

PyCell Studio, Version 4.3.3, Release Notes

Table of Contents

Introduction.....	1
New Features and Enhancements.....	2
Library Recompilation.....	2
Change Requests.....	3
2009-10-01.....	3

Introduction

Welcome to PyCell Studio, Version 4.3.3, released 2009-10-07. This is a feature release. New enhancements are based on:

- Requests from customers on commercial support.
- Capabilities needed for customer PyCells developed by Ciranova consulting services.

Highlights of Version 4.3.3 include:

- New utility for dynamic PyCell code reloading.
- New utility for updating PyCell supermaster view.
- New utility option for selective PyCell creation or update.
- Technology files for Springsoft Laker and Synopsys Custom Designer.
- Creating PyCells other than layout view.
- PyCell commands to access base cell CDF.
- PyCell commands enhanced to support bus notation.
- NumericStepConstraint – a new parameter constraint for string-valued parameters.

Highlights of Version 4.3.3 premium features include:

- Technology file versioning.
- PyCell code versioning.

See Change Requests for specific differences from Version 4.2.8.

These release notes summarize the important enhancements, bug fixes, and other changes which have been made since Version 4.2.8. For complete technical details, please see the related documentation.

New Features and Enhancements

1. New command line utility, `cnsendcmd`.
When using PyCells in other EDA applications, the PyCell author had to exit and restart the application to reload any PyCell code. New command, `cnsendcmd`, enables the author to send a message to the PyCell plug-in to reload PyCell code..
2. New command line utility, `cnupdatepycell`.
When deploying PyCells, some companies allow their customers to have limited ability to change default parameter values. New command, `cnupdatepycell`, enables updating of the supermaster view.
3. New `cnenglib` option, `--for_pycells`
When working with a large PyCell library, building the entire library can be a time-consuming task. New option, `--for_pycells`, allows the PyCell author to selectively build or update specific PyCells in a PyCell library.
4. Technology files for partner EDA applications
Technology files for Springsoft Laker and Synopsys Custom Designer are now included for Ciranova generic demonstration 180nm and 130nm process technologies.
5. PyCells can be used for other views
For building complete PDKs, PyCells can also be used for building symbol views.
6. PyCell commands for base cell CDF.
New `cni.integ.cdfUtil` module provides read-only access to base cell CDF. This simplifies the task of coordinating values in CDF with PyCell code.
7. Bus notation support.
Previous versions of PyCell Studio stored names with bus notation as scalar objects. Since Ciranova Helix, the analog device placer, needed to support bus notation, the capability was added into PyCell Studio also.
8. New `NumericStepConstraint` for parameters.
New parameter constraint, `NumericStepConstraint`, was added. This parameter constraint is the string equivalent of the `StepConstraint` for float-type parameters.

Library Recompilation

New technology versioning capability in PyCell Studio 4.3.3 requires recompilation of PyCell libraries containing Ciranova C++ smart objects, such as `Contact`, `AbutContact`, and `ArrayInstContact`. New `RectC` C++ Pcell is also added to technology libraries to support technology versioning. No PyCell code changes are required. Please rerun `cnenglib` and recreate your libraries.

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.

2009-10-01

- 419 use of Contact setMinCuts()/getNumCuts() methods can lead to core dump
- 869 Give help message when command-line options are incorrect
- 950 Persistent stretch handle support
- 1471 Documentation enhanced for example of stretch handles on instances
- 1472 Sample SKILL code should support stretch handles on instances.
- 1478 Contact to support 'minLargeViaArrayCutSpacing' rule
- 1718 ContactRing fill option modified to require enclosure of all members
- 2219 Update documentation to use DeviceContact as appropriate
- 2302 DeviceContact layer*Ext description
- 2488 Unique identifier for all releases
- 2508 PyCell Studio should have Laker and Custom Designer tech files
- 2545 Request API to access base cell CDF
- 2636 Test OA_PLUGIN_PATH, and update \$CNI_ROOT/quickstart/*rc files
- 2699 cngnlib --bundle enhancement to support only .pyc files
- 2745 layers with large numerical id cause seg fault
- 2762 Santana.tech revision support in PyCell code
- 2777 cntechconv doesn't convert rules from referenced tech libs
- 2823 PyCell Explorer using TMP variable
- 2826 Santana/Helix needs to support designs with OA bus-bit and vector-bit
- 2863 cndispconv error
- 2881 Update cngnlib help message to use CNI130 (versus CNI180)
- 2890 ArrayInstContact ruleset support inconsistent with Contact
- 2915 CNI_LOG_<TYPE> wastes cpu even when redirected to /dev/null
- 2937 cngnlib --bundle requires write permission
- 2946 Test case causes tools to abort
- 2990 fgNot() does not produce correct result with complex shapes
- 3033 DRC script creates puzzling results
- 3035 PyCell Explorer does not create library data.dm
- 3036 Tech API OA exception due to missing library data.dm
- 3053 Wing IDE integration (ciranova.py) requires update to new quickstart files
- 3055 Windows documentation update for new quickstart files
- 3058 wing "New training project" should use new files from quickstart
- 3068 PyCell code versioning support
- 3072 cngnlib support for partial library update
- 3074 0 dbu value in OFFGRID operation causes segfault
- 3077 Add cngnlib and cnsendcmd to standard (free) PyCell Studio package
- 3078 Add detail to cnsendcmd help message
- 3079 OUTSIDE operation produces wrong results
- 3086 Utility to update PyCell default parameter values
- 3088 viewType should be correctly obeyed when specified
- 3093 improve OFFGRID analyzer (special case of boundary points)