

PyCell Studio, Version 4.1.0, Release Notes

Table of Contents

Introduction.....	1
New Features.....	1
Premium Features.....	3
Changes.....	3
Change Requests.....	4
2007-05-29.....	4

Introduction

Welcome to PyCell Studio, Version 4.1.0, released 2007-05-29.

Version 4.1.0 is the next version of PyCell Studio. As denoted by the change in minor version number, this is basically a bugfix release of PyCell Studio, along with a small number of minor additions and usability enhancements. There have been some improvements and minor changes to the Python API, a number of Usability enhancements for the Pyros layout viewing tool, and several improvements in the Geometry Engine and related technology files.

See Change Requests for specific differences from Version 4.0.1.

These release notes summarize the important enhancements, bug fixes, and other changes which have been made since Version 4.0.1. For complete technical details, please see the related documentation.

New Features

1. Moving from Python 2.5 to Python 2.5.1

PyCell Studio uses the latest version of the Python language to stay current with the latest Python language developments. Note that version 2.5.1 is a bugfix release for Python 2.5.

2. Pyros Layout Viewer

Edit undo and redo buttons added for improved editing capabilities:

- Undo/Redo. After performing an editing operation, the Undo command will undo the previous editing operation, and the Redo command will redo this editing operation. Note that the undo and redo stacks are dynamically allocated, so that as many undo or redo commands can be performed as desired. This aids Pyros editing capabilities.

3. MultiPath

The MultiPath class has been significantly enhanced for the PyCell Studio 4.1.0 release. A more complete model for defining subpaths using justification and separation from the master path is now available. In addition, several additional methods have been added to this MultiPath class, as follows:

- MultiPath.createSubrectangles()
- MultiPath.getMasterPathName()
- MultiPath.getSubpathPoints()
- MultiPath.getChoppedSubpathPointLists()
- MultiPath.getJustifyPathPoints()
- MultiPath.moveBy()
- MultiPath.moveTowards()
- MultiPath.moveTo()
- MultiPath.rotate90()
- MultiPath.rotate180()
- MultiPath.rotate270()
- MultiPath.mirrorX()
- MultiPath.mirrorY()
- MultiPath.transform()
- MultiPath.snap()
- MultiPath.snapX()
- MultiPath.snapY()
- MultiPath.snapTowards()

4. PointList class changes

In conjunction with the above changes for the MultiPath class, new methods were added to the PointList class. These are the addOffsets() and genJustifyPoints() class methods.

5. Segment class changes

In conjunction with the above changes for the MultiPath class, several methods were added to the basic geometric Segment class. These are the isCoincident(), isParallel(), intersect(), hasIntersection(), exrtapIntersect(), addOffsets(), genJustifySegment(), getPosition(), moveBy(), and transform() class methods.

6. Geometry Engine and technology file changes

Technology files for the fictional CNI180 and CNI130 processes have been upgraded to support the new DRC OFFGRID operation, so that any off-grid shapes will now generate DRC errors.

Premium Features

For customers who have purchased commercial support, enhancements have been made to the existing set of additional premium features.

1. DRC Operations

Additional options (SINGULAR, OVERLAPPING) were added to the basic dimensional checking DRC operations (SPACING, WIDTH, ENCLSOYURE and OVERLAP).

Improvements were also made in the area of false error filtering.

2. Conversion programs

Design There have been some minor enhancements and bug fixes to the DRC rule deck file conversion program.

Changes

1. Pyros Layout Viewer

Numerous minor usability enhancements and bug fixes have been made for the PyCell Studio 4.1.0 release.

2. Conversion programs

The technology file conversion program (cntechconv) now supports both the newer OpenAccess technology file format, as well as the older CDB format. In addition, numerous minor usability and bug fixes have been made for the PyCell Studio 4.1.0 release.

3. Python API

Several minor changes have been made to improve usability and fix known problems in the Python API for the PyCell Studio 4.1.0 release.

For additional information on specific API changes, please refer to *Ciranova Python API Version 4.1 Code Conversion Guidelines*.

Change Requests

Like many companies, Ciranova uses Bugzilla (www.bugzilla.com) to track change requests. If your request was assigned a Bugzilla number, you should have been notified by Ciranova technical support when it was closed. Otherwise, you can check the following listing of Bugzilla requests which were closed for this release.

2007-05-29

- 363 RangeConstraint yields confusing results
- 425 CNGUI should automatically resynch for updated layer
- 497 Need better error message for invalid parameter name
- 508 Add Python API to call DRC engine with in-line rule definitions
- 618 Pyros should check for text object visibility
- 702 cntechconv should assign mask numbers to all process layers
- 802 cnexp should support –nosplash option, same as Pyros
- 817 Need support for unlimited enclosing shape
- 847 Strange NWELL overlap of ContactRing() in cni180
- 852 cntechconv should only allow layer and purpose map sections
- 853 Some cntechconv enhancements for Cadence OA tech file
- 854 Better error message for checking cntechconv input files
- 858 cntechconv should support Cadence OA technology file
- 860 cntechconv may not convert mfgGridResolution value as expected
- 874 DRC engine should support OFFGRID check
- 911 Pyros inconsistent behavior in response to DRC checking
- 933 cntechconv should also support Cadence OpenAccess tech file
- 954 Documentation should have version numbers or dates
- 964 Need to fully document DRC rule formats required by FG methods
- 981 Enhance MPP to complete spec
- 996 Remove old DRC geometries when user runs DRC in Pyros
- 1003 Text rendering in Pyros does not account for orientation
- 1018 Perform single precision checking for floating-point parameter values
- 1022 Pyros does not check layer visibility when re-calculating PyCell
- 1025 C++ DLO such as Contact does not support rulesets and contexts
- 1028 Update comments for Box expandForMinArea() method, adjust calculation
- 1032 Solaris x86 compiler bug with converting unsigned int to double in debug mode
- 1033 LPP Window is out-of-synch with viewing window after running DRC
- 1037 Ciranova smart objects not recognizing Device Contexts
- 1044 SIZE UNDEROVER produces wrong results on all-angle geometry
- 1046 Enhance cngenlib to create empty OpenAccess library

1047 PyCell Explorer should set CNI_DISPLAY_DIR for Pyros
1052 Should update copyright to include 2007 in Pyros
1053 Should update copyright to include 2007 in PyCell Explorer
1056 ContactRing enhanced to accept a Grouping argument
1057 New documentation file for Premium Product Offering
1059 Running Ciranova DRC program in Pyros needs to handle rulesets
1061 Can not obtain ranges or choices from PyCell parameters with constraints
1066 getEnclosureRule() does not detect DeviceContext
1093 Enhance ContactRing fillLayer to fillLayers
1097 Strange behavior of contacts for Metal layer
1109 Add InstTerm.getMustJoinInstTerms()
1111 Exception messages within APIs that stem from PyCells should indicate type
1114 Provide Instance._destroyOAExceptionInst(name)
1125 Pyros segfaults when 'recommended' parameter changed from gui
1131 ContactRing should have option to generate cuts for overlapping contacts
1132 AbutContact length error with off-grid via spacing values
1136 ParamSpecArray class should also have remove() method
1139 Need varactor layer for all technologies
1141 Edge Breaking and False Error Filtering in Dimensional Checks
1142 OA 2.2.6p52 pcellEval.onError() left Inst unbound on error
1146 cnrmlib to be added to external package
1147 makeArray() method should use user specified names for components
1149 Box expandToGrid() method would be very useful for off-grid errors
1151 Closing Library invalidate DloMgr::getDesign() cache
1154 Query for maxWidth rule of a layer should be provided
1164 Enhance Grid snap() methods to have optional integer multiple
1167 SINGULAR and OVERLAPPING options of Dimensional Check DRC Operations
1171 Rewrite via spacing rules in CNI130 technology file
1172 CNI180 and CNI130 technology files should include offgrid checks
1198 Core dump when generating spiral inductors and using fgPlace()
1206 GROW_EDGE generates wrong geometry
1207 Python uu2dbuArea() should return integer and dbu2uuArea() should return float
1211 RoutePath bugs with centering Contact larger than RouteTarget
INSIDE_IS_ERROR and OUTSIDE_IS_ERROR options of Dimensional Check
1213 DRC Operations
1217 Segmentation fault when running DRC on Spiral Inductor
1219 Via enclosure errors using fgDeriveLayer()