Release Options

Goals:

- model code that runs on any commercial simulator
- flexible standard interface to ALL simulators (EDA industry failed to agree on this!!)
- automated generation of reliable, complete, fast code ("model compiler")

Option 1

- Verilog-A [MAST, ?...) ==> advantages see separates slides/presentations

Note

- If the amount of effort spent just by model developers in the past 10 years on
  - model implementation,
  - learning simulator interfaces
  - fixing coding related bug and providing related support

would have been spent on developing a model compiler ...

=> we would not have this discussion anymore!!!

Release Options (cont'd)

Alternative: Model fully implemented in a circuit simulator (plus code)

- model developer (MD) owned simulator
  - **con1**: MD is not in simulator development business (which foundries/design houses left 10...15 years ago)
  - **con2**: proprietary issues
  - **con3**: support issues (different OSs, usage)
  - ***note***: commercial simulator environment is not suitable for model development

- commercial simulator such as ADS, ELDO, SPECTRE ...
  - **con1**: need to learn and deal with interface that is inconvenient for model development (bad experience at Nortel, Rockwell, ?)
  - **con2**: code and simulator are proprietary!
  - **con3**: users and EDA vendors need license for model evaluation and testing

Release Option alternatives (cont'd)

- free simulator (such as SPICE2, SPICE3, ... ?)
  - **con1**: need to learn and deal with interface (still not suitable for model development)
  - **con2**: still support issues (different OSs, usage)

Conclusion in all cases

- implementation effort on model developer side for providing ***complete*** and ***reliable*** test version is equivalent to effort at EDA vendor
- commercially acceptable model implementation is not research (work will not be accepted for graduation)
- do companies have the resources to do this work?
- additional effort to MD's own development environment

=> why would a post-doc do that work for 50% of industrial salary?

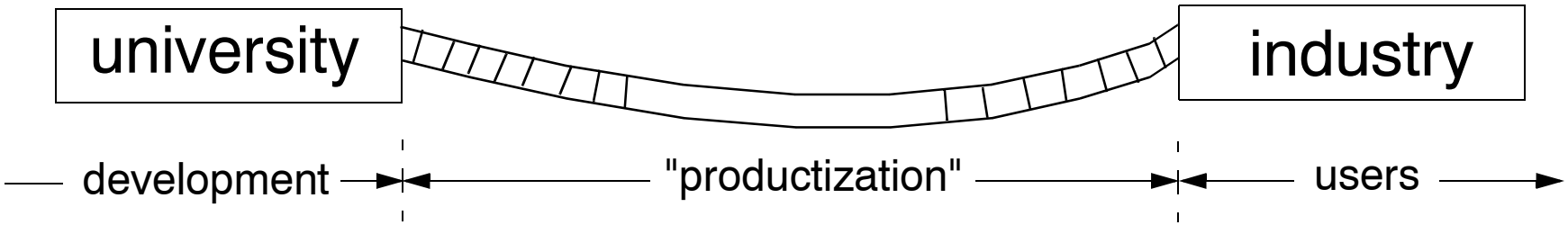
Note

derivatives are not needed for model development and solution of model equations!

Model development at universities ...

... without commercial funding (which is generally the case)

- work *must*
  - have sufficient ***research*** (theory and experiment) contents to qualify for theses/graduation
  - **not be** just a **service** competing with existing companies (to avoid law suits on cost dumping)
- model implementation
  - can only be for demonstration purposes
  - *must not require* significant manpower (examples: GUI, coding in *several* simulators, complete derivatives ...)
  - cannot use cost- or manpower-intensive EDA tools
- model release *must not require*
  - significant maintenance and user support (incl. legacy issues, extensive versioning effort ...)
  - to enter or violate legal obligations (such as responsibility for functionality other than developed case, IP disclosure, university regulations ...)



Model release format in Verilog-A

(summarized from various discussions)

- Question: Why release compact models in Verilog-A (V-A) code only ?

Answer: (i) V-A is a commonly defined simulator independent standard that allows potential model users to immediately evaluate a model in their mainstream circuit simulators for their process technologies;
(ii) V-A model compilers are now available that translate V-A code into C code that can the be linked to the simulator.
(iii) test data can be generated flexibly at each simulator vendor

- Q: Isn't Verilog-A code too slow for circuit simulation ?

A: (i) Yes, but the V-A code is only for model evaluation purposes, and not for circuit simulation and (production) parameter extraction;
(ii) Compiling the Verilog code and linking the resulting executable to the simulator will yield fast model execution.

Model release format (cont'd)

- Q: Q: As EDA vendor, I want to minimize the implementation effort and need SPICE compatible code C code!

A: All simulators have different model interfaces and data structures. Hence, a model developer can only satisfy a SINGLE simulator vendor with the model code format. However, a V-A compiler can generate code directly for each simulator (if the EDA vendor funds the development of the related compiler option.

- Q: Compiler generated V-A code is not readable, while as EDA vendor, I need read able code to be able to optimize it for performance!

A: (i) Define readable code! There is no guarantee for readable hand-written code.
(ii) Compilers can be enhanced to produce "readable" code and partially already offer some features in this direction (see ADMS's color coding of variables).
(iii) Compilers can also be enhanced to produce optimized code, which is a ***one-time*** effort compared to performing this work for every model release again.

Note: model derivatives depend on state variables which can vary depending on simulator implementation

Model release format (cont'd)

- Q: For model implementation (simulator, parameter extraction tool), I need code with ***derivatives***! Why can't the model developer provide the derivatives ?

A: (i) Calculating the derivatives is ***not required*** for (physics-based) model development and only an additional burden that slows down development.

(ii) If a model developer would use a Math tool to calculate the derivatives, he would have to ***maintain an additional code version*** and to still transfer that tool's results into C code. A model compiler does the same but does not require the additional code (which is also not necessary for model development).

(iii) Model compilers generate ***complete*** derivatives. This technique is superior to hand-calculated and -coded derivatives from the developer which have proved to be unreliable in the past.

Note: EDA industry is grossly underfunding model productization (and development)!

Model release format (cont'd)

- Q: For model implementation, are still test data required ?

  A: Ultimately, only a netlist and test model parameters should be required but not the often vast amounts of test data, since each model "implementor" can now use the Verilog-A model code to generate test data for any test set up.

- Q: Once a compiled model has been created and linked to a simulator, can that executable (of the compiled model) be distributed to model users (employing of course the same simulator) ?

  A: This seems to have to be answered on a case by case basis;
  - Cadence requires to sign an NDA for the CMI => no distribution allowed
  - ADS with Tiburon compiler: distribution seems possible
  - ELDO ? others ?
  - Is this also dependent on the compiler used (ADMS, Tiburon) ?

Model release format (cont'd)

- Q: VA compiler generated executable code is slower than executables created from hand-optimized code ?

  A: (i) The difference depends on the model and seems to be 10...20%.
  (ii) While this may be the case for the first version of existing VA compilers it is expected (by compiler developers) to be significantly reduced in upcoming versions.
  (iii) Optimization (also of "hand-written") code is simulator specific - as has been frequently pointed out by EDA vendors. Therefore, it is the EDA vendor's and not the model developer's task.

Summary:

model release in Verilog-A solves long-standing problems

=> extremely attractive approach for providing on-demand custom models