Identifying and Overcoming Multi-Die System Verification Challenges

Varun Agrawal
Product Manager
Synopsys

accellera SYSTEMS INITIATIVE™

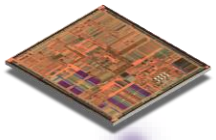
SYNOPSYS®

2023 DESIGN AND VERIFICATION™ DVCON CONFERENCE AND EXHIBITION INDIA

For the last 50 years the semiconductor industry has been driven by **Moore's Law**

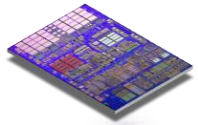
The number of transistors in a dense IC doubles about every two years

1996
Intel Pentium
Deschutes
7.5M
Transistors



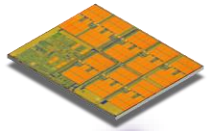
250nm

2003
IBM PowerPC 970
58M
Transistors



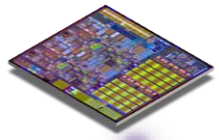
90nm

2007
Intel XEON 5400
820M
Transistors



45nm

2010
Intel Westmere-EX
2.6B
Transistors

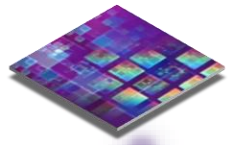


32/28nm

2018
Apple A12 Bionic
6.9B
Transistors

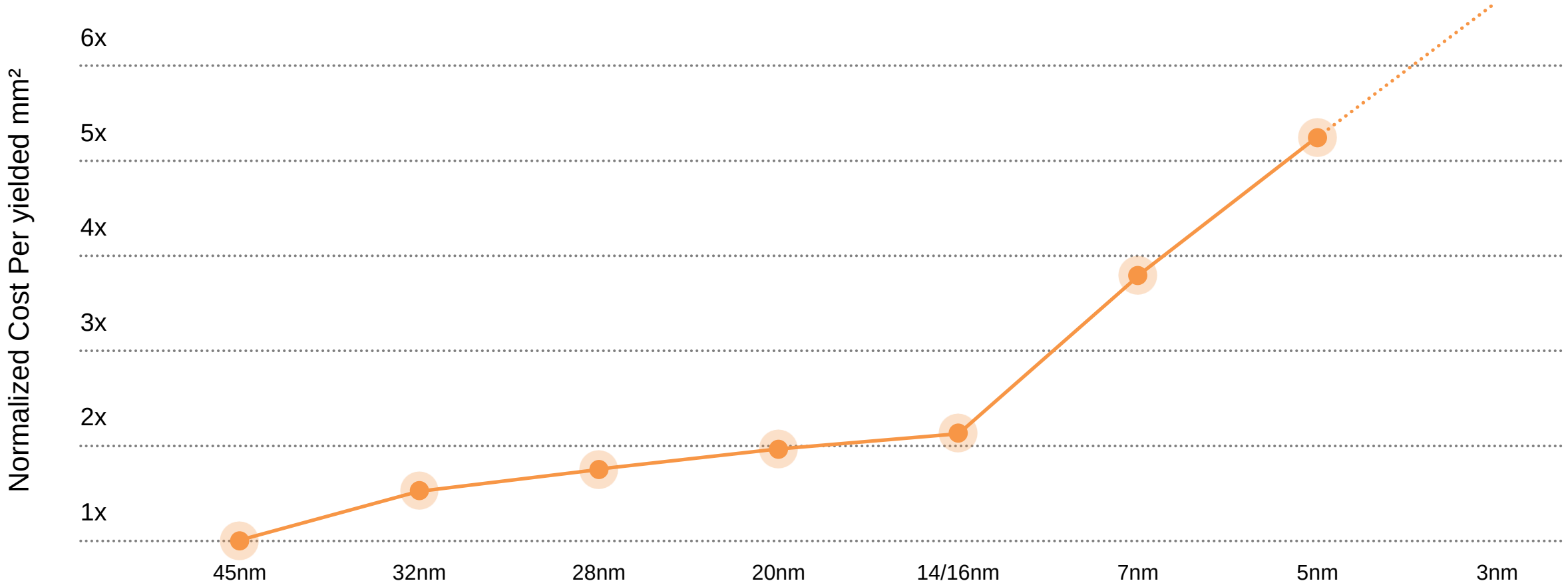
7nm

2022
Apple A16 Bionic
15B
Transistors



3nm

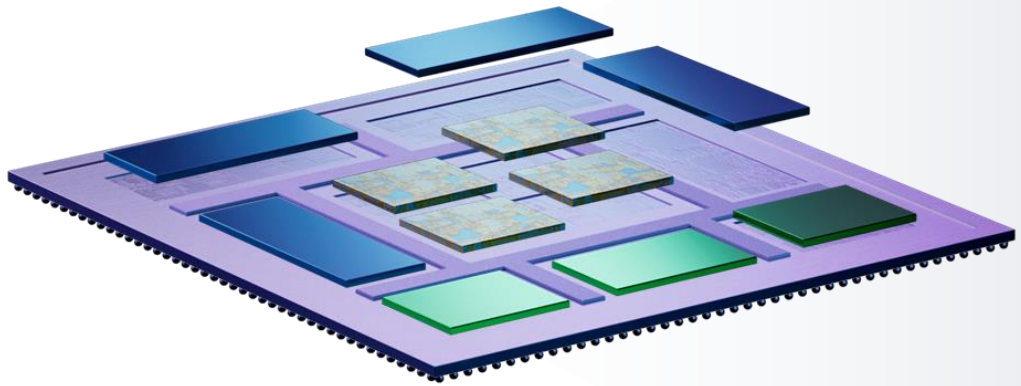
Reaching the End of Economic Viability



Silicon cost per yielded mm² for a 250mm² die

Source: Techsearch, AMD

The Drive to Multi-Die Systems



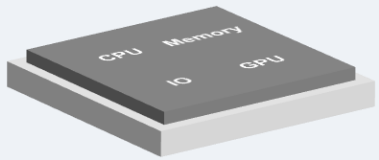
Motivation for Multi-Die Systems

- Accelerated scaling of system functionality at a cost-effective price (>2X reticle limits)
- Reduced risk & time-to-market by re-using proven designs/die
- Lower system power while increasing throughput (up to 30%)
- Rapid creation of new product variants for flexible portfolio management

Multi-Design System Challenges

Transformation from Monolithic SoC to Multi-Die Systems

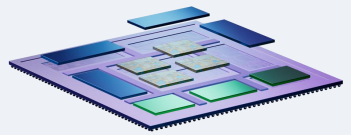
Monolithic SoC



Challenges

Heterogeneous system integration	Power & thermal management	System pathfinding
Die-to-die connectivity	Software development & modeling	Capacity/performance for system verification
System signoff analysis	Hierarchical Test & Repair	Reliability & Security
System yield	Memory utilization & coherency	

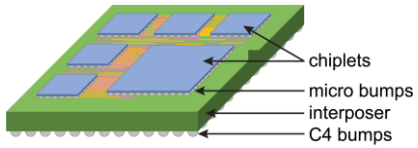
Multi-Die Systems



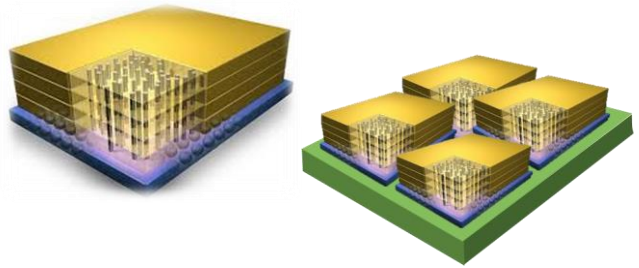
From Moore's Law to SysMoore

Evolving Trend

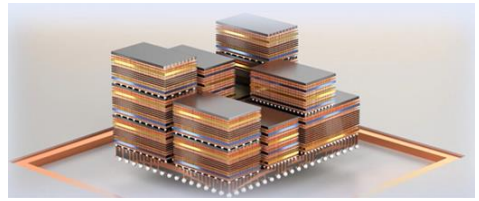
- 2.5D: Interposer-mounted chiplets



- 3D stack(s) - regular structures (memory, FPGA, …)



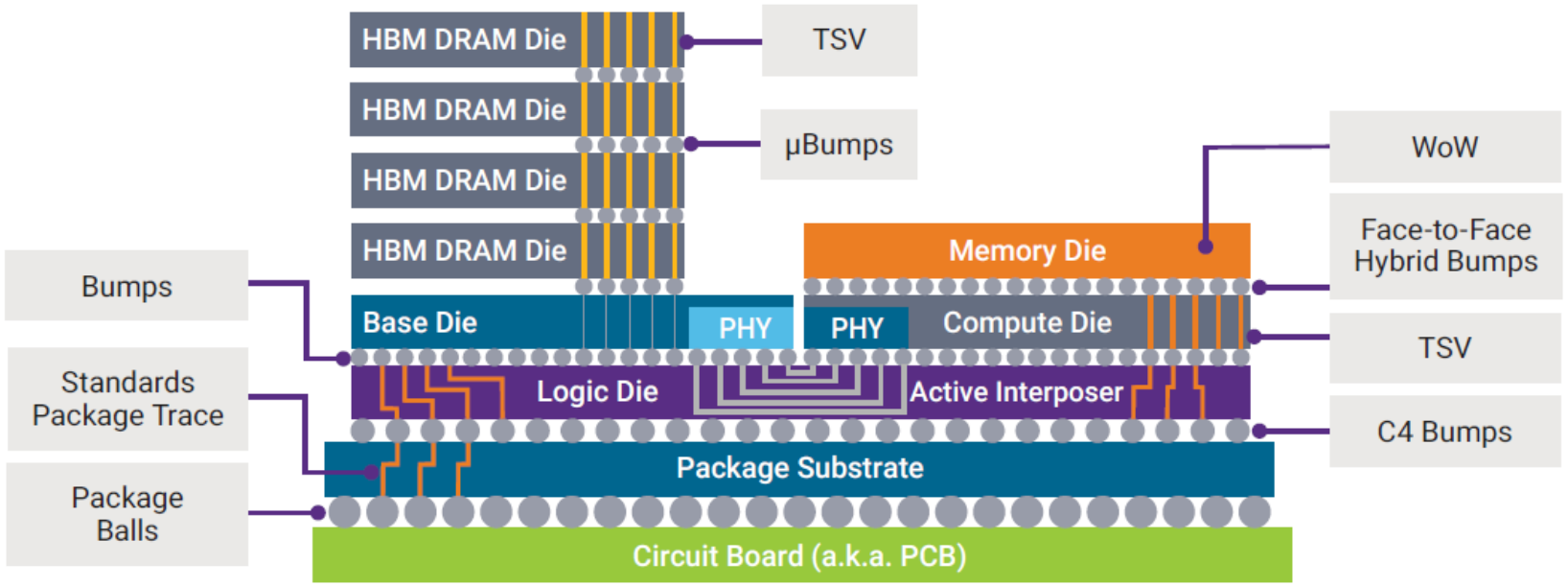
- Heterogeneous stacks mounted on interposers / bridges
- Recursive composition formulation … stacks of stacks
  - Ever larger memory and computation
  - Partition system to balance throughput and energy (EDP)
  - Tackle "Memory Wall" - Time and energy moving data to/from large off-chip memory
    - New Mantra: Move the computation, not the data → then hide the latency



Fundamental challenge = **Size**, allows building designs (already) extreme large!

Is Multi-Die System Verification Harder?

Still Need to Verify the Functionality of the Overall System



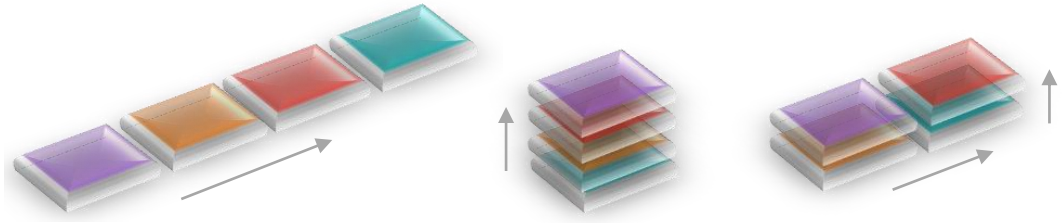
New components:

- Bumps
- µBumps
- C4 Bumps
- TSVs
- Interposers
- Interconnect Bridges
- Wafer on Wafer

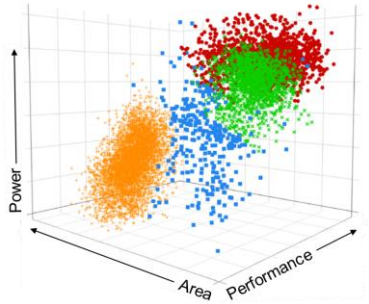
  - Mostly impacts physical layer (dies, routing, power dissipation, packaging, …)

- Increased scale and complexity exacerbates verification
- Fundamentally alters the **incremental refinement** design flow

A System-Approach is Required



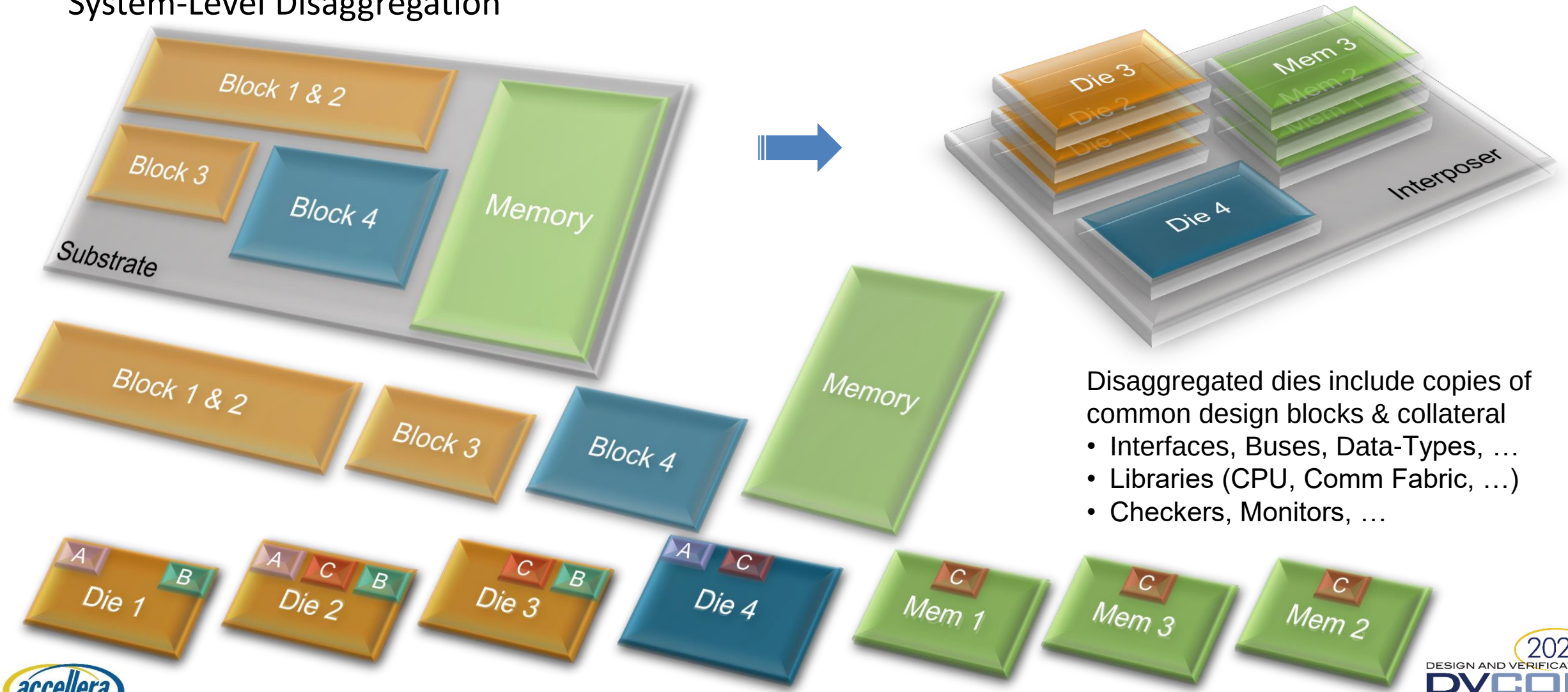
- Many organizations become possible
- Must be modeled, laid-out & verified in the context of the system
  - Design aspects and optimizations become architectural decisions
    - Tiling and placement, die-to-die communication, power, thermal, …
  - A lot of work must be done early on: Shift Left
- Need a framework for end-to-end co-exploration and co-optimization of technologies, architectures, and algorithms
  - Architectural exploration to quickly estimate PPA for a range of workloads



- Flow:
  - Full system spec → Design (& verify) individual blocks → Assemble system → Verify as a whole
  - Modularized approach – Akin to board-level verification

From Monolithic To Multi-Die System

System-Level Disaggregation



Disaggregated dies include copies of common design blocks & collateral

- Interfaces, Buses, Data-Types, …
- Libraries (CPU, Comm Fabric, …)
- Checkers, Monitors, …

Multi-Die System Verification Challenges

- System verification must validate assumptions made during architecture design
  - Must consider die-to-die communication: delay, jitter, coherency, power, guaranteed delivery and errors
    - Monolithic SoCs only consider delay
- Design size and complexity exacerbate Verification
  - Need adequate levels of capacity and performance
    - Hybrid models and traffic generators to focus on a few dies at a time
    - Very large memories bottlenecks
  - Scalability of simulation/emulation models
    - Include analog components
  - Scalable system integration methodology (system aggregation)
- Knowing when Verification is complete
  - Exhaustive verification of individual dies – Complete functional coverage (UVM)
    - Die-level bugs cannot be fixed at the system level
  - System level verification focus on scenarios – Explicit coverage model (PSS) plus SW/FW
    - Ensure data arrives in the right place and with the expected throughput and latency

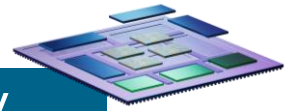
Functional Verification of Multi-Die Systems

- A Multi-Die system is not one design, it is a combination of independently manufactured designs (dies) interconnected through communication fabric(s)
- Multi-Die system-level verification needs to target
  - Complex functions spanning multiple dies
  - Performance that is a function of multi-die functionality
  - Closer to system validation of multi-chip solutions – Verify functional scenarios
- Basic functional test: Assemble and simulate the RTL of all the dies
  - How does one assemble “independent” designs into one simulation environment
    - Compile Issues: How to avoid name clashes
    - Capacity implications: Does compute server have enough memory to build and execute the simulation
  - Can the die-level testbenches be re-used and/or synchronized?
  - Can the simulation be distributed over multiple servers?

Assembling Multi-Die System for Simulation

Single Executable to Simulate System Aggregation

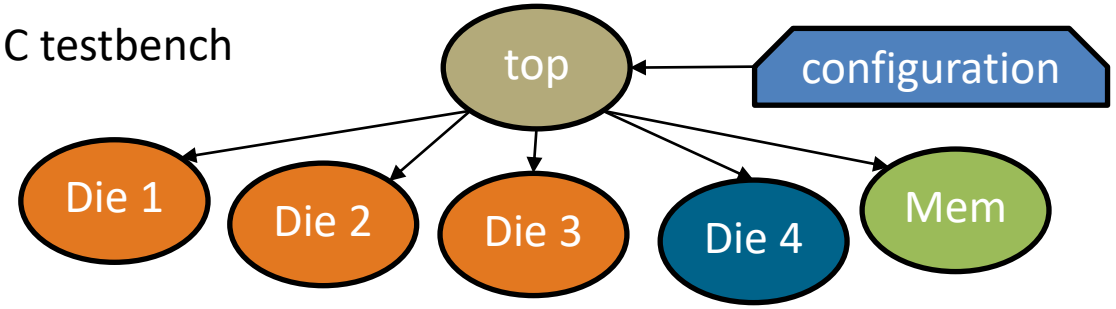
- Compiling all dies will trigger name clashes



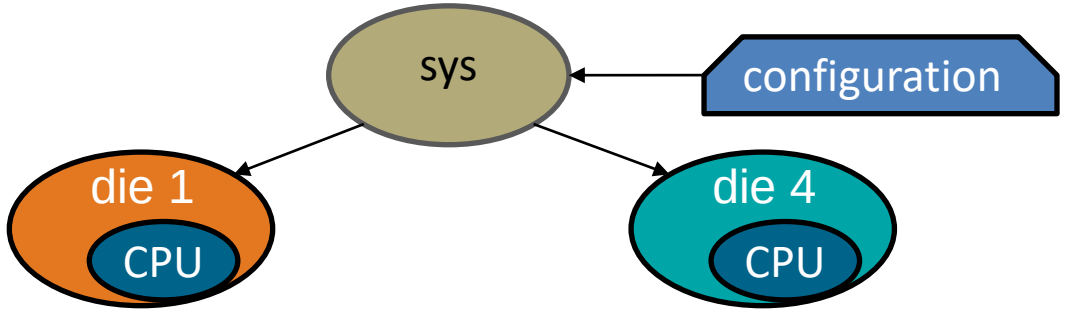
```
vcs die1.v die2.v die3.v …
```

→ A, B, C Redefined!

- Analyze each die into a separate library
  - Same names (e.g., module) may be used in multiple dies – No name clash
  - Top-level **assembly** and **configuration** files
    - No changes to per-die code
  - The multi-die system's name scope is strictly hierarchical
    - Extra care needed for C testbench



Single Executable Multi-Die System Simulation with Synopsys VCS Functional Verification Solution



die_1

```
// Target Library : DIE_1

module core();
   CPU cpu0();
endmodule

module checker();
endmodule

bind core checker ck0();

module CPU();
endmodule
```

die_4

```
// Target Library: DIE_4

module core();
   CPU cpu0();
   CPU cpu1();
endmodule

module checker();
endmodule

bind core checker ck0();

module CPU();
endmodule
```

sys

```
module sys();
   core core0();
   core core1();
endmodule
```

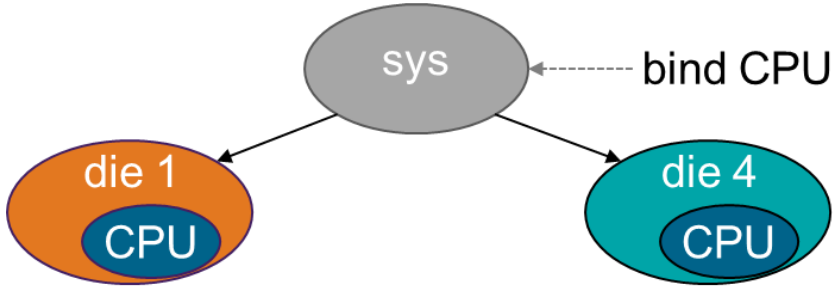
cfg

```
config cfg;
   design work.sys;
   default liblist work DIE_1 DIE_4;

   instance sys.core0 liblist DIE_1;

   instance sys.core1 liblist DIE_4;
endconfig
```

```
vlogan -work DIE_1 die_1.v
vlogan -work DIE_4 die_4.v
vlogan sys.v cfg.v
vcs cfg
```

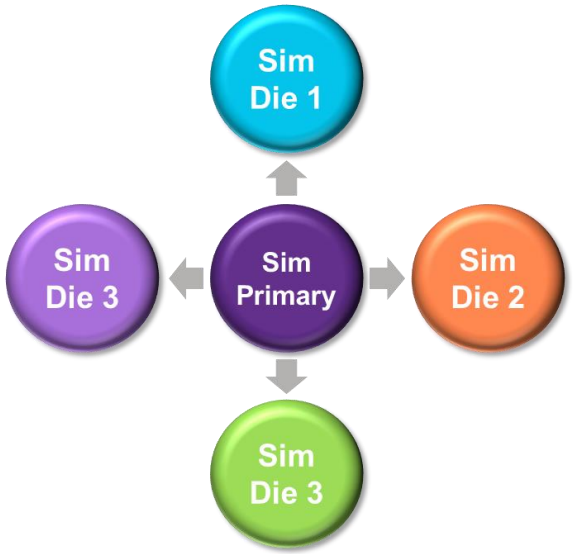


No bind to modules in sys

Which CPU? die_1 or die_4
Ambiguous → Error

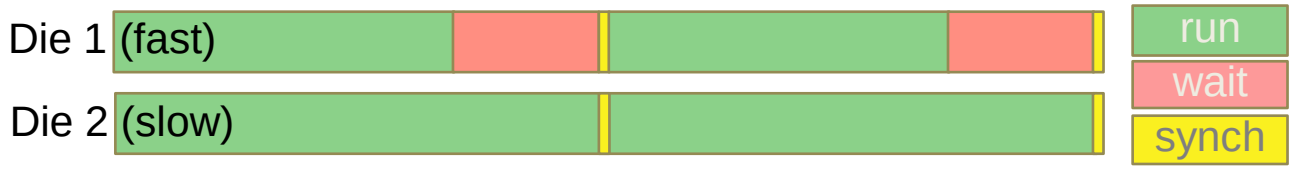
Distributed Simulation of Multi-Die Systems

Addressing Verification Capacity and Scalability

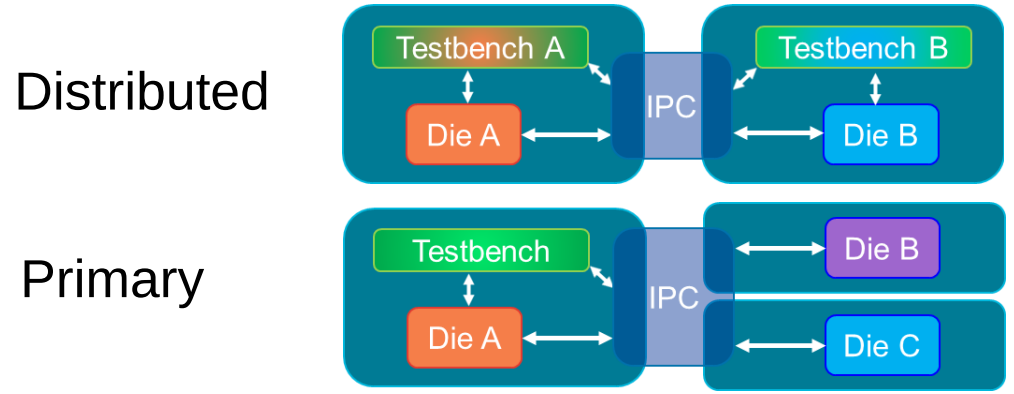


- Separate compilation – No clashes
- Concurrent execution
- Client – Server architecture (-distsim)
  - Exploits cloud elasticity & scalability
- Communication & synchronization APIs
- Configurable connectivity

- Asynchronous distributed execution
  - Kept in lock step with infrequent synch-points
    - Die-to-die communication uses SerDes
  - Breaks Von Nuemann bottleneck



- Multiple testbench modes



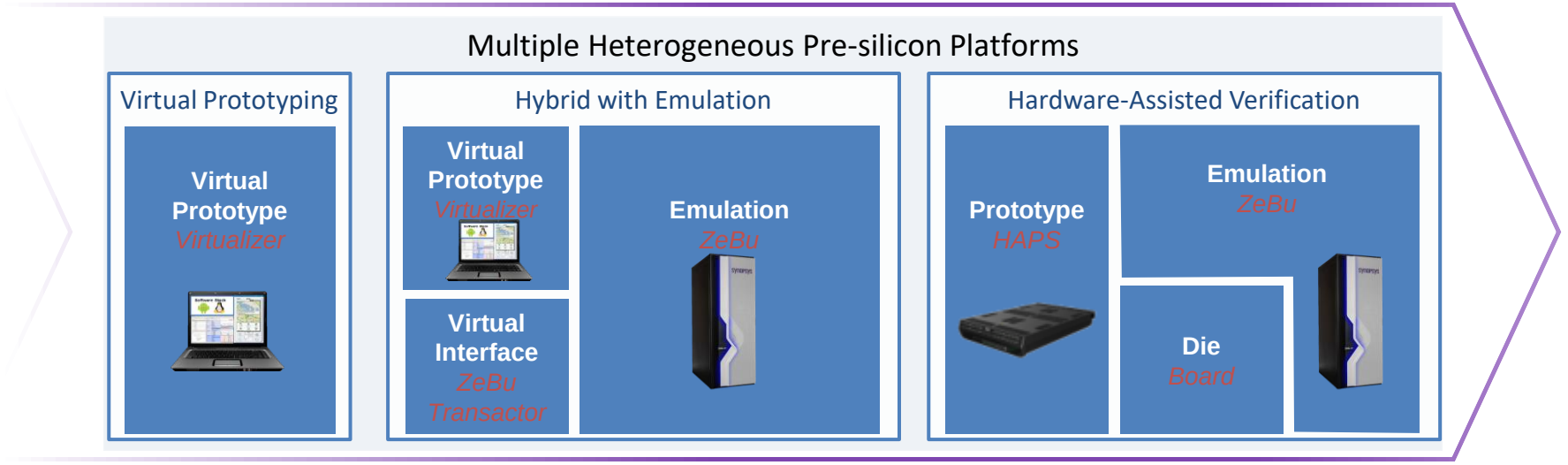
Software Development & System Validation

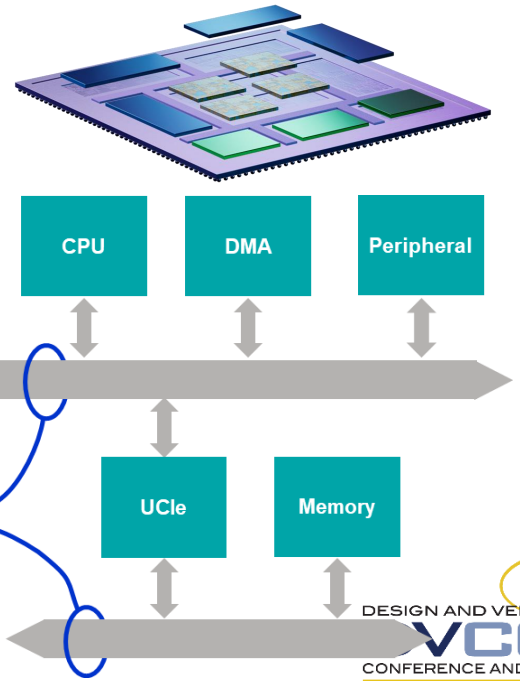
Hybrid Solutions, Approximate Models, and Traffic Generators

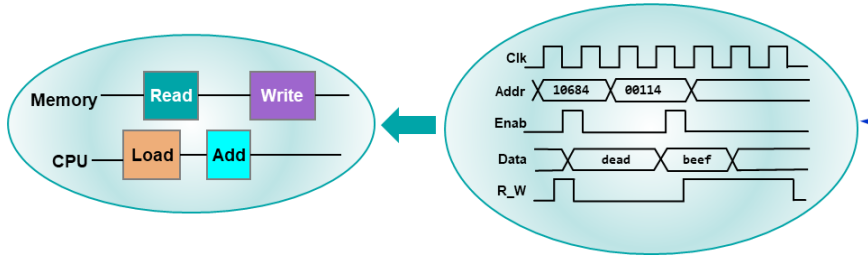
SOFTWARE BRING-UP
Early die virtual models for SW teams

SPEED & SCALABILITY
Emulation of 10s billions gates

SPEEDUP BRING-UP TIME
Models, transactors, speed adapters







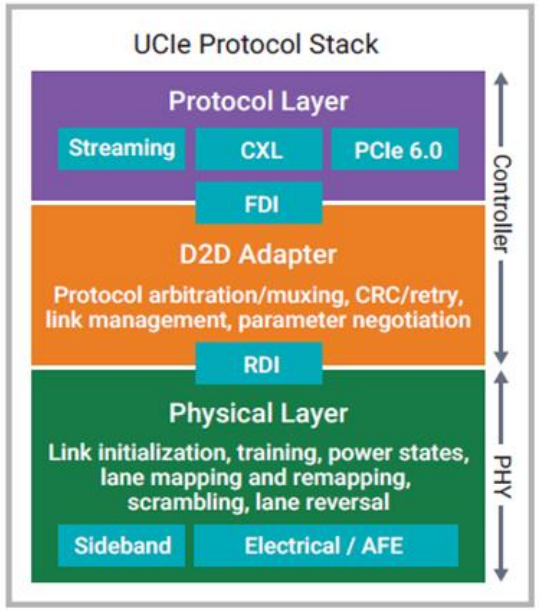
- Transaction level debug

accellera SYSTEMS INITIATIVE™

2023 DESIGN AND VERIFICATION™ VCON CONFERENCE AND EXHIBITION INDIA

Die-to-Die Communication Verification

- Need to consider delay, jitter, coherency, power, guaranteed delivery and errors
  - Generally, very difficult: Depends technologies (bumps, TSV, wires) plus system routing
- Standard die-to-die interfaces can help
  - AIB, BoW, OHBI, UCIe, XSR supporting a variety of use cases and speeds of 6-32 Gbps/pin



UCIe Controller IP

Low latency bridge for NoCs in a multi-die system

Optional security engine with encryption

UCIe PHY IP

Low latency & power for interposer & bridge designs

Extensive test & debug features for KGD

UCIe Verification IP

Scalable architecture supports verification of many DUT types

Built-in protocol checks and functional coverage at all layers

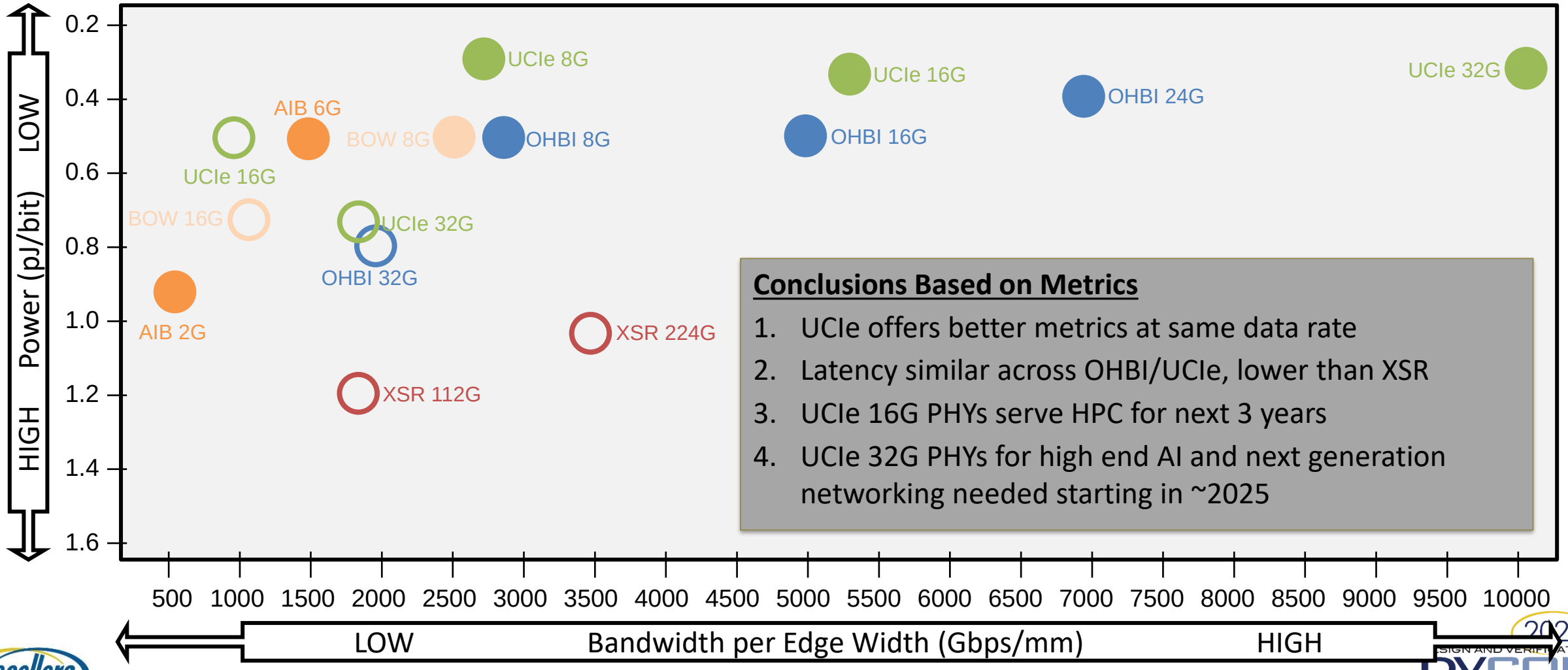
- Pick the right protocol (use-case) & verify with VIP using Synopsys VCS or ZeBu

Summary of 5 Main D2D Standards

Organic Substrate $

Interposer $$$

Key Figures of Merit: Technical (Bandwidth, Power, Latency) & Cost



Conclusions Based on Metrics

1. UCIe offers better metrics at same data rate
2. Latency similar across OHBI/UCIe, lower than XSR
3. UCIe 16G PHYs serve HPC for next 3 years
4. UCIe 32G PHYs for high end AI and next generation networking needed starting in ~2025

Why is UCIe a Preferred D2D Interface?

Technical Merits, Comprehensive Spec & Broad Eco-System

- Technical Merits (Most compelling PPAs)
  - Energy efficiency <0.3pJ/Bit
  - Edge efficiency >5Tbps/mm
  - Latency ~2ns from FDI to FDI
- Comprehensive & Futureproof
  - All use cases
  - All package types
  - Chip to Chip use case with retimer
  - Complete protocol stack
  - Future proof with support up to 32Gbps data rate per pin
- Broad Ecosystem
  - Wide range of promoters & contributors spanning all industry segments

> 110 members (June 2023)

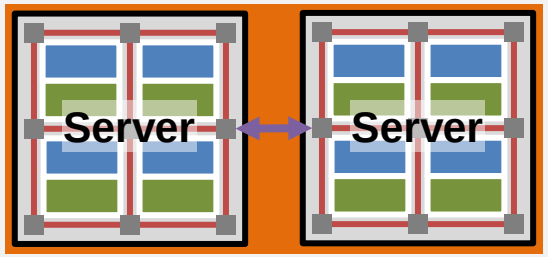
accellera SYSTEMS INITIATIVE™

UCIe Built to Enable Common and New Use Models

UCIe Retimer Enables Extension of Reach Beyond Rack (with Optical IO)

Server or AI Scaling (Homogeneous)

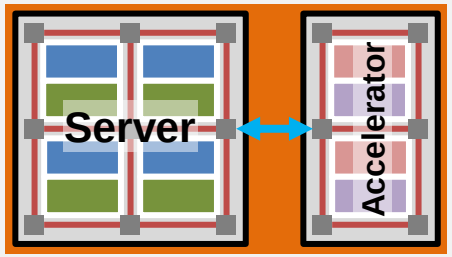
(NoC-to-NoC with low latency & coherency)



UCIe Streaming
- CXS or AXI bridge
- User Defined / Proprietary

Heterogeneous Computing (Accelerator)

(Interoperability with low latency & coherency)

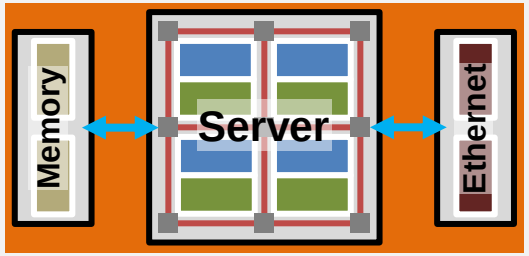


UCIe CXL or PCIe
- For interoperability

UCIe Streaming
- CXS or AXI bridge

IO or Memory Split

(Interoperability with low latency)

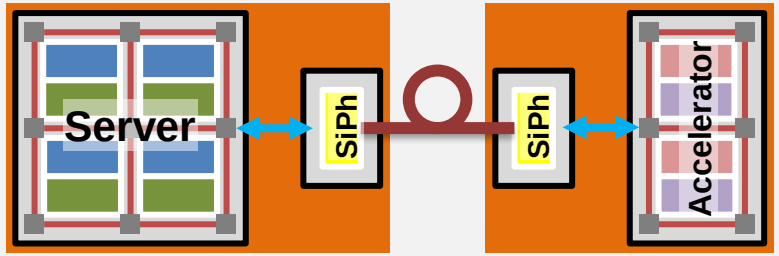


UCIe CXL or PCIe6
- Coherency for Memory

UCIe Streaming
- AXI bridge for Ethernet

Resource Aggregation (Pooling) w/Retimer

(Rack-to-Rack with low latency)



UCIe CXL or PCIe6
- For Aggregation

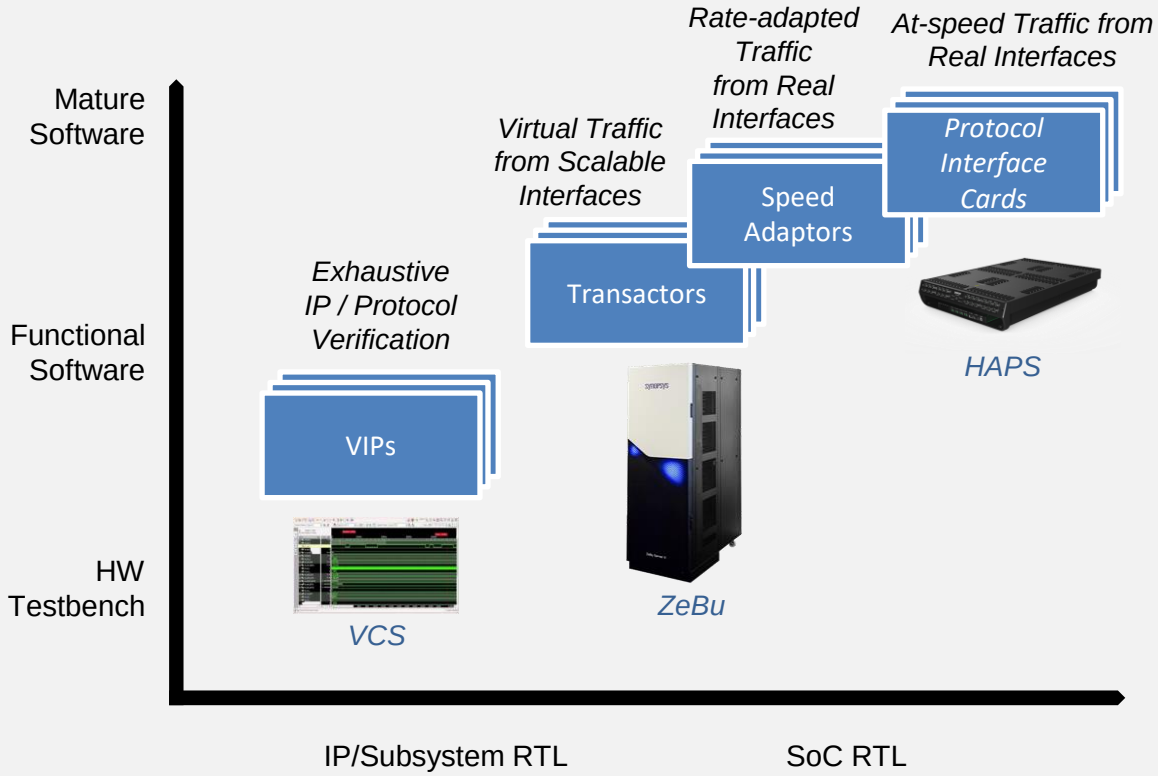
UCIe Streaming
- For CPO / Eth.

Broad Protocol Solution Across Verification Use Cases

PCIe 6.0, CXL 3.0, USB 4, HBM3, UCIe, …

Industry-first protocol solutions leverage Synopsys IP

Protocol Validation using Physical and Virtual testers

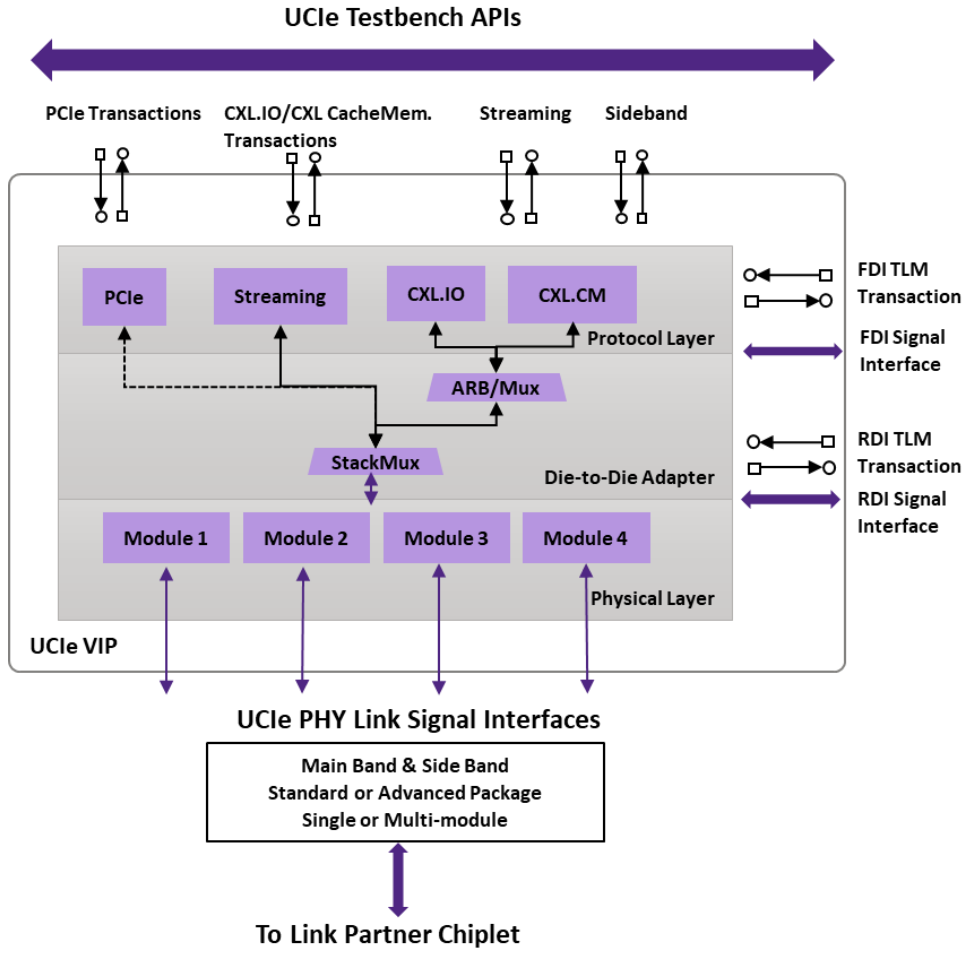


Synopsys VIP for UCIe

Architecture and Key Features

Verification IP Features

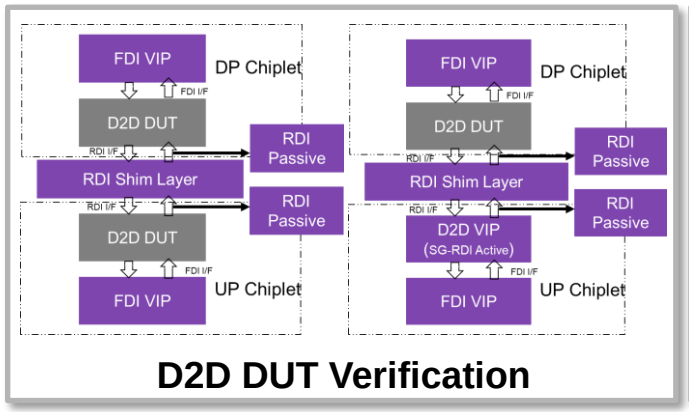
- Native SV/UVM Architecture
- Specification Version – UCIe v1.0, v1.1
- Interfaces: FDI/RDI/Logical PHY link
- Supports all topologies for various DUT types
- Operation Modes – Active, Passive
- Streaming testbench interfaces for easy traffic generation
- RDI Shim layer at D2D adapter for early test development
- APIs for traffic generation and sideband service requests
- Protocol checks and functional coverage at each layer
- Exceptions, Callback and Analysis ports for Scoreboard
- Configurable interpacket delays for mainband and sideband packets
- Reference example illustrating API usage and representative protocol scenarios



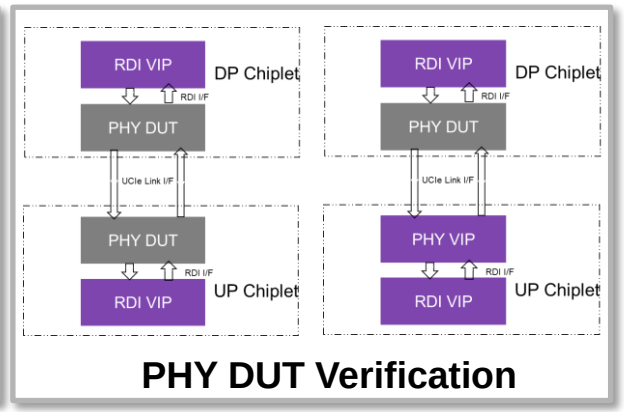
Close Collaboration with UCIe Consortium, Industry leading Synopsys IP and Key market leaders

Synopsys UCIe VIPs for all DUTs

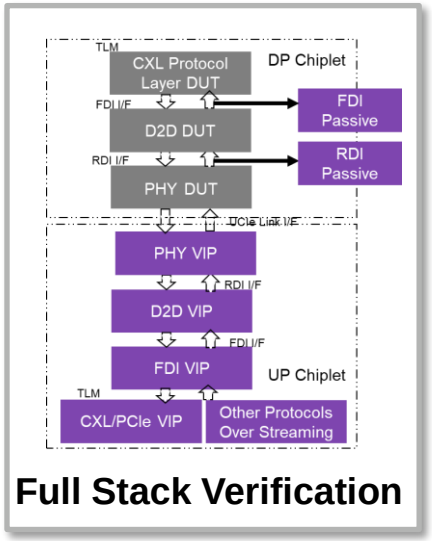
Supports All use-cases and Topologies



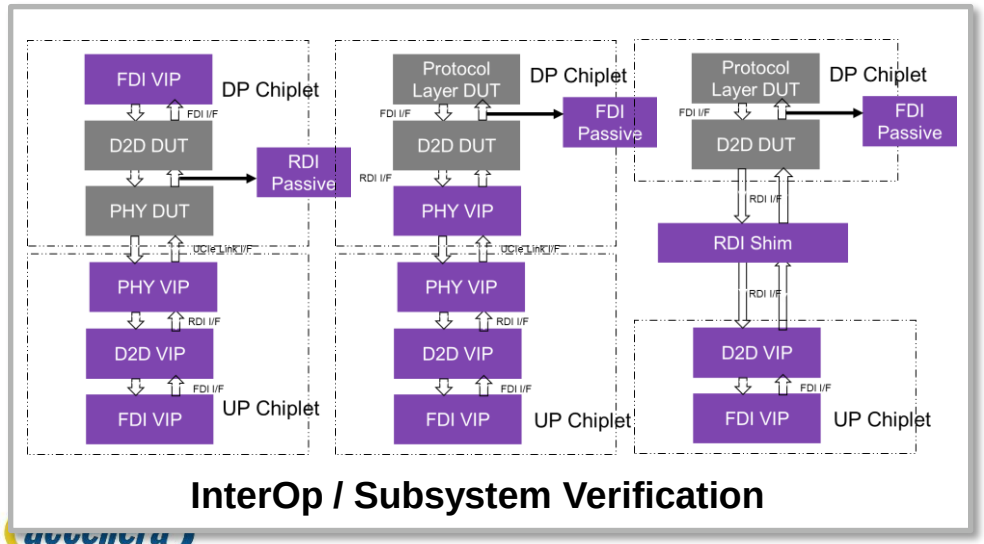
D2D DUT Verification



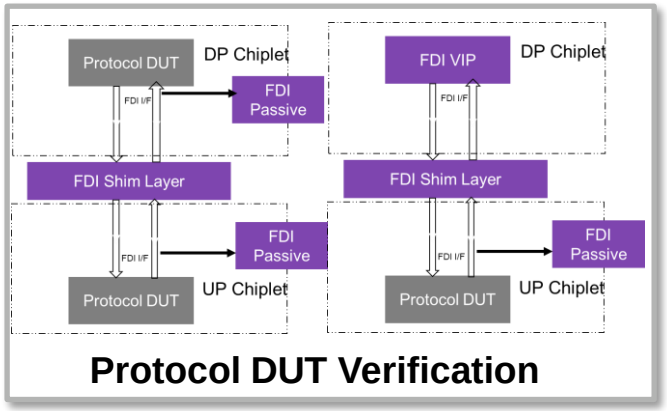
PHY DUT Verification



Full Stack Verification



InterOp / Subsystem Verification



Protocol DUT Verification

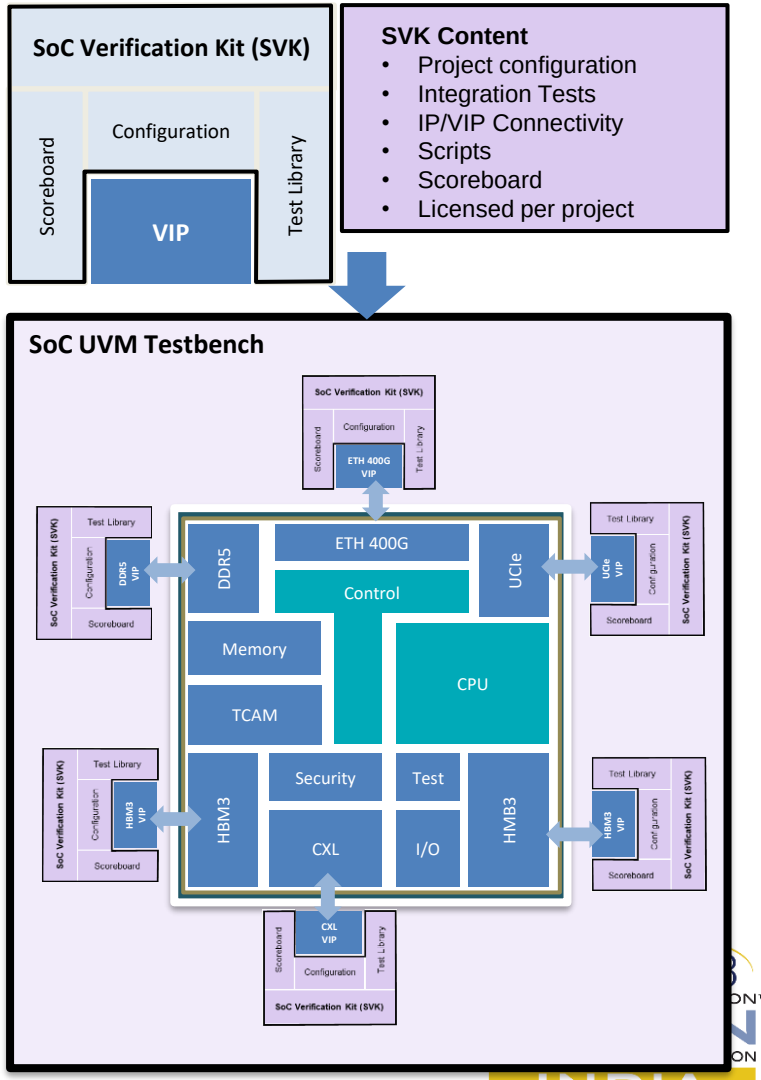
- Supports 11+ Verification Topologies for various Design types
- Supported Use-cases
  - D2D / PHY / Protocol Layer DUT
  - D2D-PHY Interoperability
  - Protocol-layer-D2D-Controller Interoperability
  - UCIe Subsystem
  - UCIe Full Stack
- Support hooks for Software Discovery and Device Enumeration
- APIs for Configuration Space Testing
- Single and Multi-node Setups
- Developed alongside Synopsys IPs and with leading customers

Synopsys SoC Verification Kit (SVK)

Accelerating SoC Verification with Synopsys IP, VIP and VCS

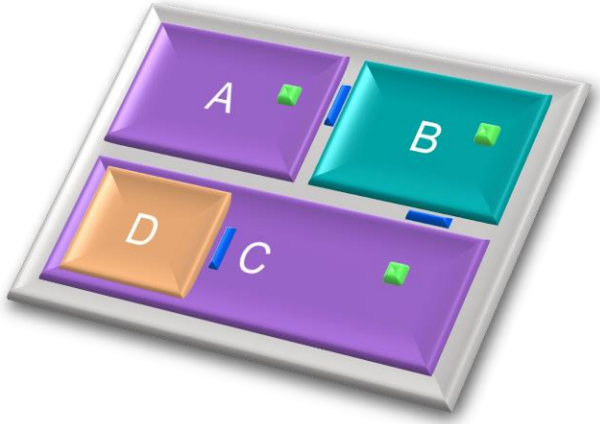
- Challenges
  - Expertise for UVM-based, scalable testbenches
  - Verification resource limitations

- Key Benefits
  - Out-of-the-box verification solution for complex protocols, tailor made for specific IP configuration
  - Accelerates the SoC testbench development for design and verification engineers
  - Enables rapid integration and testing of Synopsys IPs in Subsystem/SoC environment
  - Mitigates integration risk by providing industry-proven verification methodology by protocol experts

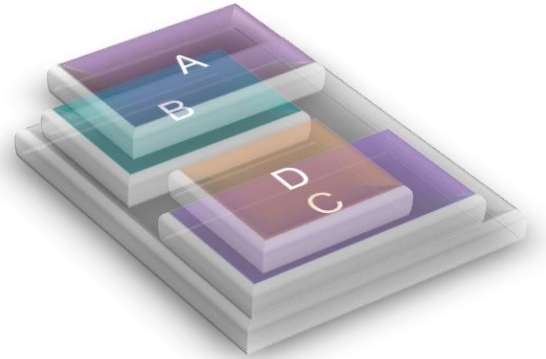


accellera SYSTEMS INITIATIVE™

Multi-Die System Power Intent Verification



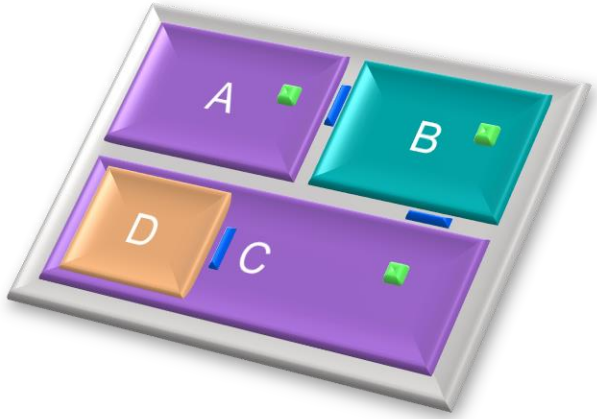
- UPF allows to overlay power intent over functional intent
- What about a Multi-Die system?



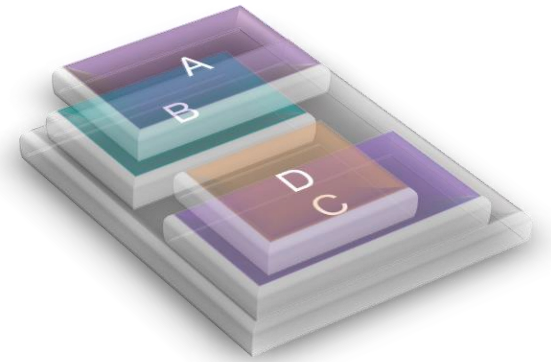
- Power Domain 1
- Power Domain 2
- Power Domain 3
- Isolation
- Retention

- That is not possible – pre-manufactured die!
- Power intent is part of the architecture : shift-left
  - Dies implement power intent - voltages & signaling ports
  - Power intent disaggregation (top-down → bottom-up flow)
- Assemble & connect power ports
  - Static verification (VC LP) can verify such connections
- System-level PST is challenging
  - Per-Die predefined “power modes” (off, standby, …)

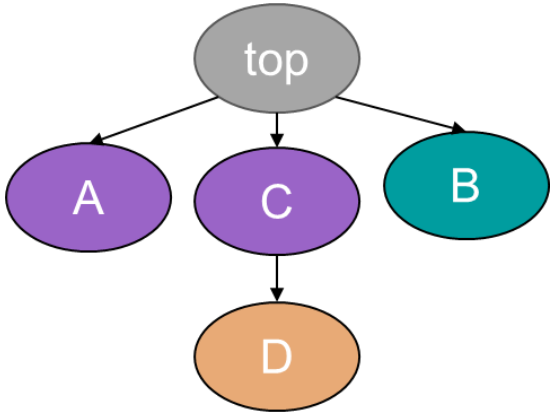
Multi-Die System Power Verification Challenges

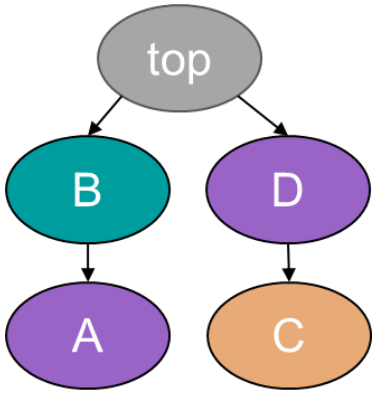


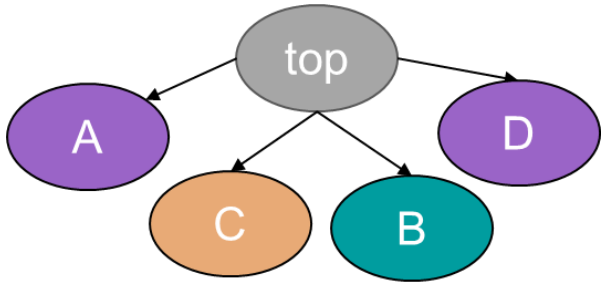
Power Domain 1
Power Domain 2
Power Domain 3
Isolation
Retention



- Verification tools will honor UPF targeting dies. No “set die”!
- Logical and physical hierarchies are not the same!

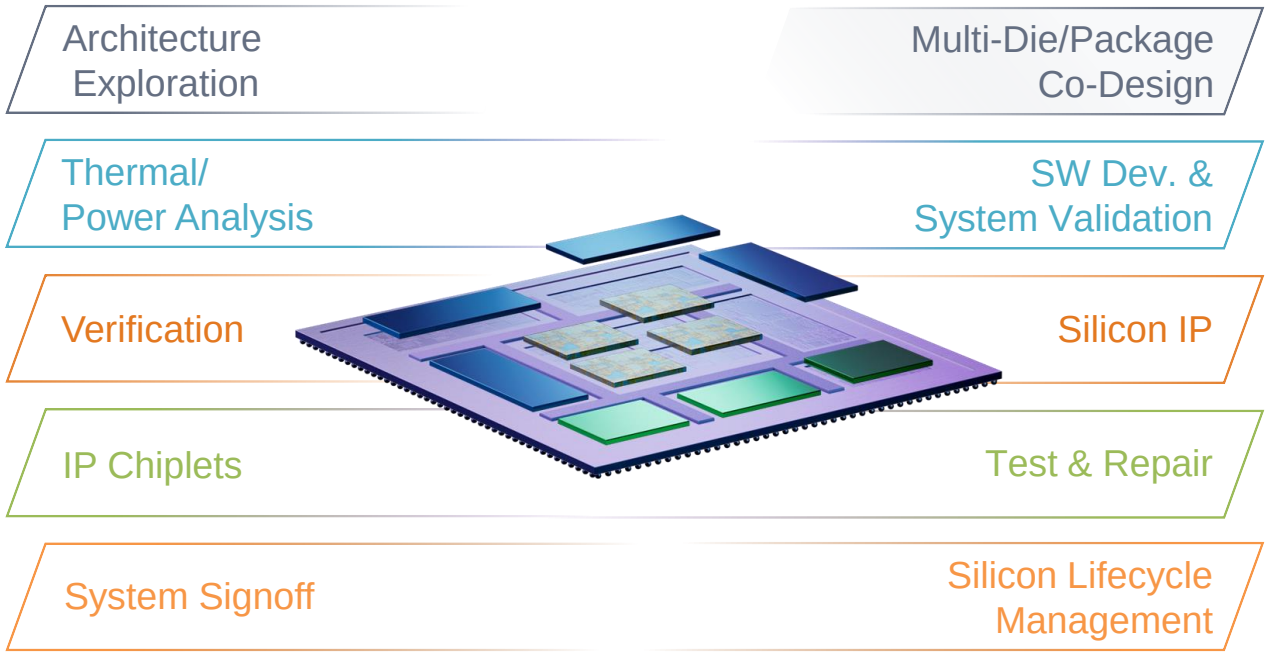






Synopsys Multi-Die System Solutions

A Comprehensive Solution for Heterogeneous Integration



Architecture Exploration	Optimize thermal, power, and performance with early exploration and partitioning
Software Dev. & Validation	Broadest IP and VIP portfolio Rapid software development and validation with high-capacity emulation & prototyping
Design Implementation	Efficient die/package co-design with unified exploration-to-signoff platform and robust IP
Manufacturing & Reliability	Improve health, security and reliability with holistic test and lifecycle management solutions

QUESTIONS?

Thank You

SYNOPSYS®