SystemC's Role in a Multilingual World

Grant Martin

Fellow, Cadence Berkeley Labs

European SystemC Users Group, Stuttgart, November 7 2003

Outline – 15 minute talk

- Alphabet Soup
- Language War?
- Where does SystemC fit?
- What about UML?
- Conclusion

Alphabet Soup

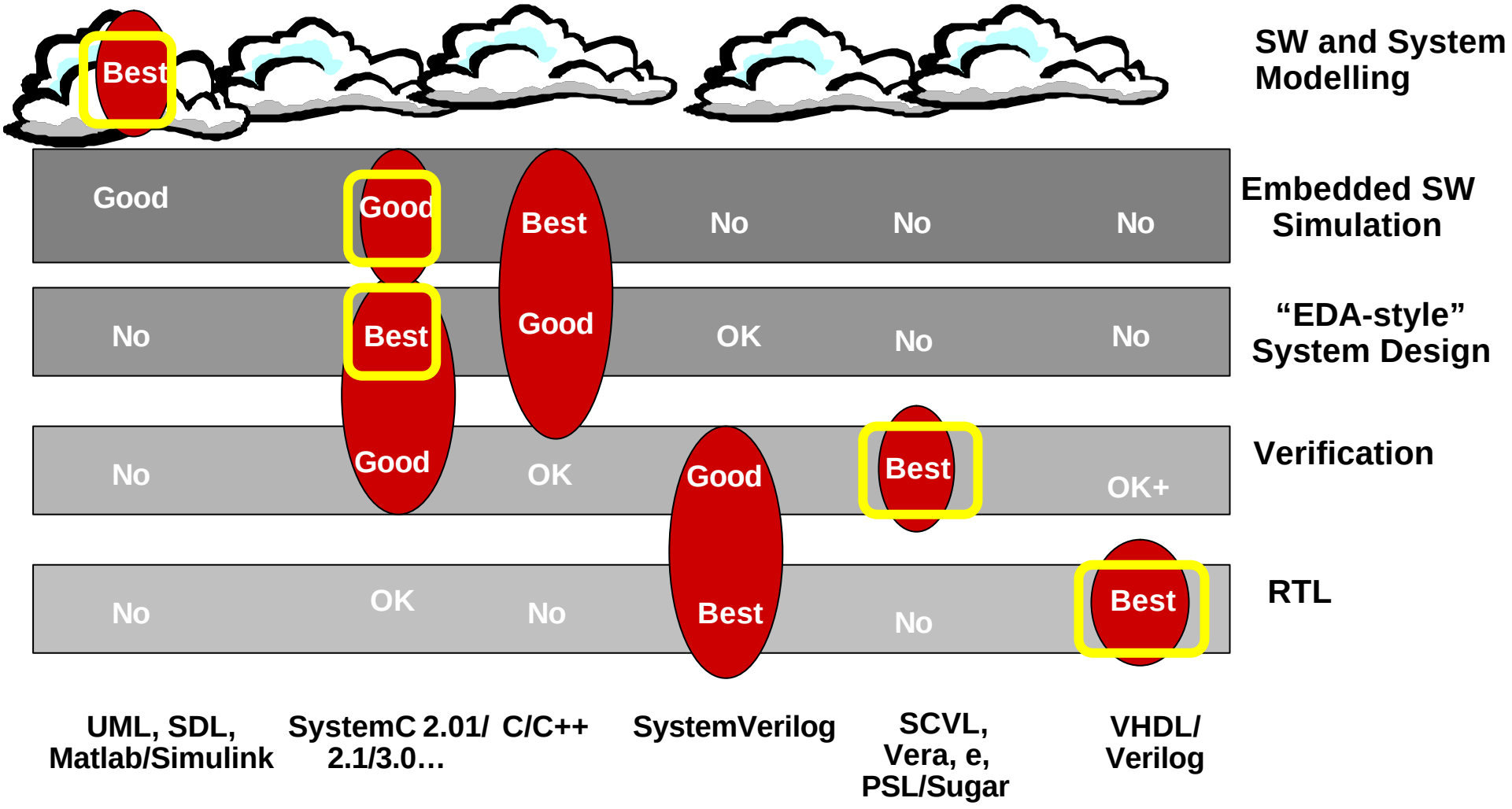
SystemC
PSL
UML
Sugar
VHDL
SDL
Vera
OVL
Verilog -A
VHDL-AMS
OpenVera
SystemVerilog
Verilog
Verilog -2005
OVA
e
Matlab/Simulink

A “typical” System-on-Chip Design Project?

- Take a microprocessor and its Instruction Set Simulator (ISS)…modelled in **SystemC**
- Add in some embedded SW – **C** or **C++**
- Add in some digital IP blocks – some modelled in **Matlab** (dataflow) and some created in **Verilog**
- Buy some 3rd party digital IP – from a **VHDL**-based supplier
- Need to have some AMS interfaces – **Spice** models
- Need to validate AMS interfaces in SoC context – **Verilog-A**
- Build some testbenches – **e**
- Create some block assertions – **PSL**

……is 9 different languages or notations enough?

Making Sense of the Alphabet Soup

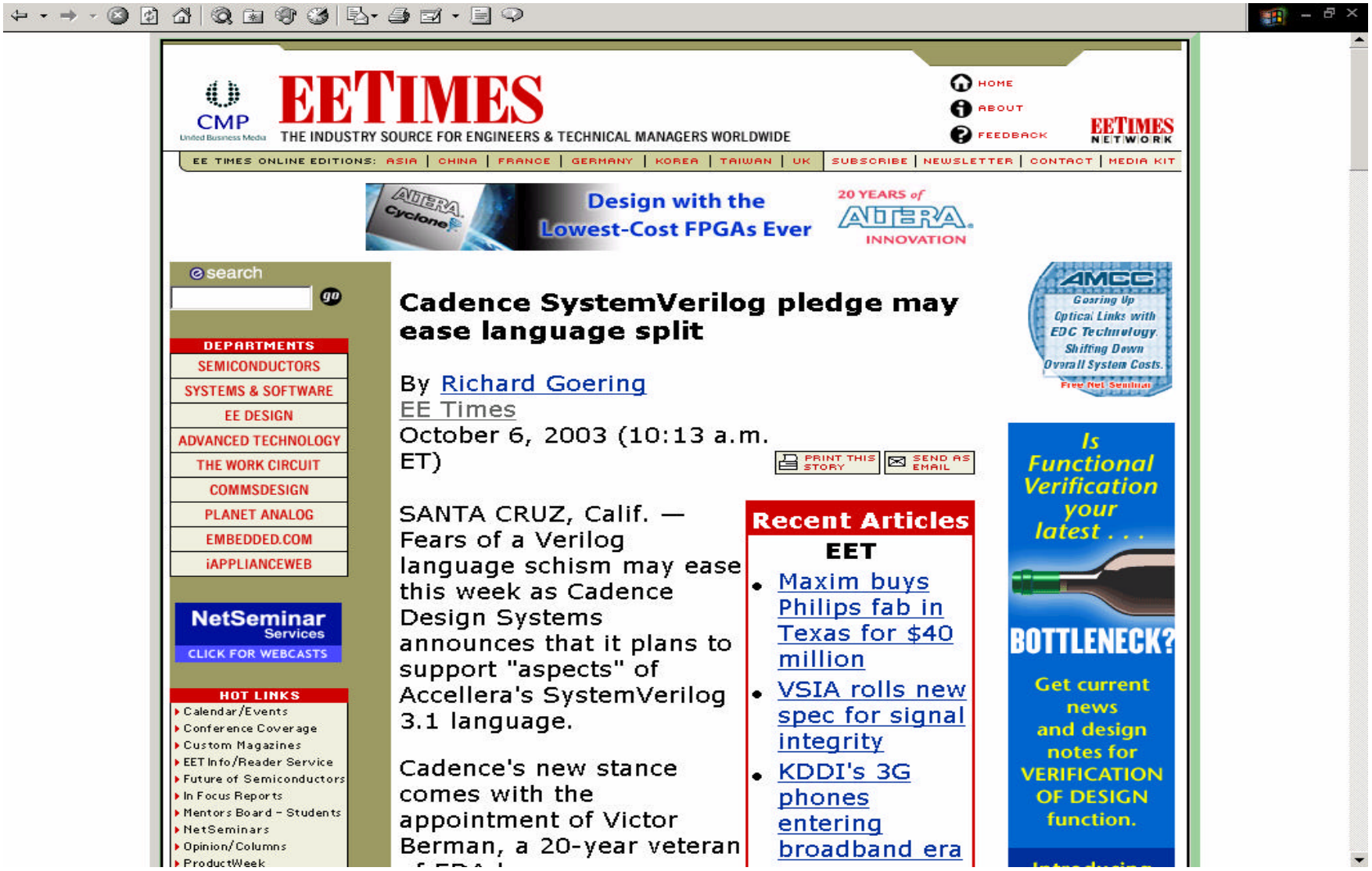


No one language does everything.

A Language War?

- **Accellera sets date for SystemVerilog donation**
  By Richard Goering, EE Times, October 20, 2003
- **SystemC won't go away quietly**
  By John Cooley, EE Times, Jun 23, 2003
- **Cadence IEEE donation overlaps SystemVerilog**
  By Richard Goering, EE Times, Jun 2, 2003)
- **EDA language dispute erupts in advance of DAC**
  By Richard Goering, EE Times, May 30, 2003
- **New EDA consortium promotes assertion language**
  By Richard Goering, EE Times, May 22, 2003
- **VHDL, the new Latin**
  By John Cooley, EE Times, Apr 7, 2003
- **Verilog 2001 compliance lacking, designer says**
  By Richard Goering, EE Times, Feb 26, 2003
- **DVCon: SystemVerilog key to new design paradigm**
  By Michael Santarini, EE Times, Feb 24, 2003 (7:45 PM)
- **C loves me, C loves me not**
  By Richard Goering, EE Times, Sep 16, 2002
- **Synopsys donation defuses assertion language war**
  By Richard Goering, EE Times, June 11, 2002

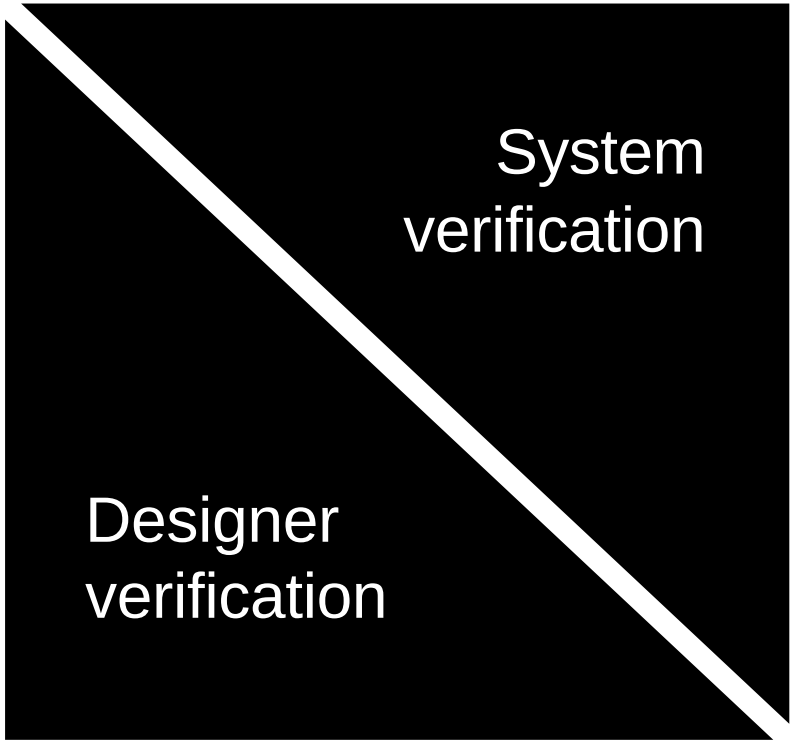
Or Peaceful coexistence?



cadence

Cadence Language Strategy

Support all standards for all designs



- Verilog & VHDL
- SystemC & SCV
- SystemVerilog
- PSL, OVL, & SVA
- Verilog/VHDL-AMS

Incisive

Encounter | Virtuoso

Incisive single-kernel architecture supports all languages natively

SystemC is for *System-Level Design*

(Hugo De Man's "7th. Heaven of Software")

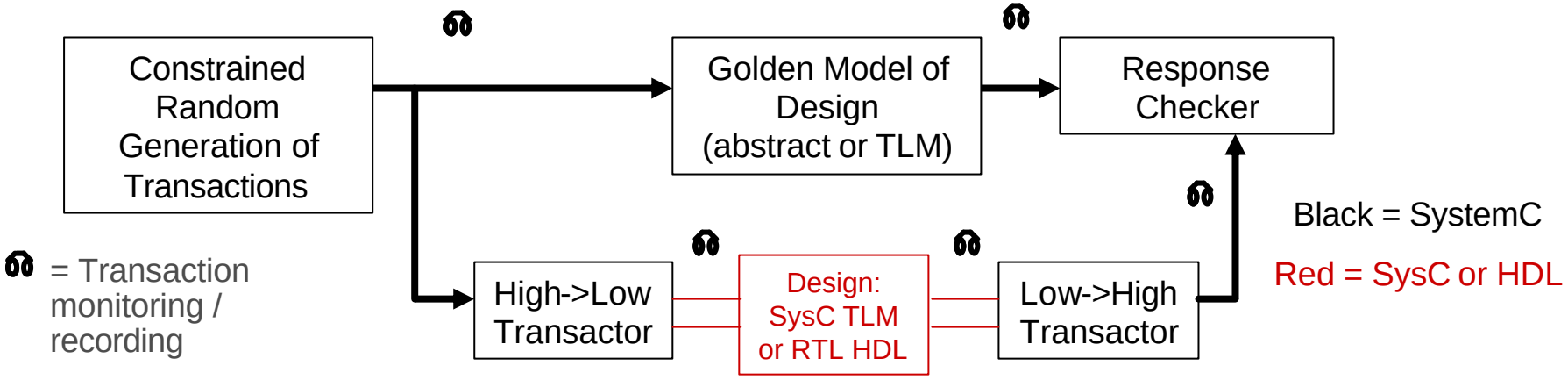
System and SW Modeling: UML, SDL, etc.

System-Level Design, Verification and System Level Integration Infrastructure: SystemC

Mere Implementation!! VHDL, Verilog, SystemVerilog

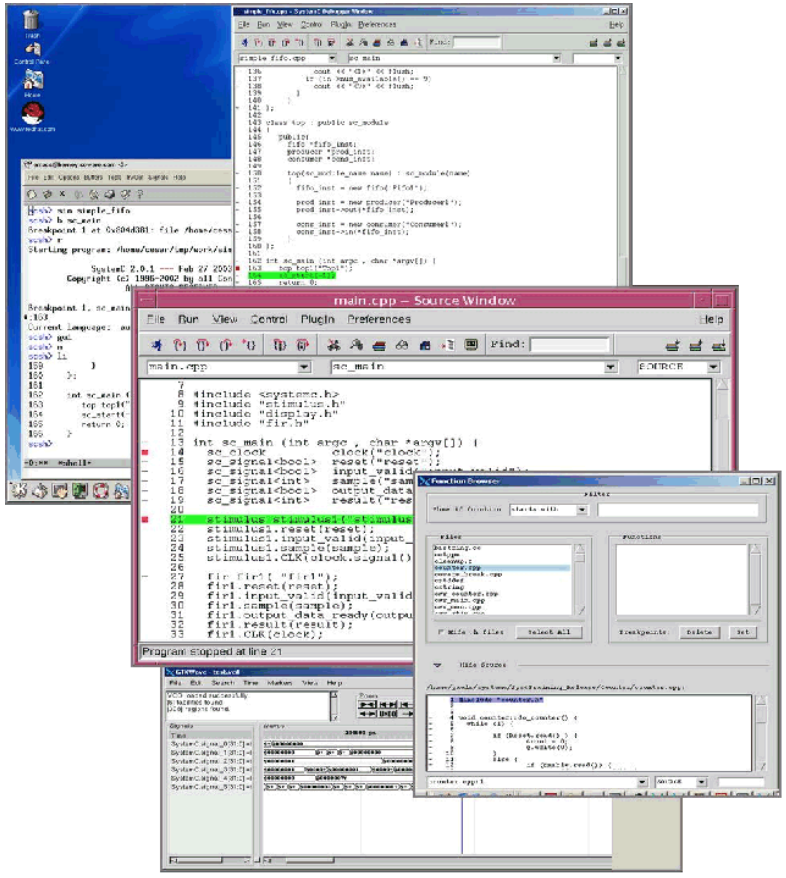
(Hugo De Man's "Deep Submicron Hell of Physics")

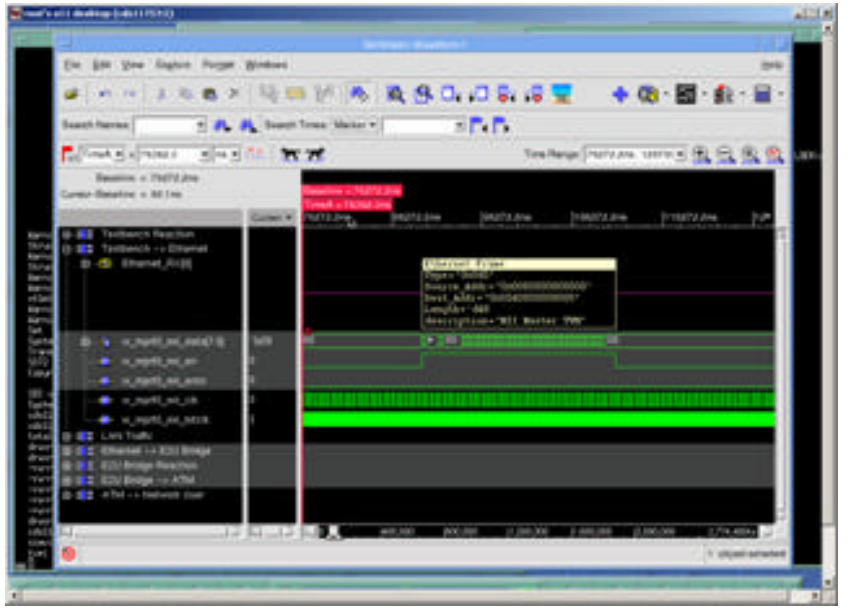
SystemC is for *System-Level Verification –e.g. transaction-based verification*



- Why do transaction-based verification in SystemC?
  - Ability to have everything (except perhaps RTL HDL) in SystemC/C++ provides great benefits: easier to learn and understand, easier to debug, higher performance, easy to integrate C/C++ code & models, open source implementation, completely based on industry standards
  - Allows you to develop smart testbenches early in the design process (before developing detailed RTL) to find bugs and issues earlier. Enables testbench reuse throughout the design process.
  - Much more efficient use of verification development effort and verification compute resources
- Transaction-Based Verification in SystemC is described in the *SystemC Verification (SCV) Standard Specification*, and in the documentation and examples included with the OSCI SCV reference implementation kit.

SystemC is also for *RTL-Level Verification via Testbench and System-Level model reuse*



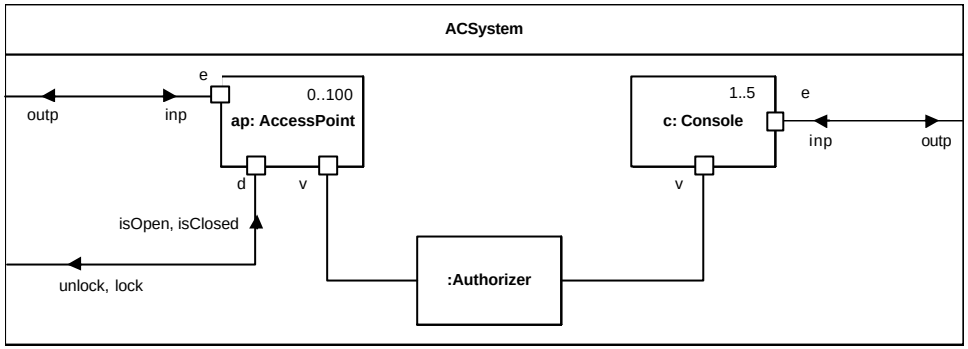


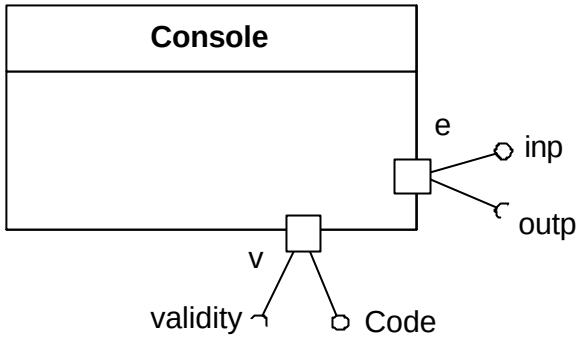
cadence

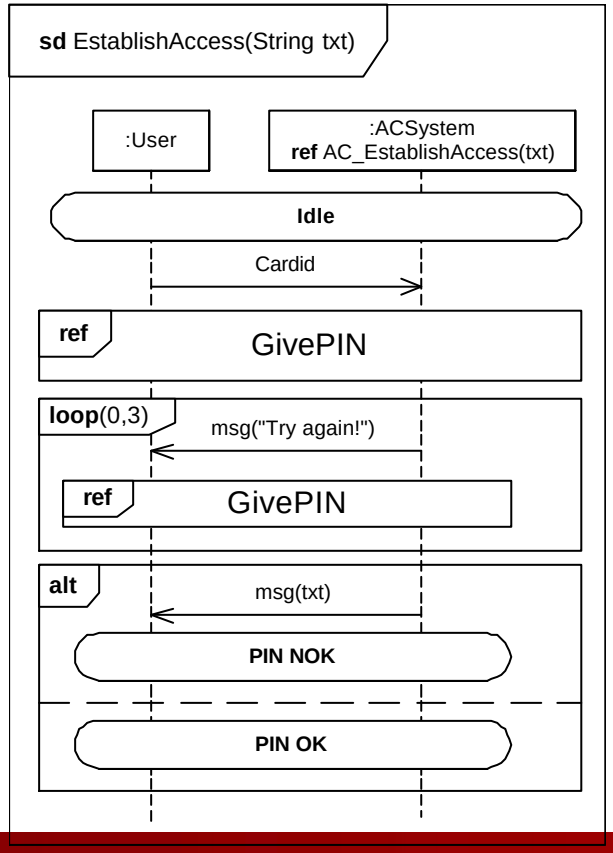
Let's not forget Software and embedded systems in our multilingual world

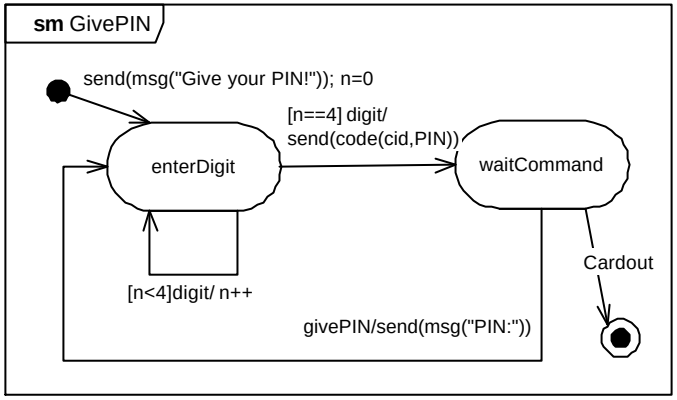
- UML moving towards finalisation in 2003 of UML 2.0
- Adds significant system modelling capabilities to UML's object-oriented software modelling capabilities:
  - Structured classes or components (what in ROOM were called capsules)
  - Ports as interfaces and service specifications
  - Hierarchical sequence diagrams with loops and alternatives
  - Hierarchical state machines
- In addition, UML profile for Scheduling, Performance and Time (SPT), and Action semantics definition, support more complete modelling of real-time aspects of systems, and more optimised code generation
- Finally, the Object Management Group (OMG)'s embrace of MDA – Model Driven Architecture – fits more closely with the EDA/HW design world notions of design methodology and flow

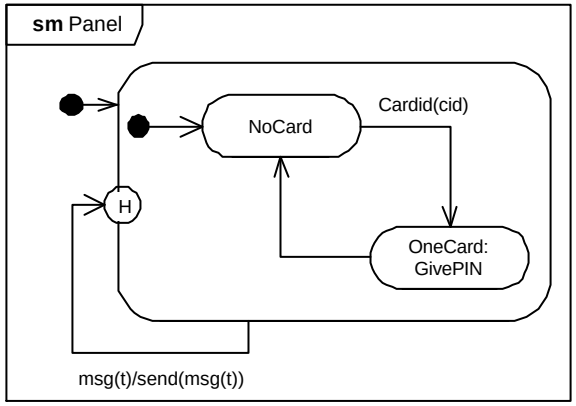
UML 2.0


ACSystem
e
outp
inp
0..100
ap: AccessPoint
d
v
1..5
c: Console
e
inp
outp
v
isOpen, isClosed
unlock, lock
:Authorizer

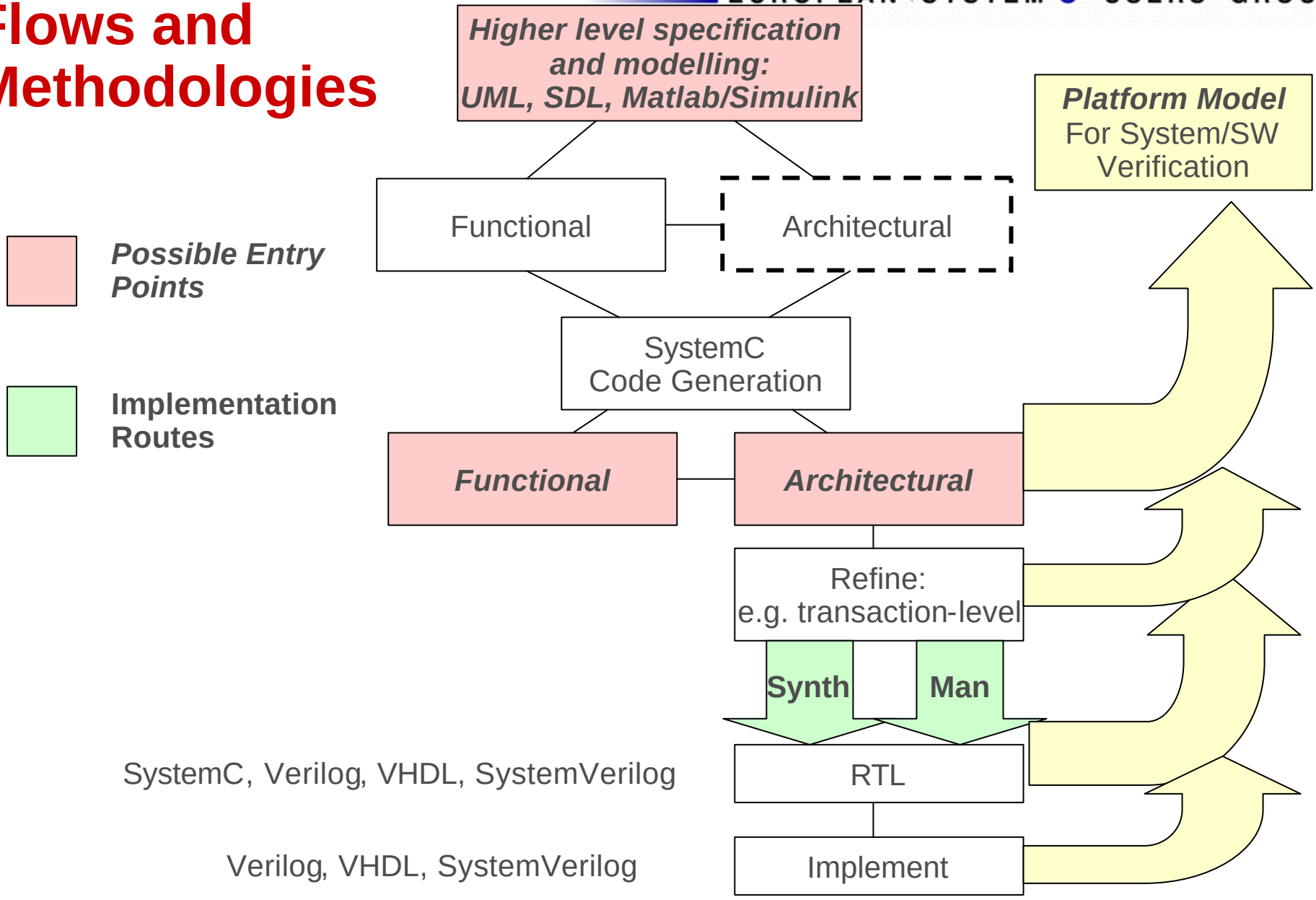

Console
e
inp
outp
v
validity
Code


sd EstablishAccess(String txt)
:User
:ACSystem
ref AC_EstablishAccess(txt)
Idle
Cardid
ref
GivePIN
loop(0,3)
msg("Try again!")
ref
GivePIN
alt
msg(txt)
PIN NOK
PIN OK

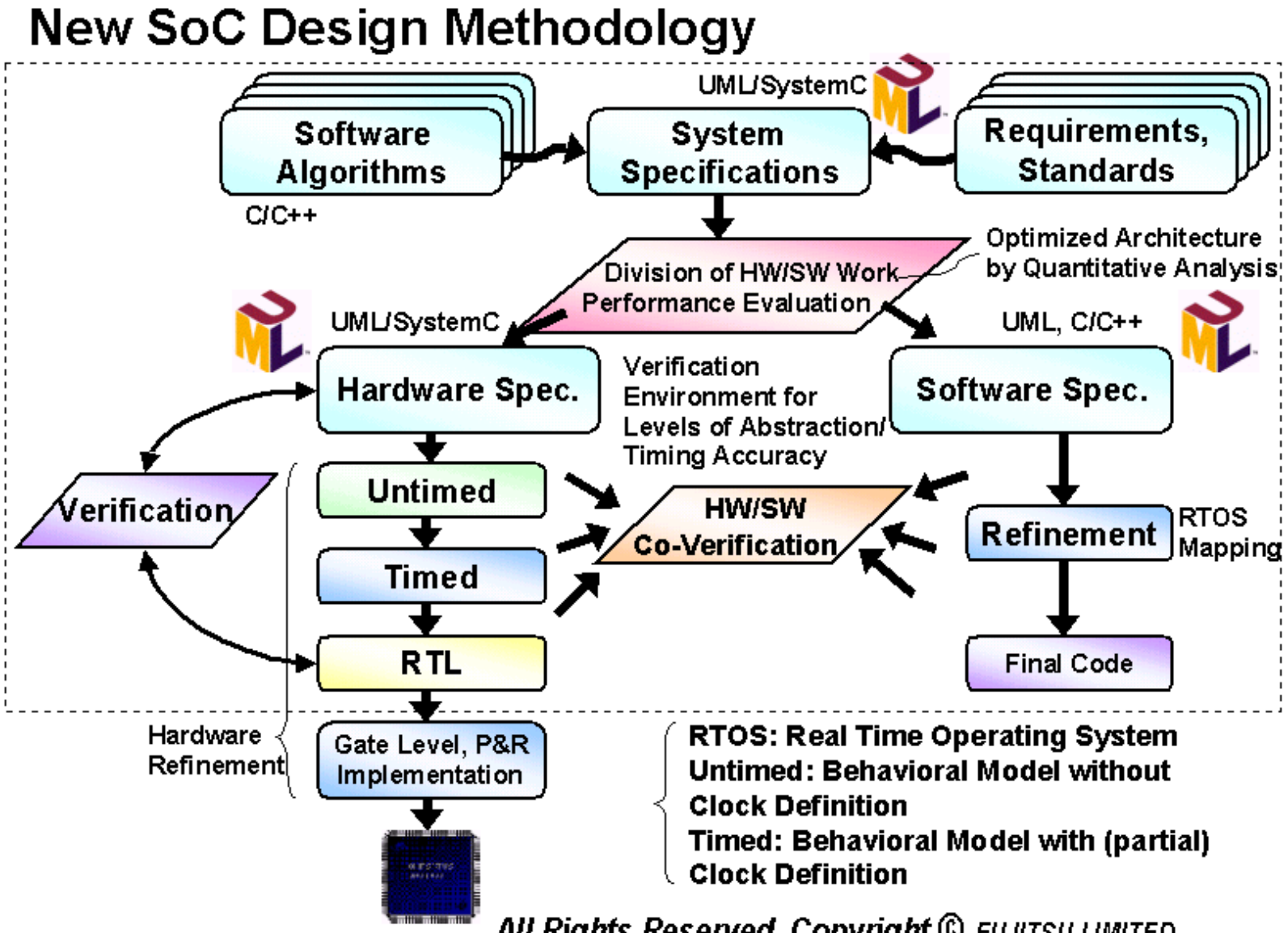

sm GivePIN
send(msg("Give your PIN!")); n=0
[n==4] digit/
send(code(cid,PIN))
enterDigit
waitCommand
Cardout
[n<4]digit/ n++
givePIN/send(msg("PIN:"))


sm Panel
NoCard
Cardid(cid)
H
OneCard:
GivePIN
msg(t)/send(msg(t))

Flows and Methodologies



Fujitsu – UML, SystemC, flow to RTL implementation



Conclusions

- The alphabet soup needs simplification
  - But simplification does not mean only one language
- SystemC is good for:
  - System Level Design
  - System Level Verification
  - A “transmission belt” for testbenches and golden system models to RTL design and verification level
- Let’s not forget those on top
  - For example, who may be looking at UML for specification
  - And who want to flow naturally into SystemC and beyond into SystemHDLs