

PyCell Studio, Version 4.2.0, Release Notes

Table of Contents

Introduction.....	1
New Features.....	1
.....	3
Premium Features.....	4
.....	4
Changes.....	4
Change Requests.....	5
2007-12-12.....	5

Introduction

Welcome to PyCell Studio, Version 4.2.0, released 2007-12-12, a holiday present from Ciranova to our customers.

Version 4.2.0 adds new features and enhancements based on customer feedback. The adoption of PyCells by a growing base of customers is an exciting trend for Ciranova. Our customers are speaking, and we are listening.

This release also marks a milestone for Ciranova. Working with our EDA partners, we've achieved a new level of PyCell interoperability. Stretch handles and auto-abutment are now supported in multiple layout editors, demonstrating Ciranova's commitment to openness and interoperability.

Highlights for Version 4.2.0 are new APIs for hierarchical design and support for OpenAccess-style constraint definitions. New API commands simplify design of hierarchical PyCells with physical operators that can reference shapes in lower levels of hierarchy. Enhancement to technology file syntax to support OpenAccess-style constraint definitions, i.e. DRC rules, eases the adoption of PyCells, when existing process design rules have already been coded for OpenAccess.

See Change Requests for specific differences from Version 4.1.0.

These release notes summarize the important enhancements, bug fixes, and other changes which have been made since Version 4.1.0. For complete technical details, please see the related documentation.

New Features

1. Hierarchical design API

Introduction of new `PhysicalComponentRef`, a reference to a `PhysicalComponent` in a lower level of the design hierarchy. Associated methods include:

- `DloGen.findCompRef()`
- `Grouping.getCompRefs()`
- `GroupingRef.getCompRefs()`
- `Instance.getCompRefs()`
- `InstanceArray.getCompRefs()`
- `InstanceArrayMember.getCompRefs()`
- `InstanceArrayMemberRef.getCompRefs()`
- `InstanceArrayRef.getCompRefs()`
- `InstanceRef.getCompRefs()`

- PhysicalCompRef.abut()
- PhysicalCompRef.alignEdge()
- PhysicalCompRef.alignEdgeToPoint()
- PhysicalCompRef.alignLocation()
- PhysicalCompRef.alignLocationToPoint()
- PhysicalCompRef.getBBox()
- PhysicalCompRef.getName()
- PhysicalCompRef.getParentPathName()
- PhysicalCompRef.getParentPathTransform()
- PhysicalCompRef.getPathName()
- PhysicalCompRef.getSpacing()
- PhysicalCompRef.getTopInst()
- PhysicalCompRef.mirrorX()
- PhysicalCompRef.mirrorY()
- PhysicalCompRef.moveBy()
- PhysicalCompRef.moveTo()
- PhysicalCompRef.moveTowards()
- PhysicalCompRef.overlaps()
- PhysicalCompRef.place()
- PhysicalCompRef.rotate180()
- PhysicalCompRef.rotate270()
- PhysicalCompRef.rotate90()
- PhysicalCompRef.snap()
- PhysicalCompRef.snapTowards()
- PhysicalCompRef.snapX()
- PhysicalCompRef.snapY()
- PhysicalCompRef.transform()

2. Definition of stretch handle and auto-abutment protocol.

The Ciranova standard protocols enable any EDA vendor to develop or enhance applications to support PyCells with graphical stretch handles and auto-abutment. The requirements for PyCell authors are documented in *Ciranova Python API Reference Manual*. Corresponding requirements for EDA vendors are found in *Ciranova PyCell EDA Tool Integration Guidelines*.

3. New technology file syntax for OpenAccess-style constraints, i.e. DRC rules.

New syntax added for physicalRules() section of Santana.tech technology file, to match all design rule constraints as of OpenAccess 2.2p072.

- *minSpacing* rule name used for one-layer rules.
- *minClearance* rule name used for two-layer rules.
- *minAdjacentViaSpacing* and *minLargeViaArrayCutSpacing* rule names replace *minSpacing* rule with *neighbours* parameter.
- *minExtension* rule name replaces with *minEnclosure* rule name.
- *minDualExtension* rule name replaces *minEnclosure* rule with *endOfLine* and *4sides* parameters.
- Rule value can be a number, pair of numbers or list of pairs.

Older PyCell code with older-syntax Santana.tech will continue to function. Newer PyCell code, such as latest IPL code will work with new-syntax Santana.tech. Older PyCell code may need

updates to work with new-syntax Santana.tech.

Please refer to the documentation on matching changes in `Tech.getPhysicalRule()` command.

4. Binary Santana.tech
Optional binary format available for Santana.tech technology file, per request of our foundry partners. To create the binary format, use `cngenlib --force_binary_techfile`.
5. Interoperable PCell Library (www.iplnow.com)
As always, a copy of the latest IPL code is included with PyCell Studio, and can be found in the directory, `$CNI_ROOT/quickstart/IPL_Python_code`. Latest IPL code includes new PGuardRing and NGuardRing PyCells.
6. OpenAccess version
PyCell Studio Version 4.2.0 ships with OpenAccess 2.2p072.

Premium Features

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

1. PCell Xtreme

New commands, *cenablecache* and *cndisablecache*, simplify the most common tasks of enabling and disabling a PCell library for caching.

Changes

1. Import/export of GDSII

cnstrmin/cnstrmout shellscripts added to simplify use of OpenAccess *strm2oa/oa2strm*. These shellscripts automatically set the environment variables for the Ciranova PyCell plug-in.

2. Simplified installation

The installation script has been renamed from *configure* to *installer*. Installation script simplified to always install all packages, removing the most common source of error for new customers, an incomplete installation.

3. Pyros Layout Viewer

When viewing large hierarchical PyCells, Pyros can exhibit slow redraw performance. When the environment variable, `CNI_PYROS42_HIER_DEBUG` is set to `False`, Pyros will redraw hierarchical PyCells faster, but Wing IDE breakpoints in lower levels of hierarchy will prevent Pyros from redrawing the layout.

Pyros will be enhanced in upcoming releases to improve redraw speed, and this environment variable will be removed.

Change Requests

Like many companies, Ciranova uses Bugzilla (www.bugzilla.org) to track change requests. As yet, Ciranova does not have a publicly accessible request tracking system. 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.

2007-12-12

- 199 uses of Instance::getParams() need to be fixed
- 210 Lib object should have method to return list of cells in library
- 285 addEnclosureRects() should not include Text bbox in calculation
- 290 Request method(s) to support hierarchical traversal
- 436 Request Dlo() usage in batch context
- 563 Enhance installer with default mode which installs everything
- 794 directory name should be stripped of white space pre-install
- 804 Minimizing use of CNI_ROOT
- 834 Installer tries to copy files into unrelated directory
- 849 More precise error and help messages for cndispcnv
- 851 cndispcnv problems with trying to convert Cadence CDB technology file
- 855 cndispcnv should perform basic check on input files
- 856 cndispcnv program not displaying exception messages for user
- 857 Misleading help and usage message for cndispcnv
- 861 Typo in cndispcnv error message
- 862 Some display purposes being generated twice by cndispcnv
- 863 Help message and consistency issues with cndispcnv
- 873 cndispcnv output is wrong when pattern is single element list
- 917 cndispcnv error - outputs purpose definitions twice
- 977 cndispcnv incorrectly converts cni130 technology files
- 980 cnversion reports incorrect path errors
- 1024 __version__ for all Python modules
- 1030 need summary after cnDRC engine run
- 1055 cnpy abort with Segmentation Fault
- 1064 Segmentation fault when creating lots of PyCells in Pycell Explorer
- 1065 Need to examine parameters for Spiral Inductor PyCell
- 1084 fgAddEnclosingRects() should support env option
- 1089 cnpy segfaults on single processor machine
- 1091 Need reduced __init__.py for making training projects in Wing
- 1106 cnpy aborts when unable to access a graphics display
- 1133 fgAddEnclosingRects needs to provide environment parameter
- 1202 Add missing tech/cni*/{santanaTech, combinedTech}/.oalib files
- 1220 Seg fault on solaris 8 creating design in optimized mode
- 1260 cndispcnv creates errors in the converted SantanaDisplay.tech file
- 1285 Tech API query for conditional enclosure versus encloser width
- 1301 pyros segfault on other than linux32 platforms
- 1325 Make the SMALLTEST cases in quickstart kit really small
- 1333 Need Ciranova trademark handling for all reference manuals
- 1339 segfault when viewer is used with re-exploring same design
- 1387 cnenablecache should allow relative paths for cache directory
- 1398 Request API to ungroup everything in DloGen
- 1401 cndisablecache raises exception when cache directory does not exist

1423 Smart functions and objects should filter non-drc Layers automatically
1448 Dlo.props has no description
1462 cnexp should support --no_core_dlos option
1464 Rule query results should be defined for duplicate cases
1469 Update parameters for Spiral Inductor baseKit PyCell
1474 cryptic error message when PYTHONPATH is not set correctly
1479 Please add oaGetVersion to external and internal packages
1480 installer fails on Solaris 2.8 with tesh and PATH=./bin:/usr/bin
1522 Double headers in \$CNI_ROOT/bin script
1523 Remove cni/devices/*.py from packaging
1525 installer aborts if LD_LIBRARY_PATH not set
1526 Correct sample tcshrc file to check for LD_LIBRARY_PATH
1527 Rename "configure" script to "install"
1531 Add cnstrmin/cnstrmout to internal & external packages
1534 cnpy segmentation fault for invalid ParamSpec default value
1540 Wing IDE fails to build library correctly when __init__.py less than 15 lines
1542 Wing IDE is not able to create tutorial project
1543 Problem with source tracing if the trace points to a file from pylib packages
1553 remove error message of readline, as it is not always available
1554 Enhance cnenablecache and cndisablecache messages
1575 cngenlib does not work in debug mode
1577 binary Ciranova techfile