How to Verify Complex FPGA Designs for Free

Sebastian Dreßler

Nikos Anastasiadis

Thomas Richter

accellera SYSTEMS INITIATIVE

swarm64

2016 DESIGN AND VERIFICATION™ DVCON CONFERENCE AND EXHIBITION EUROPE

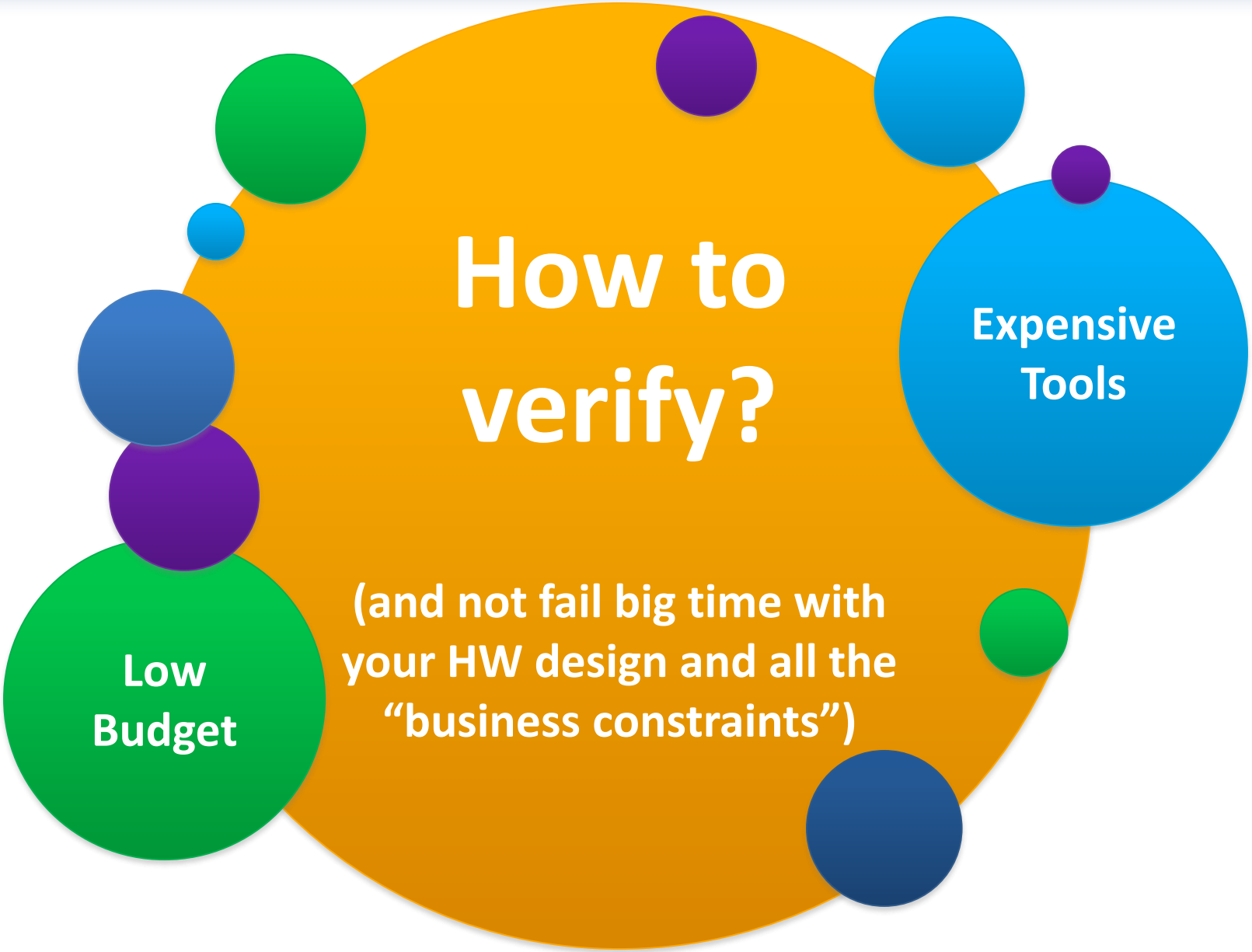
Swarm64 Mission: SQL DBs for Big Data

Full Stack Solution

16x SSDs

Swarm64 SDA

Swarm64 SQL DB Plugin


How to verify?
(and not fail big time with your HW design and all the "business constraints")
Expensive Tools
Low Budget

OUR SOLUTION

TOOLSET

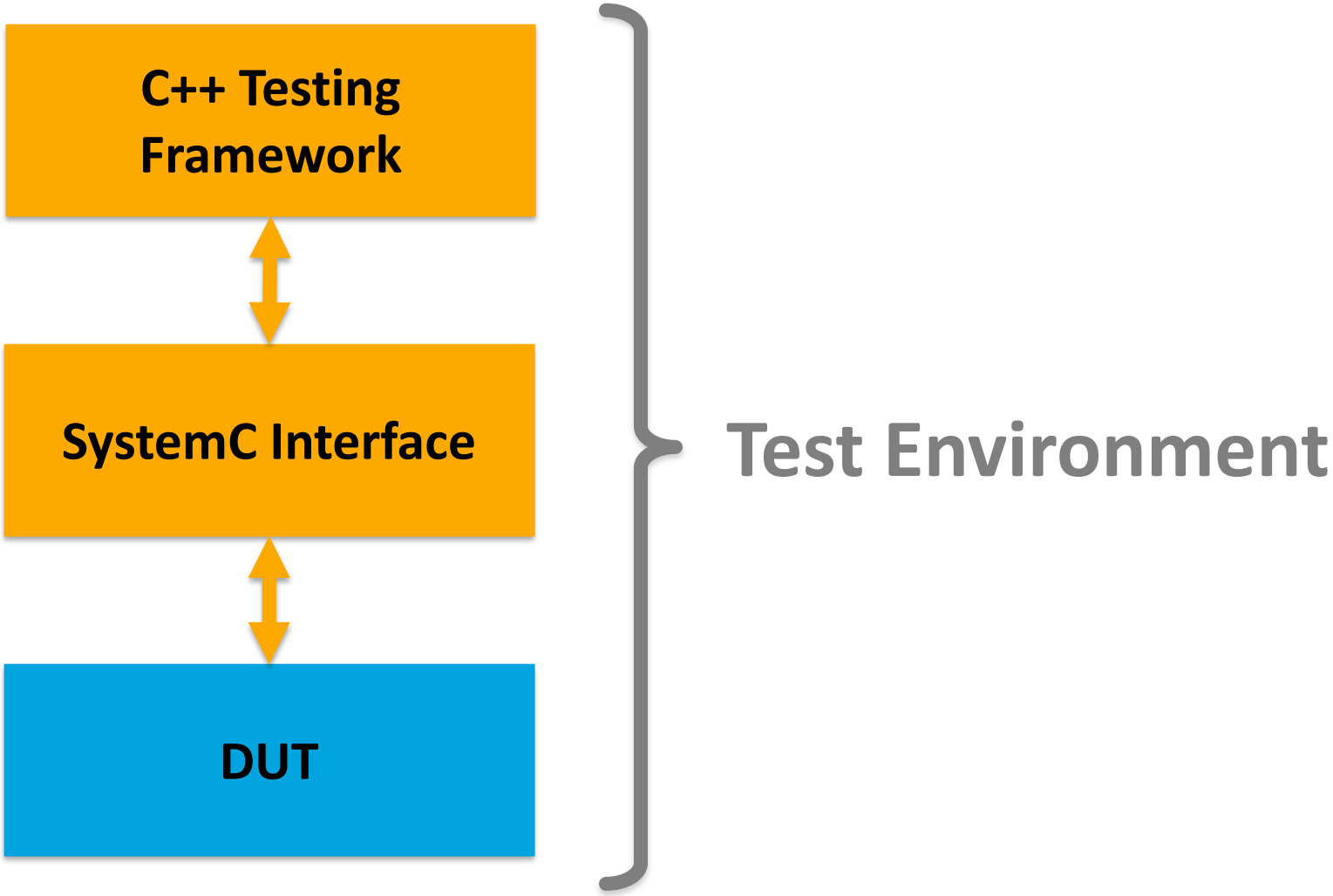
- Language: C++(11/14) & SystemC
- Open Source Software

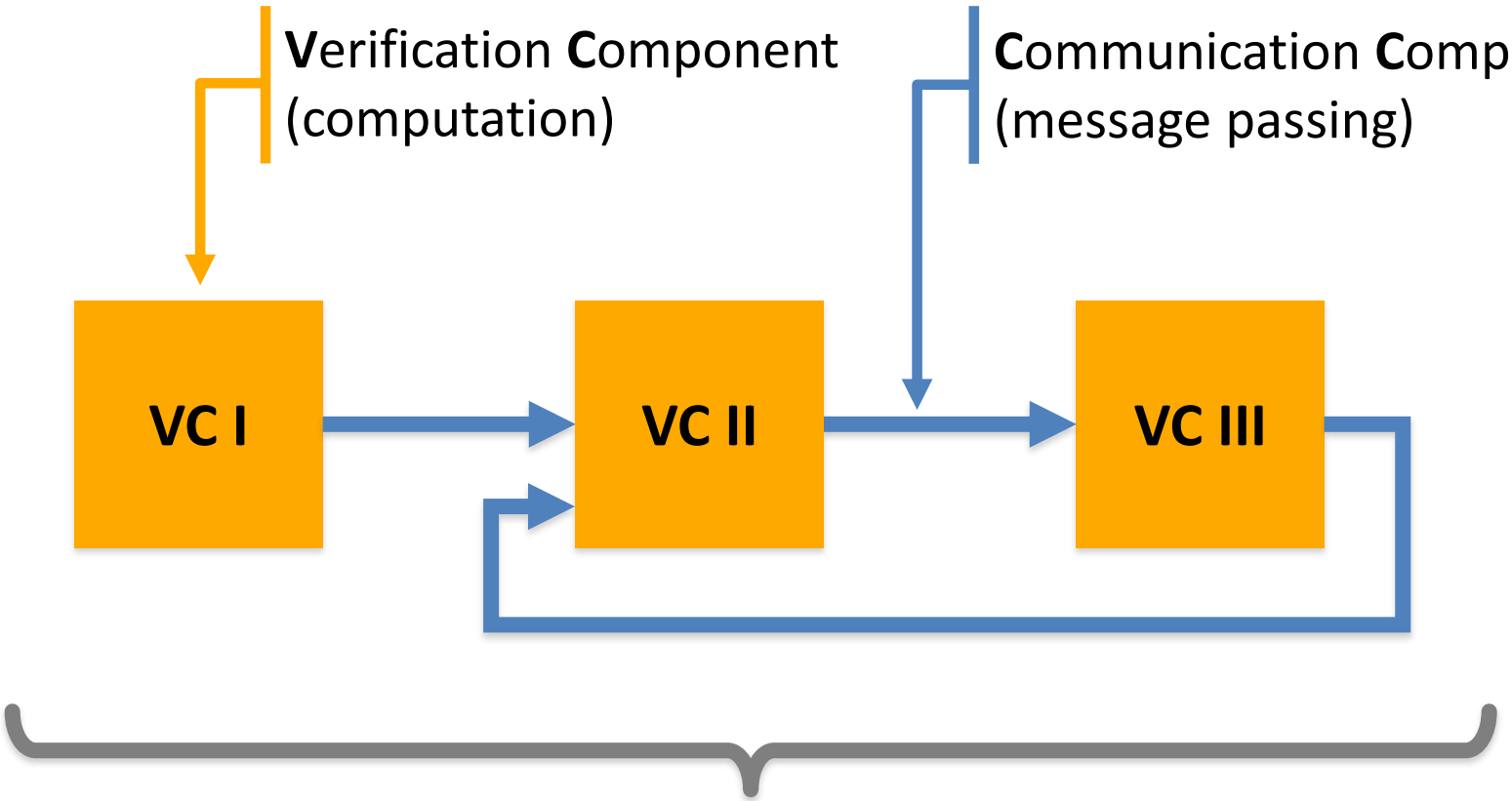
  Verilator (simulator), **GTKWave** (waveform viewer)
- Engineers: have to know C++; get help by SW

VERILATOR?

- Written & maintained by Wilson Snyder
- Not a “classic” simulator
- Transforms Verilog to C++
- Provides a runtime
- Compatible with SystemC datatypes

TEST FRAMEWORK





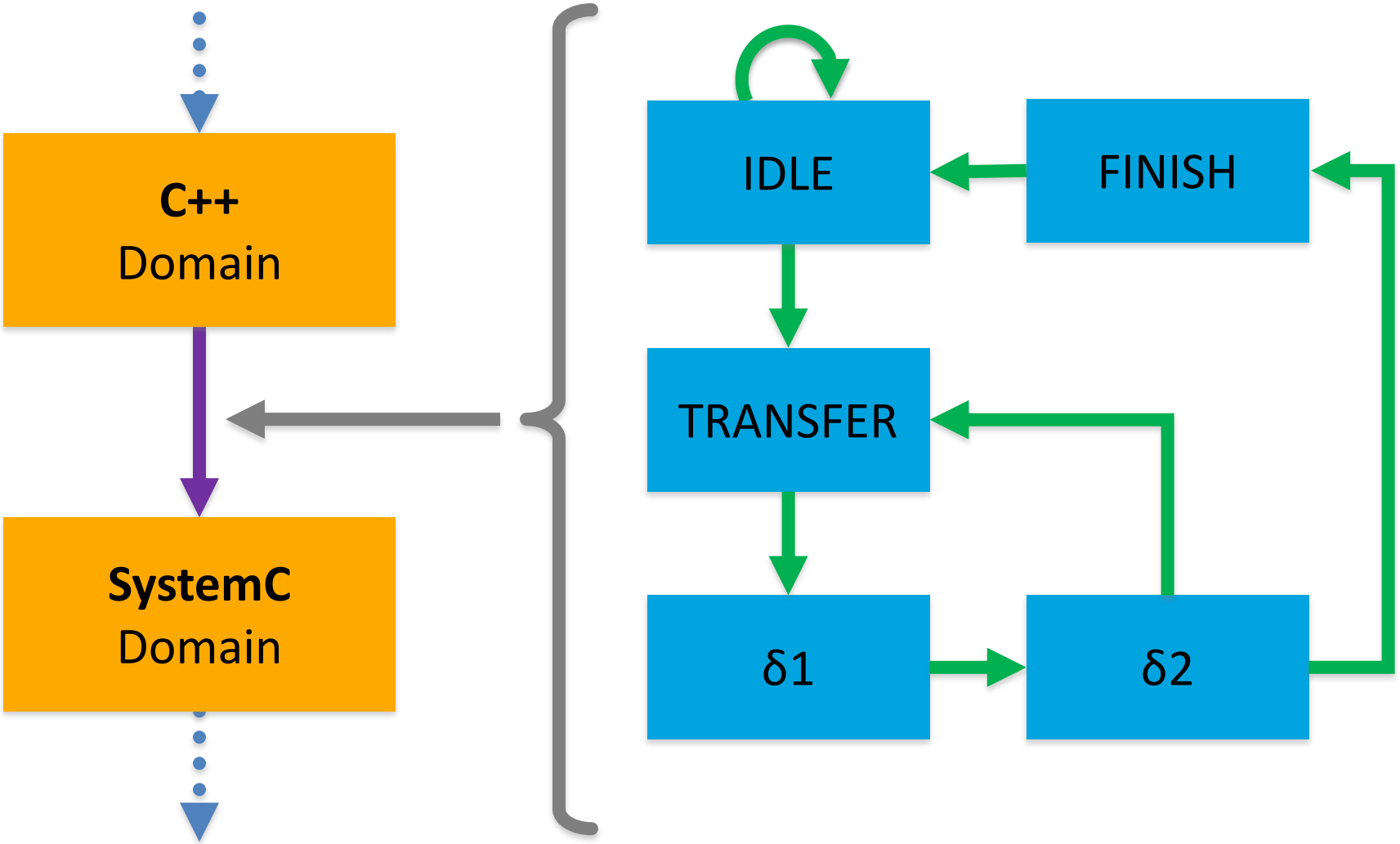
C++ Test Structure Graph

VCs

- Command generator
- Command registry
- Result verificator
- …

CCs

- Simple queue
- Reordering queue
- Delay queue
- …


C++
Domain
SystemC
Domain
IDLE
FINISH
TRANSFER
δ1
δ2

Plain C++

- Only SystemC, no other 3rdparty libraries
- No verification language needed

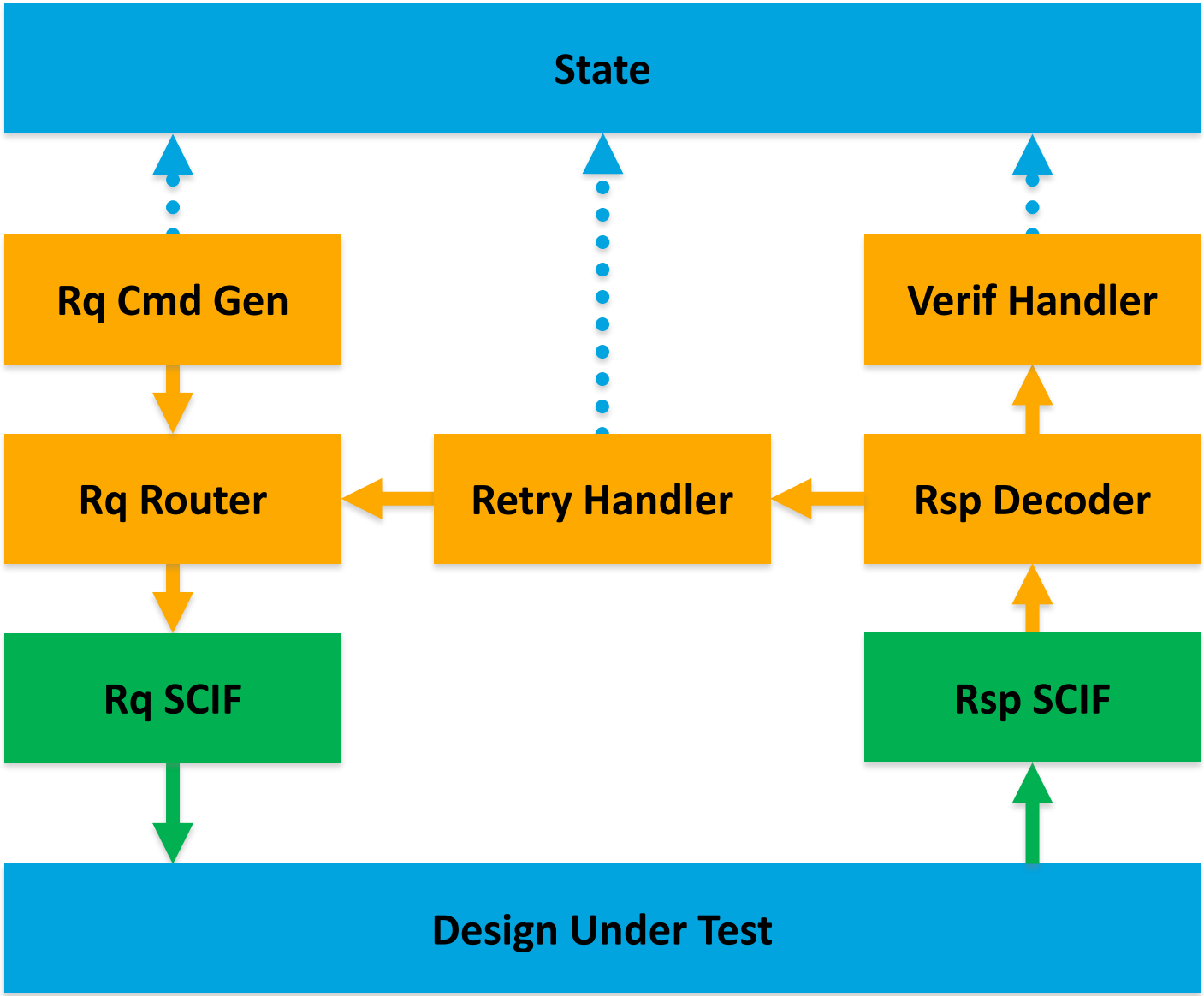
Components can be re-used

- Write tests quickly with quality
- Avoids code duplication, enables code reuse
- “Autoconnect”: no members in headers needed

A test produces a single executable

- Fast simulation
- Multicore CPU: multiple tests/seeds in parallel

EXAMPLE


State
Rq Cmd Gen
Verif Handler
Rq Router
Retry Handler
Rsp Decoder
Rq SCIF
Rsp SCIF
Design Under Test

PERFORMANCE

- With optimal settings, Verilator closely follows performance of commercial tools
- Time to compile can be an issue
- Single executable, can be run in parallel, i.e. many seeds in parallel

NUMBERS

	LoC	Coverage*
Module A	4655	92.0%
Module B	3668	92.0%
System	31250	88.6%

*Line toggle coverage

THANK YOU.

FAST CONTACT: SEBASTIAN@SWARM64.COM