

Standard Modeling Languages

- **C code for compact models**
 - is like doing graphics in HPGL (or on graph paper)
 - is like writing documents in PostScript
 - is like using a hammer instead of a nail gun
- **Why is Matlab so popular?**
 - Allows engineers to write at a high level of abstraction, and not get bogged down
 - Enables efficiency and fewer errors
 - Lets engineers concentrate on the real problems and not details of solution

Standard Modeling Languages

- C code is not standard for every simulator, but Verilog-A is!
- Capability to re-use blocks is exponentially enabled by Verilog-A
- Allows arbitrary generation of “golden” results by running directly
 - this is a huge advance over publishing limited test results with a model
 - noise, transient, HB, ... not provided
- **The only way to go**

Impediments

- **Fracturing model developers by anointing a plethora of competing languages as standard**
 - we need to all use only one, Verilog-A
- **Barrier to introduction to the compact model development club is removed**
 - still need a “bozo” filter ...
- **Anarchy: gives modeling groups the ability to deliver equations, not just model parameters**

Why not used now?

- **Infrastructure for compact modeling in high level languages is immature**
- **The light bulb is only just going off for many people**
 - **CS knowledge varies widely in EEs**
 - **lots of misconceptions and lack of understanding of basic concepts**
- **It takes years for changes like moving from C to Verilog-A to happen**

My Experience

- I personally find that once I get used to a language I do not like to change
- I have personally defined or helped define several versions of languages for compact model definition
- After seeing Verilog-A, I completely switched to that and have no desire to use anything else