IC Compiler II Library Preparation Reference Methodology

R-2020.09

IC Compiler II

Sep 2020

IC Compiler II LIBPREP-RM R-2020.09

What's New?

- This release includes 1 new feature for read_lef
  1. Option -configure_frame_options is always part of read_lef to configure the appropriate frame options

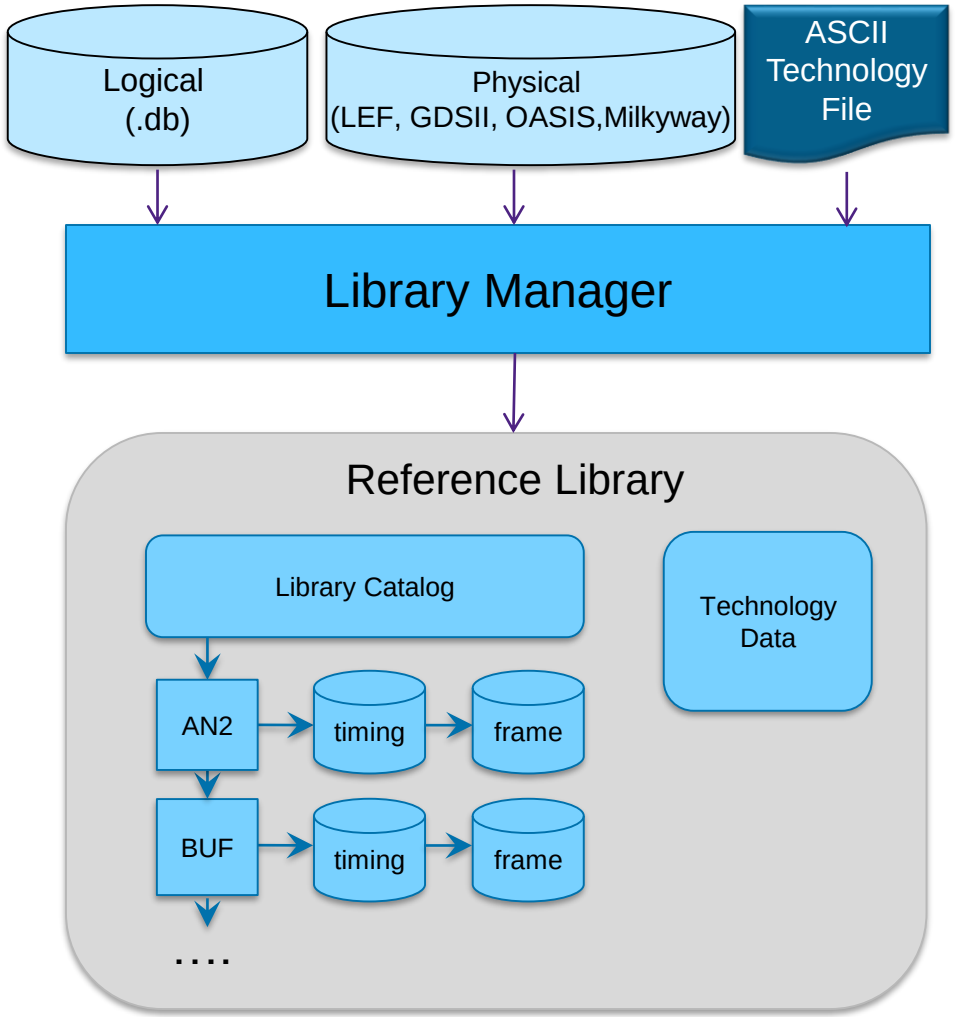
  2. Update RM in

     -read_physical.tcl

IC Compiler II Reference Library Management

Library Manager

- Library preparation is done in a tailored application, called **IC Compiler II Library Manager.**
- Unified logical-physical models are constructed and stored in a **reference library.**
- The library manager automatically explores and aligns all input data.
- Data inconsistencies are reconciled as much as possible.
- You can edit and adjust the input data if necessary.



SYNOPSYS®

IC Compiler II LIBPREP-RM R-2020.09

Four Source Types

- The IC Compiler II LIBPREP-RM supports four source types:

$source_type: gds_lef_oasis

- Uses the GDSII, LEF, or OASIS files as inputs
- Supported flows: normal, physical, frame, exploration, etm

$source_type: ndm

- Uses existing cell libraries as inputs
- Supported flows: normal, exploration, etm, aggregate

$source_type: combined

- A fixed flow in two steps (frame + exploration)
- Uses the GDSII, LEF, or OASIS files to generate the reference library by using the frame flow, and then imports the logic library files by using the exploration flow

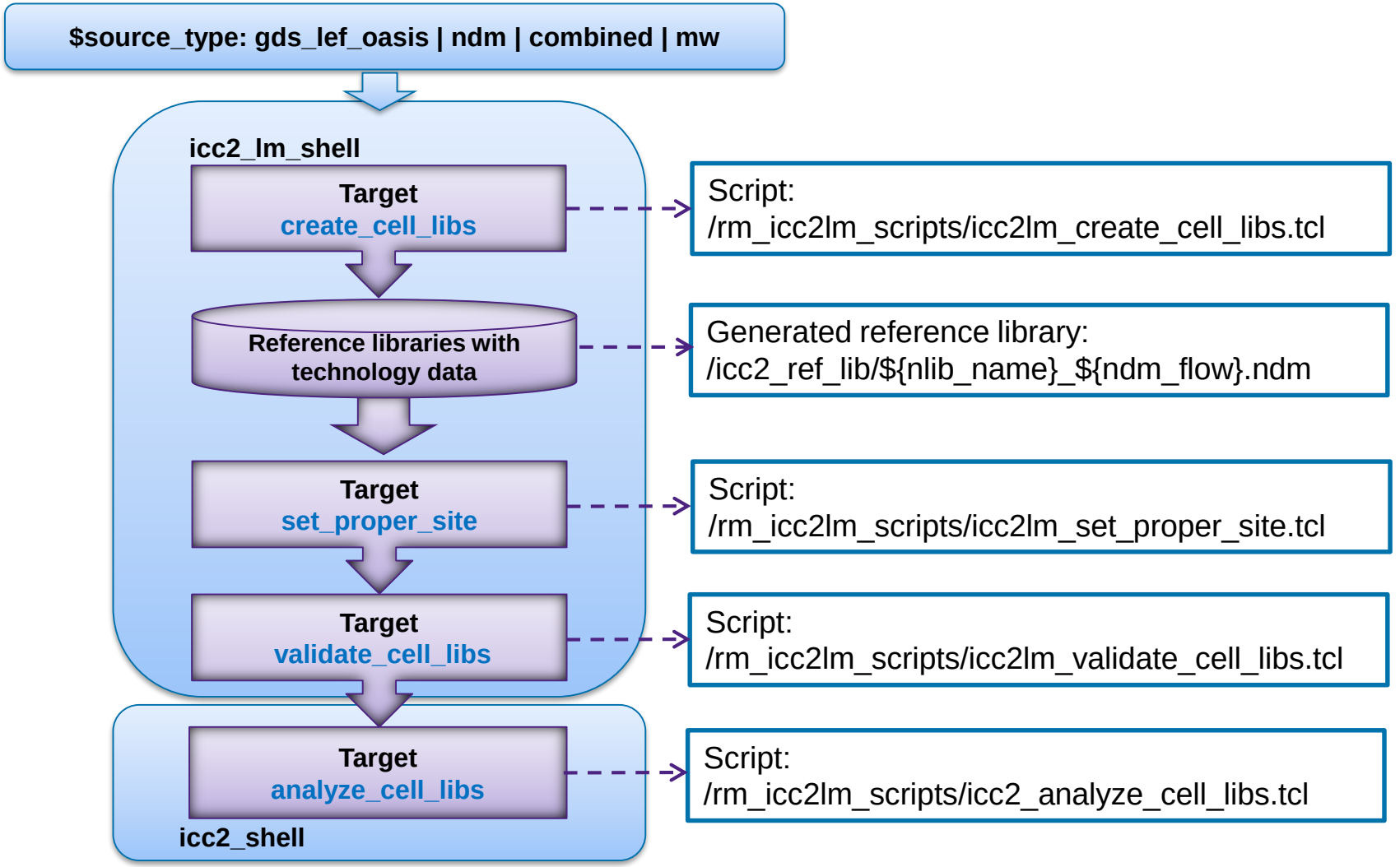
$source_type: mw

- A fixed flow in two steps (import_icc_fram + exploration)
- Uses the Milkyway reference library to generate the frame-only library by using the Milkyway migration flow, and then imports the logic library files by using the exploration flow

SYNOPSYS®

IC Compiler II LIBPREP-RM R-2020.09

Flow Chart

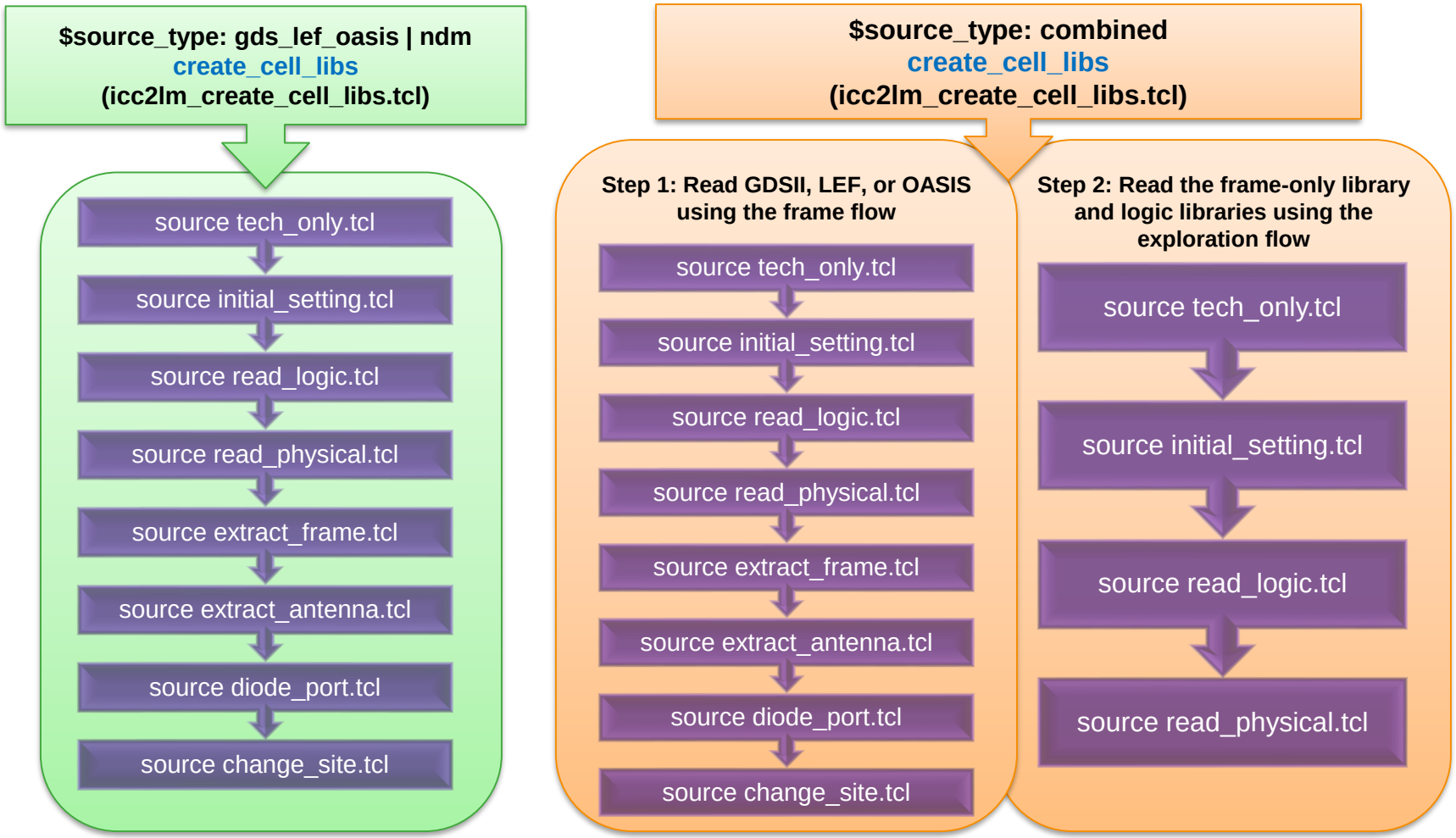


SYNOPSYS®

IC Compiler II LIBPREP-RM R-2020.09

create_cell_libs Target

- This step creates cell libraries within a library workspace

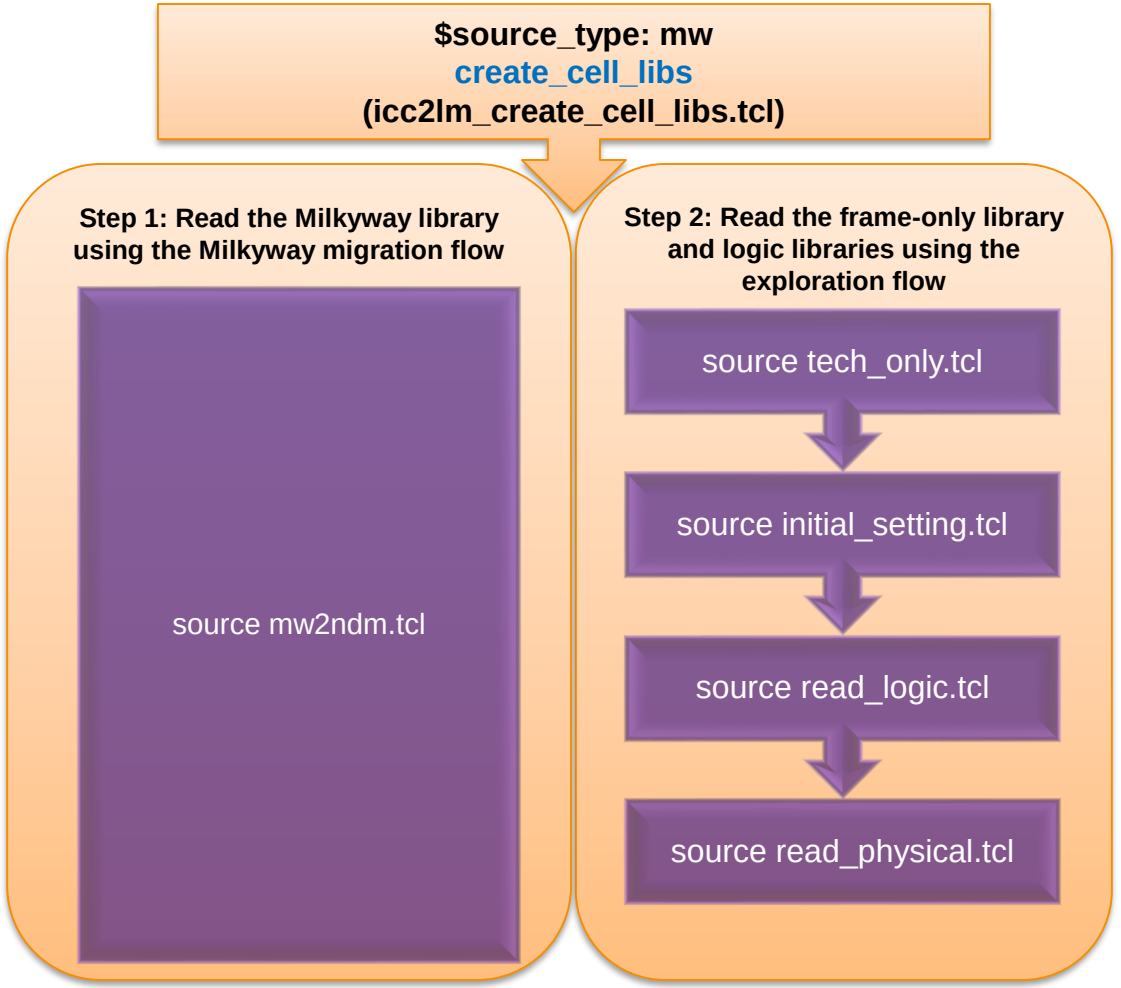


SYNOPSYS®

IC Compiler II LIBPREP-RM R-2020.09

create_cell_libs Target (Continued)

- This step creates cell libraries within a library workspace

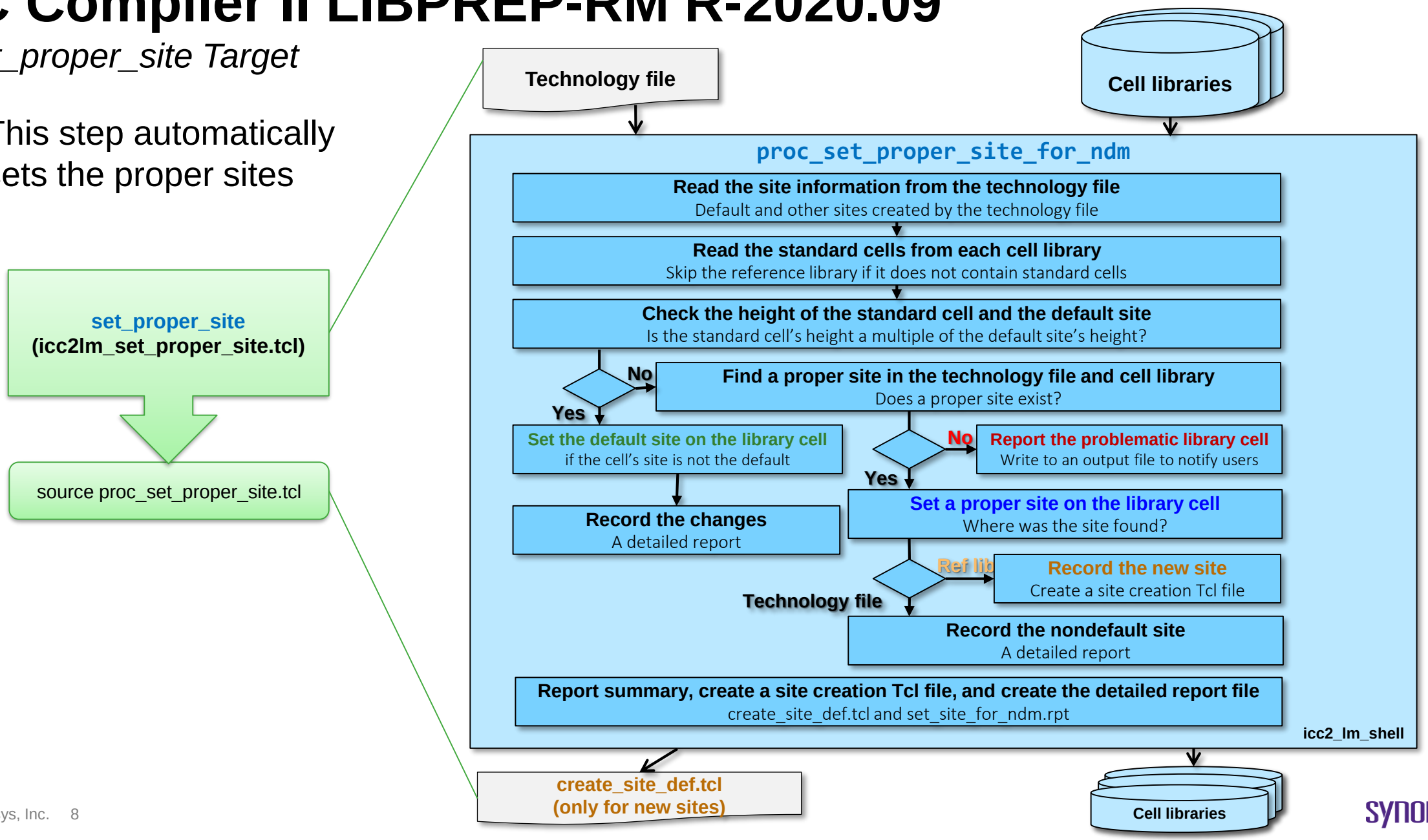


SYNOPSYS®

IC Compiler II LIBPREP-RM R-2020.09

set_proper_site Target

- This step automatically sets the proper sites

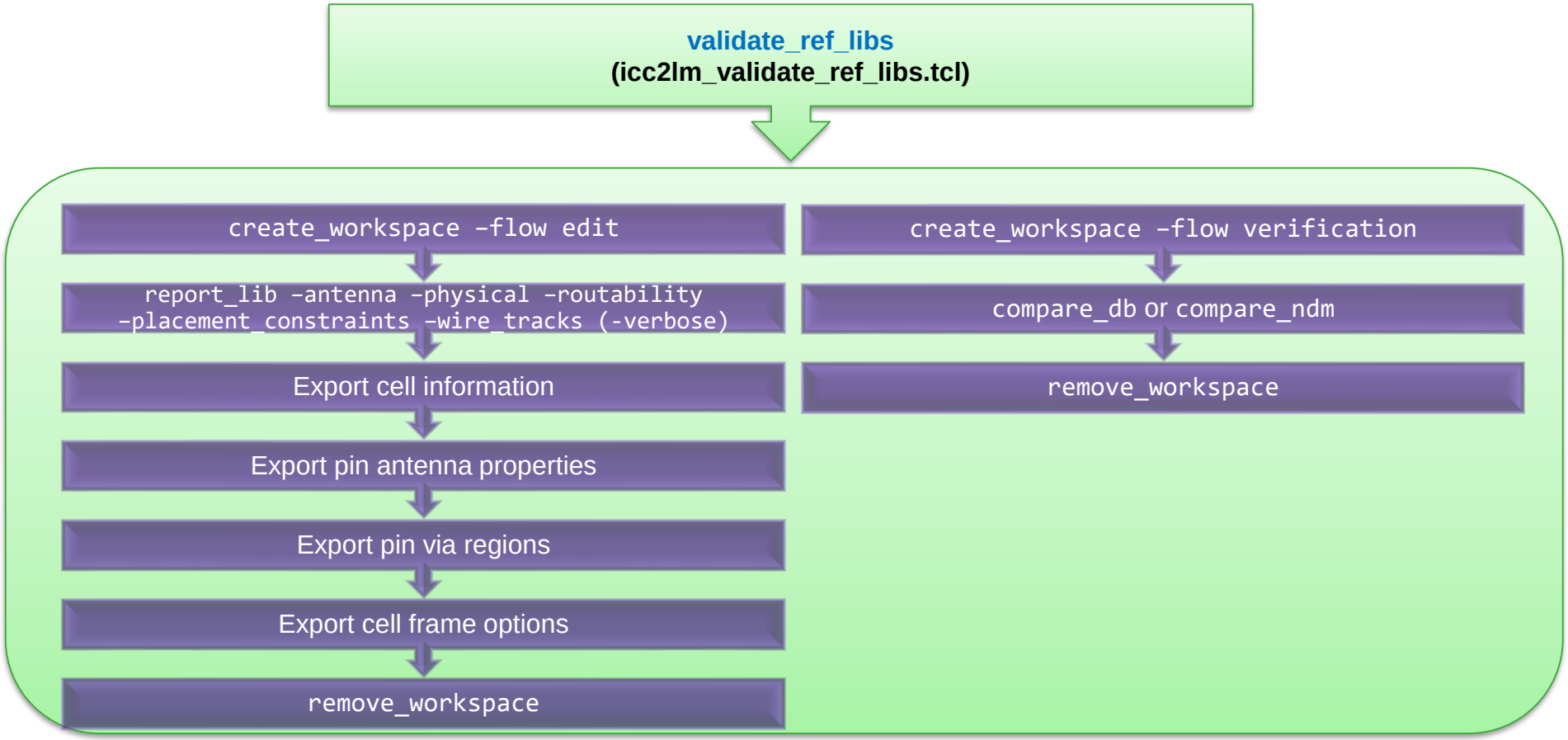


SYNOPSYS®

IC Compiler II LIBPREP-RM R-2020.09

validate_cell_libs Target

- This step validates cell libraries by using the `report_lib`, `compare_db`, and `compare_ndm` commands, as well as the manual scripts

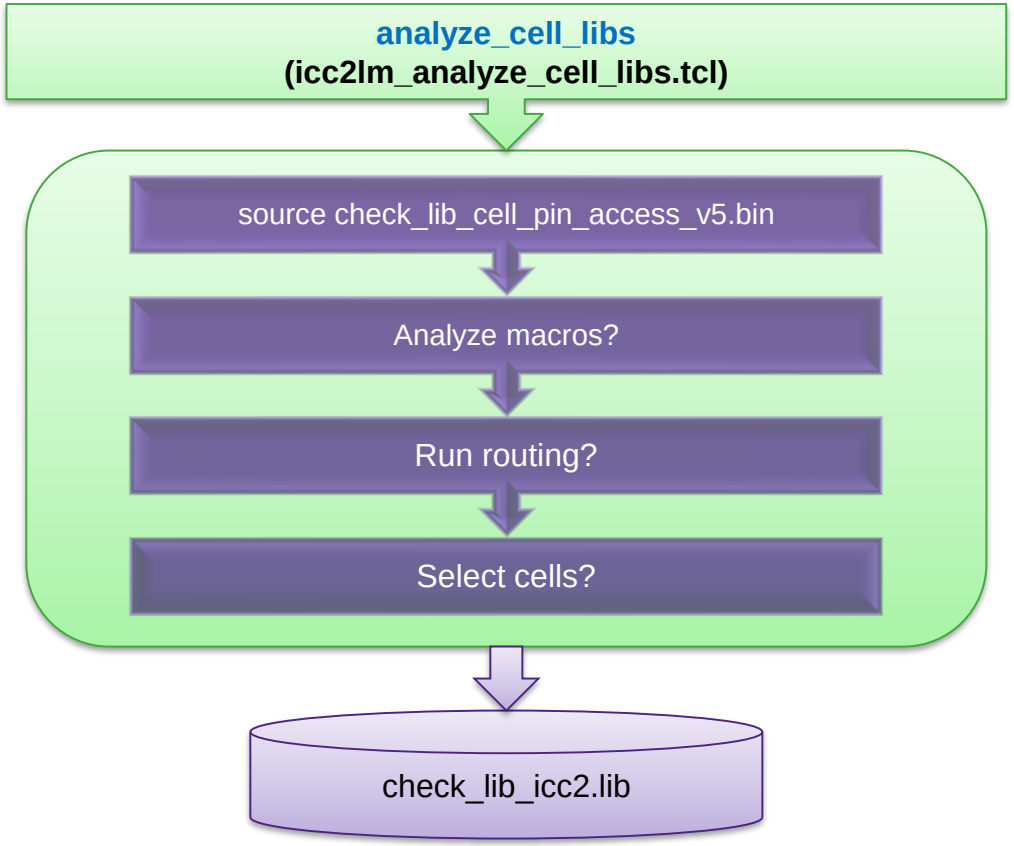


SYNOPSYS®

IC Compiler II LIBPREP-RM R-2020.09

analyze_cell_libs Target

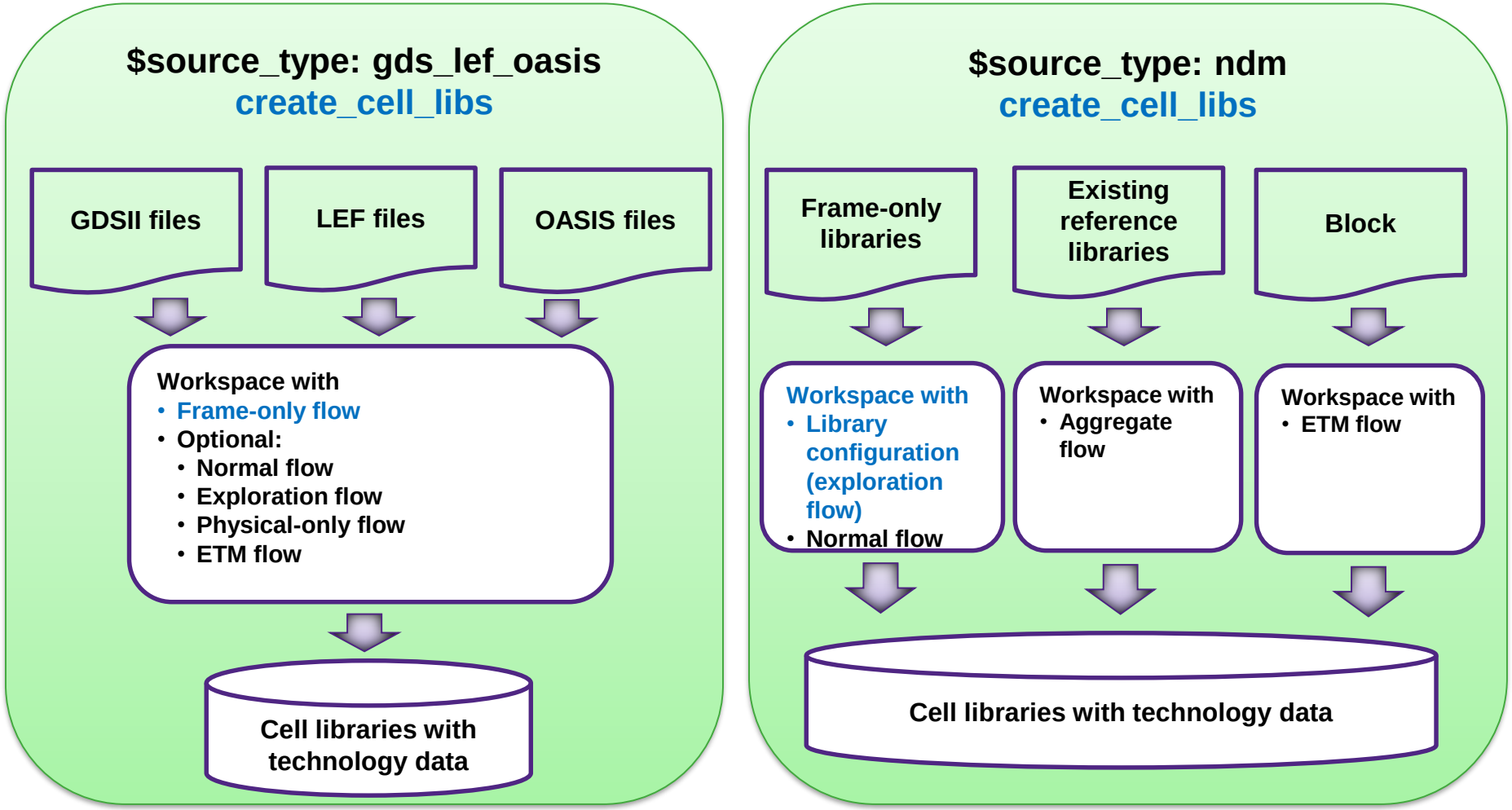
- This step analyzes reference libraries by using the pin access checking utility and creates an IC Compiler II design library, named check_lib_icc2.lib, for analysis



SYNOPSYS®

IC Compiler II LIBPREP-RM R-2020.09

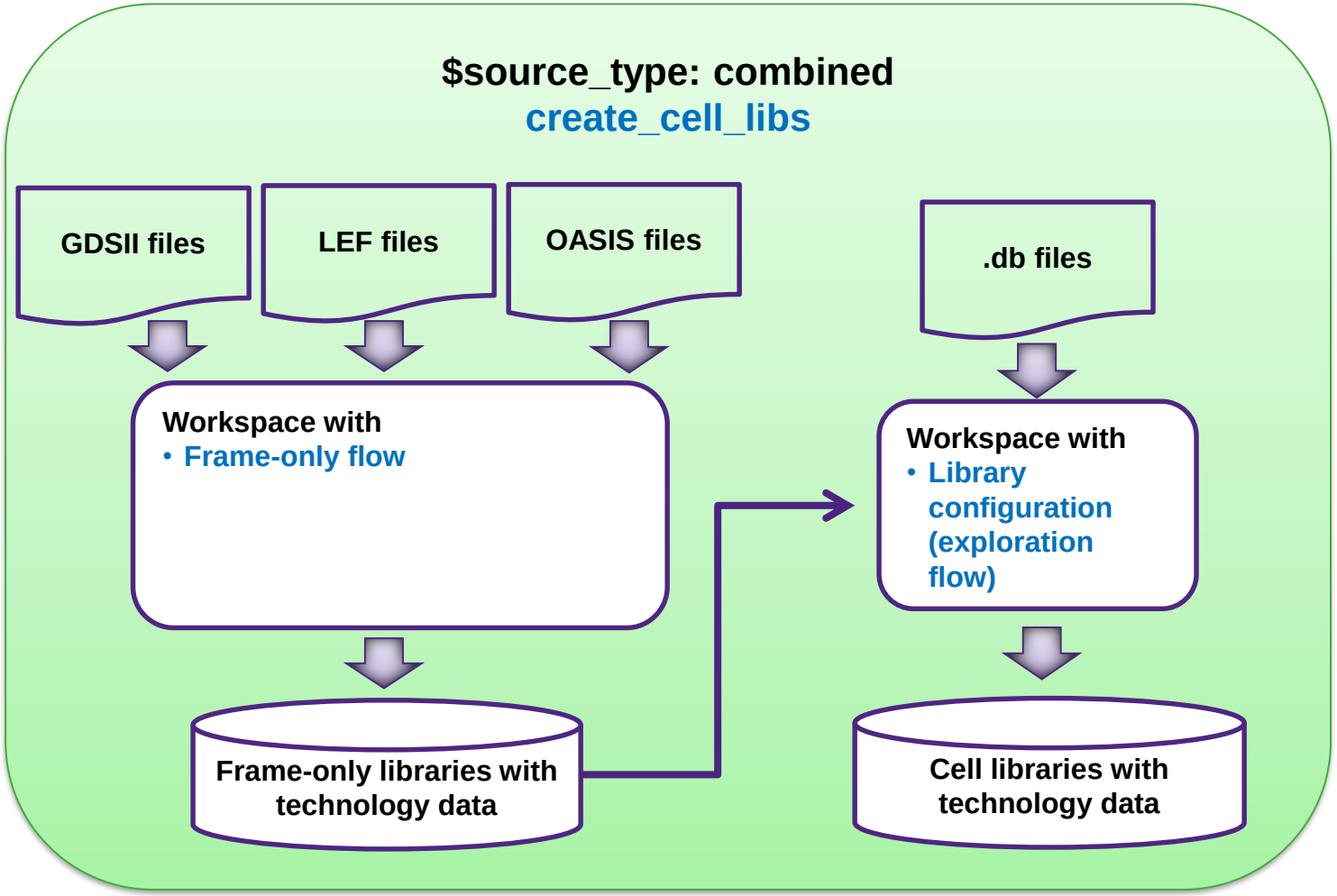
Workspace Flow Support



SYNOPSYS®

IC Compiler II LIBPREP-RM R-2020.09

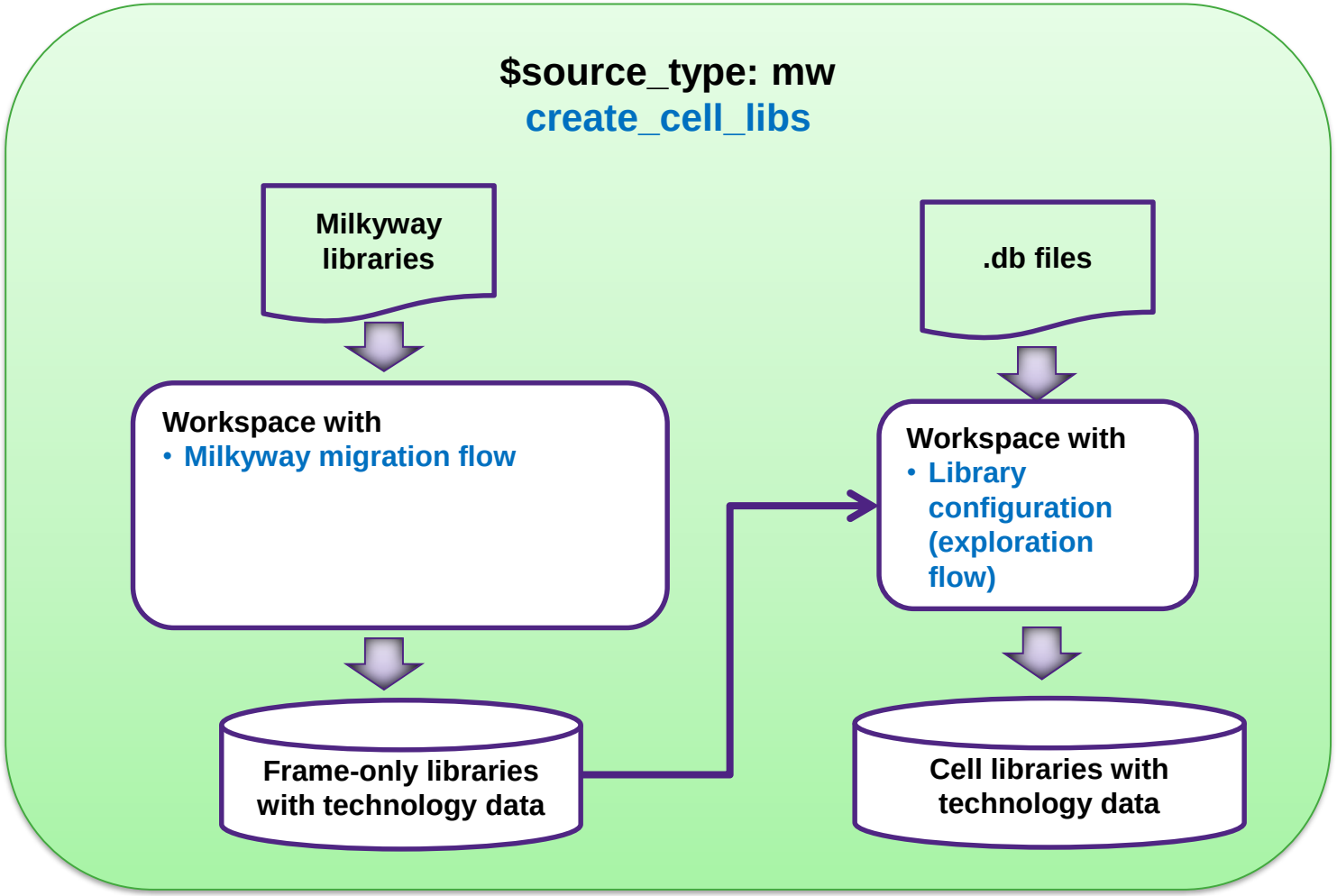
Workspace Flow Support (Continued)



SYNOPSYS®

IC Compiler II LIBPREP-RM R-2020.09

Workspace Flow Support (Continued)



SYNOPSYS®

IC Compiler II LIBPREP-RM R-2020.09

Flows Supported by $source_type: gds_lef_oasis

- `$source_type: gds_lef_oasis` supports the following library preparation flows:
  - **`normal`** flow
    Use this flow if you have a single logic library file or a group of logic library files, each of which contains the same set of cells but has timing data for a different characterization point, and physical library files that contain a superset of the cells in the logic library files. You would typically use this flow for all standard cells and pad cells.
  - **`exploration`** flow
    Use this flow to automatically analyze the library source files and generate a script that you can use to perform library preparation.
  - **`physical_only`** flow
    Use this flow to create a separate physical-only reference library for the cells that exist only in a physical library file and do not exist in any of the logic library files.
  - **`frame`** flow
    Use this flow if you want to create a file that contains only frame views (and optionally, other physical views, controlled with application options).
  - **`etm_moded`** flow
    Use this flow to build a library from extracted timing models (ETMs). Each library loaded with `read_db` requires a mode label on the command line. This flow allows you to merge multiple PVTs and modes for the same model into a single library cell.

IC Compiler II LIBPREP-RM R-2020.09

Flows Supported by $source_type: ndm

- `$source_type: ndm` supports the following library preparation flows:
  - **`normal`** flow
    Use this flow if you have a single logic library file or a group of logic library files, each of which contains the same set of cells but has timing data for a different characterization point, and physical library files that contain a superset of the cells in the logic library files. You would typically use this flow for all standard cells and pad cells.
  - **`exploration`** flow
    Use this flow to automatically analyze the library source files and to generate a script that you can use to perform library preparation.
  - **`aggregate`** flow
    Use this flow to create an aggregate library, which combines several separate reference libraries into a single reference library.
  - **`etm_moded`** flow
    Use this flow to build a library from extracted timing models (ETMs). Each library loaded with `read_db` requires a mode label on the command line. This flow allows you to merge multiple PVTs and modes for the same model into a single library cell.

IC Compiler II LIBPREP-RM R-2020.09

Flows Supported by $source_type: combined

- `$source_type: combined` supports a two-step library preparation flow:

  Step 1: **`frame`** flow
  Creates a file that contains only frame views (and optionally, other physical views, controlled with application options).

  Step 2: **`exploration`** flow
  Automatically analyzes the library source files and generates a script that you can use to perform library preparation.

SYNOPSYS®

IC Compiler II LIBPREP-RM R-2020.09

Flows Supported by $source_type: mw

- `$source_type: mw` supports a two-step library preparation flow:

  Step 1: Milkyway migration flow
  Creates a file that contains only frame views (and optionally, other physical views, controlled with application options).

  Step 2: **`exploration`** flow
  Automatically analyzes the library source files and generates a script that you can use to perform library preparation.

IC Compiler II LIBPREP-RM R-2020.09

Directory Structure

ICC2LM_LIBPREP-RM_R-2020.09.pdf

README.ICC2LM_LIBPREP-RM.txt

Release_Notes.ICC2LM_LIBPREP-RM.txt

/rm_setup

- Makefile
- icc2lm_setup.tcl
- adv_frame_setup.tcl

/rm_icc2lm_scripts

- icc2lm_create_cell_libs.tcl
  - tech_only.tcl
  - initial_setting.tcl
  - read_logic.tcl
  - read_physical.tcl
  - extract_frame.tcl
  - extract_antenna.tcl
  - diode_port.tcl
  - change_site.tcl
  - load_antenna_tcl_file.tcl
  - load_antenna_clf_file.tcl
  - mw2ndm.tcl
- icc2lm_set_proper_site.tcl
  - proc_set_proper_site.tcl
- icc2lm_validate_cell_libs.tcl
  - main_validate_cell_lib.tcl
- icc2_analyze_cell_libs.tcl
  - check_lib_cell_pin_access_v5.bin

SYNOPSYS®

IC Compiler II LIBPREP-RM R-2020.09

Setup Files

- Edit the setup files to configure the run
  - **`/rm_setup/icc2lm_setup.tcl`**
    - This is the primary configuration file
    - Use this setup file to define the variables that are used to run the flow
    - You can create several configuration files, and use them by changing the `FLOW_CONFIG` variable either in the makefile or on the make command line
  - **`/rm_setup/adv_frame_setup.tcl`**
    - Use this optional setup file to explicitly define the frame view generation options

IC Compiler II LIBPREP-RM R-2020.09

Set Up the Tool Environment

- Set up your environment to run the IC Compiler II Library Manager and IC Validator executables
  - Modify `$ICC2LMCODE` and `$ICC2CODE` in the rm_setup/Makefile file

```
ICC2LMCODE=icc2_install_dir/bin/icc2_lm_shell
ICC2CODE=icc2_install_dir/bin/icc2_shell
```

  - Modify the `PATH`, `ICV_HOME_DIR`, and `ICV_INCLUDES` environment variables in the rm_icc2lm_scripts/extract_antenna.tcl file

```
setenv PATH "icv_install_dir/bin/AMD.64:$env(PATH)"
setenv ICV_HOME_DIR "icv_install_dir"
setenv ICV_INCLUDES "icv_install_dir/include"
```


IC Compiler II LIBPREP-RM R-2020.09

Running a Target

- To run the IC Compiler II LIBPREP-RM flow, use the following command:

```
% make -f rm_setup/Makefile clean init complete
```

- If a makefile target fails, examine the log file and make the necessary changes to the configuration file. Then rerun the following command:

```
% make -f rm_setup/Makefile complete
```

- You can also run each target by using the following command:

```
% make -f rm_setup/Makefile target
```

- To rerun a specific target, delete the touchfile and then run the target
  - For example, to rerun the `validate_ref_libs` target, use the following commands:

```
% rm touchfiles/validate_cell_libs
% make -f rm_setup/Makefile validate_cell_libs
```


IC Compiler II LIBPREP-RM R-2020.09

Generated Reference Libraries and Files

- The IC Compiler II LIBPREP-RM generates the following files:
  - Reference libraries
    - In the **`$cell_lib_dir`** directory
  - Log files
    - In the **`log`** directory
  - Generated files
    - In the **`export_rm_tcl`**, **`export_creation`**, and **`export_validation`** directories
  - Touch files
    - In the **`touchfiles`** directory

IC Compiler II LIBPREP-RM R-2020.09

Useful Generated Files in the export_rm_tcl Directory

- The flow writes the following files to the **`export_rm_tcl`** directory. These files are created from the technology information and during the `create_cell_libs`, `set_proper_site`, `validate_cell_libs` and `analyze_cell_libs` steps. They can be reused in several parts of the flow and are useful for debugging.

File name	Description
/export_rm_tcl/icc2lm_create_cell_libs_export.tcl	- Generated by the `create_cell_libs` step - A Tcl script for cell library preparation
/export_rm_tcl/icc2lm_set_proper_site_export.tcl	- Generated by the `set_proper_site` step - A Tcl script to set the proper sites
/export_rm_tcl/icc2lm_validate_cell_libs_export.tcl	- Generated by the `validate_cell_libs` step - A Tcl script for cell library validation
/export_rm_tcl/icc2lm_analyze_cell_libs_export.tcl	- Generated by the `analyze_cell_libs` step - Tcl scripts for cell library analysis
/export_rm_tcl/icc2lm_tech.tcl	- Generated for wire track checking and pin access checking - Tcl scripts for a design library
/export_rm_tcl/import_icc_fram.tcl	- Generated for importing the needed data to create frame-only NDM

IC Compiler II LIBPREP-RM R-2020.09

Useful Generated Files in the export_creation Directory

- The flow writes the following files to the **`export_creation`** directory. These files are created during the `create_ref_libs` and `set_proper_site` steps. They are useful for debugging.

File name	Description
/export_creation/icc2lm_setup.rep	- Reports the configuration variable settings from icc2lm_setup.tcl
/export_creation/adv_frame_setup.rep	- Reports the configuration variable settings from adv_frame_setup.tcl
/export_creation/create_site_def.tcl	- Generated by the set_proper_site step - A Tcl script for site creation
/export_creation/${clib_name}_${ndm_flow}_report_app_options.rep	- Reports all application options

IC Compiler II LIBPREP-RM R-2020.09

Useful Generated Files in the export_validation Directory

- The flow writes the following files to the **`export_validation`** directory. These files are created during the `validate_ref_libs` step. They can be reused in several parts of the flow and are useful for debugging.

File name	Description
/export_validation/${clib_name}_${ndm_flow}_report_lib.rep	- Generated by the `validate_ref_libs ste` - Validation report generated by the `report_lib` command for the reference libraries
/export_validation/${clib_name}_${ndm_flow}_cell_lib_info.rep	- Generated by the `validate_cell_libs` step - A script-based validation report for the reference libraries
/export_validation/${clib_name}_${ndm_flow}_antenna_prop.tcl:	- Generated by the `validate_cell_libs` step - A Tcl script for pin antenna properties
/export_validation/${clib_name}_${ndm_flow}_via_region.rep	- Generated by the `validate_cell_libs` step - A pin via region report generated by the `report_via_regions` command
/export_validation/${clib_name}_${ndm_flow}_cell_frame_options.tcl	- Generated by the `validate_cell_libs` step - A Tcl script for cell frame options

IC Compiler II LIBPREP-RM R-2020.09

An Example of the Generated File

/export_rm_tcl/icc2lm_create_cell_libs_export.tcl

```
############################################################
### Synopsys(R) IC Compiler II Library Manager(TM) Library Preparation Reference Methodology
### Specify NDM flow: normal
############################################################

### Print commands from initial_setting.tcl
set sh_continue_on_error true
set_app_options -as_user_default -name design.bus_delimiters -value {<>}
…
############################################################
### Create normal flow NDM
############################################################
create_workspace -flow normal -technology ../../../INPUT/test.tf icc2lm_std_normal
### Print commands from read_logic.tcl
set_app_options -as_user_default -name lib.logic_model.require_same_opt_attrs -value false
…
```


IC Compiler II LIBPREP-RM R-2020.09

An Example of the Generated File

/export_rm_tcl/icc2lm_set_proper_site_export.tcl

```
set technology ../../../INPUT/test.tf
set ndm_list [glob /SOURCE_GDS/flow_normal/icc2_cell_lib/*.ndm]
source /flow_normal/rm_icc2lm_scripts/proc_set_proper_site_for_ndm.tcl
proc_set_proper_site_for_ndm -technology $technology -ndm_list $ndm_list
exit
…
```


IC Compiler II LIBPREP-RM R-2020.09

An Example of the Generated File

/export_rm_tcl/icc2lm_validate_cell_libs_export.tcl

```
source /rm_icc2lm_scripts/main_validate_cell_libs.tcl
set all_ndms [glob /SOURCE_GDS/flow_normal/icc2_cell_lib/*.ndm]
icc2lm_cell_libs_validation -ref_libs $all_ndms -output_dir
export_validation -detail
exit
```


IC Compiler II LIBPREP-RM R-2020.09

An Example of the Generated File

/export_rm_tcl/icc2_analyze_cell_libs_export.tcl

```
set all_ndms [glob /flow_normal/icc2_ref_lib/*.ndm]
set tech_option "-technology"
set tech ../../../INPUT/test.tf
set preplace_tech /flow_normal/export_rm_tcl/icc2lm_tech.tcl
set tcl_file /flow_normal/rm_icc2lm_scripts/check_lib_cell_pin_access_v5.bin
source $tcl_file
check_lib_cell_pin_access_icc2 -def_unit 10000 -ref_lib_path $all_ndms
$tech_option $tech -preplace_option_file $preplace_tech -
num_cells_per_parallel_run 100 -pin_access_check_tcl_path $tcl_file -
skip_routing
exit
```


IC Compiler II LIBPREP-RM R-2020.09

An Example of the Generated File

/export_validation/${clib_name}_${ndm_flow}_cell_lib_info.rep

```
The detailed antenna report: icc2lm_std_normal_antenna_prop.rep
The detailed via region report: icc2lm_std_normal_via_region.rep
+---------------------------------+-------------------------------------------------------------------------------------------------+
|                    Library Name | icc2lm_std_normal                                                                               |
+---------------------------------+-------------------------------------------------------------------------------------------------+
|                    Default Site | unit (W:0.2000 H:1.8000 Symmetry: Type:core)                                                    |
+---------------------------------+-------------------------------------------------------------------------------------------------+
|                Non-default Site |                                                                                                 |
+---------------------------------+-------------------------------------------------------------------------------------------------+
|               Routing Direction | ME1 -- ME2 -- ME3 -- ME4 -- ME5 -- ME6 -- ME7 -- ME8 -- ME9 -- ME10 --                          |
+---------------------------------+-------------------------------------------------------------------------------------------------+
|                    Track Offset | ME1 -- ME2 -- ME3 -- ME4 -- ME5 -- ME6 -- ME7 -- ME8 -- ME9 -- ME10 --                          |
+------------------+--------------+-----------+--------+-----------+-----+---------------+-----------+--------------+------------+
|        Cell Name | Boundary Size | Cell Site | Height | Cell Type | Pin | Pin Direction | Port Type | Antenna Prop | Via-Region |
+------------------+--------------+-----------+--------+-----------+-----+---------------+-----------+--------------+------------+
|    CELL1.frame   |   W:0.8 H:1.8 |      unit |    1xh |  lib_cell |   A |            in |    signal |          Yes |          6 |
|    CELL1.frame   |   W:0.8 H:1.8 |      unit |    1xh |  lib_cell |   Z |           out |    signal |          Yes |          3 |
|    CELL1.frame   |   W:0.8 H:1.8 |      unit |    1xh |  lib_cell | VDD |       unknown |     power |          Yes |         -- |
|    CELL1.frame   |   W:0.8 H:1.8 |      unit |    1xh |  lib_cell | VSS |       unknown |    ground |          Yes |         -- |
+------------------+--------------+-----------+--------+-----------+-----+---------------+-----------+--------------+------------+
```


IC Compiler II LIBPREP-RM R-2020.09

An Example of the Generated File

/export_validation/${clib_name}_${ndm_flow}_report_lib.rep

```
#BEGIN_REPORT_PHYSICAL
...
Cell              Views                              Cell Type       Allowable Orientation                    Mask Shiftable       Site         Height
Boundary                                                             Cell Area(um^2)
-------------------------------------------------------------------------------------------------------------------------------------------------------
-
CELL1             timing design layout frame         lib_cell        R0 R90 R180 R270 MX MXR90 MY MYR90       true                 unit         1.8u
{0.0000 0.0000} {0.0000 1.8000} {1.4000 1.8000} {1.4000 0.0000}      2.5200
...
Cell Name: AOI22M1N.frame
-------------------------------------------------------------------------------------------------------
Pin Name        Direction        Port Type        PG Type        Bias Type        Secondary PG        Diode
-------------------------------------------------------------------------------------------------------
A1              in               signal           --             --               false               false
...
#END_REPORT_PHYSICAL
#BEGIN_REPORT_ROUTABILITY
Routability report:
Warning: List of pins without optimal routability. (FRAM-014)
-------------------------------------------------------------------------
  Cell Name        Design Type   Missing Property   Pin Name(s)
-------------------------------------------------------------------------
  CELL1            lib_cell      via_region         ENL

Total number of pins without optimal routability: 2 (out of 67)

#END_REPORT_ROUTABILITY
#BEGIN_REPORT_PLACEMENT_CONSTRAINTS
Placement constraint report:
Total number of cells without placement layers and constraints: 0 (out of 18)
#END_REPORT_PLACEMENT_CONSTRAINTS
```


IC Compiler II LIBPREP-RM R-2020.09

An Example of the Generated File

/export_validation/${clib_name}_${ndm_flow}_antenna_prop.tcl

```
set _library [get_libs]
set library [get_object_name $_library]
if {[string match {} [current_lib -quiet]]} {
set view design
} else {
set view frame
}

set lc [get_lib_cells -quiet ${library}/CELL1/${view}]
set lp [get_lib_pins -quiet -all -of_objects $lc -filter "name == CK"]
    set_attribute -objects $lp -name diff_area -value {ME1 0 ME2 0 ME3 0 ME4 0 ME5 0 ME6 0 ME7 0 ME8 0 ME9 0 ME10 0}
    set_attribute -objects $lp -name gate_area -value {oxide1 ME1 0.1242 oxide1 ME2 0.1242 oxide1 ME3 0.1242 oxide1 ME4
0.1242 oxide1 ME5 0.1242 oxide1 ME6 0.1242 oxide1 ME7 0.1242 oxide1 ME8 0.1242 oxide1 ME9 0.1242 oxide1 ME10 0.1242}
    set_attribute -objects $lp -name mode1_area -value {oxide1 ME1 0.2466 oxide1 ME2 0 oxide1 ME3 0 oxide1 ME4 0 oxide1
ME5 0 oxide1 ME6 0 oxide1 ME7 0 oxide1 ME8 0 oxide1 ME9 0 oxide1 ME10 0}
    set_attribute -objects $lp -name mode2_ratio -value {oxide1 ME1 0 oxide1 ME2 0 oxide1 ME3 0 oxide1 ME4 0 oxide1 ME5 0
oxide1 ME6 0 oxide1 ME7 0 oxide1 ME8 0 oxide1 ME9 0 oxide1 ME10 0}
    set_attribute -objects $lp -name mode3_area -value {oxide1 ME1 0.2466 oxide1 ME2 0.2466 oxide1 ME3 0.2466 oxide1 ME4
0.2466 oxide1 ME5 0.2466 oxide1 ME6 0.2466 oxide1 ME7 0.2466 oxide1 ME8 0.2466 oxide1 ME9 0.2466 oxide1 ME10 0.2466}
    set_attribute -objects $lp -name mode4_area -value {oxide1 ME1 0.2466 oxide1 ME2 0 oxide1 ME3 0 oxide1 ME4 0 oxide1
ME5 0 oxide1 ME6 0 oxide1 ME7 0 oxide1 ME8 0 oxide1 ME9 0 oxide1 ME10 0}
    set_attribute -objects $lp -name mode5_ratio -value {oxide1 ME1 0 oxide1 ME2 0 oxide1 ME3 0 oxide1 ME4 0 oxide1 ME5 0
oxide1 ME6 0 oxide1 ME7 0 oxide1 ME8 0 oxide1 ME9 0 oxide1 ME10 0}
    set_attribute -objects $lp -name mode6_area -value {oxide1 ME1 0.2466 oxide1 ME2 0.2466 oxide1 ME3 0.2466 oxide1 ME4
0.2466 oxide1 ME5 0.2466 oxide1 ME6 0.2466 oxide1 ME7 0.2466 oxide1 ME8 0.2466 oxide1 ME9 0.2466 oxide1 ME10 0.2466}
```


IC Compiler II LIBPREP-RM R-2020.09

An Example of the Generated File

/export_validation/${clib_name}_${ndm_flow}_via_region.rep

```
****************************************
Report : Report via regions
Design : CELL1
Version: K-2015.06
Date   : Thu Jun  4 04:21:09 2015
****************************************
ViaRegion
         Owner       ViaDef    Type    Rect Rotate
--------------------------------------------------------------------------------
A1/VR_0  A1          VI1_1_HV  System  Yes  --
A1/VR_1  A1          VI1_1_HV  System  Yes  Yes
A1/VR_2  A1          VI1_2_VV  System  Yes  --
A1/VR_3  A1          VI1_2_VV  System  Yes  Yes
****************************************
Report : Report via regions
Design : CELL2
Version: K-2015.06
Date   : Thu Jun  4 04:21:09 2015
****************************************
ViaRegion
        Owner       ViaDef    Type    Rect Rotate
--------------------------------------------------------------------------------
A/VR_0  A           VI1_1_HV  System  Yes  --
A/VR_1  A           VI1_1_HV  System  Yes  Yes
A/VR_2  A           VI1_2_VV  System  Yes  --
```


SYNOPSYS®

IC Compiler II LIBPREP-RM R-2020.09

An Example of the Generated File

/export_validation/${clib_name}_${ndm_flow}_cell_frame_options.rep

```
# Frame version: K-2015.06-SP5-BETA
set_app_options -block icc2lm_std_normal/CELL1/design -name lib.physical_model.block_all -value {auto}
set_app_options -block icc2lm_std_normal/CELL1/design -name lib.physical_model.connect_within_pin -value
{true}
set_app_options -block icc2lm_std_normal/CELL1/design -name lib.physical_model.hierarchical -value {true}
set_app_options -block icc2lm_std_normal/CELL1/design -name lib.physical_model.merge_metal_blockage -value
{false}
set_app_options -block icc2lm_std_normal/CELL1/design -name lib.physical_model.preserve_metal_blockage -
value {auto}
# Frame version: K-2015.06-SP5-BETA
set_app_options -block icc2lm_std_normal/CELL2/design -name lib.physical_model.block_all -value {auto}
set_app_options -block icc2lm_std_normal/CELL2/design -name lib.physical_model.connect_within_pin -value
{true}
set_app_options -block icc2lm_std_normal/CELL2/design -name lib.physical_model.hierarchical -value {true}
set_app_options -block icc2lm_std_normal/CELL2/design -name lib.physical_model.merge_metal_blockage -value
{false}
set_app_options -block icc2lm_std_normal/CELL2/design -name lib.physical_model.preserve_metal_blockage -
value {auto}
# Disable next line manually if no need to re-create frame view.
set_attribute -objects [get_lib_cells icc2lm_std_normal/CELL1/design] -name frame_update -value true
set_attribute -objects [get_lib_cells icc2lm_std_normal/CELL2/design] -name frame_update -value true
```



SYNOPSYS®
Silicon to Software™