Complementing EDA with Meta-Modelling & Code Generation

Wolfgang Ecker, Michael Velten, Leily Zafari, Ajay Goyal

**Motivation**
- EDA tools normally cater to a large number of customers and create generic tools to improve design flows through automation.
- Development cost of EDA tools is very high.
- EDA cannot focus on domain-specific tools as design companies cannot afford to pay for such tools.
- Automation is the key to make designers more efficient so that such tools cannot be ignored.
- Designer normally does not have expertise to build such tools.

**Levels of Abstraction**
- Meta-Meta-Model
  - Defines Structure of Meta-Model
  - Generates structural meta-model
  - Generates executable meta-model or access model
- Meta-Model
  - Defines Structure of Model
  - Generates executable meta-model or access model
- Model
  - Defines Structure of Data independent of the view format
  - Generates view or access model (where necessary)
- View
  - Implementation of content

**Infrastructure & Features**
- Reusability
  - UML, Timing, power, etc., are integrated in the infrastructure.
- Extensibility
  - Model can be easily extended by extension mechanism.
- Linkability
  - Multiple models can be linked to create complex data structures.
- Transferability
  - Conceptual models can be easily exchanged and understood between designers as data structures have a visual look.

**Flexibility**
- Same model used for multiple input & output formats.

**Features**
- Reusability
- Extensibility
- Linkability
- Transferability

**Application & Benefits**
- IO Pad Area Estimator
- IO Lib Info Collector

**Benefits**
- Productivity Improvement
- Quality Improvement
- Consistency Improvement
- Easier to incorporate late specification changes

**Challenges of Technology**
- Learning curve for designers to learn the modeling infrastructure.
- Designer must know his/her job very well in order to automate the same.
- Designer must understand what can be automated in design flow so that he can re-shape his flow towards automation.
- Takes sometime for designer to learn the infrastructure and integrate it in his flow. Management needs to understand and support.
- The current framework is generated in Python. There are constant language wars. Perl, Python, Tcl etc.
- The automatic task in the design flow such that the designer becomes more interested in code generation than designing.

**Conclusion**
- This technology is not to create already existing EDA tools but to complement the same.
- It is mainly used to create domain-specific tools and to help increase efficiency of using EDA tools.
- It provides a mechanism to extend EDA and helps making designers more efficient.
- It provides from Meta-Modelling features to assure flexibility, extensibility, and reusability of the design flow.
- Similar to any other technology, the designer should invest some time to understand this technology before he can use it productively.