



SystemVerilog: Getting Started with SystemVerilog for Design and Verification

Tom Carlson
Surrendra Dudani
Joao Geada
Phil Moorby
Karen Pieper
Dave Rich
Arturo Salz

Moderated by
Cliff Cummings,
Sunburst Design, Inc.

New SystemVerilog Capabilities

SystemVerilog 3.0
Major enhancements:

RTL & behavioral
design enhancements

New data types for
simulation and co-
simulation with 'C' code

Most of the **Superlog**
language

SystemVerilog 3.1
Major enhancements:

**Hardware Verification
Language (HVL)**
capabilities added

Enhanced event
scheduling

Immediate and
concurrent **assertions**

C-language
integration

SystemVerilog 3.1a
Major enhancements:

Coverage
enhancements

Some additional
verification
enhancements

VHDL-like packages

Bluespec language
capabilities for high-level
design abstraction and
synthesis

Next Slide

Verilog Race Conditions

IMPORTANT !!

- SystemVerilog is fully backward compatible with Verilog-2001
- **Verilog RTL races are still SystemVerilog RTL races !!**
 - New SystemVerilog features reduce simulation-synthesis mismatches
- New SystemVerilog event regions reduce races between
 - RTL
 - Verification Enhancements
 - Assertions
- SystemVerilog does not compensate for poor coding styles or bad coders

**SystemVerilog reduces
races between RTL models
and HVL testbenches!**

**Poor coding styles can be
fixed with good training**

**Hopelessly bad coders should
consider new career counseling!**



"When to Switch?"

- ESNUG moderator, John Cooley, quoted the question

"Are there any users out there who are contemplating switching to SystemVerilog?"

This is the wrong question!!

SystemVerilog is 100% backward compatible with Verilog-2001

You can start using SystemVerilog features **NOW using Synopsys tools**

You do **NOT have to change your Verilog & VHDL code**

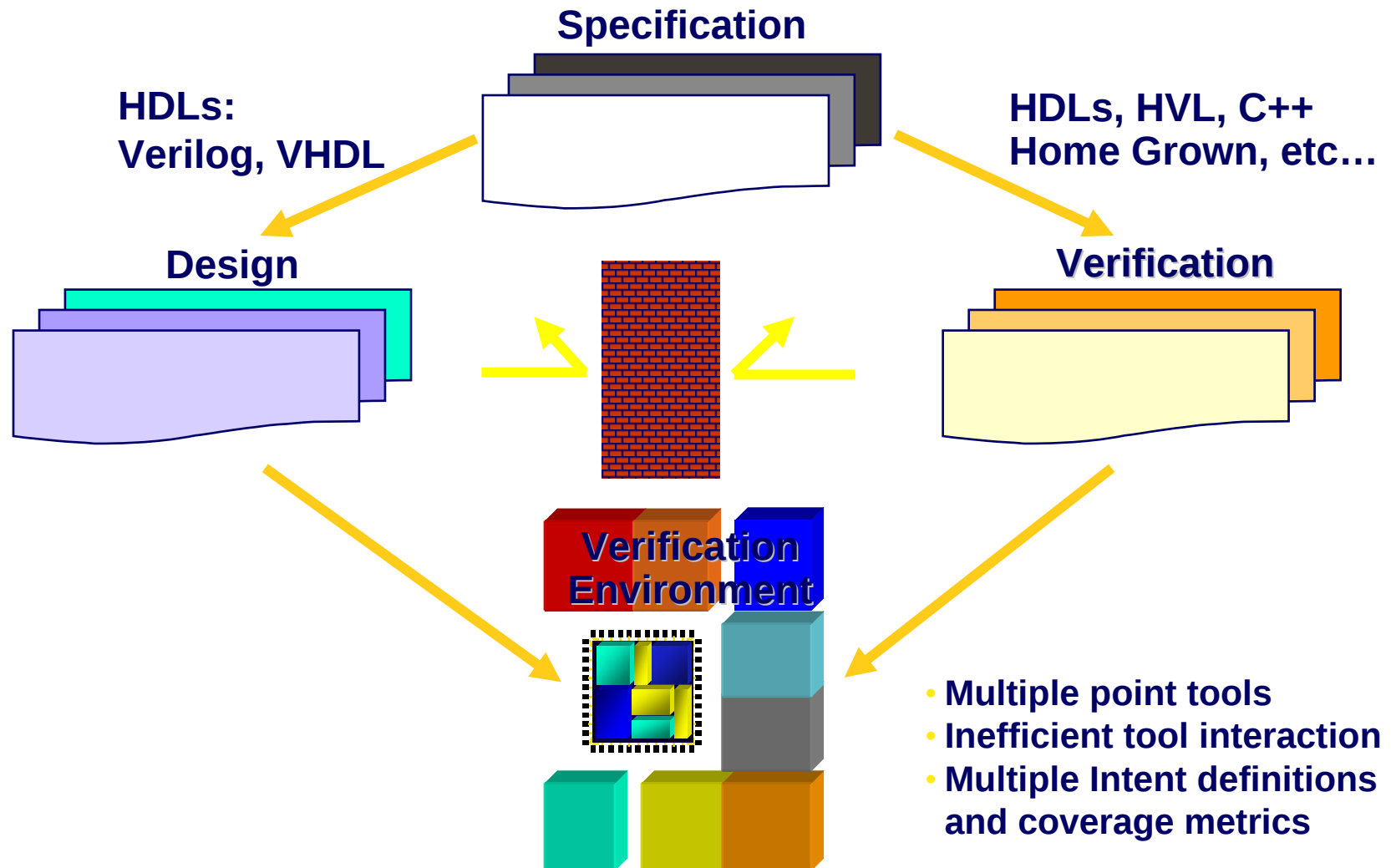
Find your favorite SystemVerilog features and start using them **NOW**



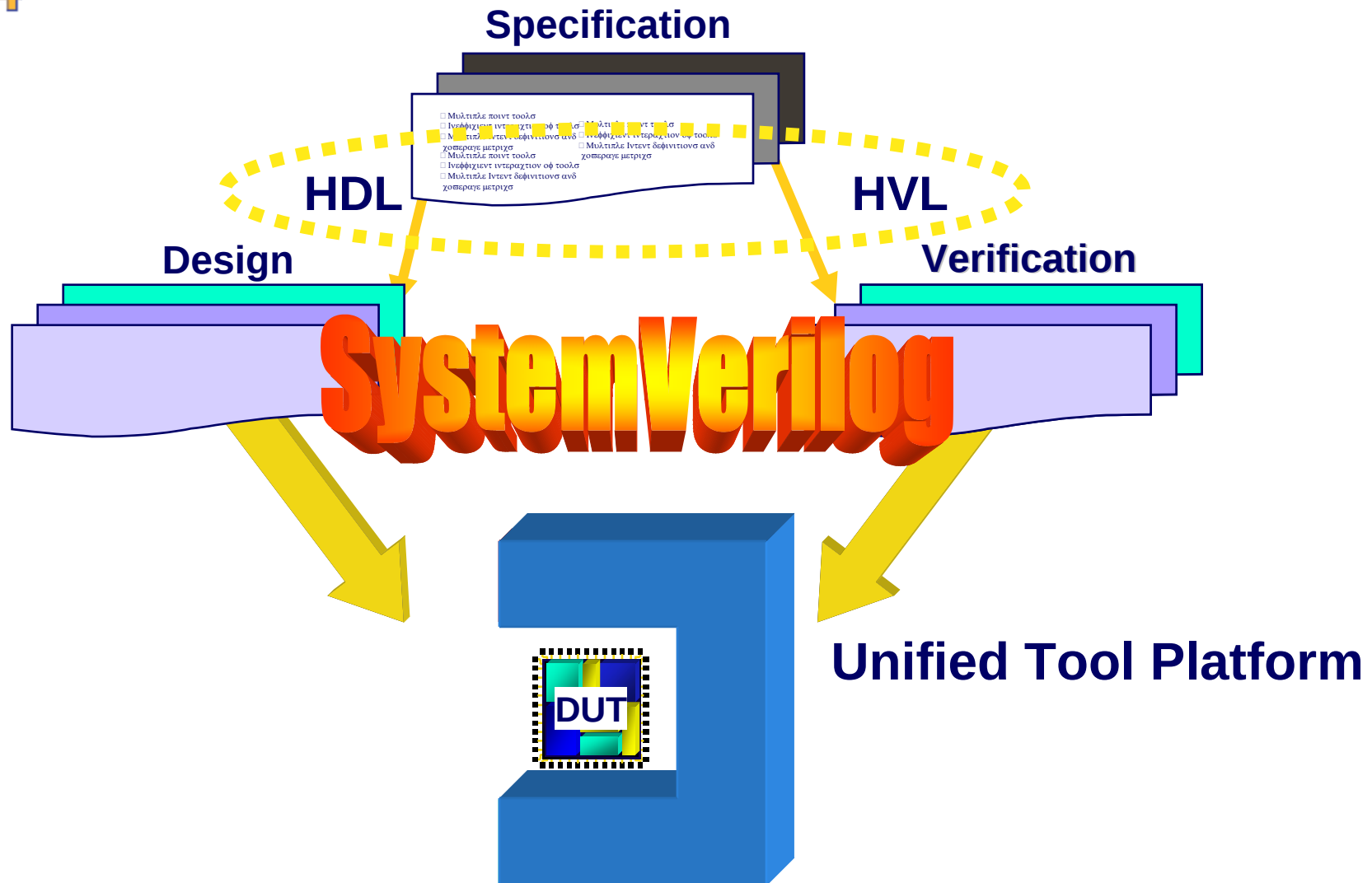
Our Synopsys Panel

- David Rich – SV-BC Design
- Arturo Salz – SV-EC Testbench
- Surrendra Dudani – SV-AC Assertions
- Joao Geada – SV-CC C-Interface
- Karen Pieper – P1800 Errata Chair
 - HDL Compiler
- Phillip Moorby
 - Verilog Creator
- Tom Carlson – Formality

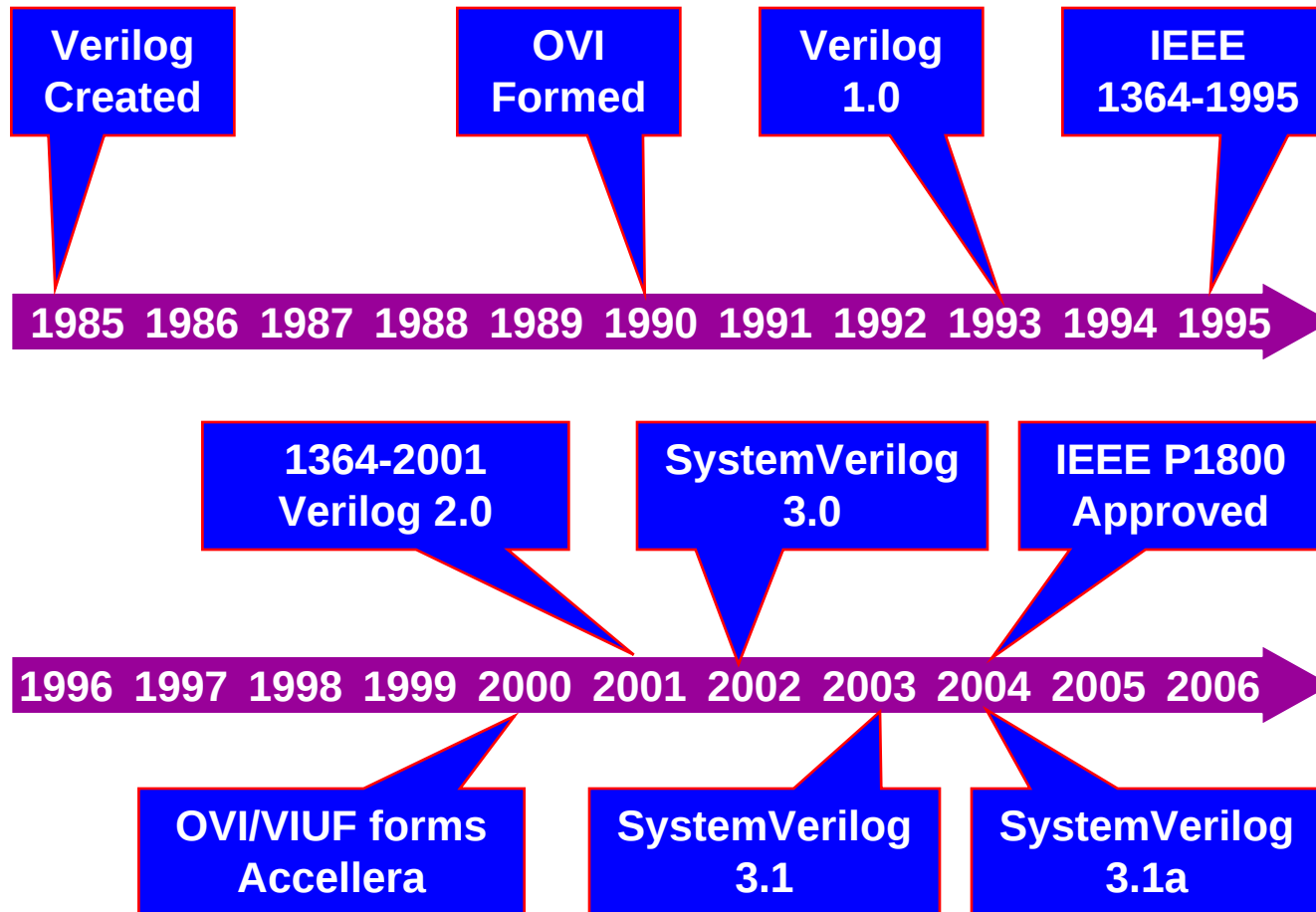
RTL Design and Verification Today



Unifying Design and Verification



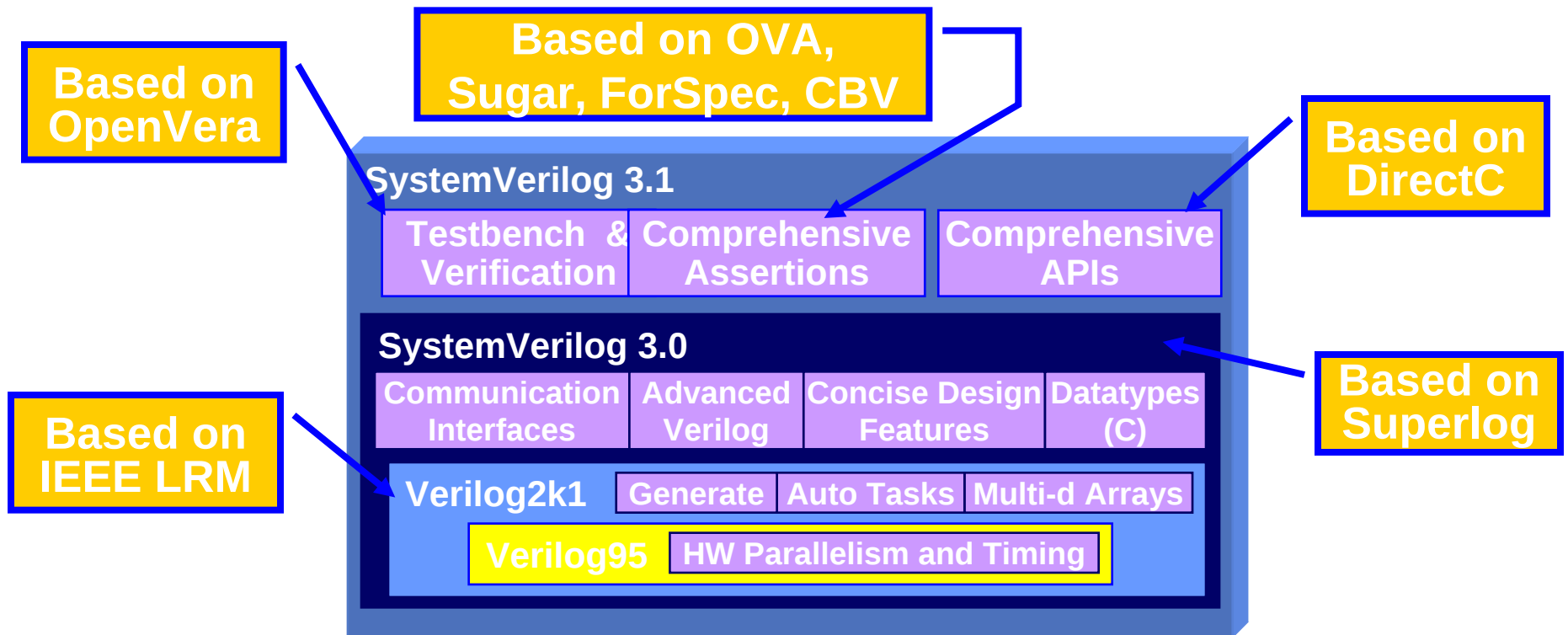
Quick History of Verilog



SystemVerilog Key Components

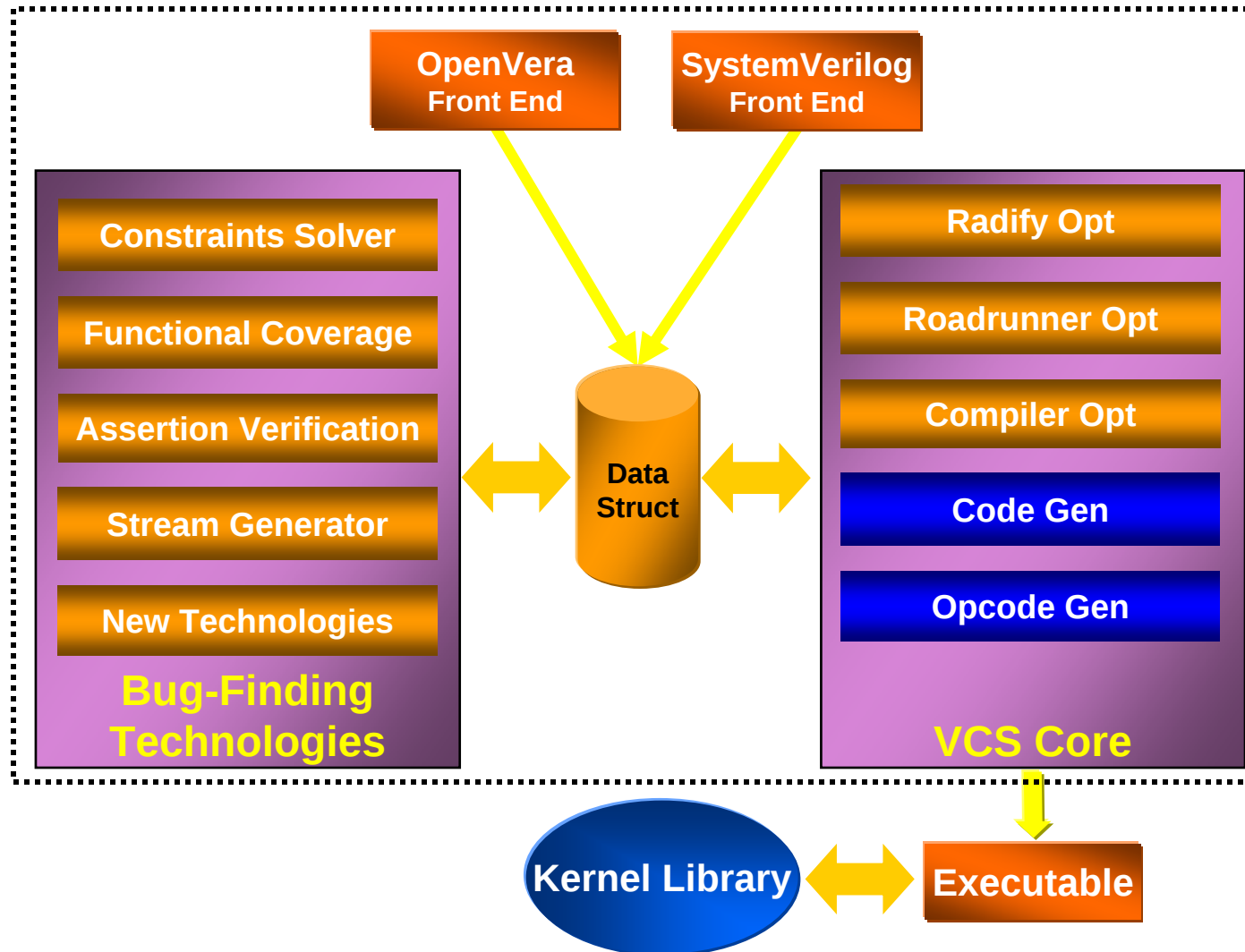
- Accellera Standard, Next Generation Verilog, HDVL Language

Based on technologies proven by EDA vendors and users

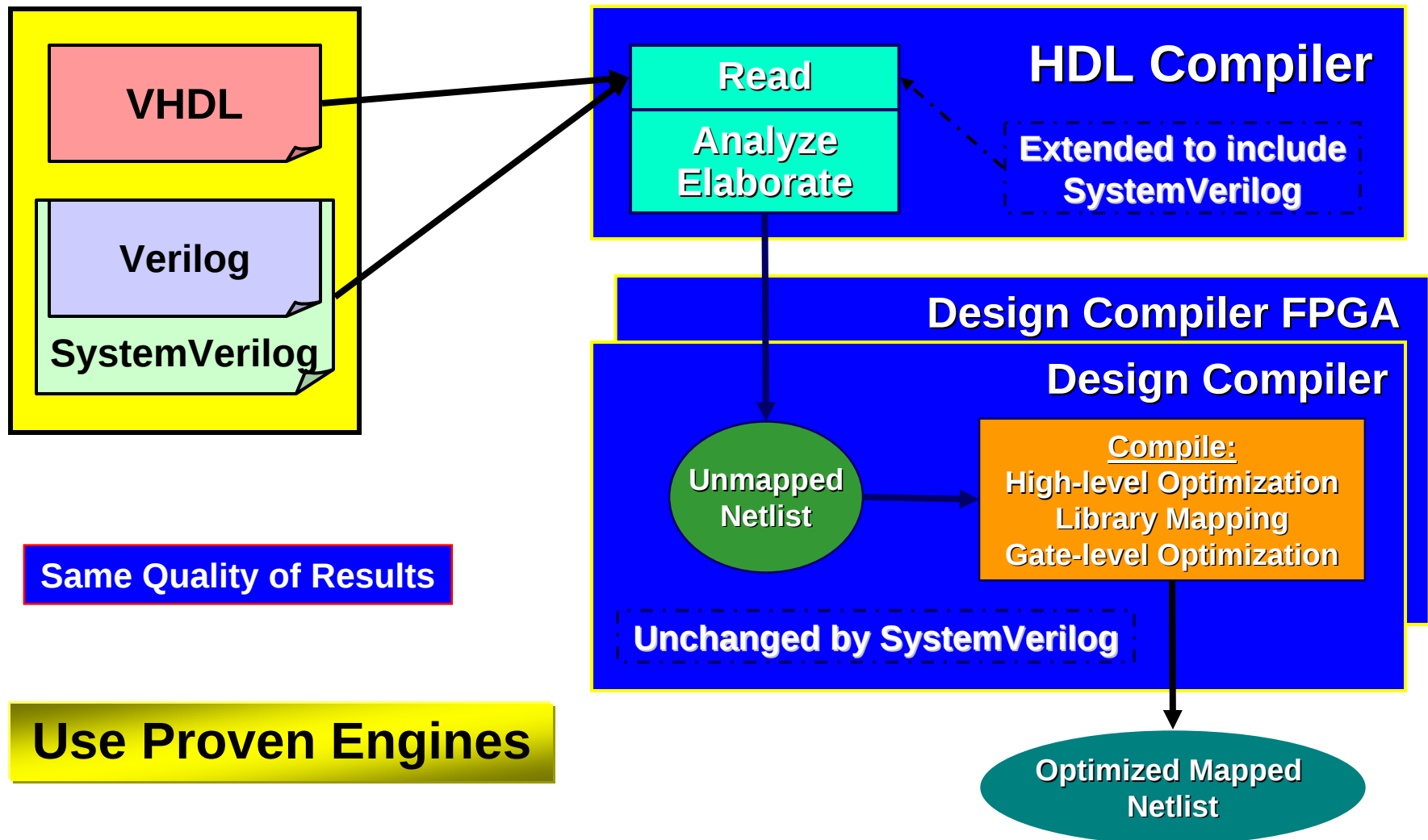


VCS Native Simulation Architecture

Bug-finding technologies independent of front-end



Same Synthesis Environment Design Compiler



Synopsys Offers Comprehensive SystemVerilog Design & Verification Solution

Product	SystemVerilog Support	Availability
VCS®	RTL Design Assertions & Checker Library DPI	☒ ☒ ☒
Design Compiler®	RTL Design	☒
Magellan™	Assertions (beta) RTL Design (beta Q4/04)	☒
Formality®	RTL Design (beta)	☒
Leda®	RTL Design Assertions	☒