GenISys
Advancing the Standard

BEAMER

What‘s new BEAMER 7.6

Retrieving Data from VIEWER

Python module extends its VIEWER integration with the BEAMERGUIpy library allowing to fetch data from measure and picking

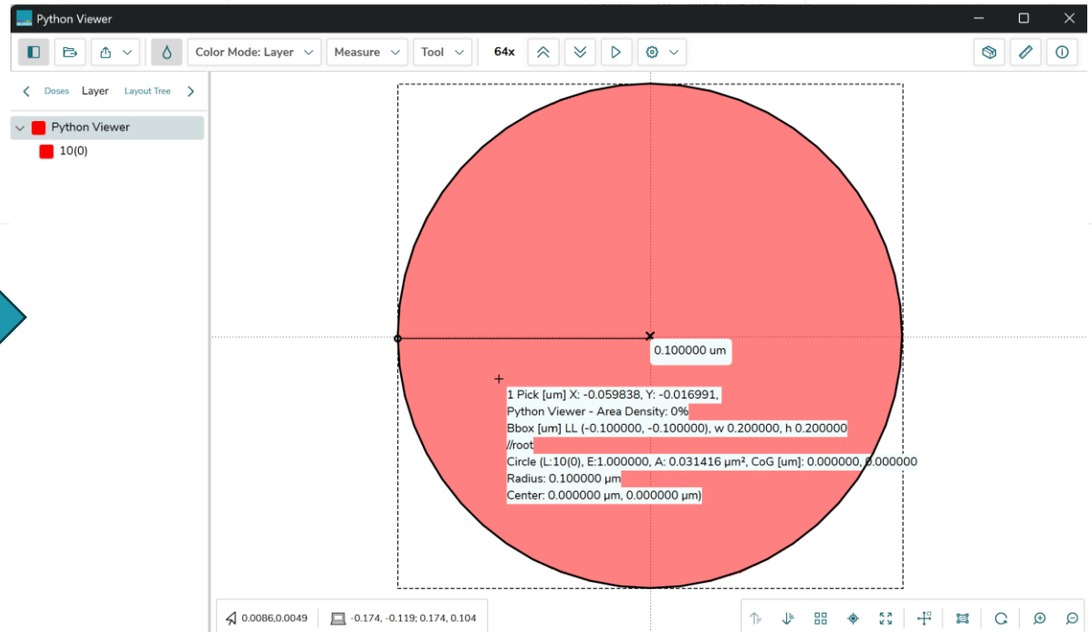
Python Dialog

Python Script | Python GUI Script | Comment

```
1
2   from LAYOUTpy import *
3
4   with Database() as db, Transaction(db) as txn:
5       txn.insert(Circle( (0, 0), 100, layer=10))
6   obj1 = db.togobject()
7
8
9   from BEAMERGUIpy import *
10  viewer_data = BEAMER_GUI.open_viewer(obj1)
11
12  for m in viewer_data.get_measurements():
13      print("center   : " + str(m.center_x) + ", " + str(m.center_y))
14      print("distance x/y: " + str(m.distance_x) + ", " + str(m.distance_y))
15      print("distance : " + str(m.distance))
16      print("snapped  : " + str(m.snapped))
17      print()
18
19  for p in viewer_data.get_picks():
20      print("position : " + str(p.position_x) + ", " + str(p.position_y))
21      print("info : " + str(p.info))
22      print()
23
24
```

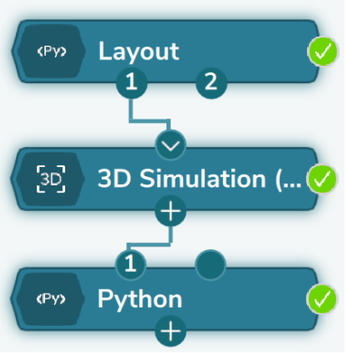
```
center   : -0.05, -0.001
distance x/y: 0.1, 0.0
distance : 0.1
snapped  : True

position : -0.059838, -0.016991
info : b'Python Viewer - Area Density: 0%,Bbox [um] LL (-0.100000, -0.100000), w 0.200000, h 0.200000
```



3D Export Command

BEAMERpy introduces a command that **exports 3D simulation** results to **CSV** improving integration between **BEAMER** and **Python**



Python Dialog

Python Script
Python GUI Script
Comment

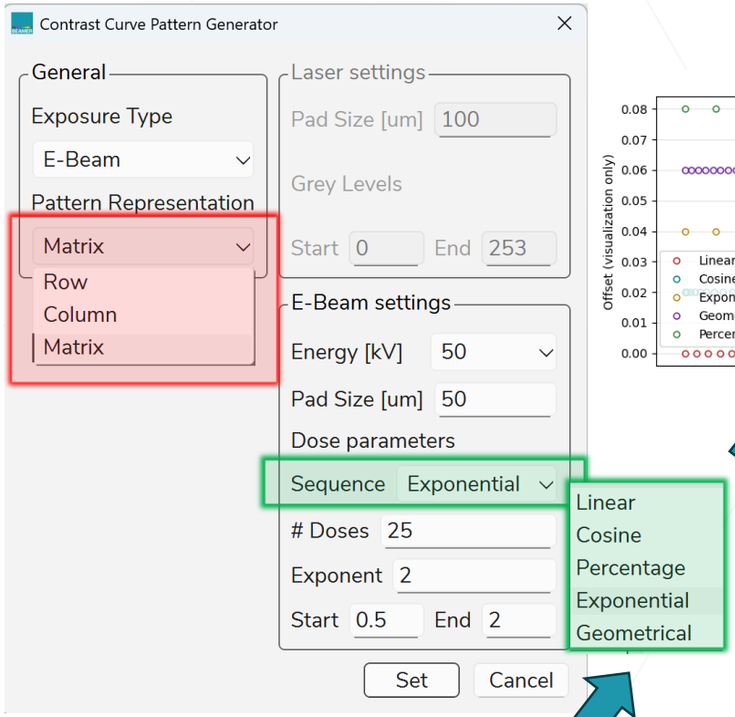
```
1
2  file = "C:/Users/Desktop/results/3D_image.csv"
3
4  csvParams = BEAMER.export_3d_csv_options()
5  csvParams["Filename"] = file
6  csvParams['ImageType'] = 'BulkImage'
7  csvParams['Selected Region'] = 'R1'
8  csvParams['Z'] = 0
9  csvParams['X min'] = 0
10 csvParams['X max'] = 0.5
11 csvParams['Y min'] = 0
12 csvParams['Y max'] = 0.5
13
14 BEAMER.export_3d_csv(in1, csvParams)
```

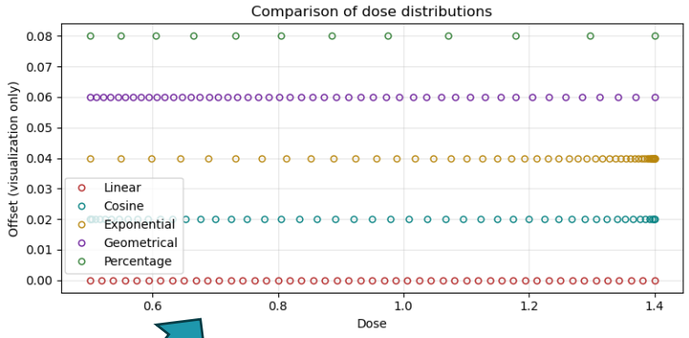
A1

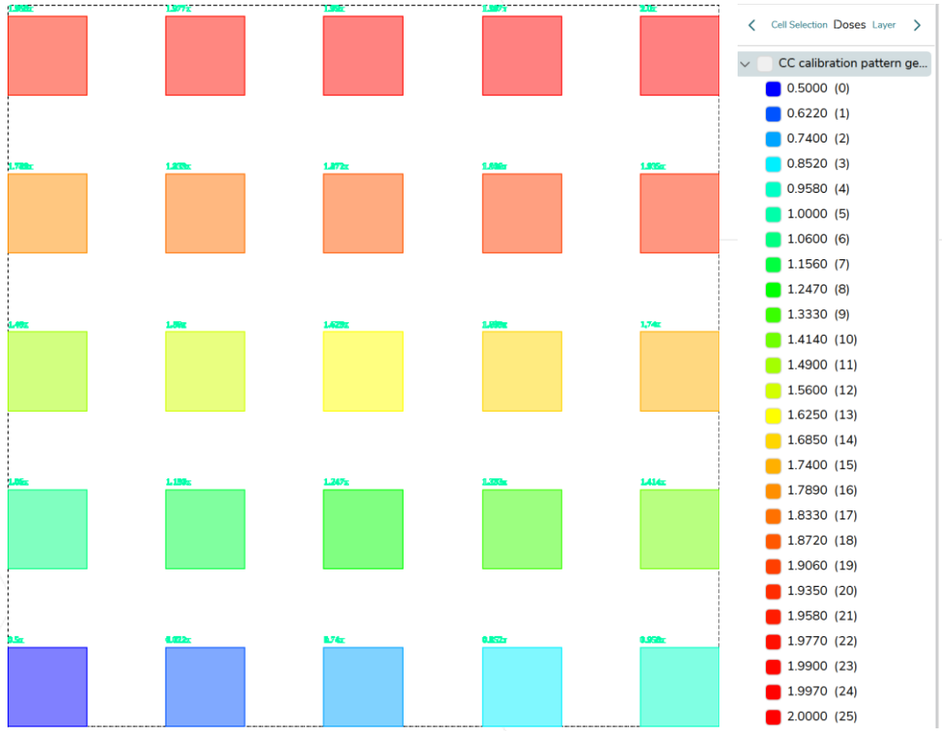
	A	B	C	D
1	# Region: R1	Column1	_1	_2
2	# Intensity[mJ/cm^2]			
3	0.0468527	-150045	-15004	-150035
4	-0.541548	281621	283175	284787
5	-0.49252	283208	2849	286669
6	-0.443491	284886	286738	288681
7	-0.394463	286664	288692	290837
8	-0.345435	288539	290767	293156
9	-0.296406	290508	292973	295665
10	-0.247378	292575	295321	298382

Contrast Curve Script

BEAMER facilitates **generating** a **Contrast Curve Pattern** via a user-friendly interface





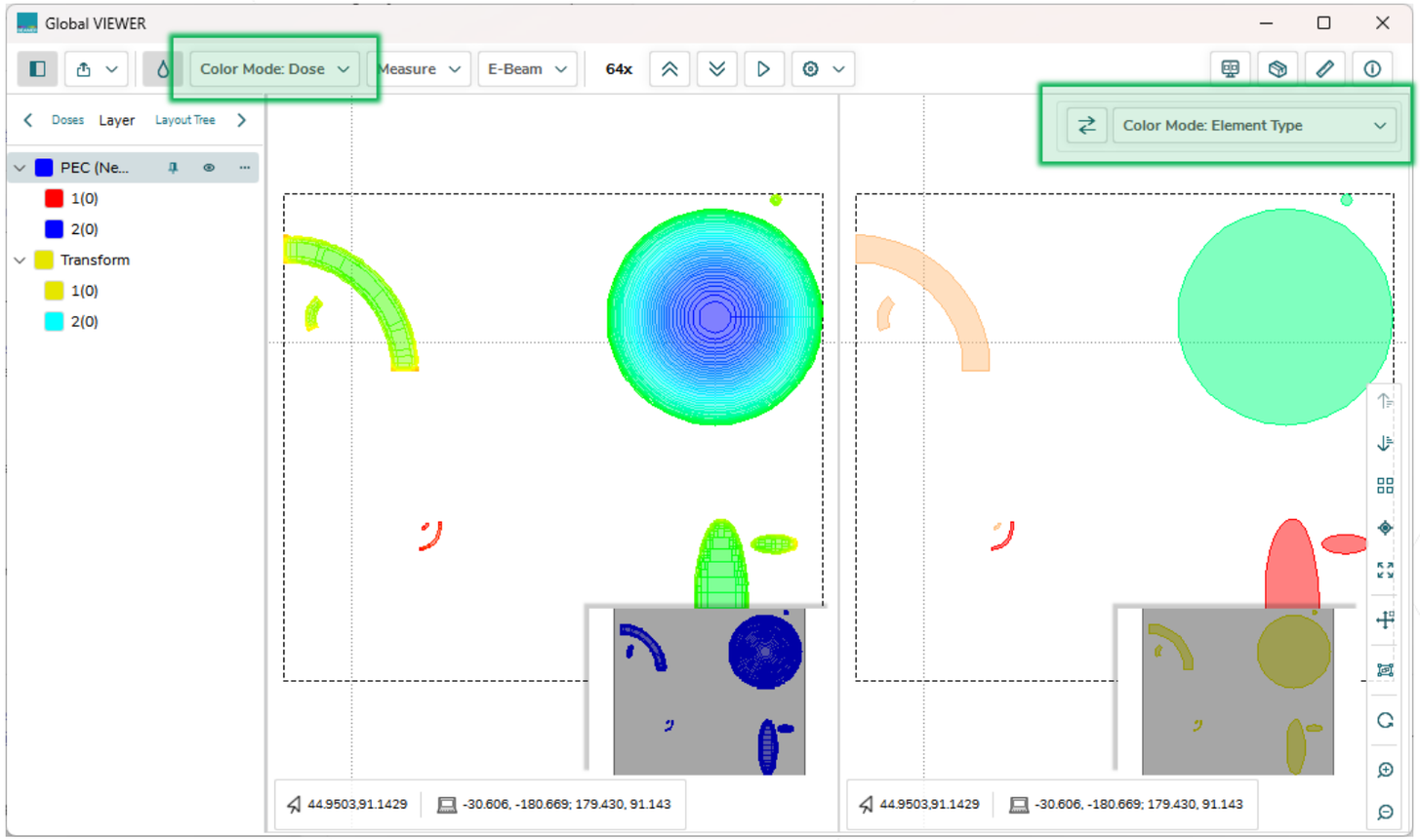


Possible to choose different dose ramping

Multiview Improvement

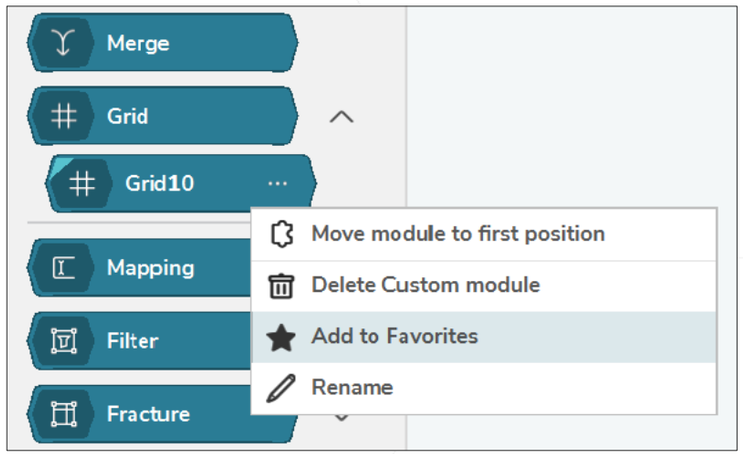
VIEWER **Multiview mode** adds support side-by-side viewing

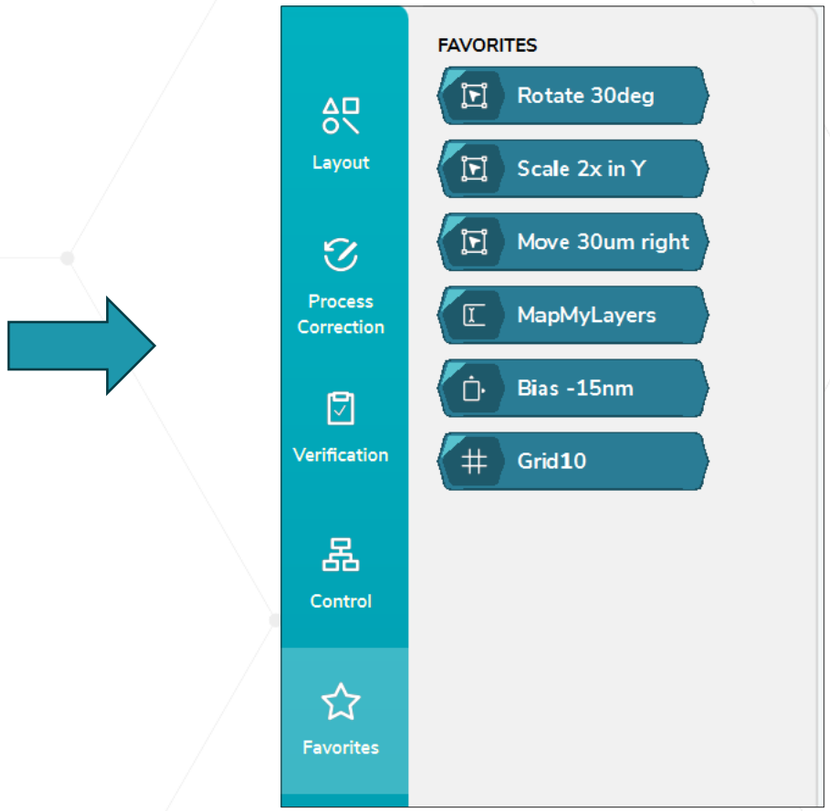
- Allows **different** combinations of **Color modes**
- Eases **Field** Order or **Subfields comparison**



Favorite Custom Modules

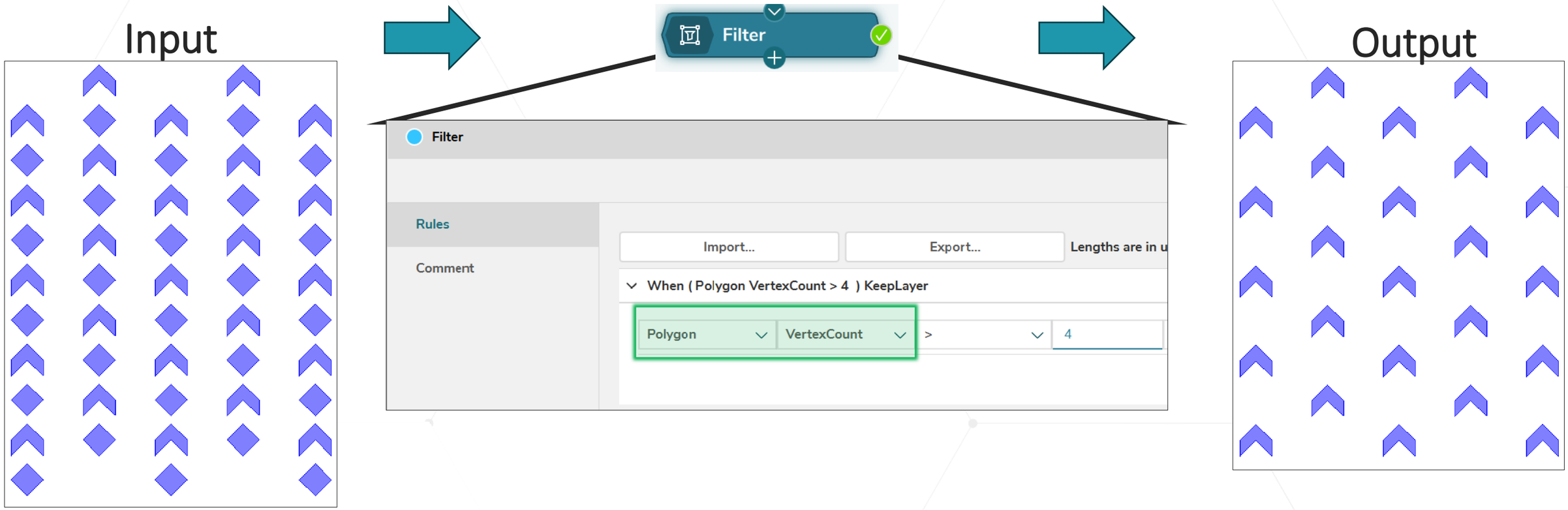
Keep your preferred **custom modules** accesible in the **Favorites tab**





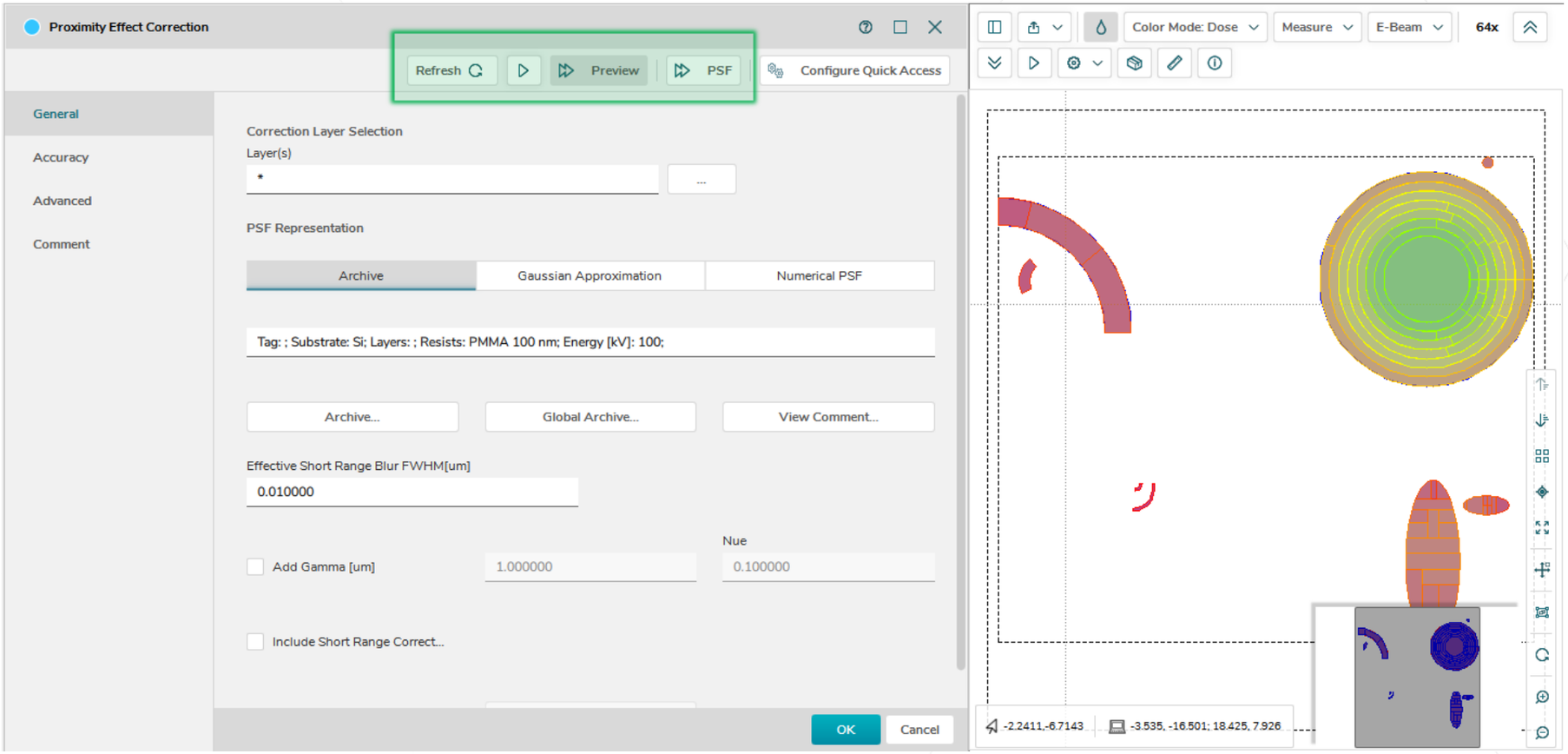
Filter by Vertices

Filter module incorporates the **attribute Vertex Count** for Polygons



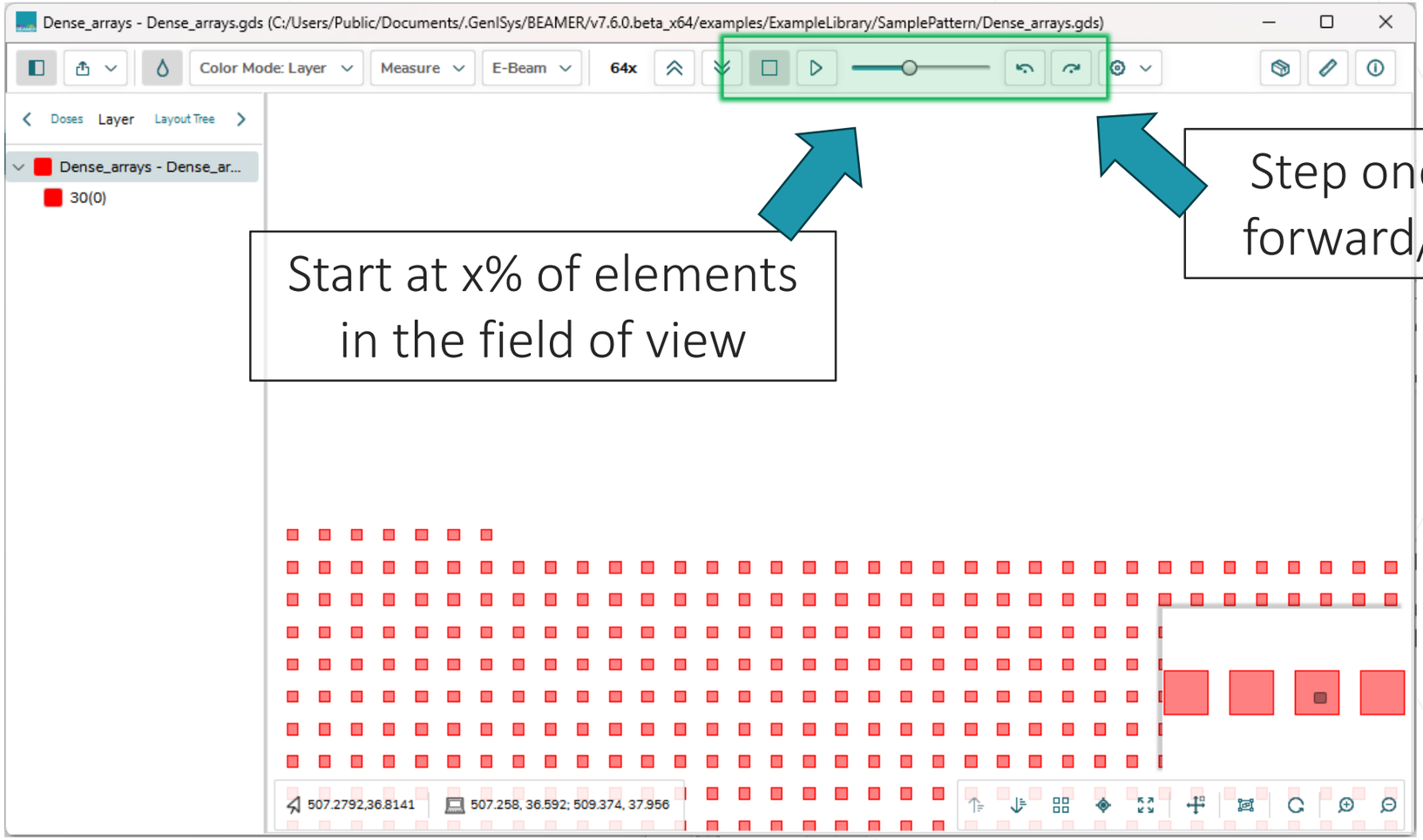
Preview PEC Results

Preview the impact of PEC settings on the dose and fracture of a small region of a layout



Improved Write Sequence View

Manage **view reproduction** using the improved **write-sequence controls**



Pixel Size in QR Code

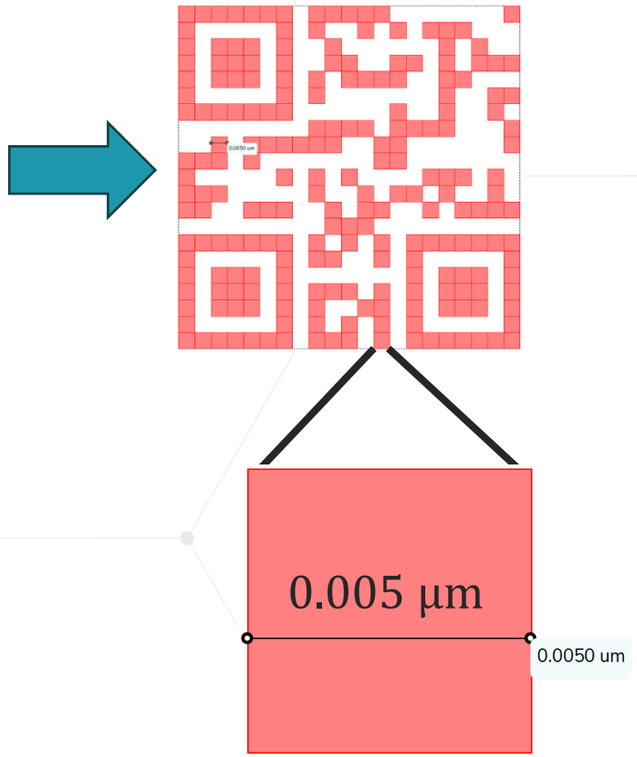
Extended the QRCode LayoutPy function in Python module with a minPixelSize parameter to specify the minimum pixel size

(as an integer of database units)

Python Dialog

Python Script
Python GUI Script
Comment

```
1  from LAYOUTpy import *
2
3  db = Database()
4  with Transaction(db) as txn :
5      # QRcode((Center_x, Center_y), Size, Value_to_encode, Layer, Dose, Minimum_pixel_size)
6      txn.insert(QRcode((-2900, 1200), 100, 'This is a test.', layer = 7, minPixelSize = 5))
7
8  db.close()
9  out1 = db.togobject()
```



Thank You!

help@genisys-gmbh.com

Headquarters
GenISys GmbH
Inselkammerstr. 5
D-82008 Unterhaching (Munich)
GERMANY
+49 (0)89 954 5364 0
+49 (0)89 954 5364 99
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 (45) 530 3306
+81 (45) 532 6933
apsales@genisys-gmbh.com