

PyCell Studio, Version 3.2.0, Release Notes

Table of Contents

Introduction.....	1
New Features.....	1
Changes.....	3
Change Requests.....	4
2006-07-24.....	4

Introduction

Welcome to PyCell Studio, Version 3.2.0, released 2006-08-09.

See Change Requests for specific differences from Version 3.1.1.

These release notes summarize the important enhancements, bug fixes, and other changes which have been made since Version 3.1.1. For complete technical details, please see the related documentation.

New Features

1. Connectivity API completely rewritten

In response to the strong user feedback from Version 3.0 and 3.1, the Connectivity API has been completely rewritten. Highlights:

- Constructors added for Net, Term, and Pin.
- Classes added for InstTerm, InstPin, ShapeRef, RectRef, and PolygonRef.
- Pin model enhanced to support multiple shapes per Pin.
- Instance connectivity is specified as a Python dictionary, instead of a string.

2. Location class added

New Location class replaces confusing use of Direction to specify the 9 standard locations on a Box. Location values are also used to specify Text justification. The enumerated values for Location are:

- Location.LOWER_LEFT
- Location.CENTER_LEFT
- Location.UPPER_LEFT
- Location.LOWER_CENTER
- Location.CENTER_CENTER
- Location.UPPER_CENTER
- Location.LOWER_RIGHT
- Location.CENTER_RIGHT
- Location.UPPER_RIGHT

Affected methods include: Box.getHandlePoint(), Bar constructor, and moveTo() method for DloGen, Grouping, PhysicalComponent, and CompoundComponent.

3. Better alignment methods

align() method replaced with alignLocation(), alignEdge(), alignLocationToPoint(), alignEdgeToPoint(). These new alignment methods provide greater flexibility and clarity.

4. Named shapes
Shapes may be assigned names. Related methods are `Shape.getName()` and `Shape.setName()`. `Shape.find()` can be used to locate a named Shape.
5. Additional methods for shape classes
Basic methods added to many of the shape classes for completeness. Affected classes include Arc, Donut, Dot, Ellipse, Line, Path, PathSeg, Polygon, Rect, and Text.
6. Python API support for DRC
`DloGen.fgDrc()` method added to enable DRC execution to be initiated from Python. Previously, DRC could only be manually initiated from Pyros. Later enhancements for `fgDrc()` will enable use of DRC operations in layout generation.
7. OpenAccess Technology DB
When creating a technology library, `cnngenlib` now creates an OpenAccess technology database, `tech.db`, with information about layers, purposes, and `viewType` database units.
 - Enables use of layer mapping table when exporting to GDSII.
 - Supports database units other than OpenAccess default of 1000 dbu/uu.
8. Standardized command line options
All Ciranova programs, except `cnpy`, support standard command line options:
 - `-V, --version`
 - `-h, --help`
9. Enhancements to parameter constraint methods
 - `ParamConstraint`
New flexible parameter constraint with user-defined Python function.
 - `StepConstraint`
New constraint method requires parameter to be an integer multiple of step value. Typically used for checking transistor dimensions as a multiple of layout snap grid.
 - `RangeConstraint`
Enhanced to support single sided constraints.
10. Pyros
 - Direct creation of submasters
Pyros enhanced to support direction creation of submasters by changing parameter value in supermaster view. A separate design containing `PyCell` instances is no longer required to test behavior when changing parameter values.
 - Zoom-in and zoom-out buttons added
Previously, Pyros supported zoom-out only using the mouse scroll wheel. An astute user pointed out that some people don't have a scroll wheel mouse, especially Solaris users. New zoom buttons added for users with impaired mice. :-)
 - Display files translator
`endispcnv` provided to translate display files for Pyros.

Changes

1. SimplePath and MultiPath
SimplePath renamed to Path. Path renamed to MultiPath. New names better reflect the conventional industry terminology.
2. getRefPoint()
getRefPoint() method removed. Use Box.getHandlePoint().
3. Technology library binding
Use of library property, cnTechLibName, to bind design library to technology library has been replaced with standard OpenAccess library property, techLibName. The earlier use model, where the OpenAccess technology library could be different from the Ciranova technology library, was deemed too confusing to be useful. Now, the OpenAccess technology library and the Ciranova technology library must be the same.
4. Route methods
Route.connect() renamed to Route.Connect() for consistency. Route.FlightLine() supports an optional layer argument. Route.LShape(), Route.ZShape(), and Route.Connect() no longer support an option Net argument.
5. Topology.asString()
Topology.asString() replaced with Topology.getName() and Topology.setName().
6. compress() method replaced with PointList.compress()
compress() method removed from MultiPath, Path, and Polygon. Instead, Pointlist.compress() method should be called before creating these shapes.
7. DloGen methods
 - anchor() method removed. Use setOrigin().
 - Added currentDloGen() method, used to determine current context when writing Python CompoundComponents.

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.

2006-07-24

ID#	Description
43	Verify pin/terminal functionality
47	Add package interconnect to nets
60	Pin creation and access functions
80	Open Design dialog should use current L-C-V as default
88	Would like to see hierarchical name of selected single shape/instance easily
101	Enhanced Zoom commands for Santana GUI
109	Function to create instance terminals
110	Enhancements for terminal creation function
111	Function to create nets
112	Function to get all shapes associated with a pin
113	Function to get all pins associated with a terminal
135	CNGUI "Open Design" dialog should have default button set
177	API enhancement to add additional methods for shapes
191	Net does not wrap oaNet
207	Route.FlightLine() method enhanced for layer argument
213	Instance & Text methods for setting orientation
220	Provide width and height accessor methods for Dot object
228	Additional methods needed for Donut physical component
229	Additional methods needed for Text Physical component object
230	Additional methods needed for PathSeg physical component class
239	Incorrect exception name being used for DloGen::addToPin() method
248	Route LShape() and ZShape() methods not using Net parameter
252	Enhance addPin() methods for DloGen to return pin object
262	Pin object needs Pin::setAuthorPrefLayer() method
275	Pin object creation method needs better error message
276	Pin::getAccessDir() method needs better error message
277	Pin methods need to generate meaningful error messages
279	oaPin::find() method fails for several Pin object methods
282	Instance creation should make specification of nodelist optional
286	fgAddEnclosingRects() should only create on-grid geometries
305	fgPlace() enhancement to support alignment directive
306	Improve rendering of Text objects
312	Instance.create() method does not detect change of viewName
313	Instance.create() nodeList default parameters do not work as expected
315	Instance::Create() method should require nodeList parameter to be specified
325	Review removal of compress option from Polygon, SimplePath constructors
327	Transistor devices should follow SPICE terminal order convention
328	Correct X stipple pattern
335	cngui should support --help and --version
337	Remove DloGen anchor() method
341	cngenlib should create basic oaTech

359 CNGUI should update Layer window after drawing object in interactive mode
373 Layer pane does not refresh select/unselect status for a group of layers
387 Santana tech-binding property changed from cnTechLibName to techLibName
393 Need to strip leading and trailing blanks from file paths for dialogs
396 DLO Explorer and CNGUI not properly handling Python exceptions
442 Viewer coordinates do not correctly account for DBU-per-UU
462 Tracking Connectivity API bugs
466 PyCell Studio releases to include additional OpenAccess libraries
467 PyCell Studio releases to include Python .h files
470 cngui aborts with Segmentation Fault
474 moveTowards() method should support negative increments
486 Bad error message from Route.StraightLine()
487 Request utility to translate of display.drf to Ciranova display file
495 Need API capability for absolute rotate, Xmirror and Ymirror transformations
496 Constraint support for stepped values
506 Request change DIF to NOT for boolean operations in technology file
507 Request support for DRC rule description field
516 fgAddEnclosing*() should have same behavior when no enclosure rule exists
528 Running DRC in pyros gives unexpected error message
529 help(cni.dlo.ContactRing) causes an exception
530 help(cni.exp.dlo) raises an exception
531 Add test for help()
535 Support for Santana layer purposes in GUI
543 Provide Python API method for running DRC on a design
554 Add a README or VERSION file to indicate software version
560 Rename Route.connect() to Route.Connect()
561 Information panel mixes reported info for instance and shape
565 Rename path classes, SimplePath->Path, Path->MultiPath
573 Support for WITH EDGE layer operation
575 Support for RECTANGLE ENCLOSURE
581 "blank" stipple should be built-in to Pyros
582 Pyros should report errors encountered in display files
585 fgPlace/place() to support alignment directive
586 Enhance our Pcell evaluator to support submaster caching
587 Rename align() to alignEdge()
588 Need to be able to change SuperMaster and SubMaster parameters
589 Need to be able to move stretch handles in SuperMasters and SubMasters
592 Add ParamConstraint with checking in Python
593 PyCompoundComponent.clone() invokes PyCompoundComponent.__init__()
594 enhance alignEdge() to support second Direction argument
595 alignEdge() updated to use ShapeFilter()
596 place() updated to use ShapeFilter()
597 New Location class needs to be exported to Python
598 API changes for Location class
599 New alignLocation() method
600 New alignLocationToPoint(), alignEdgeToPoint()
601 Remove redundant alignToBox() method
603 cnexp to support standard ciranova options for executables
604 cngenlib to support ciranova standard options for executables
605 cngenlibgui to support standard ciranova options for executables
609 All variants of Contact should specify connectivity as optional dictionary

610 Method to get current or owning DloGen
616 Missing Python bytecode files after installation
619 If layer is defined but purpose is not, then pyros shows both fields as numbers
621 Enhance ParamArray to have remove() method
622 Contact incorrectly enforces minArea rule when routeDir*!=NONE
623 cngenlib error when creating viewType=schematics with -view option
625 Remove cn_BlockTerm_Order property for masters
629 Correct DRC Dialog Tree Output
630 Remove getRefPoint() function from Python and C++ code
632 Location class has incorrect values for Boolean methods
635 change converter to output "cds" instead of "cadence"
636 cnTechConv.py should check for correct command-line options
640 Documentation files included in automated build for release
642 Document limitations in overridden methods
643 Typo in error message about non-existent layers
646 Remove mandatory connectivity from ContactRing, Bar, MultiPath
647 cndispcnv to print out more information
648 Cryptic error message creating Instance with wrong connectivity argument
649 TermType.INOUT to be renamed to TermType.INPUT_OUTPUT
651 ContactRing() error when node=None
652 Bar() error when node=None
655 cntechconv help message should be improved
661 fgAddEnclosingRects have problems with derived layers
663 Support of WITH WIDTH operation
665 Need to change name for the Net class isGlobal property
671 Layout object setName() and PhysicalComponent.find(name)