



# AMI BIRD Feedback

**March 2010**

Ken Willis, Kumar Keshavan  
Sigrity



# Summary of Proposed Changes

- Change Reference Flows
- Remove Branches
  - Reserved\_Parameters
  - Model\_Specific
- Remove Reserved Parameters
  - Tx\_Jitter
  - Rx\_Clock\_PDF
- Add Reserved Parameters
  - Init\_Returns\_Filter
- Remove Keywords
  - Format
  - Gaussian
  - Table
  - DjRj
  - Dual-Dirac
- Add Keywords
  - Array



# Change Reference Flows

- AMI\_GetWave flow is clarified for Tx usage, which is good
- Proposed to add “Init\_Returns\_Filter”
- Could be problematic:
  - For pure LTI systems, you can already do time domain or statistical analysis, using the existing “Init\_Returns\_Impulse” API
    - De-convolution is the tool’s challenge, not the model’s
    - Should put minimum burden on the model / model developer
  - If using “AMI\_GetWave” you **cannot** guarantee LTI, and time domain results should be considered “golden” vs. statistical results
  - So if both AMI\_GetWave and Init\_Returns\_Filter existed, different results could be produced from the **same** model!
- This appears to add no new value but could add confusion very quickly
- Model itself could always be made to output a particular filter description, even with today’s specification
- In favor of adding documentation on AMI\_GetWave for Tx
- Since “Init\_Returns\_Filter” would impact reference flow with new functionality, should be handled in its own BIRD



# Remove Branches

- Proposal to remove (or make optional) existing branches
  - Reserved Parameters
  - Model Specific
- Position:
  - We feel this was well thought out in 5.0 and provides clear delineation between EDA tool (Reserved Parameters) and model (Model Specific)
  - Also ruins backward compatibility of existing models
  - Not in favor of this change



# Remove Reserved Parameters

- Proposal to remove some Reserved Parameters
  - Tx Jitter
  - Rx Clock PDF
- In favor of removing these parameters to reduce complexity, unless people using these already



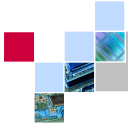
## Add Reserved Parameters

- See earlier slide on “Change Reference Flows”



# Remove Keywords

- Proposal to remove Keywords:
  - Format
  - Gaussian
  - Table
  - DjRj
  - Dual-Dirac
- In agreement with all of above with exception of “Table”
- Sigrity has found this useful and used this syntax successfully in many models, ex:
  - (Iffe (Usage In) (Type Float) (Format Table (0.25 1.0 0.5)) (Description " The normalized tap limits."))
- Would violate backwards compatibility
- In favor of removing all but “Table”, unless people already using these



# Add Keywords

- Proposal to add new keywords “Array”, “Step”, “Increment”
- Array
  - Is this only for handling the type “Tap”? Should it be a sub-parameter?
  - We handle tap coefficients with today’s spec
  - What unique capability or significant advantage does a new “Array” keyword bring?
  - Would like to see a differentiating example, where something new is achieved
- Step and Increment
  - Need to see real-world example of how these add value over and above current capability





## Recommendations

- Focus initially on a simple “clarification” AMI BIRD, with no new syntax introduced
- Make updates to documentation portions
- Make minor updates to existing syntax if needed and justified (ex. get rid of “Format”)
- Once this is finished and a stable baseline is set, consider introduction of new syntax
- Adopt “MatLab” philosophy of few / general / powerful data types, to avoid chasing constant syntax updates



## Recommendations (cont.)

- Test any new syntax proposals with some standard criteria:
  - Does this allow us to do something that can't be done today?
  - Is there some major advantage to the modeler (not tool) in introducing this?
- For any additions to “Reserved Parameters” should require all EDA vendors to produce same results on same testcase
  - This was the approach in original IBIS spec for VI table handling



*Thank You!*

