SIEMENS EDA

How to Verify Waivers on Designs With Updated Cell Names

2023.3

© Siemens 2023 | Siemens Digital Industries Software | Where today meets tomorrow.

Outline

- Objective
- Included Files
- Description
- Directions
- Results
- Conclusion

© Siemens 2023 | Siemens Digital Industries Software | Where today meets tomorrow.

Objective

In this Support Kit, you should learn how to verify your waivers database on designs with updated cell names using **waiver_util** supported in the auto-waivers flow

© Siemens 2023 | Siemens Digital Industries Software | Where today meets tomorrow.

Included Files

File	Description
layout.gds	Layout file
layout.gds.layerprops	Layer properties file
waived.gds	Original waivers database
drc.rules	DRC rules deck
waiver_criteria.file	Waiver criteria file
waivers_ver_setup_org.file	Waiver verification setup file with original waivers database specified
waivers_ver_setup_new.file	Waiver verification setup file with new waivers database specified
cell_rename_file	Text file that contains cell renaming instructions
runme	Script to generate and verify waivers and run Calibre nmDRC
mv-results.csh	Script to move the results of each run into the designated directory
clean	Script to cleanup directory

© Siemens 2023 | Siemens Digital Industries Software | Where today meets tomorrow.

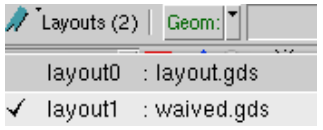
Description
>> Layouts

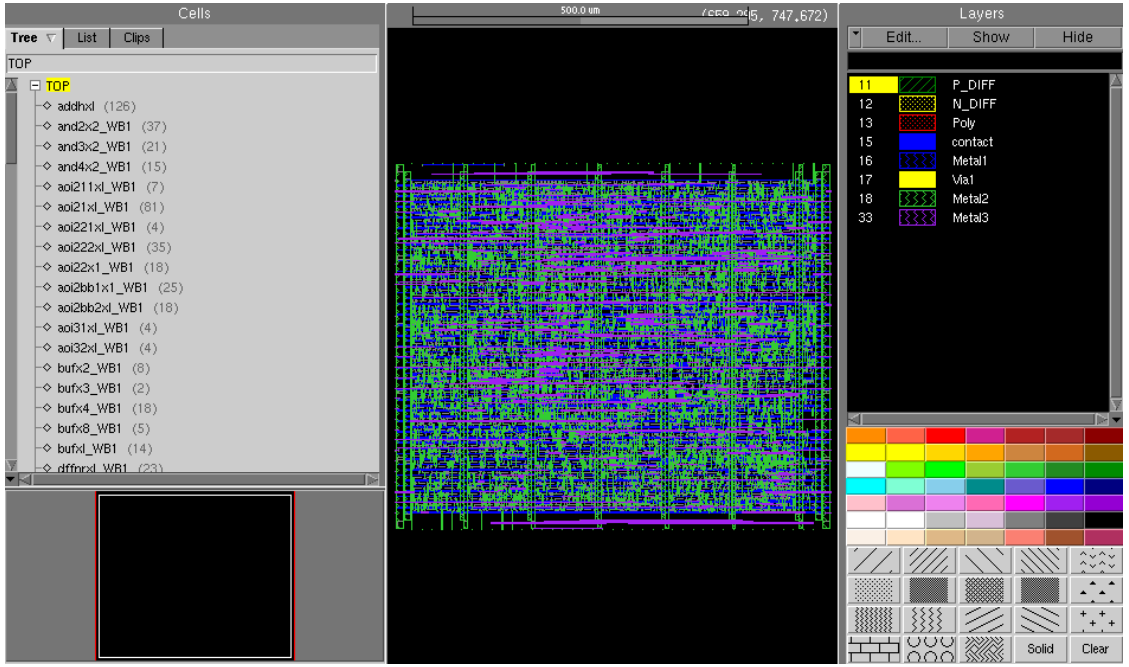
- Type the following command in your terminal to open Calibre DESIGNrev with the design and waivers databases:

  calibredrv -individaul layout.gds waived.gds

- In the design database "layout.gds“, investigate the name of the cells in the design database. The original waivers database should waive the violations in the following cells: addhxl, aio2x1_WB1, and3x2_WB1, dffrx4_WB1, mx2x2_WB1 and mx2x4_WB1

- Switch to viewing the original waivers database in Calibre DESIGNrev by doing the following:
  - From the layout tab → choose the "layout1 :waived.gds" from the pull-down menu:





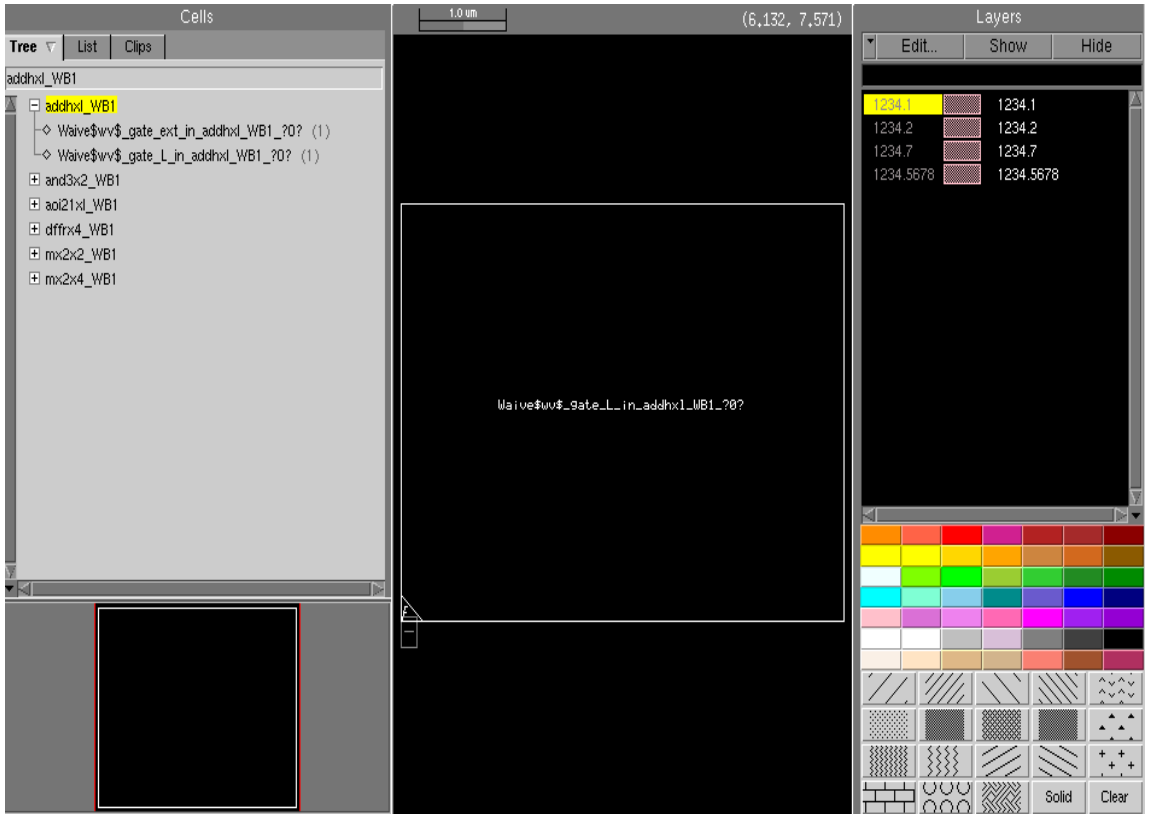
© Siemens 2023 | Siemens Digital Industries Software | Where today meets tomorrow.

Description
>> Layouts

- In the *“waived.gds”,* investigate the cells in your waivers database. Each cell should contain sub-cells that correspond to the generated waiver for each check/cell, having the following naming conventions:
  Waivewv_**check**_**name**_in_**cell**_**names**_?0

- Expand cell *“addhxl_WB1”.* It contains two sub-cells that correspond to the waiver shapes generated for each of the following rule checks:
  gate_ext →
  Waivewv_**gate**_**ext**_in_**addhxl**_**WB1**_?0?
  gate_L →
  Waivewv_**gate**_**L**_in_**addhxl**_**WB1**_?0?

- Notice that the waivers database contains a different cell name *“addhxl_WB1”* from the design database *“addhxl”*



- After investigating the layout, close Calibre DESIGNrev

© Siemens 2023 | Siemens Digital Industries Software | Where today meets tomorrow.

Description

- As designs go through multiple cycles, the cell names in the design database could differ from the original cell names with which the waivers were generated

- Hence, it is important to update the waivers database with the new design database cell names using a mapping file, otherwise, the waivers will not be applied, and the DRC results database will contain violations that should be waived

- The waiver utility **“waiver_util”,** which is included in Calibre software tree, performs certain functions on an input GDSII or OASIS input file specified by optional arguments and outputs an updated GDSII or OASIS file

- We can use the **waiver_util** command to rename waiver cell names to match the cell names found in the updated design using **-r** argument with **rename_cells_file** which contains the cell names that will be changed

```
waiver_util -mergedesc -i org_waiver_database -o updated_waiver_database -r rename_cells_file
```

Where **-mergedesc** is a required argument that specifies adding or updating the waiver criteria text objects in waiver cells in the input file and writing the text objects to the output file.

© Siemens 2023 | Siemens Digital Industries Software | Where today meets tomorrow.

Description
>> Rule Deck

```
LAYOUT SYSTEM GDS
LAYOUT PRIMARY TOP
LAYOUT PATH "layout.gds"
DRC RESULTS DATABASE "results.db"
DRC MAXIMUM RESULTS ALL
DRC CELL NAME YES CELL SPACE XFORM
```

Only part of the DRC rule deck is shown in this slide

- The following SVRF statement **must** be added within the rule decks whenever they are used to run the auto-waivers flows:

  DRC CELL NAME YES CELL SPACE XFORM

- This SVRF statement appends the cell name and cell space coordinates associated with each DRC result in the hierarchical ASCII DRC database. Moreover, the XFORM keyword, appends the transformation data to be used by other tools, such as RVE, to allow error highlighting in both the cell and top-level contexts

© Siemens 2023 | Siemens Digital Industries Software | Where today meets tomorrow.

Description
>> Files Used During the Auto-waivers Run

- The waiver setup file should contain the name of the waivers database on which the waiver verification flow will be done

- The cell rename file used in the waiver utility using the ***-r*** argument contains the following:

```
addhxl_WB1 addhxl
```

  Indicating that the waiver cell name *“**addhxl_WB1**”* found in the input file *“waived.gds”* will be changed to *“addhxl”.* This change will be reflected on the output waivers database “*waived_updated.gds*”

- The first waiver setup file “waiver_ver_setup_org.file” uses the original waivers database:

```
WAIVER_CRITERIA waiver_criteria.file
LAYER_NUMBER 1234
DATATYPE_NUMBER 5678
WAIVER_DATABASE waived.gds
MERGE NO
```

- The second waiver setup file “waiver_ver_setup_new.file” uses the updated waivers database:

```
WAIVER_CRITERIA waiver_criteria.file
LAYER_NUMBER 1234
DATATYPE_NUMBER 5678
WAIVER_DATABASE waived_updated.gds
MERGE NO
```


© Siemens 2023 | Siemens Digital Industries Software | Where today meets tomorrow.

Description
>> Runme

```
setenv RUN_TYPE verify_waiver_with_org_waivers
calibre -drc -hier -waiver waiver_ver_setup_org.file drc.rules | tee waiver_v1.log
source mv-results.csh

setenv RUN_TYPE verify_waiver_with_updated_waivers
waiver_util -mergedesc -i waived.gds -o waivers_for_renamedcells.gds -r cell_rename_file
calibre -drc -hier -waiver waiver_ver_setup_new.file drc.rules | tee waiver_v2.log
source mv-results.csh
```

- The runme file invokes two Calibre Auto-waivers runs:
  1. With “RUN_TYPE” = **verify_waiver_with_org_waivers**, it will invoke a DRC waiver run with the original waiver database *“waived.gds”*
  2. With “RUN_TYPE” = **verify_waiver_with_updated_waivers**, it will invoke the following:
     - Waiver utility to rename the cell names in the original waivers database stated in the *“cell_rename_file”* and output it to the new waivers database
     - DRC waiver run with the updated waivers database *“waived_updated.gds”*
- After each Auto-waivers run, the mv-results.csh script will create a directory with the RUN_TYPE variable name and move the generated databases, log, and summary files into it

© Siemens 2023 | Siemens Digital Industries Software | Where today meets tomorrow.

Directions
>> Results with Original Waivers Database

- Invoke the runme file by typing the following command in the terminal:

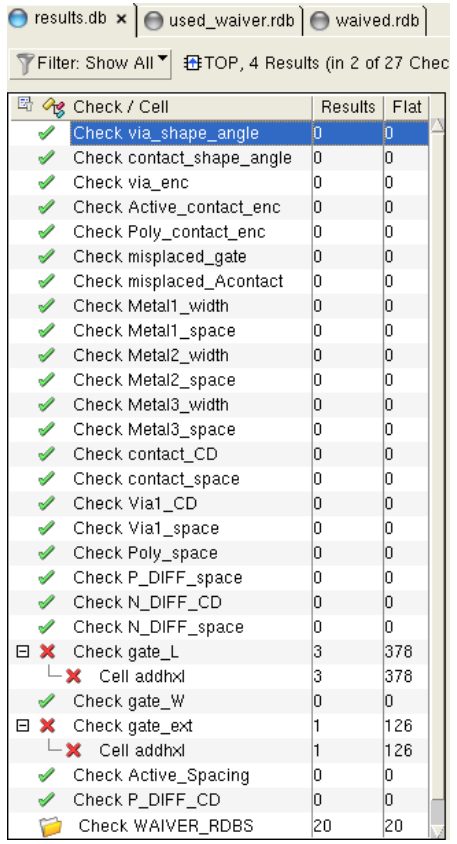
  source runme

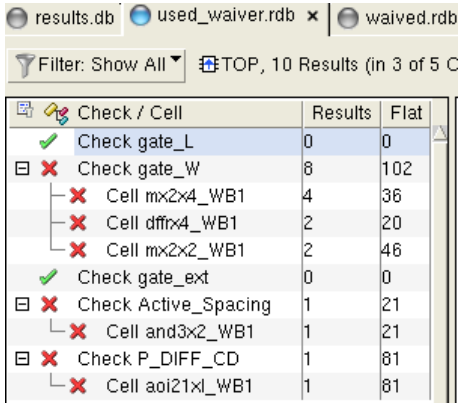
- Open the DRC results database from the first run in Calibre RVE by typing the following command:

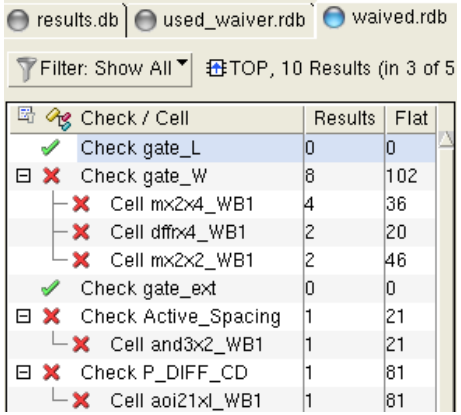
  calibre -rve verify_waiver_with_org_waivers/results.db

- We notice that the violations of **gate_ext** and **gate_L** checks haven't been waived

- Close Calibre RVE after investigation







© Siemens 2023 | Siemens Digital Industries Software | Where today meets tomorrow.

Directions
>> Results with Updated Waivers Database

- Open the “waived_updated.gds” through Calibre DESIGNrev by typing the following command:

  calibredrv waived_updated.gds

- Investigate the cells in the updated waivers database and expand cell *“addhxl”.* it contains the two sub-cells that correspond to the waiver shapes generated for each of the following rule checks:

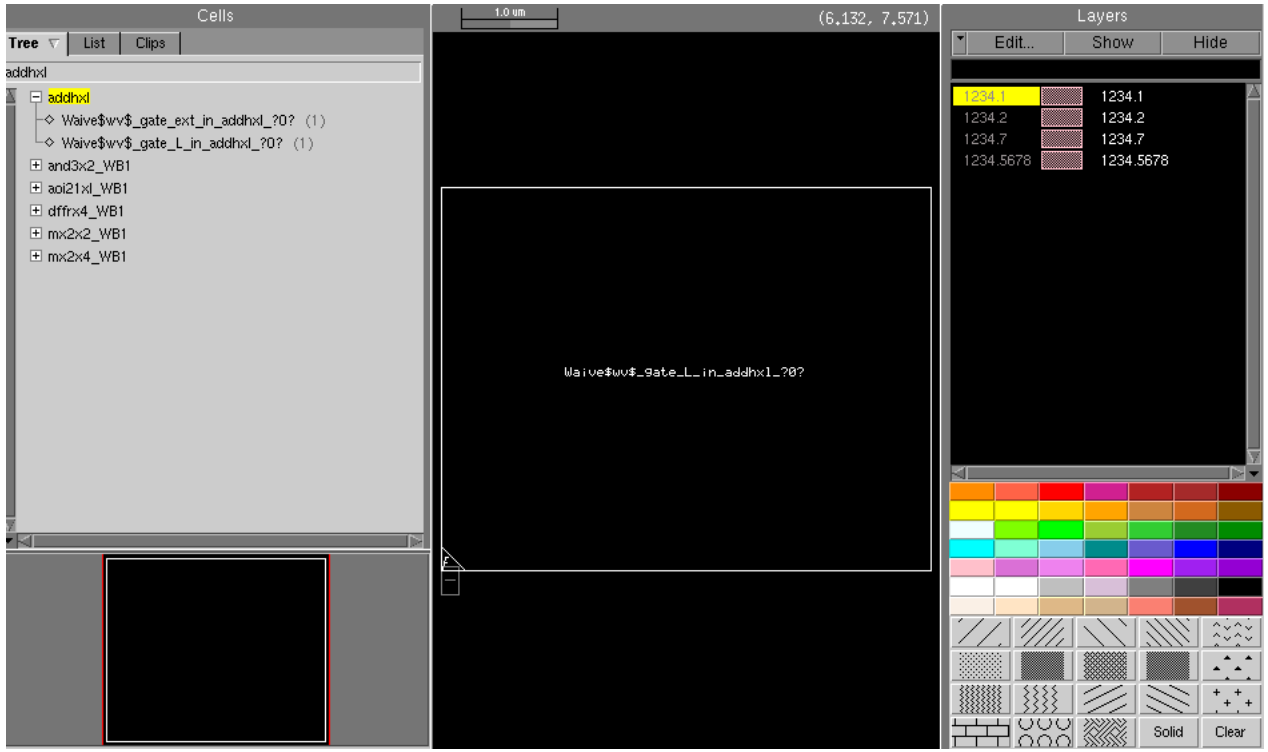
  gate_ext →

Waivewv_**gate**_**ext**_in_**addhxl**_?0?

  gate_L →

Waivewv_**gate**_**L**_in_**addhxl**_?0?

- Notice that the waivers database contains the same cell name *“addhxl”* from the design database *“addhxl”*



- After investigating the layout, close Calibre DESIGNrev

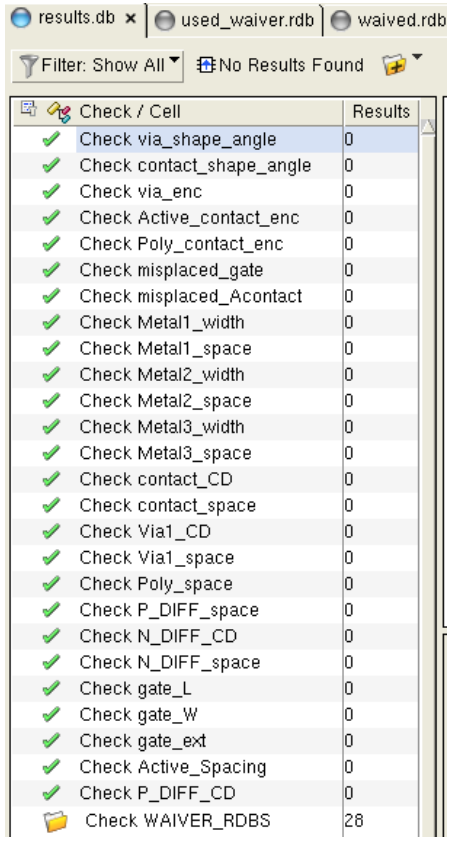
© Siemens 2023 | Siemens Digital Industries Software | Where today meets tomorrow.

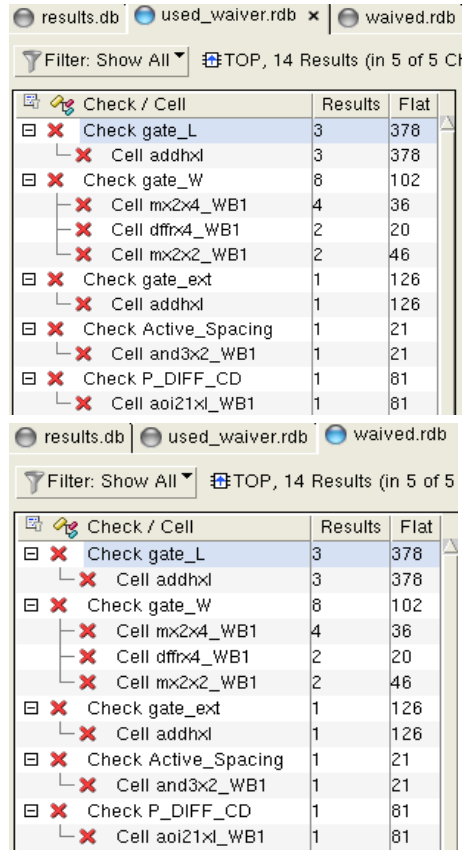
Directions
>> Results with Updated Waivers Database

- Open the DRC results database from the second run in Calibre RVE by typing the following command:

```
calibre -rve verify_waiver_with_updated_waivers/results.db
```

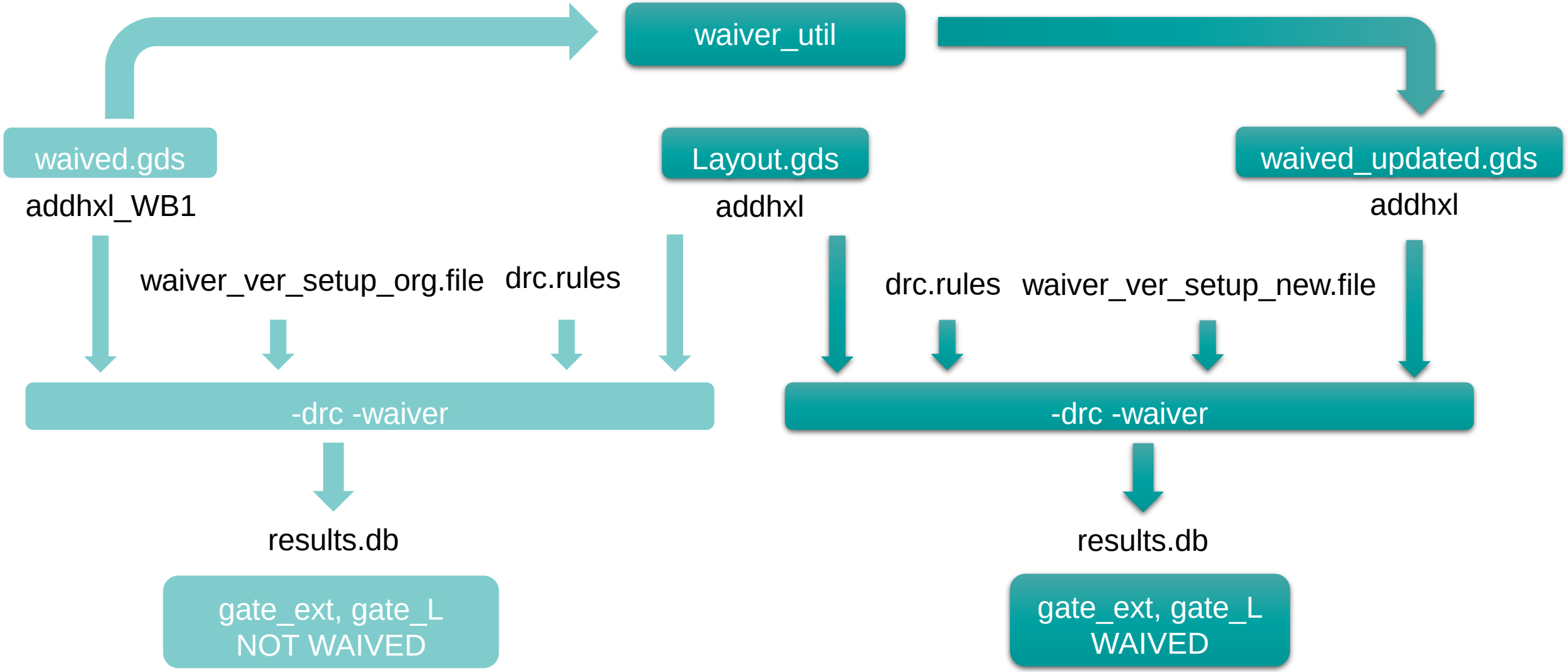
- We notice that all waivers are used as we changed the hierarchal cell names to be the same using the waiver utility





© Siemens 2023 | Siemens Digital Industries Software | Where today meets tomorrow.

Description
>> Waiver Utility Flow



© Siemens 2023 | Siemens Digital Industries Software | Where today meets tomorrow.

Conclusion

- As designers update their design databases, it is important to have Calibre Auto-waivers relate the current cell names in the design database to the original cell names with which the waivers database was generated, otherwise the waivers will not be applied, and the DRC results database will contain violations that should be waived

- Calibre Auto-Waivers provides a useful utility to update the waiver database from the old Calibre Auto-Waivers run with the new design database cell names that will be verified using a mapping file

- For more information about Calibre Auto-Waivers, please refer to **Calibre® Auto-Waivers™ User's and Reference Manual**

© Siemens 2023 | Siemens Digital Industries Software | Where today meets tomorrow.

Unpublished work. © 2023 Siemens

This software or file (the “Material”) contains trade secrets or otherwise confidential information owned by Siemens Industry Software Inc. or its affiliates (collectively, “SISW”), or SISW’s licensors. Access to and use of this information is strictly limited as set forth in one or more applicable agreement(s) with SISW. This material may not be copied, distributed, or otherwise disclosed without the express written permission of SISW, and may not be used in any way not expressly authorized by SISW.

Unless otherwise agreed in writing, SISW has no obligation to support or otherwise maintain this Material. No representation or other affirmation of fact herein shall be deemed to be a warranty or give rise to any liability of SISW whatsoever.

SISW reserves the right to make changes in specifications and other information contained herein without prior notice, and the reader should, in all cases, consult SISW to determine whether any changes have been made.

SISW MAKES NO WARRANTY OF ANY KIND WITH REGARD TO THIS MATERIAL INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, AND NON-INFRINGEMENT OF INTELLECTUAL PROPERTY. SISW SHALL NOT BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, CONSEQUENTIAL OR PUNITIVE DAMAGES, LOST DATA OR PROFITS, EVEN IF SUCH DAMAGES WERE FORESEEABLE, ARISING OUT OF OR RELATED TO THIS PUBLICATION OR THE INFORMATION CONTAINED IN IT, EVEN IF SISW HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

TRADEMARKS: The trademarks, logos, and service marks (collectively, “Marks”) used herein are the property of Siemens AG, SISW, or their affiliates (collectively, “Siemens”) or other parties. No one is permitted to use these Marks without the prior written consent of Siemens or the owner of the Marks, as applicable. The use herein of third-party Marks is not an attempt to indicate Siemens as a source of a product, but is intended to indicate a product from, or associated with, a particular third party. A list of Siemens’ Marks may be viewed at: http://www.plm.automation.siemens.com/global/en/legal/trademarks.html

Support Center: https://support.sw.siemens.com/
Send Feedback on Documentation: https://support.sw.siemens.com/doc_feedback_form