

PyCell Studio, Version 4.0.1, Release Notes

Table of Contents

Introduction.....	1
New Features.....	1
Premium Features.....	2
Changes.....	3
Change Requests.....	4
2007-01-20.....	4
2006-12-20.....	4

Introduction

Welcome to PyCell Studio, Version 4.0.1, released 2007-01-20.

Version 4.0.1 marks the first year anniversary of the commercial release of PyCell Studio. As denoted by the change in major version number, there are significant additions, as well as changes, to the APIs. During the first year, the product strategy was to rapidly assimilate user feedback, and to introduce revolutionary technology. Reflecting our growing user base, and commercial support commitments, stability and backwards compatibility now become equally important. Subsequent versions of PyCell Studio will not introduce incompatible changes without an appropriate transition period. For users who have been using PyCell Studio from the beginning, we thank you for your patience, and for your valuable feedback, during 2006.

See Change Requests for specific differences from Version 3.2.1.

These release notes summarize the important enhancements, bug fixes, and other changes which have been made since Version 3.2.1. For complete technical details, please see the related documentation.

New Features

1. Moving from Python 2.4.2 to Python 2.5
Besides staying current with the Python language developments, PyCell Studio uses new Python 2.5 language constructs to improve code clarity for new device contexts and new multiple rulesets in the technology API.
2. Wing 2.1.3-1
Latest version of Wing IDE supports Python 2.5, and is more closely integrated with Pyros layout viewer. The new version will work with existing Wing IDE licenses.
3. Pyros layout viewer
Object delete, rotate, and mirror buttons added. Improved debugging capabilities:
 - Source code tracing. When enabled, a shape can be selected in Pyros, and the matching PyCell code which created that shape will be highlighted in Wing IDE.

To enable, choose “Enable Source Tracing” under the Ciranova menu in Wing IDE, when building supermaster default view. Otherwise, choose “Source Code Tracing” in the Pyros tool bar, or check “Source Code Tracing” box in Pyros Open Design.
 - Hierarchical debugging. Setting breakpoints in lower level PyCells now works. Wing IDE can be used to step into any level of the design hierarchy during PyCell code

evaluation.

- On-demand debugging. The older use paradigm requires setup of a Wing project prior to debugging PyCells. A newer, more convenient, use paradigm is introduced which allows Pyros to connect to an existing Wing IDE session. On-demand debugging will continue to be refined in future releases.

4. RoutePath

Route class replaced by RoutePath class. RoutePath implements an improved interface for creating paths; in comparison, a true router performs obstacle avoidance. RoutePath static methods:

- RoutePath.StraightLine
- RoutePath.StraightLineToBar
- RoutePath.CShape
- RoutePath.LShape
- RoutePath.ZShape
- RoutePath.Connect
- RoutePath.FlightLine

5. New style technology files

Technology files for fictional processes, CNI180 and CNI130, have been upgraded to new style. The older keyword-value format continues to be supported, but new style DRC commands enable more sophisticated modeling of process design rules.

A simple technology file checker has been added. See `cni.integ.techcheck`.

6. PhysicalComponent.makeArray()

Creating repeated placements of shapes simplified by `PhysicalComponent.makeArray()` method.

7. MosUtils

A set of `CompoundComponent` building blocks is provided to simplify writing MOS transistor PyCells. These building blocks include different varieties of source/drain contacts with support for abutment. Examples using `MosUtils` are provided in the `PyCellKit`, which can be separately downloaded, or found in `$CNI_ROOT/quickstart`.

8. PyCellKit adds support for multiple process technologies

`PyCellKit` is a collection of Makefiles and scripts for managing PyCell development. `PyCellKit` has been enhanced to support verification with multiple process technologies.

Premium Features

For customers who have purchased commercial support, additional premium features are available.

1. Device Contexts

Device contexts allows rules to be substituted depending on the context of the device being constructed. For example, contact-to-gate spacing is larger for a high voltage transistor than for a regular transistor. Device contexts simplify generation of the many different types of transistors commonly used in modern 90nm processes.

2. Multiple Rulesets

Design rules in `Santana.tech` technology file may be organized into multiple rule sets. Different rulesets may be activated within PyCells to enable the same code to generate different layouts. A

typical usage would be creating layout based on minimum design rules versus recommended DFM rules.

Changes

1. Consistent API signatures

All APIs involving two PhysicalComponents reviewed, and appropriate changes made to ensure consistent naming of keyword options.

2. DloGen methods moved to PhysicalComponent

DloGen methods, such as DloGen.fgPlace(), moved to PhysicalComponent class. This reorganization reflects better object-oriented design, and simplifies writing

CompoundComponents. Affected methods are:

- DloGen.abut()
- DloGen.alignEdgeToPoint()
- DloGen.alignLocation()
- DloGen.alignLocationToPoint()
- DloGen.fgAbut()
- DloGen.fgAddEnclosingPolygon()
- DloGen.fgAddEnclosingRects()
- DloGen.fgAnd()
- DloGen.fgFill()
- DloGen.fgMinSpacing()
- DloGen.fgNot()
- DloGen.fgOr()
- DloGen.fgPlace()
- DloGen.fgSize()
- DloGen.fgXor()
- DloGen.getSpacing()

For additional information on specific API changes, please refer to *Ciranova Python API Version 4.0 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.

2007-01-20

- 954 Documentation should have version numbers or dates
- 992 Should add Python API Conversion Guide document to ship list
- 1000 Segmentation fault for Pyros when errors in lib.defs library definition file
- 1001 Enhance Text class to have method to set drafting mode
- 1002 Text class should default drafting mode to False
- 1004 pyros help to point to correct documentation
- 1005 wingdbstub.py file not found by Pyros for Wing debugging
- 1008 cnp.py.exe silently aborts when msg files are missing
- 1015 Add dloPython/cni/devices/{__init__.py, MosUtils.py}

2006-12-20

- 68 Need getInst(instName) to get instance in current cell
- 85 lower-level pcell exception not reported in hierarchical pcell debugging
- 168 Perpendicular method for Direction object should handle more directions
- 180 Grid class enhancements
- 249 Instance Class object Improvements needed
- 330 roundOff() function should handle floating-point numbers
- 332 Contact.stretchTo(), stretchTowards() signatures
- 333 Move adjustAspectRatio() into sample devices
- 354 Route class LShape() and ZShape() methods need error checking
- 355 Route class StraightLine() methods need error checking
- 356 Route methods should be consistent when route can not be generated
- 377 Route needs to clean up invalid contacts
- 378 Inconsistent method names should be fixed
- 426 Contact should generate error for incorrect parameter names
- 430 CiraNova executables need consistent command-line option handling
- 451 Should try to consolidate multiple versions for DloGen addPin() method
- 456 Provide message to user when loading PyOA for the first time
- 497 Need better error message for invalid parameter name
- 500 Need log file for DRC run in CNGUI
- 502 Need rule descriptions in cni180 & cni130 tech files
- 508 Add Python API to call DRC engine with in-line rule definition
- 512 Change CiraNova spelling to Ciranova in C++ error messages
- 513 Change CiraNova spelling to Ciranova in Python error messages
- 520 Progress bar while waiting to open a large design
- 527 Box.{hasNoArea(),transform()} inconsistent with OA review
- 550 cntechconv should support INCLUDE directives
- 552 Rule without error producing commands should be discarded
- 562 Point class += and -= operator overloading returns None
- 567 "Assertion `position <= 0' failed" message for cni130

569 Reconsider use of aliases for rule names in Technology Class
579 Need Santana technology file manual
580 Need Santana display file manual
584 Consolidate overloaded C++ abut() methods into one abut() method
607 cnTechConv.exe to support standard ciranova options for executables
611 Select single solution when align/place odd-grid vs even-grid Comp's
614 clone() method should copy instance attribute of Python classes
624 Tutorial PDF isn't readable with Linux xpdf
662 Should clean up C++ API to agree with Python API as necessary
689 Consistent rendering of Instance of Submaster with PyCell eval error
691 Need Layer('error') to render instance and submaster with errors
692 Core Dump when using ParamArray Python code
694 fgMinSpacing() should have same signature as fgPlace()
695 fgMinSpacing() should not modify geometries to perform its calculation
696 place() should have consistent signature as fgPlace()
697 cnreplay should support standard command line options
718 Tech DB does not specify line number and file names for some reported errors
719 Missing error message in technology file parser
720 Provide grid resolution information in Layer class
722 request Box.getArea() method
733 OA bug with oaPolygon.isOrthogonal()
747 Remove on-grid constraint from fgAddEnclosingRect and fgAddEnclosingPolygon
748 cnpy help message misleading
749 fg*() functions support for 2 ShapeFilters
750 Consistent support for 2 ShapeFilters
751 alignLocation() should support 2 ShapeFilter options
752 alignEdge() should support 2 ShapeFilter options
753 place() should support 2 ShapeFilter options
756 PyCell Studio Tutorial is wrong on page 8, Project Name
759 PyCell Studio Tutorial page 8 is wrong, lib.defs change
781 cndisponv generates incorrect SantanaDisplay.drf
784 fgAddEnclosing*() methods should support edges also
786 fgAbut() should support optional align, same as fgPlace()
787 Bar does not automatically stretch to Route wire
789 Consistent keywords for fgPlace, place, fgAbut, abut, align*
790 Enhance snap() methods to support direction
791 rule discarded when uppercase and lowercase layer name used
792 Unique.Name() does not generate unique names
799 Improve reporting of errors for technology file problems
806 Duplicated entries of LPP pairs in Layer/Purpose subwindow
808 Bar.stretchTo(), Bar.trim() require existent of Contact
812 utlFormat and utlString conversion for "8-byte" long long int
817 Need support for unlimited enclosing shape
818 Request ShapeFilter support for fgAddEnclosing*()
822 Remove or fix constructor from Tech
824 New global function getSpacing() needs to be exported to Python
826 Decide on global functions for the Python API
827 place() method should use same keyword names as fgPlace()
828 Add help message for methods with two ShapeFilter parameters
829 New Box expand() methods should consider more directions
830 Rename Box Handle Point methods to use Location

831 Provide additional align methods for Box object
832 Need to update sed script for Python help messages
835 DloGen addPin() method should not take default layer parameter
836 Better error message for fgPlace(), fgMinSpacing(), getSpacing()
841 trailing space in comments
843 Consistency issue between place() and fgPlace() methods
850 Python Exception for endispconv program
864 fgPlace should consistently check for allowed direction values
866 Better handling of cases where comp and refComp are the same
868 cntechconv not properly handling exception
871 Wrapped C++ DLO Contact error
876 cntechconv fails when LPP found in *Rules() section.
881 Problem viewing ArrayInstance in pyros
882 Incorrect behavior in explorer/pyros when a new session starts
884 Changing parameters cause strange behavior
886 cntechconv to be part of external package
892 ContactRing API unexpected behavior when using addLayer and enclosingBox features simultaneously
895 Should try to consolidate multiple versions for DloGen addPin() method
897 PhysicalComponent doesn't support using the Transform class
898 Provide optional offset parameter for 2D alignment operations
901 cngenlib Segmentation Fault when comparing Direction.NORTH == None
902 Introduce design rule sets in Santana environment
903 Introduce device contexts in Santana environment
904 Evolution of ShapeFilter Class: methods
905 fgAbut() implementation enhancement
909 Replace "minEnclosureEnd" rule
910 Segment class contains() method does not accept keyword arguments
913 RoutePath findAdjacentInterconnectLayer() method should sort layer list
914 Better Python wrapping for RouteTarget getLayers() method
915 Pin layer methods should not raise exceptions when no Layers present
916 All OpenAccess shared libraries should be packaged for internal & external
918 add cnUtilOaBase to internal and external packaging list
919 add the new cnpy files to the release
922 Consolidating Point.snap() method with PhysicalComponent.snap()
923 cngenlib error message when running Wing
926 Getting members of enumerated class in Python
927 Error messages from RoutePath should be reported in UU, not DBU
928 All enumerated constants should be exported in cni.constants
929 Provide findInstTerm and findInstPin methods for InstTerm and Instance classes
930 Provide fgMerge() convenience method for Python API
931 Request consistency check for OpenAccess versions in PCell PlugIn
935 cngenlibgui has been removed from packaging list
939 PhysicalComponent method for repeated placements
940 How to compare cni.utils.list_of_str
941 Grouping.ungroup() enhanced to transfer ownership to other Grouping
943 need to add Helix and associated .so's to nightly build/release
944 Provide ability for user to specify choice of LShape Route
945 Provide CShape route method for RoutePath class
946 cntechconv wrapper script should be generated properly
948 Need Python API to directly create submasters

- 949 Need Python API to directly purge submasters
- 952 Inconsistent handling of exceptions in Pyros
- 961 DloGen makeGrouping() method ignores user-supplied name
- 962 Need to package pylib/cni/geo.py
- 967 New files to be packaged.
- 968 Grouping setName() method should perform better name checking
- 970 Python wrapping of temporary Box contains random data
- 973 broken pipe when using help() in cnpy
- 975 Should Instance connectivity be updated automatically when using Instance.setParams()
- 979 Do not reset all parameter values in parameter window upon error