

PyCell Studio, Version 4.2.6, Release Notes

Table of Contents

Introduction.....	1
New Features and Enhancements.....	2
Changes from 4.2.5.....	2
Available in 4.2.5, but omitted from the Release Notes.....	2
Change Requests.....	3
2008-11-19.....	3

Introduction

Welcome to PyCell Studio, Version 4.2.6, released 2008-11-19. This release has only a few, but important, new features. The goal is to support the *production* deployment of PyCells by design teams and silicon manufacturers.

Highlights of Version 4.2.6 include:

- Support for PyCell code bundling and encryption.
engenlib, the utility for creating PyCell libraries, adds a --bundle option. When specified, the PyCell code is packaged with the OpenAccess PyCell library, in either source or encrypted form.
- Stretch handle extensions
Extensions added to the stretch handle protocol to support unitized string parameters and multi-valued parameters.
- Support for OpenAccess 22.04.028 (DM4).
We're continuing to work closely to align our OpenAccess support with software releases from Cadence, Mentor Graphics, Springsoft, and Synopsys.

See Change Requests for specific differences from Version 4.2.5.

These release notes summarize the important enhancements, bug fixes, and other changes which have been made since Version 4.2.5. For complete technical details, please see the related documentation.

New Features and Enhancements

1. Command enhancements

- `engenlib --bundle=OPTION` where `OPTION` is one of: `source`, `bytecode`, `encrypted`
PyCell code is packaged with the OpenAccess PyCell library. The package name is preserved. Python restrictions for package names apply. Do not use two different packages with the same name in different PyCell libraries.
- `cnchecklocks`
Identify OpenAccess designs which are locked for edit.
- `Pyros`
Ciranova layout viewer, `Pyros`, adds new buttons to simplify the process of reviewing DRC errors.

2. API Enhancements.

- `AttrDict`
Use of `Dictionary` from `cni.integ.common` is deprecated, in favor of faster `AttrDict` from `cni.utils`. `AttrDict` implements a dictionary whose key/value pairs are conveniently accessible using attribute syntax.
- `cFloat`
Python represents floating point numbers in double precision. Use `cFloat` for setting OpenAccess parameters or properties as single precision. Mostly used for support of legacy PCells.
- `Instance()` and `InstanceArray()`
`Instance()` and `InstanceArray()` add an optional *trans* argument, simplifying the task of setting the location and orientation of an instance.
- `Instance.setMaster()`
New method added to allow changing of instance masters.
- `PhysicalCompRef.getProps()`
New method added to hierarchical component references, to support property retrieval. Reference components cannot have properties set upon them.
- `stretchHandle()`
Adds an optional *key* argument, for support of multi-value parameters, as would be required for a dynamic number of stretch handles. This case occurs when the number of stretch handles depends on the value of another PyCell parameter. Also, protocol definition is extended to require applications to support string-type parameters with units, such as “10u”.

Changes from 4.2.5

Minor correction, making `Instance.setParams()` consistent with `Instance()`. When creating a PyCell instance, illegal parameter names were flagged as an error. In 4.2.5, when setting parameters of an existing PyCell instance, illegal parameter names were silently ignored. In 4.2.6, setting illegal parameter names will be flagged as an error, for consistency.

Available in 4.2.5, but omitted from the Release Notes

`DeviceContact`, a smart object derived from `CompoundComponent`, was introduced for construction of devices such as resistors and MOS transistors. Unlike `Contact`, `DeviceContact` is not an PCell instance, but contains shapes in the current design, a necessary requirement for modeling pins in OpenAccess.

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.

2008-11-19

- 538 Would like ability to step from one DRC error to the next in Pyros
- 953 Contact should allow user to specify cuts in both X and Y directions
- 990 Consistent help message and option syntax for utilities
- 1242 Method and class names should be listed alphabetically in Python API manual
- 1487 PyCell code encryption
- 1489 cnhelp utility should provide list of available topics
- 1491 If help information does not exist for topic, cnhelp generates traceback
- 1571 ShapeFilter methods should not be exported to Python
- 1642 ENCLOSURE with OUTSIDE_IS_ERROR doesn't report errors when outer layer is empty
- 1854 Remove the need for specifying path to deviceKit in PYTHONPATH
- 1986 cngenlib should also support tilde notation
- 2015 Need to get tech name (tech info) during definePcells()
- 2048 Enhance license notification to provide accurate day countdown
- 2054 Open design view-only - can't remove rulers
- 2061 copy.copy() should be disabled from copying DLO objects
- 2087 Instance*() constructor to have Transform argument
- 2088 PyCell parameter passing should allow C float type
- 2117 \$CNI_ROOT/tech/lib.defs must be packaged and installed
- 2119 Dlo() should support "w" or create mode
- 2122 Tracking of documentation changes for 4.2.6 release
- 2135 ContactRing method getContact returns reference, should return pointer
- 2141 ContactRing should use fillLayers to influence spacing
- 2145 Pin and InstPin getLayers() method do not filter out identical layers
- 2146 String representation for Layer objects ignores purposes
- 2158 Support storage of PyCell code in supermaster view
- 2161 Display information conversion issues.
- 2178 Enhance installer to require empty installation destination
- 2180 Grouping.overlaps() raises exception and aborts cnexp
- 2190 Need to audit Santana.tech viaLayers() mask number
- 2198 Dlo() should support retrieval of Grouping when made persistent
- 2208 Weird stuff APIs to be re-organized
- 2209 Hotfix to support DM data in PyCell API
- 2211 cngenlib gives errors
- 2214 missing diode in IPL code
- 2217 \$CNI_ROOT/pylib/cni/quickstart.py should not be packaged with PyCell Studio
- 2234 Request Instance.setMaster()
- 2237 When layer is found, but not defined, print readable error message
- 2244 Need documentation and example of Numeric class
- 2250 Change cntools.ils to cntools.ext
- 2273 Documentation on Path.setStyle() is incorrect
- 2287 cni explorer should not require GUI
- 2299 INSIDE_OF option of WITH_NEIGHBOR to be supported

2311 DRC SELECT CHECK needs to be converted
2312 TOUCH !=2 output is wrong
2324 cnchecklocks to be added to santana package
2348 ContourAttrs.id() should return polygon ID
2372 Improve error messages for Abutment Control PCell
2377 Stretch handle enhancement for string valued parameters
2382 Stretch handle enhancements to be documented
2383 Pyros DRC browsing enhancements to be documented
2384 code bundling and encryption to be documented
2385 class AttrDict to be documented
2387 Instance.setMaster() to be documented
2388 *Contact() documentation or API change?
2389 transform argument for Instance and InstanceArray
2391 Line - API change or documentation fix
2392 cFloat needs to be documented
2393 Point - API change or documentation fix
2420 Rename layer 'instance' to 'sys_instance'.
2421 Rename select, background and grid layers
2458 ciranova.py does not work with newer versions of wing
2464 fgDeriveLayer() needs to give better error messages for exceptions
2471 PhysicalComponent.clone() source tracing reports line for original shape, not clone()