



Open Design Constraints

19th Synopsys EDA Interoperability Developers' Forum

What Are Constraints?

- ⊕ Viewed from the ground:
 - Constraints describe designer intent to EDA tools

- ⊕ Viewed from 30,000 feet:
 - Constraints describe HOW the designer wants the tools to implement his design
 - Constraints describe designer intent NOT described by
 - Circuit topology (schematics)
 - Device sizing
 - Constraints pass intent from one level of abstraction to the next

- ⊕ Constraints are as important as design data
 - It is difficult to generate credible physical implementations from schematics without user defined physical constraints

Constraint Producers and Consumers

- ⊕ Many roles touch constraints
 - Circuit designers
 - Layout engineers
 - Process engineers – mostly global constraints
- ⊕ Tools may consume and produce constraints
 - Placers
 - Routers
 - DFM systems
 - Physical analysis tools (parasitics, thermal, etc.)

Sample Analog Layout Constraints

⊕ Object Constraints

- Alignment & orientation
- Symmetry
- Pair layouts
- Shielding
- Boundary area
- Number of contacts
- Dummies
- Clustering

⊕ Routing Constraints

- Signal Type
- Net (Pin) Grouping/ Ordering
- Min/ Max Wire Length
- Matched/balanced routing
- Equal length routing
- Area/layer blockages & halos
- IR drop threshold
- Current density limit
- RC limit
- Sensitive net specifications
- Width of Vdd and Vss

⊕ And many, many more...

What Ciranova is Doing

- ❖ Ciranova offers solutions for automation of analog physical design in OpenAccess
 - Next generation Parameterized Cell development & deployment
 - Device placement optimized for area
 - Device placement optimized for routing
- ❖ Ciranova tools work as components in established design cockpits & flows that support OpenAccess
 - We believe and practice component based software architecture
- ❖ In the Market Place:
 - PyCell Studio is available to all for free
 - Our technologies conform to OpenAccess & other open standards
 - Integrate into existing flows

Ciranova's Strategy for Constraints

- ❖ Text based format for representing analog physical design constraints
- ❖ Constraint entry GUI paradigm can be specific to a design cockpit
 - Schematic based
 - Spreadsheet based
- ❖ Text representation readable by humans and tools

Fundamental Principals of Open Constraints

- ⊕ Constraints are Customer IP
- ⊕ Constraints should be open to all tools from all vendors
- ⊕ Constraints should be supported within the OpenAccess data model
- ⊕ Textual representation for constraints external to OpenAccess is equally important
 - Can be compiled into OA constraints as needed

It's Time to Collaborate on Constraints!

- ❖ Currently, OA constraints are evolving around a limited set of tools
- ❖ If we don't collaborate, the result will be
 - Closed constraints
 - Electrical & physical design tools from different vendors won't interoperate in one flow
 - Broken OpenAccess flows that cannot share design intent between multiple vendors' tools
 - Same old... same old...
- ❖ If the goal is an open, vendor-neutral constraint standard, then we need:
 - Collaboration between vendors committed to open standards
 - Input from users with constraint-driven flows
 - A process for integrating with OpenAccess