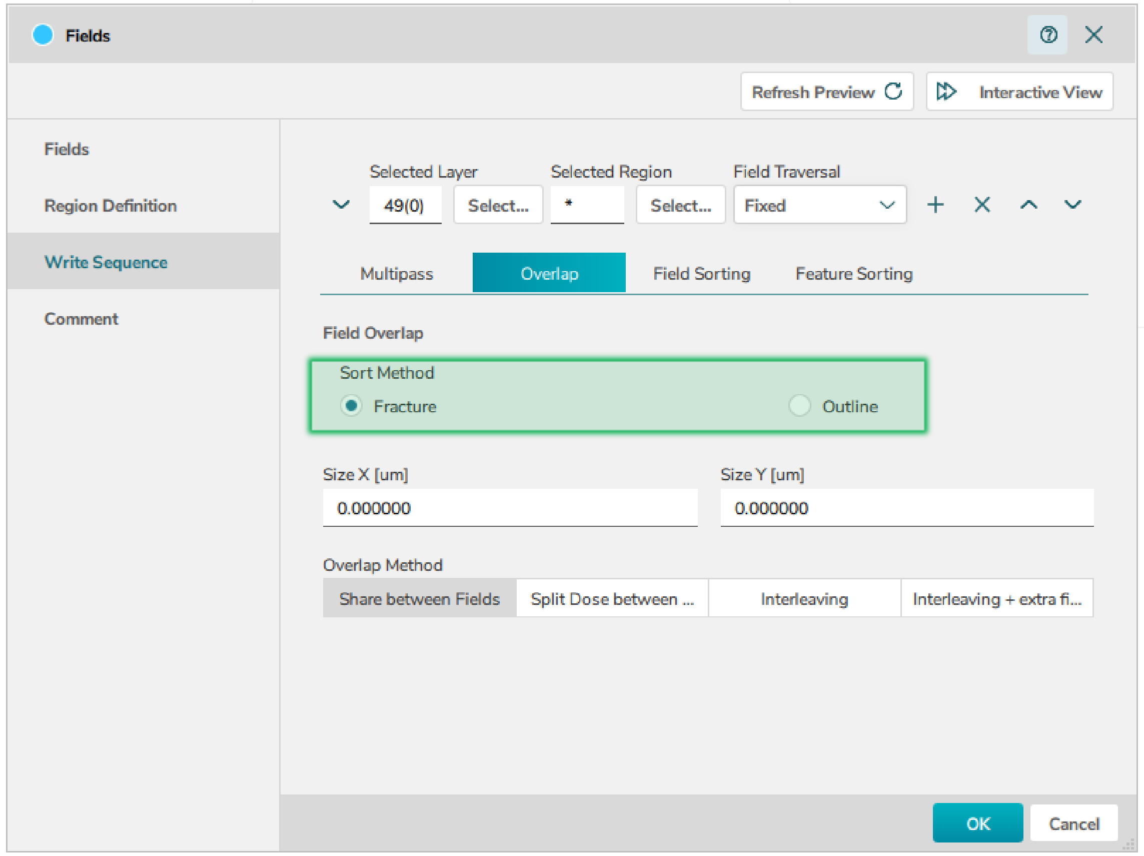
Color Mode: Layer | Measure | E-Beam | 64x

...osition(Layout Origin)[um]: 1052.4980,1536.3479 ...rea [um]: -554.235, 0.500; 2177.802, 1624.955

Fields module - overlap

Overlap treatment

- Sort method: Fracture / Outline



Fields

Refresh Preview | Interactive View

Fields
Region Definition
Write Sequence
Comment

Selected Layer		Selected Region		Field Traversal
49(0)	Select...	*	Select...	Fixed

Multipass | Overlap | Field Sorting | Feature Sorting

Field Overlap

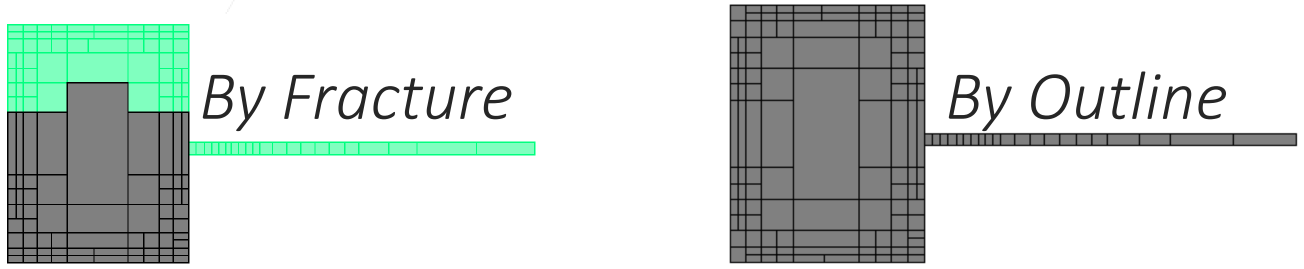
Sort Method: [✓ Fracture] Outline

Size X [um]: 0.000000

Size Y [um]: 0.000000

Overlap Method: Share between Fields | Split Dose between ... | Interleaving | Interleaving + extra fi...

OK | Cancel

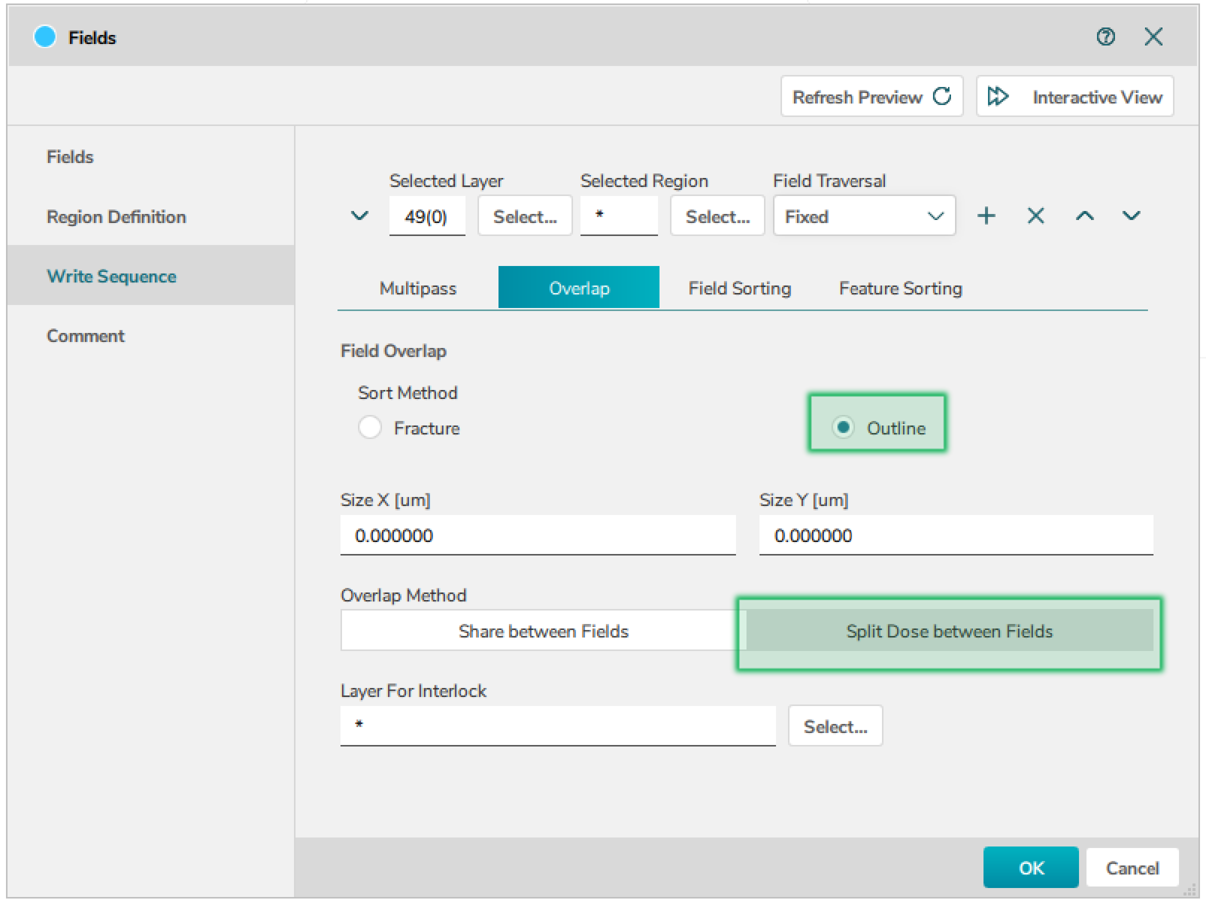


*Color by Fields

GenISys
Advancing the Standard

Fields module - overlap

Sort method “Outline” in combination with “Split dose”


Fields

Refresh Preview | Interactive View

Fields
Region Definition
Write Sequence
Comment

Selected Layer		Selected Region		Field Traversal
49(0)	Select...	*	Select...	Fixed

Multipass | Overlap | Field Sorting | Feature Sorting

Field Overlap

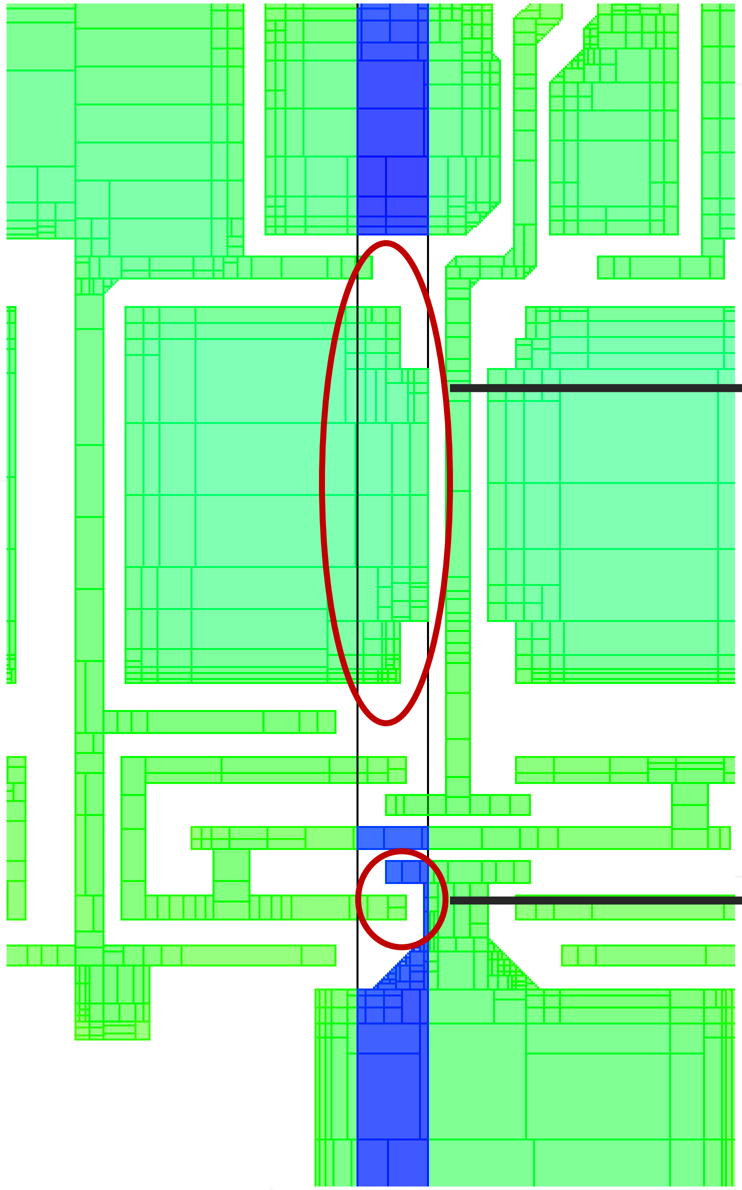
Sort Method
Fracture [✓ Outline]

Size X [um]: 0.000000
Size Y [um]: 0.000000

Overlap Method
Share between Fields [✓ Split Dose between Fields]

Layer For Interlock: * Select...

OK Cancel



No split dose, because the structure fits completely within the overlap region

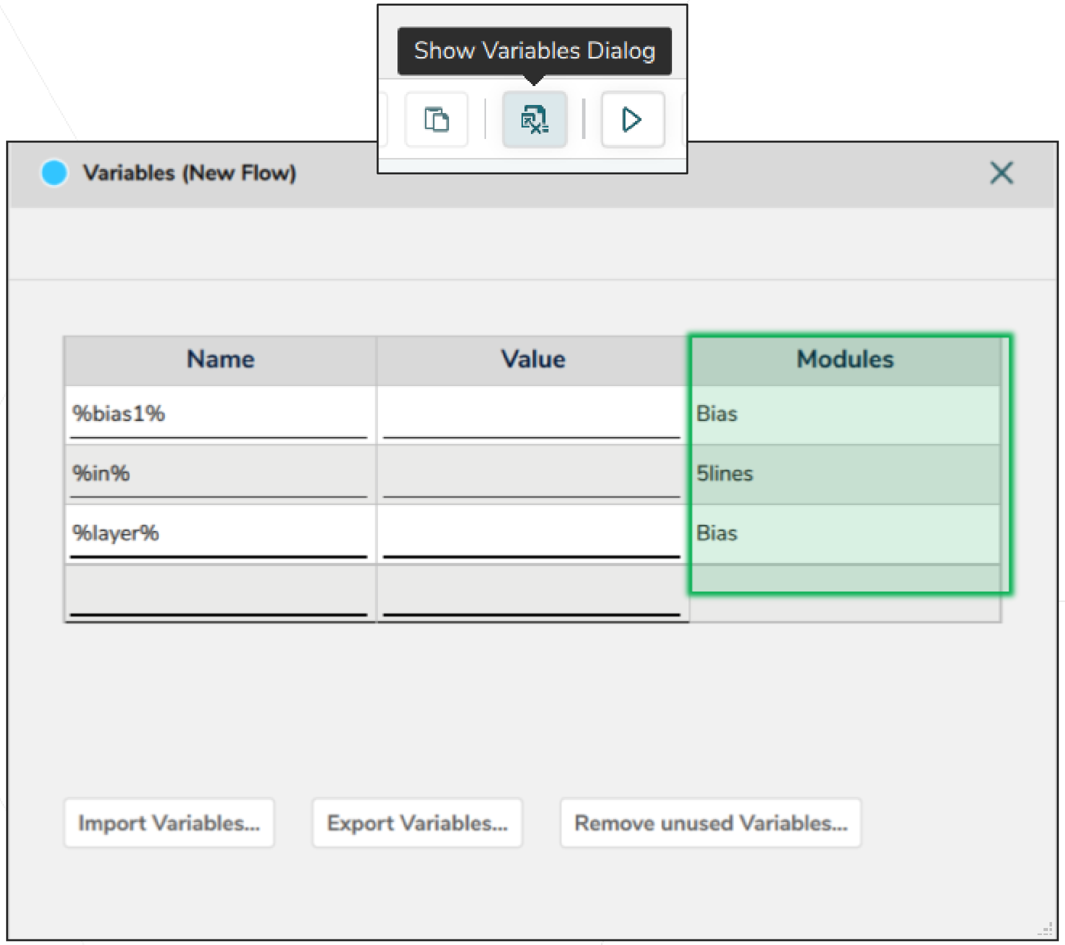
split dose, because the outline of the shape is crossing the overlap region

GenISys
Advancing the Standard

Variables

The dialog for the variable definition has been extended to also show the related module.

Unused Variables can be removed directly in the table

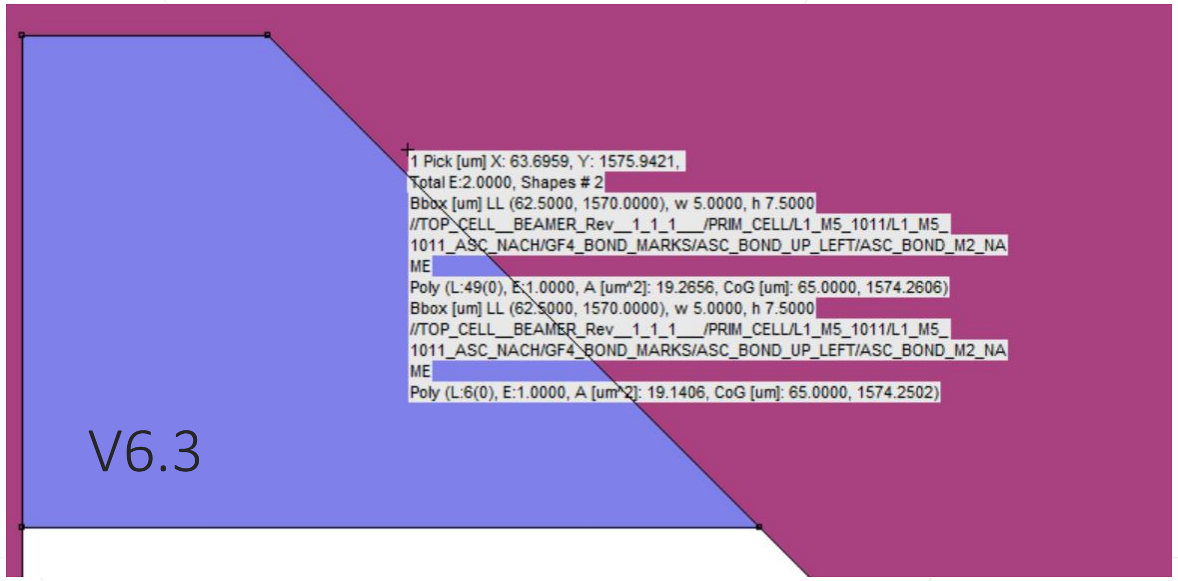

Show Variables Dialog

Variables (New Flow)

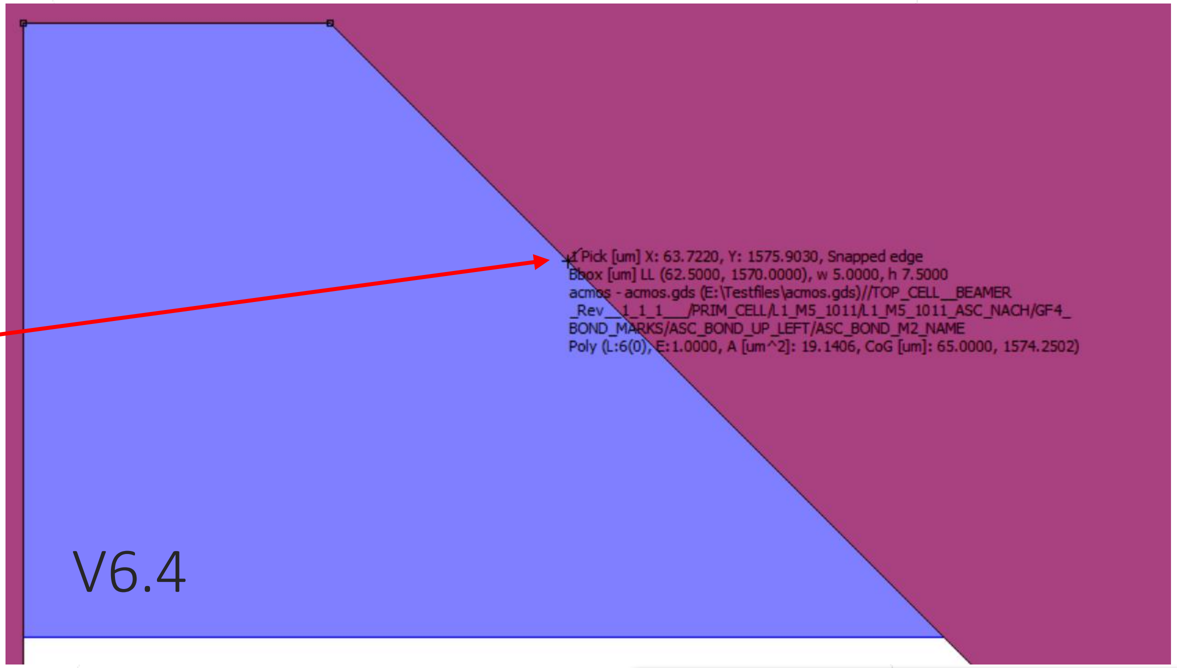
Name	Value	Modules
%bias1%		Bias
%in%		5lines
%layer%		Bias

Modified the picked data behavior in viewer

- Designated to identify shapes in overlapping layout scenarios
- The operation selects the shape next to the given pick-point
- Reporting is restricted to only one single shape
- In case of ambiguous shape edges the 1st reported is selected
- The measurement option is available via
  - CTRL + SHIFT + Right-double-click



```
1 Pick [um] X: 63.6959, Y: 1575.9421,
Total E:2.0000, Shapes # 2
Bbox [um] LL (62.5000, 1570.0000), w 5.0000, h 7.5000
//TOP_CELL__BEAMER_Rev__1_1_1___/PRIM_CELL/L1_M5_1011/L1_M5_
1011_ASC_NACH/GF4_BOND_MARKS/ASC_BOND_UP_LEFT/ASC_BOND_M2_NA
ME
Poly (L:49(0), E:1.0000, A [um^2]: 19.2656, CoG [um]: 65.0000, 1574.2606)
Bbox [um] LL (62.5000, 1570.0000), w 5.0000, h 7.5000
//TOP_CELL__BEAMER_Rev__1_1_1___/PRIM_CELL/L1_M5_1011/L1_M5_
1011_ASC_NACH/GF4_BOND_MARKS/ASC_BOND_UP_LEFT/ASC_BOND_M2_NA
ME
Poly (L:6(0), E:1.0000, A [um^2]: 19.1406, CoG [um]: 65.0000, 1574.2502)
```

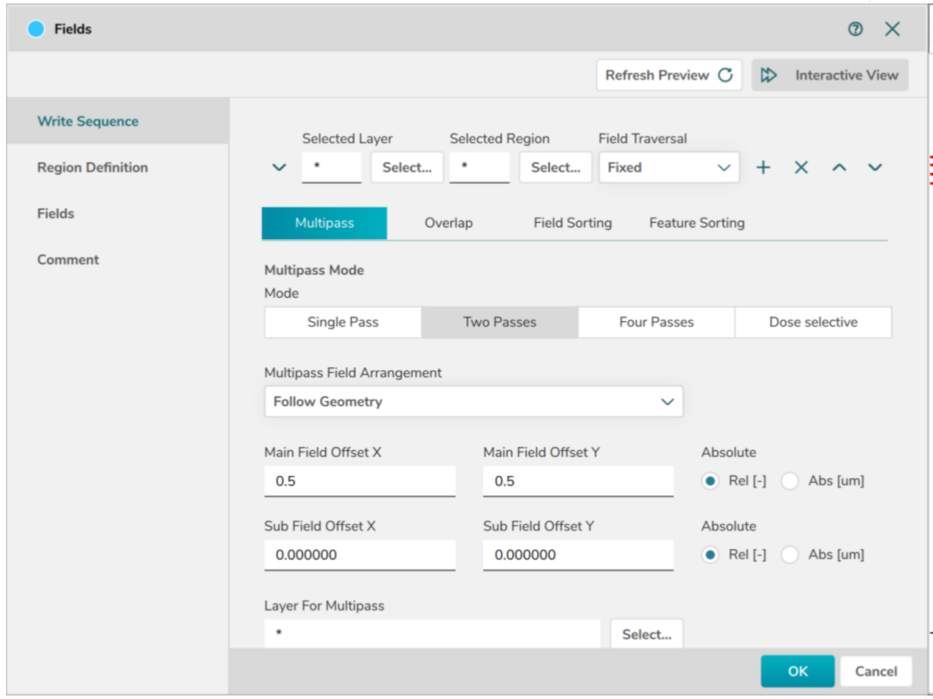


```
1 Pick [um] X: 63.7220, Y: 1575.9030, Snapped edge
Bbox [um] LL (62.5000, 1570.0000), w 5.0000, h 7.5000
acmos - acmos.gds (E:\Testfiles\acmos.gds)//TOP_CELL__BEAMER
_Rev__1_1_1___/PRIM_CELL/L1_M5_1011/L1_M5_1011_ASC_NACH/GF4_
BOND_MARKS/ASC_BOND_UP_LEFT/ASC_BOND_M2_NAME
Poly (L:6(0), E:1.0000, A [um^2]: 19.1406, CoG [um]: 65.0000, 1574.2502)
```

CTRL + SHIFT + right-double-click near edge

GenISys
Advancing the Standard

Improved field shifting and sorting – Multipass for waveguides


Fields

Refresh Preview | Interactive View

Write Sequence
Region Definition
Fields
Comment

Selected Layer	Selected Region	Field Traversal
*	*	Fixed

Multipass | Overlap | Field Sorting | Feature Sorting

Multipass Mode
Mode: Single Pass | [✓ Two Passes] | Four Passes | Dose selective

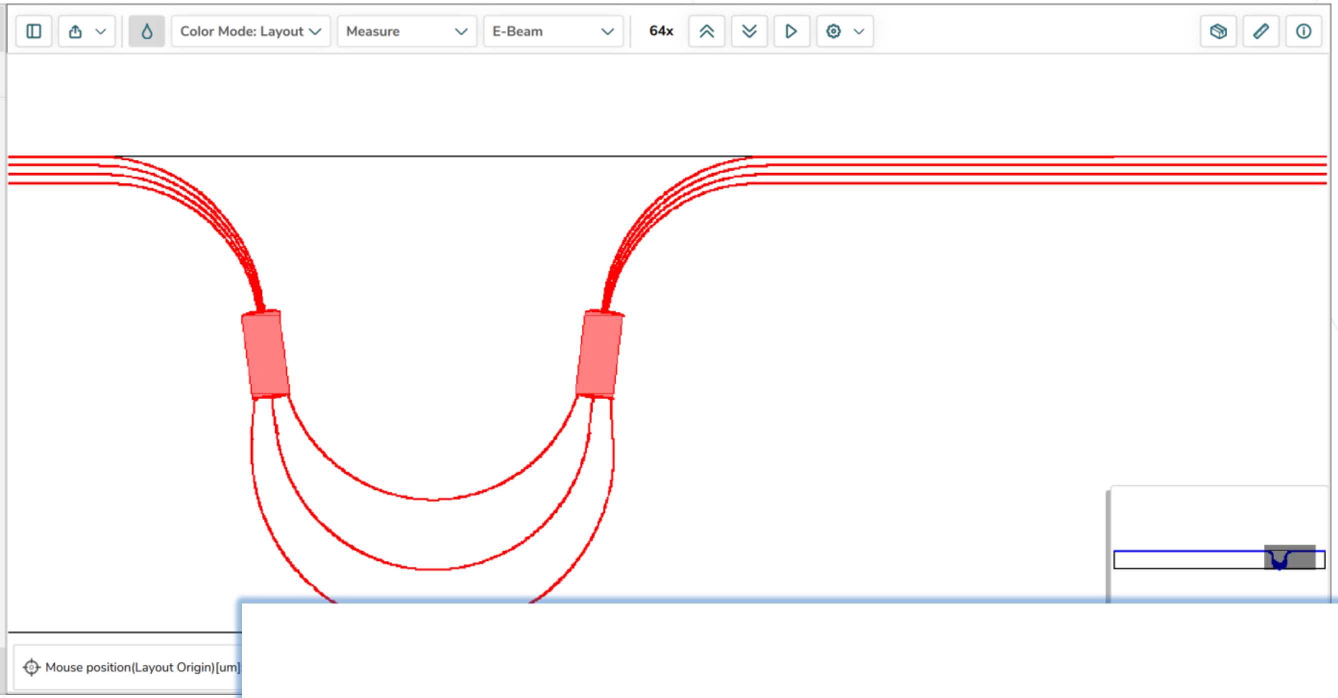
Multipass Field Arrangement: Follow Geometry

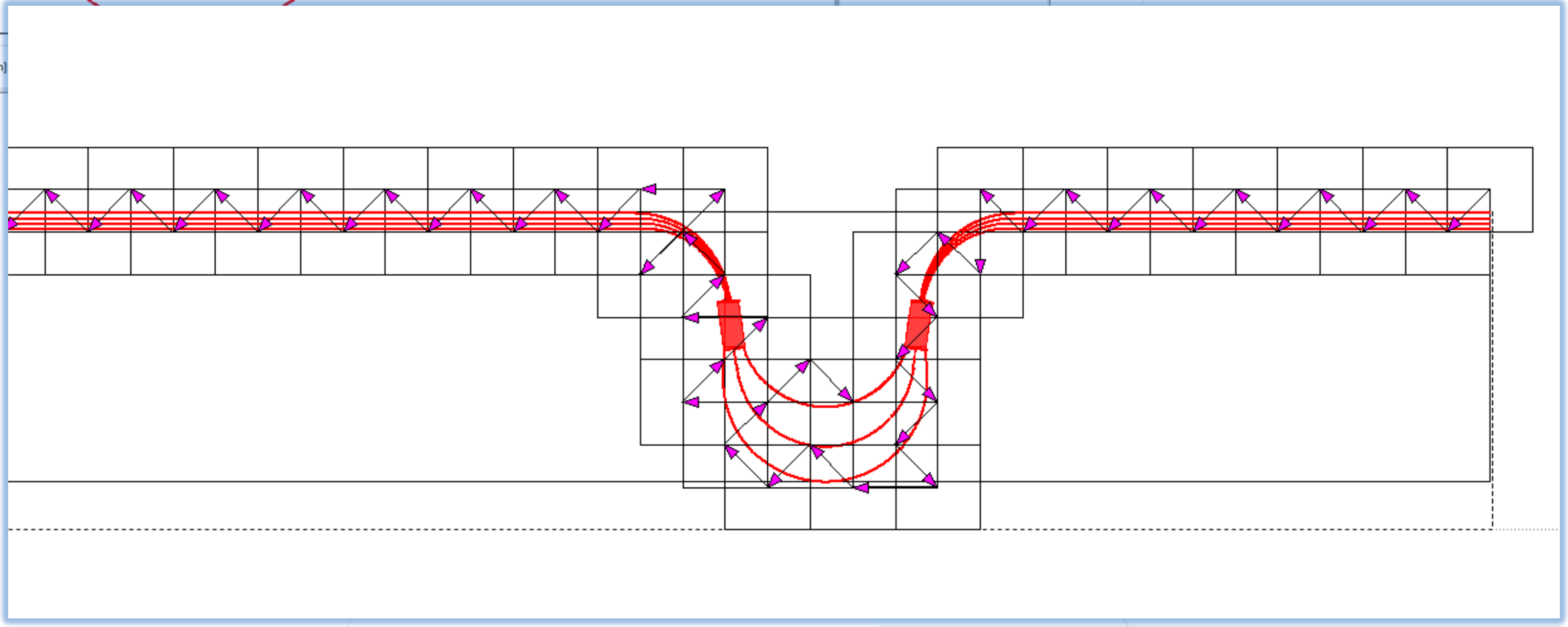
Main Field Offset X	Main Field Offset Y	Absolute
0.5	0.5	[✓ Rel [-]] Abs [um]

Sub Field Offset X	Sub Field Offset Y	Absolute
0.000000	0.000000	[✓ Rel [-]] Abs [um]

Layer For Multipass: *

OK | Cancel





Field shifting and sorting has been improved, minimizing the transitions between fields

Python module

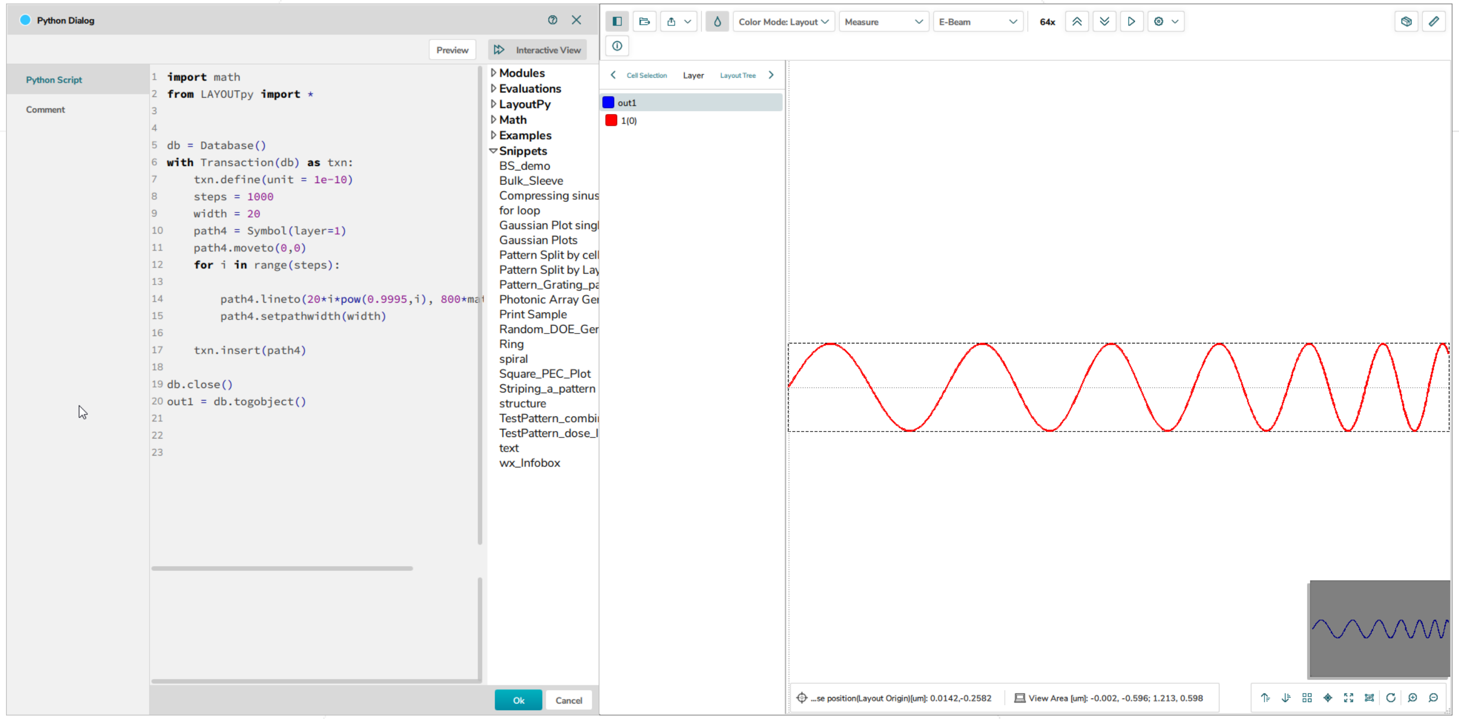
GenISys
Advancing the Standard

New Python Module

Python

The new Python module allows Python scripts to run within a BEAMER flow. Scripting enables more complex computation and flow control than flows built with just discrete modules. Python modules can take multiple input layouts and produce multiple output layouts.

- Scripts can be edited directly when a Python module is opened.



```python
import math
from LAYOUTpy import *


db = Database()
with Transaction(db) as txn:
    txn.define(unit = 1e-10)
    steps = 1000
    width = 20
    path4 = Symbol(layer=1)
    path4.moveto(0,0)
    for i in range(steps):

        path4.lineto(20*i*pow(0.9995,i), 800*ma
        path4.setpathwidth(width)

    txn.insert(path4)

db.close()
out1 = db.togobject()
```

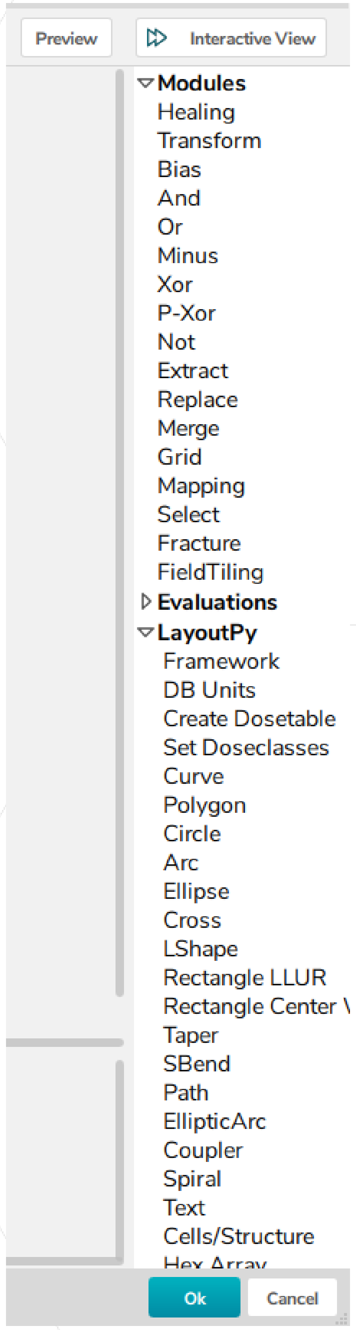
...se position(Layout Origin)[um]: 0.0142,-0.2582
View Area [um]: -0.002, -0.596; 1.213, 0.598

Not available for RedHat 6

GenISys
Advancing the Standard

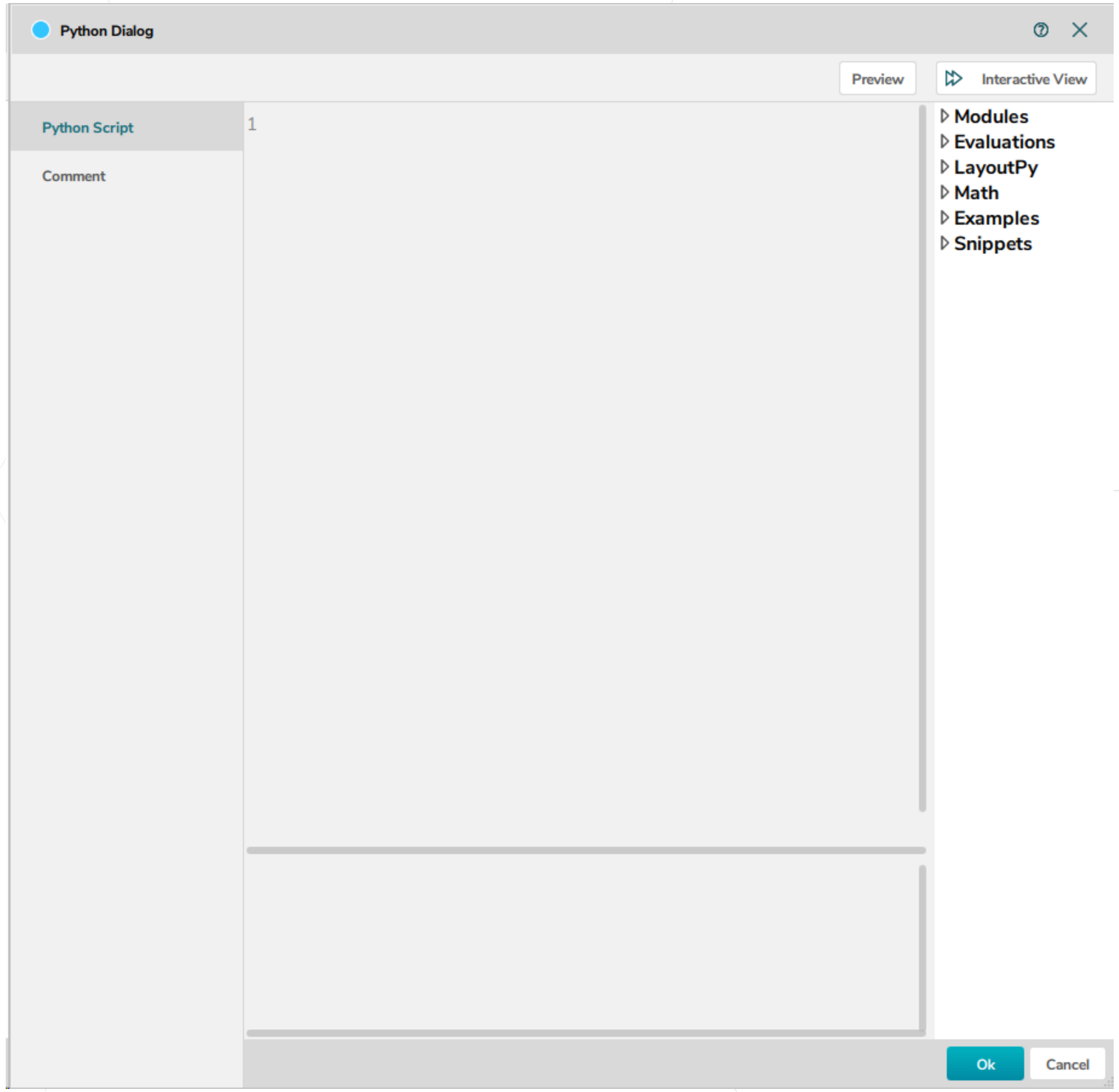
Python Module

- Script pane
  Here one can program full python code using basic Python libraries and anything what can be done within Python
  Supported Libraries are:
  BEAMERpy, LayoutPy, numpy, math, sys, PIL, matplotlib, wx, xlsxwriter, xlwt
- Snippet pane
  The snippet pane allows easy access to relevant functions. These are arranged in a tree type layout
  - Math
  - Module – contain all layout operations of BEAMER
  - Evaluations – contain the pattern analysis routines of the IF module
  - LayoutPy – offers layout generation via scripting
  - Examples – some pre-scripted LayoutPy and general script samples
  - Snippets – a custom repository of any code snippet the user wants to store



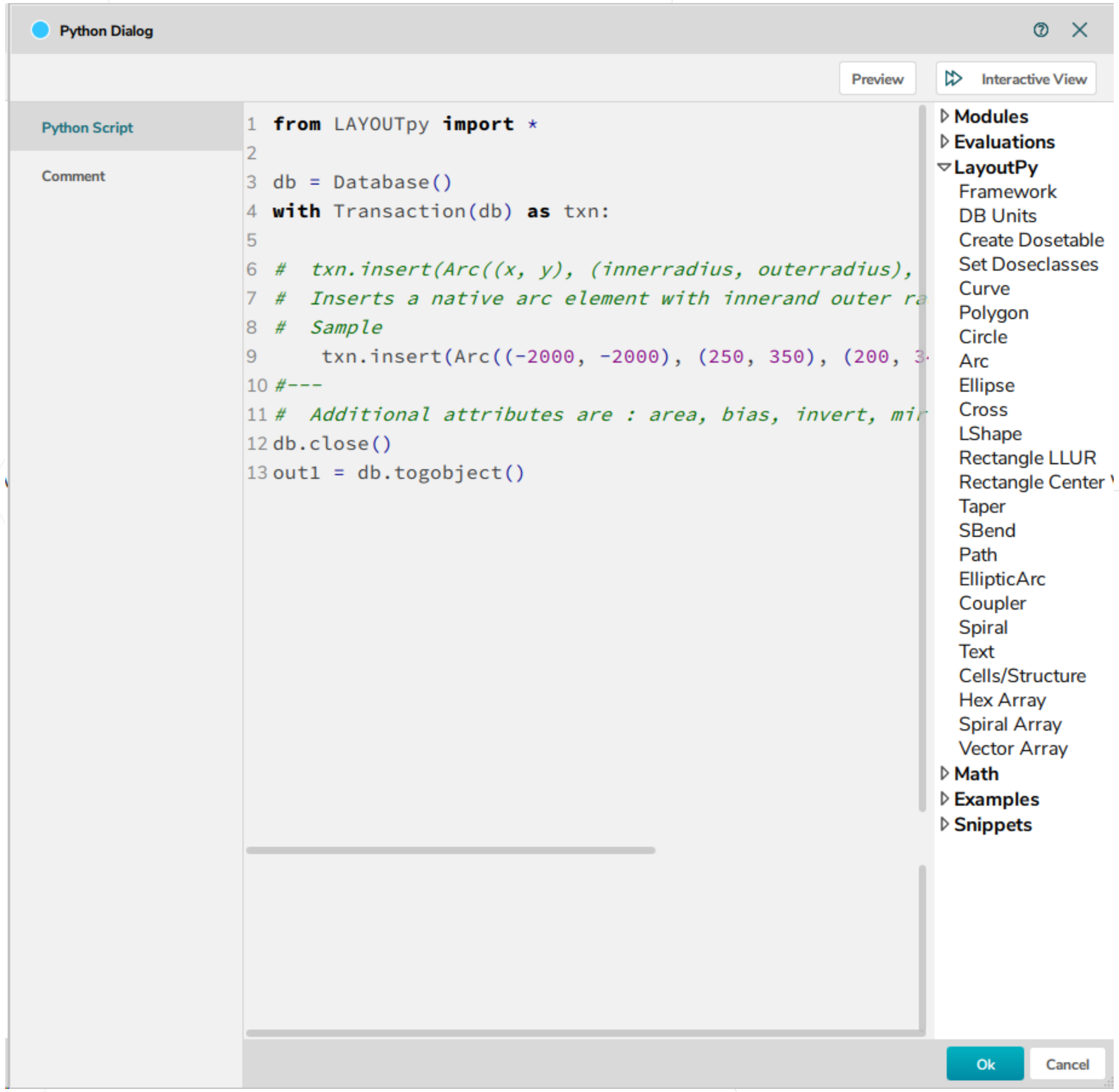
Enhancements for Python Module

- Reworked LayoutPy samples
- Access to new database elements
- Log output panel



Enhancements for Python Module

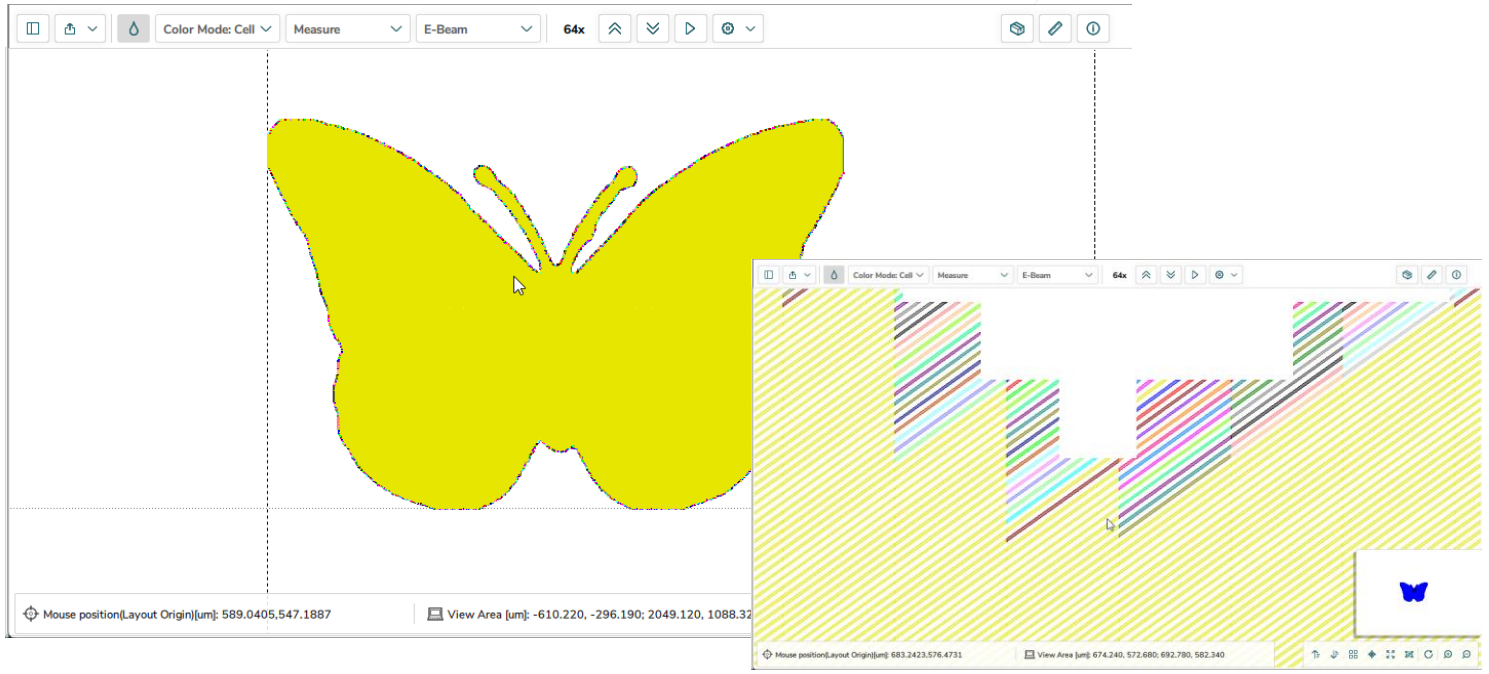
- Reworked LayoutPy samples
  - All samples have been reworked for consistent drag & drop usage
  - Documentation on each sample has been improved following a scheme of
    - Parametric sample
    - Description
    - Practical sample
    - Optional parameters
- Access to new database elements
- Log output panel



```
from LAYOUTpy import *

db = Database()
with Transaction(db) as txn:

#  txn.insert(Arc((x, y), (innerradius, outerradius),
#  Inserts a native arc element with innerand outer ra
#  Sample
    txn.insert(Arc((-2000, -2000), (250, 350), (200, 3
#---
#  Additional attributes are : area, bias, invert, mir
db.close()
out1 = db.togobject()
```

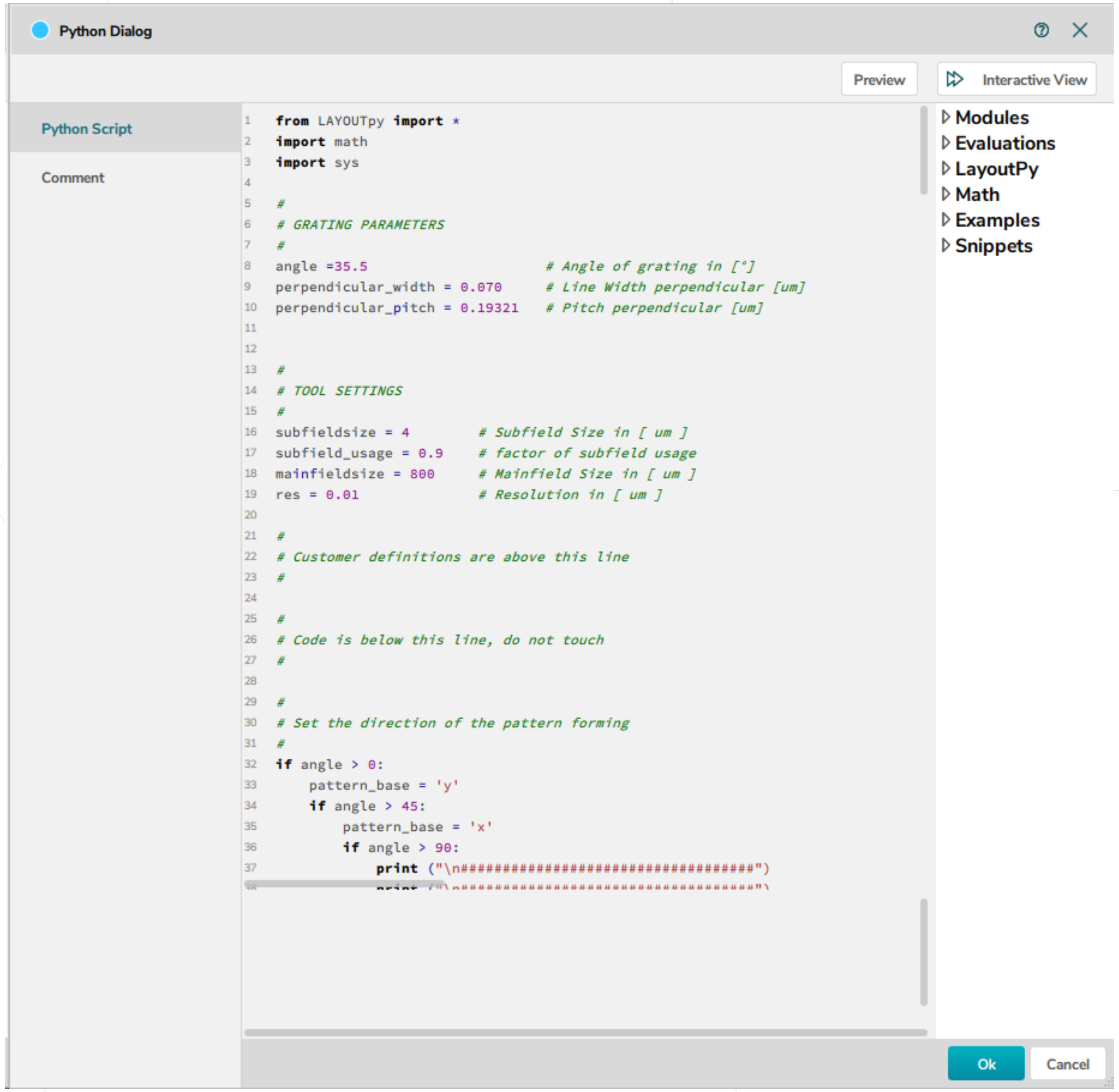
Enhancements for Python Module

- Reworked LayoutPy samples
  - Complex samples are also added
    - Grating generator for arbitrary shapes loaded
- Access to new database elements
- Log output panel



Mouse position(Layout Origin)[um]: 589.0405,547.1887
View Area [um]: -610.220, -296.190; 2049.120, 1088.32

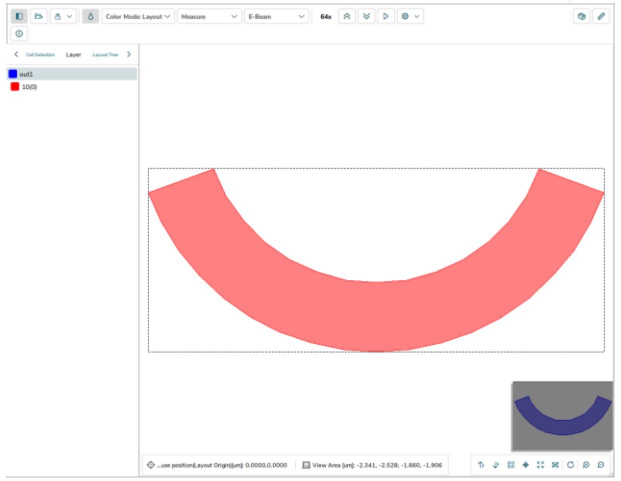
Mouse position(Layout Origin)[um]: 683.2423,576.4731
View Area [um]: 674.240, 572.680; 692.780, 582.340

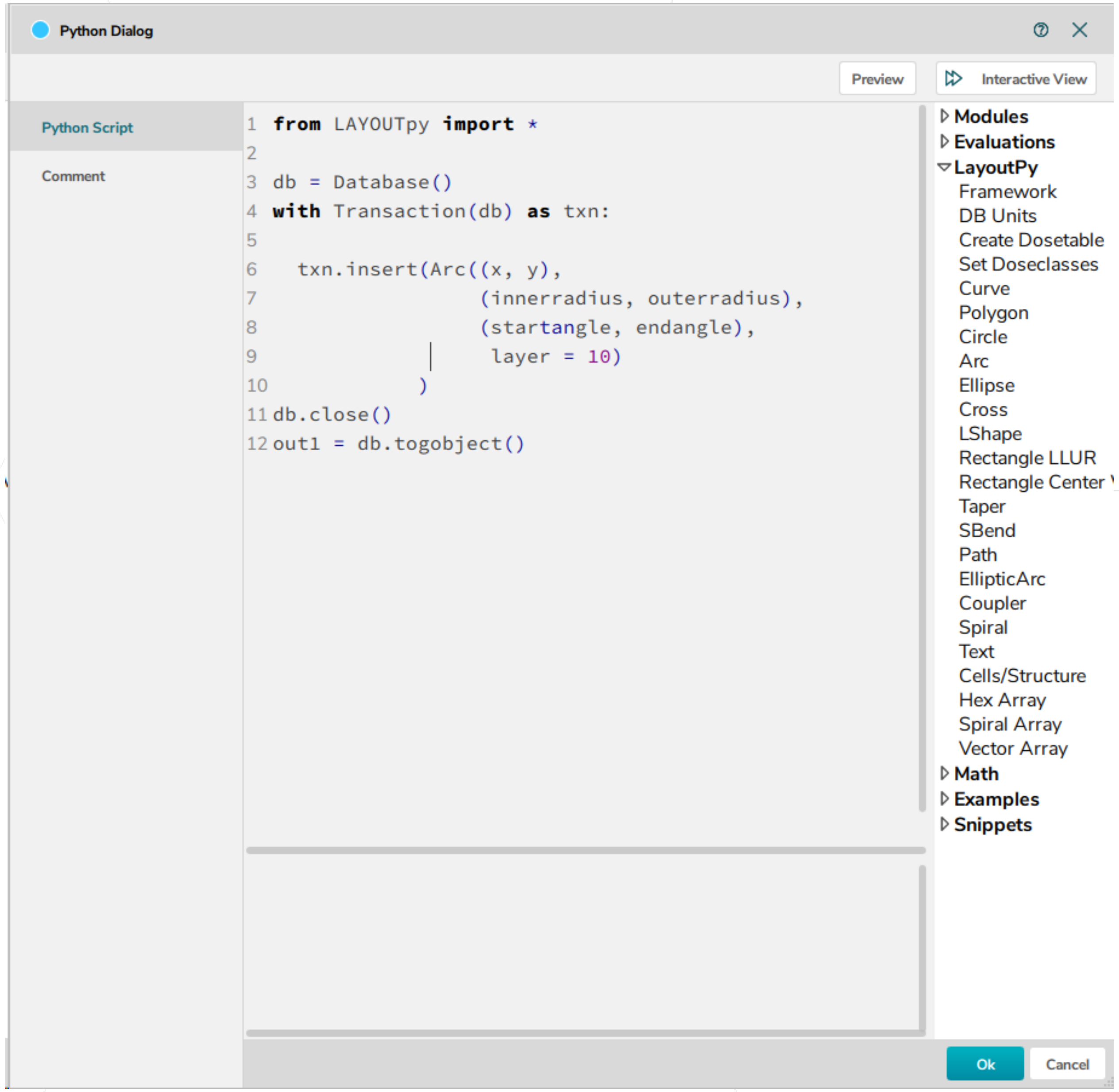


```
from LAYOUTpy import *
import math
import sys

#
# GRATING PARAMETERS
#
angle =35.5                          # Angle of grating in [°]
perpendicular_width = 0.070          # Line Width perpendicular [um]
perpendicular_pitch = 0.19321        # Pitch perpendicular [um]


#
# TOOL SETTINGS
#
subfieldsize = 4        # Subfield Size in [ um ]
subfield_usage = 0.9    # factor of subfield usage
mainfieldsize = 800     # Mainfield Size in [ um ]
res = 0.01              # Resolution in [ um ]

#
# Customer definitions are above this line
#

#
# Code is below this line, do not touch
#

#
# Set the direction of the pattern forming
#
if angle > 0:
    pattern_base = 'y'
    if angle > 45:
        pattern_base = 'x'
        if angle > 90:
            print ("\n####################################")
```

Enhancements for Python Module

- Reworked LayoutPy samples
- Access to new database elements
  - Arc has been reworked from polygon based arc to native database element
- Log output panel

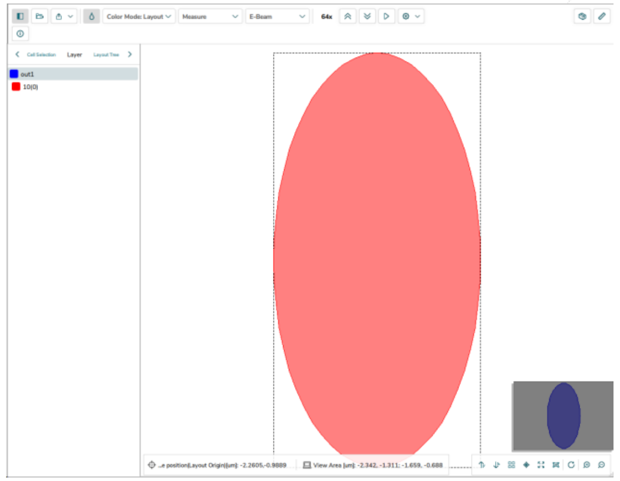


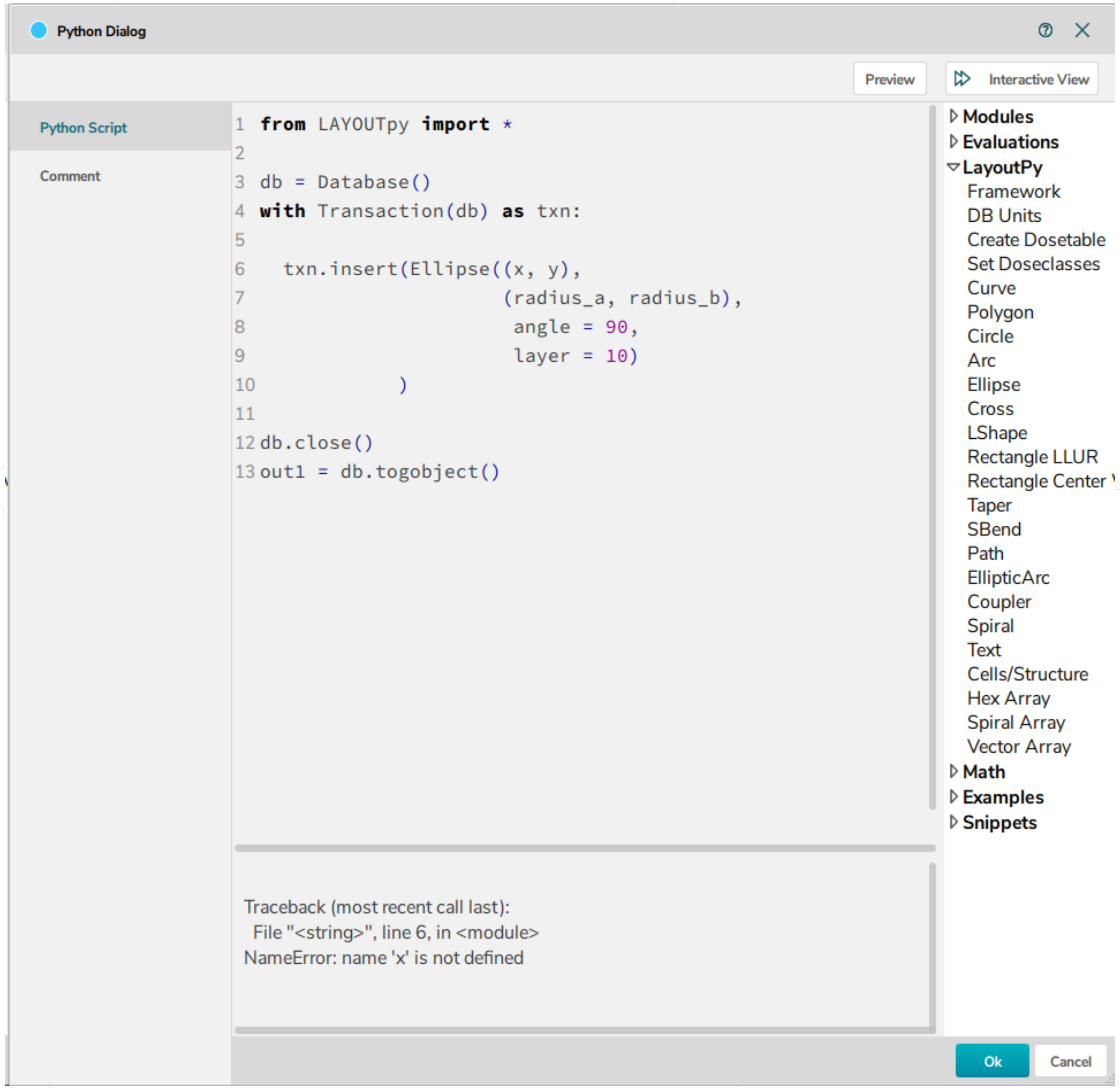


```
from LAYOUTpy import *

db = Database()
with Transaction(db) as txn:

  txn.insert(Arc((x, y),
                 (innerradius, outerradius),
                 (startangle, endangle),
                  layer = 10)
            )
db.close()
out1 = db.togobject()
```

Enhancements for Python Module

- Reworked LayoutPy samples
- Access to new database elements
  - Ellipse has been added as native database element
- Log output panel



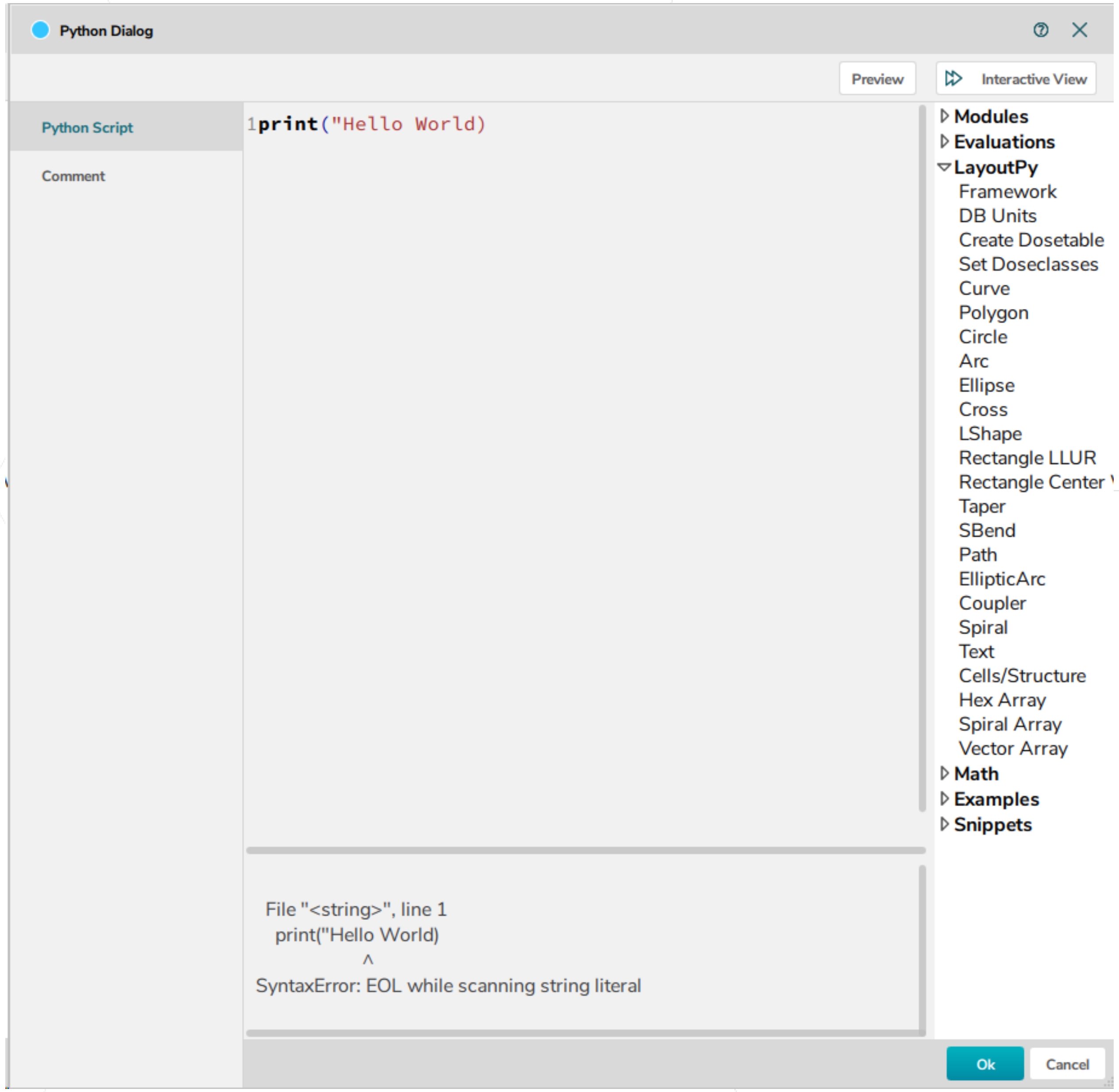


```
from LAYOUTpy import *

db = Database()
with Transaction(db) as txn:

  txn.insert(Ellipse((x, y),
                      (radius_a, radius_b),
                       angle = 90,
                       layer = 10)
             )

db.close()
out1 = db.togobject()
```

```
Traceback (most recent call last):
  File "<string>", line 6, in <module>
NameError: name 'x' is not defined
```

Enhancements for Python Module

- Reworked LayoutPy samples
- Access to new database elements
- Log output panel
  - The log output is now visible within the module itself to allow easier debugging
  - Syntax errors are highlighted and pointing to corresponding lines in the code

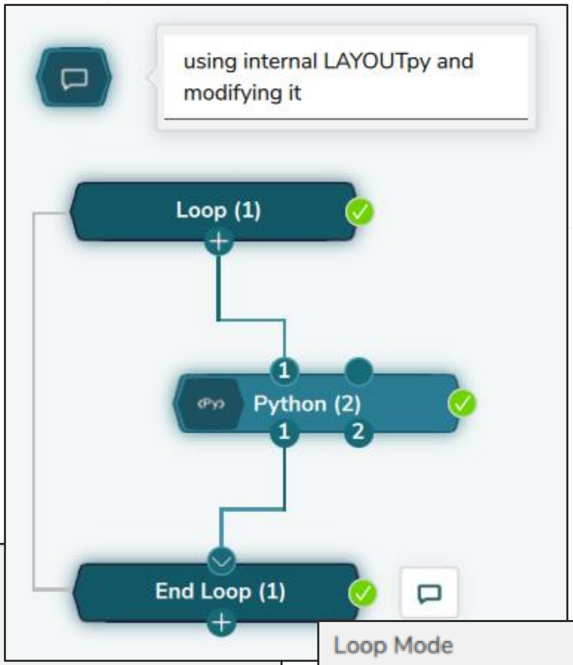


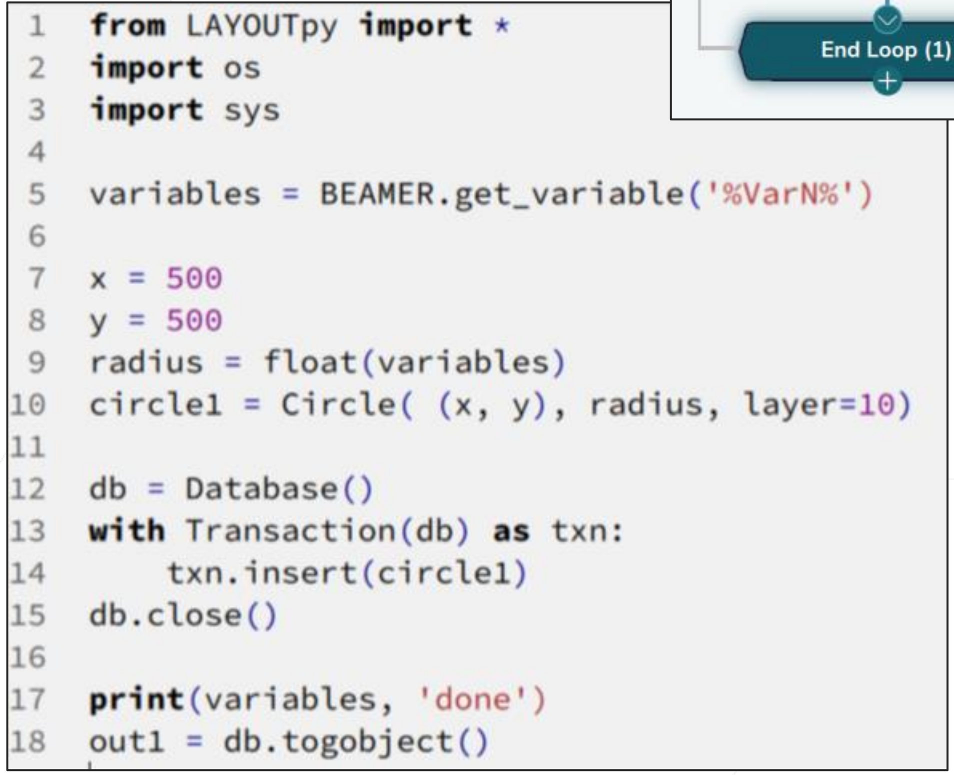
```
print("Hello World)
```

```
File "<string>", line 1
  print("Hello World)
                    ^
SyntaxError: EOL while scanning string literal
```

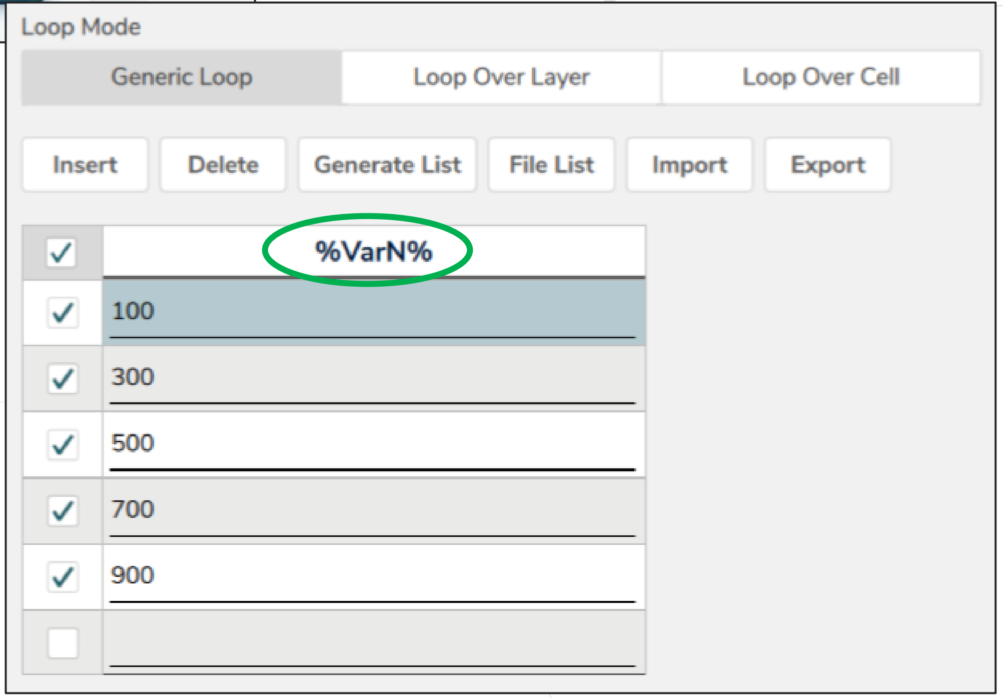
Enhancements for Python Module

- Variables can be included in Python. Shown on the right is a flow including a Python module which uses a variable to create circles of increasing radii.
- Result is shown below:





```
from LAYOUTpy import *
import os
import sys

variables = BEAMER.get_variable('%VarN%')

x = 500
y = 500
radius = float(variables)
circle1 = Circle( (x, y), radius, layer=10)

db = Database()
with Transaction(db) as txn:
    txn.insert(circle1)
db.close()

print(variables, 'done')
out1 = db.togobject()
```

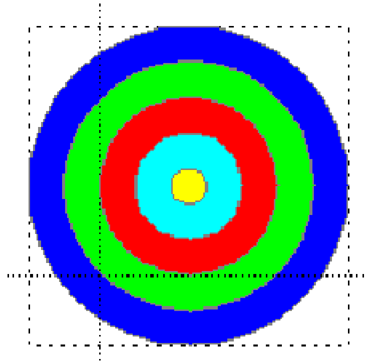


Loop Mode

Generic Loop | Loop Over Layer | Loop Over Cell

Insert | Delete | Generate List | File List | Import | Export

✓	%VarN%
✓	100
✓	300
✓	500
✓	700
✓	900



GenISys
Advancing the Standard

BEAMER v7 Framework requirements

- The support of OpenGL a requirement for our new BEAMER v7 GUI framework.
- This support is ideally hardware based with a graphics card available, either dedicated or on-chip.
- Software based OpenGL is also available. Please refer to your OS support pages on how to enable this option.
- To streamline the process the rpm and deb packages will be made the default install option for Linux since these allow dependency checks to prevent any issues.

GenISys
Advancing the Standard

Thank You!

support@genisys-gmbh.com

BEAMER LAB TRACER MASKER ProSEM VIEWER

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