[redacted]

[redacted]

Emulation User Problems to be Solved by SCE-API

- All emulators on the market today have proprietary API’s.
  - Restricts the availability of emulation solutions to users.
  - Leads to low productivity and low ROI for emulation users who build their own solutions.
- The emulation ‘API’s’ which exist today are oriented to gate-level and not system-level verification.
- Users need an API which takes full advantage of emulation performance.

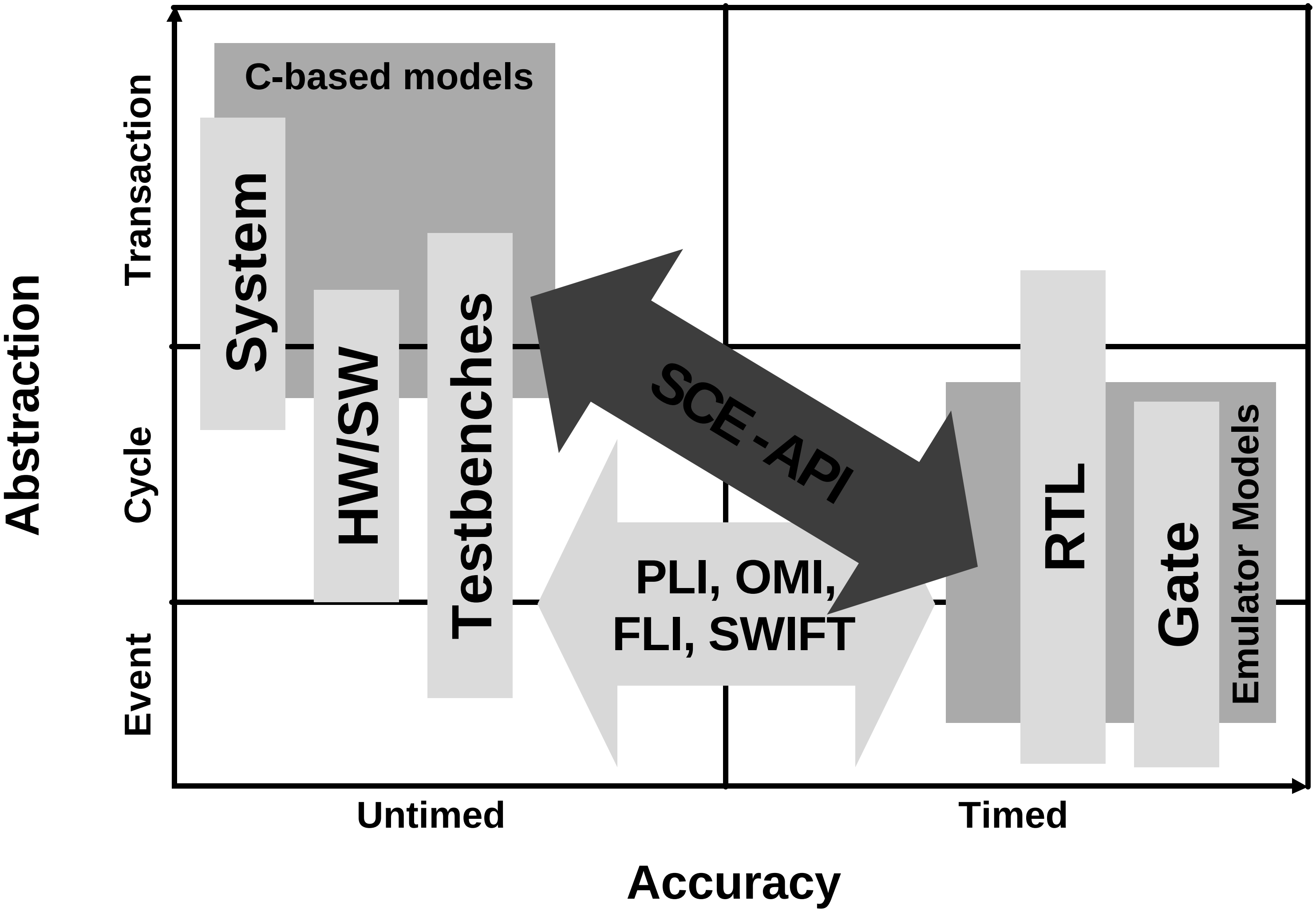
Emulation Supplier Problems to be Solved by SCE-API

- Users are reluctant to invest in building applications on proprietary API’s.
- Traditional simulator API’s like PLI and VHPI slow down emulators.
- Third parties are reluctant to invest in building applications on proprietary API’s.

SCE-API History

- June 2000, first meeting, at DAC in Los Angeles
- July 2000, Steering and Technical Groups established
- November 2000, first specification draft from Technical Group
- February 2001, public announcement of SCE-API
- April 2001, posting of SCE-API 1.0 on SystemC open source web site
- May 2001, proposal to become an Accellera working group accepted

Where SCE-API Fits in the Modeling World



SCE-API Consortium, 2001

SCE-MI

- A message passing interface
  - Designed with system level communication in mind
    - C/System Design vs HDL
    - **System Transactions** vs Pin Events
  - Wide
  - Simple terminals
  - Multiple channels
  - Designed for low latency and high bandwidth
  - Up to full emulation speeds (1MHz+)
- Based on IKOS ‘Co-Modeling’ technology

SCE-MI

- Bridges high-abstraction models to models with implementation detail
  - ‘Untimed’ to ‘Timed’ bridging
- Reduces communications overhead between models
  - Optimized for system-level transactions
  - Allows increased performance up to full emulation speeds

Applications of SCE-MI

- Software model to emulator or simulator interface
- Software model to software model interface
- Software model examples
  - C/C variant models
    - E.g. SystemC
  - Intelligent testbenches
  - Processor/DSP ISS models
  - HDL simulators

[redacted]

Changes and Learning since 2000

- SCE-MI 1.0 has had some success with its goals.
- Some adoption of transaction based methods.
- Increasing complexity – protocols, ip, verification
  - Implies increasing complexity of transactors and verification environments.
- Total cost of ownership / verification.
- Need for verification IP
- Economics of verification IP
- Continued adoption of C and System C as a modeling environment for transaction level models.
- Evolution and maturation of the System Verilog standard including interfaces and modeling constructs.

Goals, Issues, Problems to be solved.

- A viable verification IP market based on standards supporting both simulation and emulation and accessible by average to above average engineers would benefit vendors and users.
  - Enlarges the pie for vendors
  - Increases the ROI on acceleration for customers.
- Reluctance to build emulation-only verification ip
  - Building completely separate simulation and emulation environments is cost prohibitive.
  - Building verification IP for emulation requires a great deal of emulation expertise. Adoption requires reorganization.
  - SCE-MI is seen as an emulation-only standard.
  - SCE-MI requires A+ grade engineers.
  - Most if not all current “external” verification IP exists for simulation only.
- Do all of this without (great) sacrifice in the original goals.

Discussion from 3/2/2005

- Adherence to original goals needs to be emphasized.
  - Existing SCE-MI 1.1 models must continue to function in any new environment.
  - Model writers investment in SCE-MI 1.1 models must be protected.
- Good general agreement and benefits from solving the problem.
  - Broadening the market for acceleration suppliers.
  - Broadening the market for VIP providers
  - Move usage earlier in the process.
  - Reduce specialization required to write 'universal' models.
- Some concern over the scope of the problem as a whole.
- Interface, modeling capabilities and usability are necessary to solve the complete problem.
  - The right interface capabilities determine the overall verification architectures which are supported and well supported.
    - PLI / FLI generally leads to architectures which are simulation specific.
    - SCE-MI 1.1 generally leads to (somewhat) emulation centered transaction architectures.
  - Modeling capabilities determine how much effort is required in porting a given architecture.
    - SCE-MI 1.1 leads to emulation centered models.
    - Although the spec is neutral on modeling, de facto, it uses least-common-denominator style.
  - We would like to keep these independent.
  - Currently SCE-MI 1.1 blends them somewhat.
- ITC is at a crossroads and can choose broadening the scope of the verification IP market to include simulation and emulation over simply enhancing the emulation offering.

Potential Activities

- There is good consensus on the goals of the previous slides.
- Based on these goals, 4 activity areas have been suggested.
  - Interface
  - Modeling
  - Debug
  - Other
- Compliance checking is an operational necessity that has also been raised.
- Status of Discussion
  - Good agreement to do interface work.
  - Mixed views on modeling.
  - Good agreement on compliance suite, but not on timeframe for it.
  - May be a consensus to postpone debug.
  - “Other” may be an empty set.

Interface Work

- ITC can work on setting up an interface
  - which supports a verification environment and model architecture which is ‘natural’ for both simulation and emulation uses.
    - Better performance in simulation
    - Better productivity of IP creation in acceleration and simulation
  - Smooth Transition from Simulation to Acceleration
    - Interfaces and architecture do not need to be changed.
    - Only individual HDL modeling constructs may need to be changed
    - No HVL/C constructs need to be changed.
  - Simulation Implementation of Accelerated Environments
    - Interfaces and architecture do not need to be changed.
    - No HDL modeling constructs need to be changed.
    - No HVL/C constructs need to be changed.
    - Although less desirable, simulator mode may need to be changed.
- The heavy intellectual lifting is here.
  - Pretty good agreement on goals – increasing pie possibility
  - Abstraction bridging
  - Language bridging
  - Architectural concerns
  - Productivity concerns
- This has value in itself, but more value when combined with the next page.

Modeling Work

- System Verilog, VHDL and/or PSL groups can work on an ‘acceleration subset’, i.e. compilation for acceleration. Perhaps this standard can be multi-leveled.
  - Modeling constructs which are trivially implemented.
  - Modeling constructs which are guaranteed to be as efficient as RTL in acceleration.
  - Modeling constructs which are supported, but not as efficient as RTL in acceleration.
  - Modeling constructs which are not supported.
  - Modeling include both procedural (traditional HDL) elements and declarative (property based) elements.
- The heavy political lifting is here.
  - Agreement to support various subsets – LCD, zero sum issues.
  - Differing behavioral compilation capabilities and resources
  - Differing commitment to properties, SV, …
  - User productivity versus ease of implementation.
  - One proposal to eliminate political roadblocks is to blend efficiency and ease of implementation constructs.
- This has value on it’s own, but more combined with the previous page.

Debug Activities

- Transaction based debug?
- More unified debug adds to usability.
- There may be secondary portability benefits.

Other

Compliance Activities

- In support of the first goal on slide 12, it is important that compliance of models and implementations to the standard be verifiable.
  - Compliance suite is a possibility.

Summary

- A viable verification IP market based on standards supporting both simulation and emulation and accessible by average to above average engineers would benefit vendors and users. (slide 12)
- Long term, eliminate the need for “emulation-only” verification IP (slide 12)
- An improved SCE-MI *Interface* goes a long way towards reaching this goal. (slide 15)
  - Allow average to above average engineers to create and use verification IP which is suitable for general use in simulation and emulation. (slide 12)
    - Eliminate the need for specialized interface knowledge for acceleration.
    - The only specialized knowledge required for acceleration would be synthesis / compilation for acceleration.
    - Eventually displace PLI as the dominant verification IP interface standard.
  - Make movement from simulation to acceleration purely an HDL side model refinement problem (slide 15)
    - Interfaces and architecture do not need to be changed.
    - No HVL/C constructs need to be changed.
    - Only individual HDL modeling constructs may need to be changed.
  - Allow movement from acceleration to simulation with no (or minimal) see slide 15.
  - Ease of modeling effort
    - Specify a common interface that supports simulation and emulation
    - Simplify software-side/hardware-side synchronization mechanism
  - Determinism
    - Introduce no non-determinism beyond that already inherent in HDLs
  - Additional functionality to increase performance opportunity
    - Without negatively impacting Verification IP portability
- Interface work should not preclude work to improve modeling capabilities (see slides 15-16).
- backward compatibility with SCE-MI 1.1 (slide 13)
  - Existing SCE-MI 1.1 models must continue to function in any new environment.
- Maintain SCE-MI 1.x Goals (slide 13)
  - Performance
    - Interface should not be the performance bottleneck
  - Software-side language neutrality
    - Use C as least-common-denominator interface
  - Hardware-side support of Verilog (& SV), VHDL
    - Provide bindings to all three major HDLs.