Accellera Breakfast and Panel Discussion:
UVM 1.2 Roundtable with John Aynsley
UVM Roundtable Panelists

- Amol Bhinge, Senior SoC Verification Manager, Freescale
- Colin McKellar, Senior Director of HW Engineering, Imagination
- John Aynsley, CTO, Doulos
- Mohamed Elmalaki, Design Verification Architect, Intel
- Rich Newton, Senior Manager Design Automation, Ericsson
Migration to UVM …… Journey so far!

- Managing global SoC verification team in Austin, Israel and India (100+)
  - Taped out a flagship product with 2B transistors in 2013 using internal methodology, with one of the IP teams (SoC IP team) adopting UVM.
  - Chose migration to UVM 1.1 for the next generation product to be taped out in 2014
- Why UVM? When should we migrate? Who should migrate? Is there a magical tool?
  - Usual Schedule, Quality, Resources and Complexity (SQRC) equation
- Benefits to overcome SQRC challenges
  - Cleaner environment with better support from vendors including third party IPs/VIPs
  - Possibly fine-tuned simulation and debugging performance
  - Attracting and retaining talent.
- Challenges
  - No clear direction on register modeling enforcing co-existence of multiple formats
  - Reuse for stimuli from IP to sub-system to SoC to silicon and vice versa
  - Several IPs not yet moved to UVM making it impossible to reuse their collateral
  - Co-existence with internal methodology
  - Reading between lines: may not be plug-n-play among implementations

- Migration to UVM was a blessing in several contexts but not yet absolute victory.
Mohamed Elmalaki
Design Verification Architect – Intel Corp
UVM WG Contributor – Accellera

UVM Direction

Freeze Features
Very high bar for any extra feature to be added to UVM base library. Strict backward compatibility

Focus on Quality
Increase testing/Enhance Documentation/IEEE Standardization

Encourage Community Contributions
Build over UVM rather than into UVM. e.g.: MLWG, AMS, Emulation
Rich Newton, Ericsson
IP Systems ASIC Group, San Jose, CA

• Group moving to UVM for next project
  – Currently using in-house SV class library

• Concerns
  – UVM style – which is “best”?  
  – Performance
    • UVM_REG scalability
    • Logging, config db
  – Moving target of standardization