2023 DESIGN AND VERIFICATION™ DVCON CONFERENCE AND EXHIBITION EUROPE 10 YEAR ANNIVERSARY

# Towards a Hybrid Verification Environment for Signal Processing SoCs

Jan Hahlbeck, Steffen Löbel, Chandana G P

NXP Semiconductors Germany GmbH

NXP

accellera SYSTEMS INITIATIVE

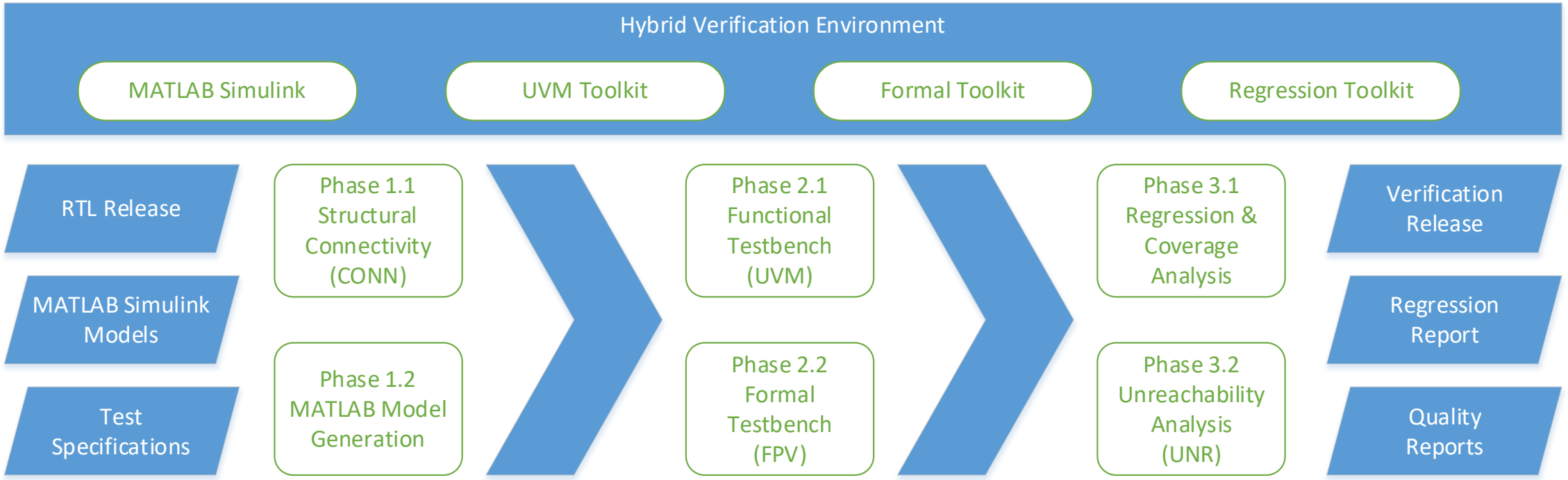
# Agenda

- Motivation – Why are we using a hybrid approach?
- Introduction to our Hybrid Verification Environment
  - Walking through the six phases of our verification cycle
- Apply Hybrid Verification approach on an exemplary DUT
  - Results
- Summary / Outlook

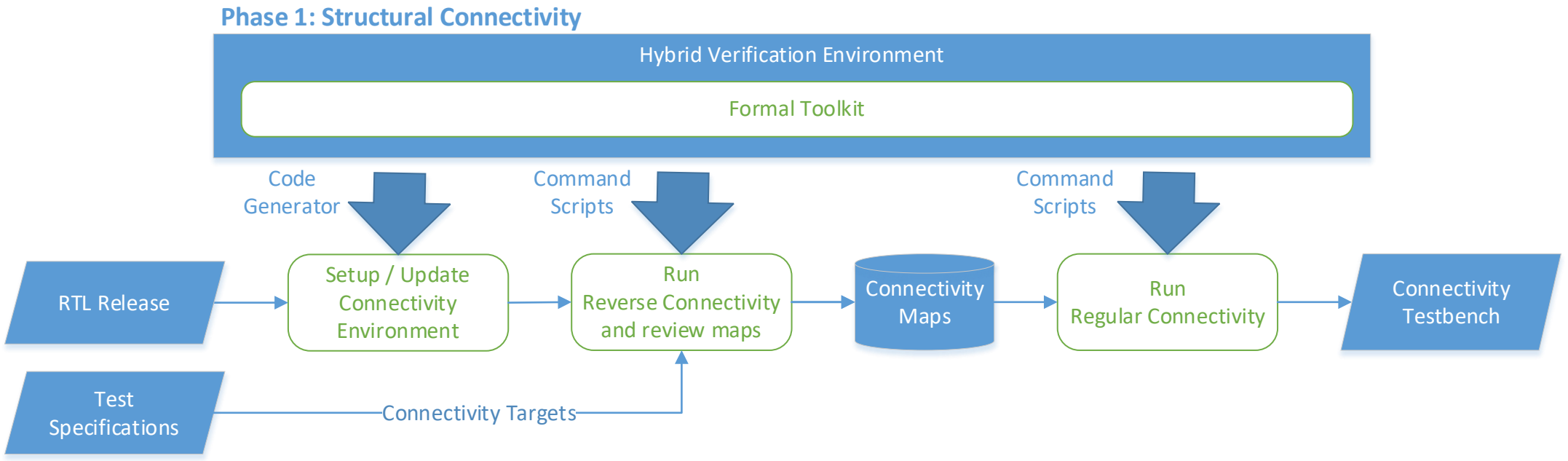
# Motivation

- Dealing with radio SoC verification in automotive domain
- Demand for combination of traditional functional verification flow with formal verification and MATLAB signal processing reference models -> Hybrid Verification
- Be prepared for further methodology shift
  - Formal Property Verification
  - Automated Formal Verification
  - Motivate engineers to try out formal related apps and techniques
- Tackle verification tasks from different angles
- Verification engineers sleep better with a formal proof!

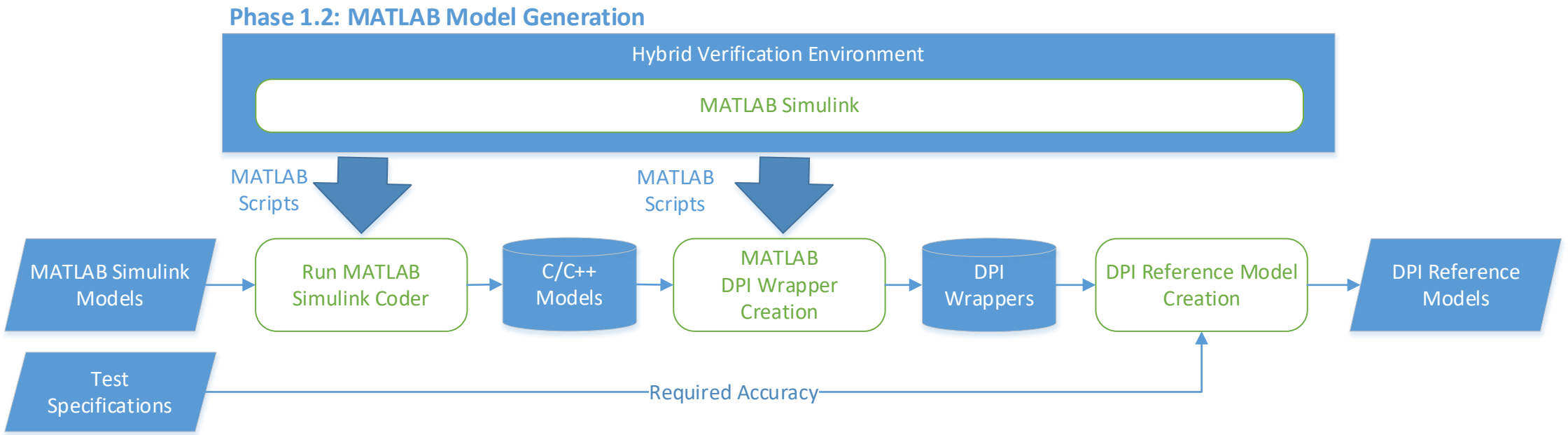
# Hybrid Verification Environment - Overview



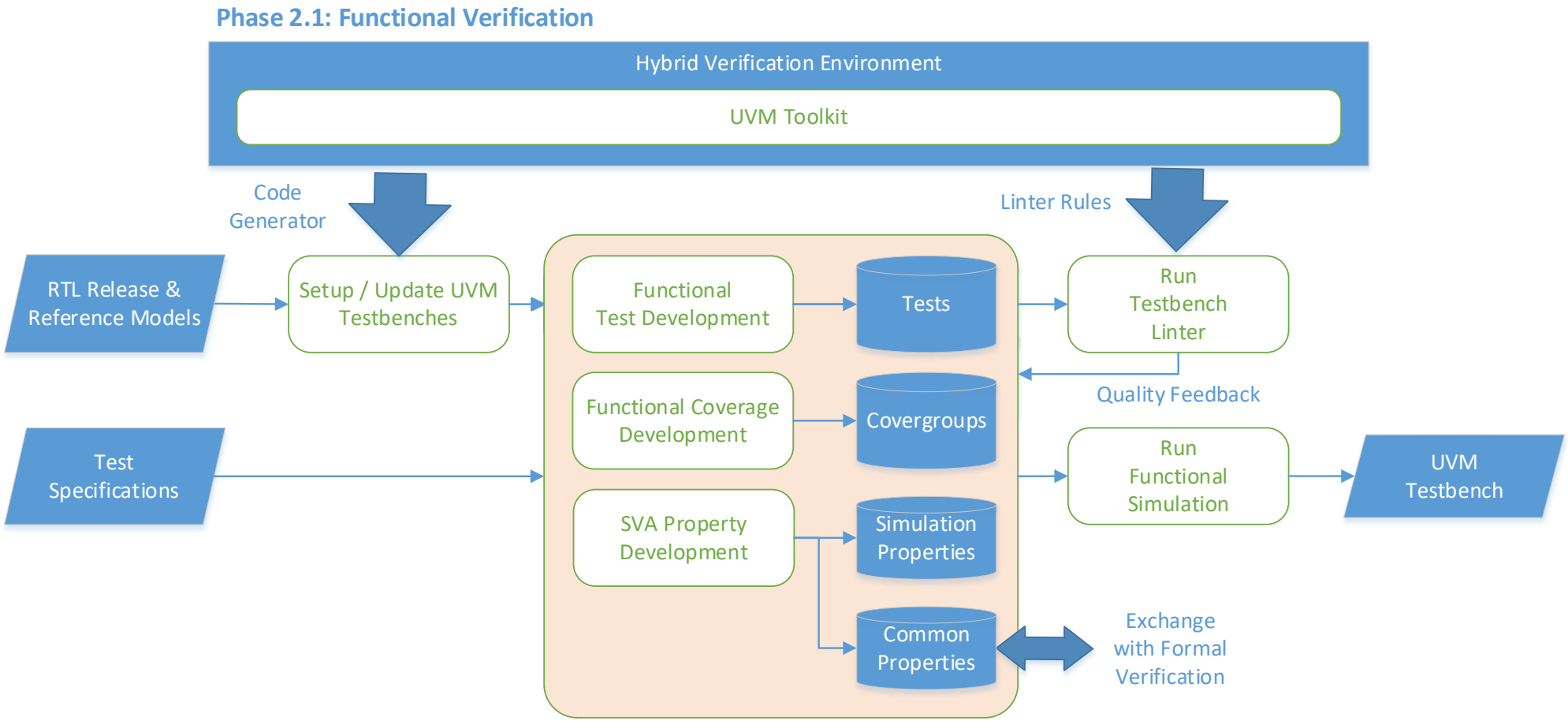
# Phase 1.1: Structural Connectivity



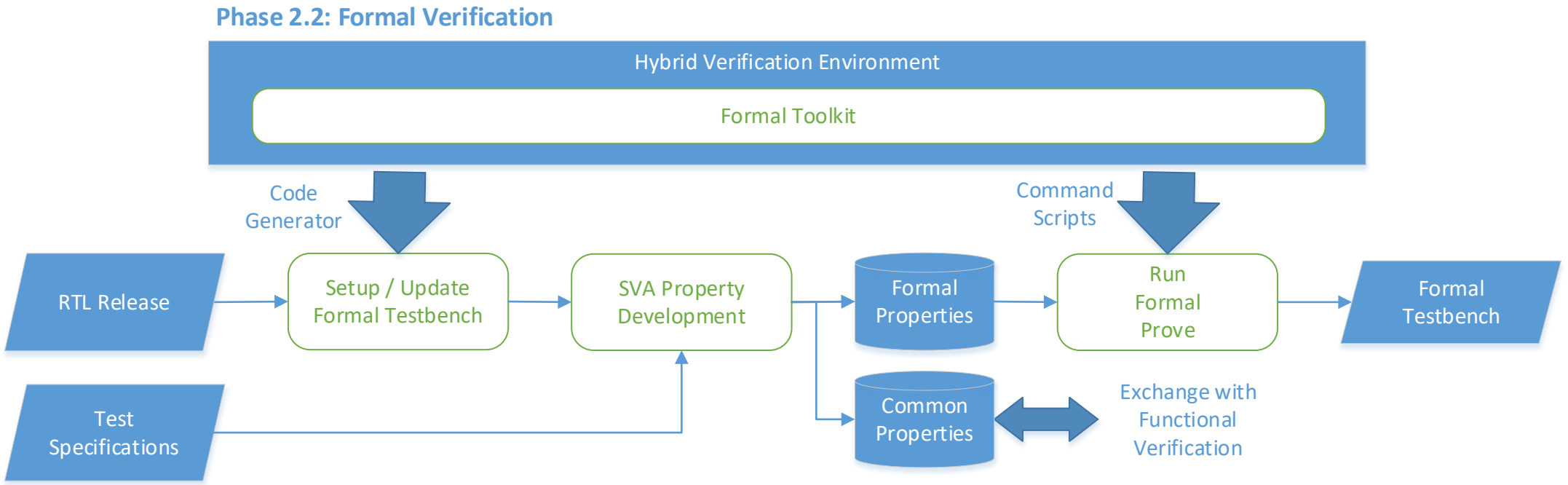
# Phase 1.2: MATLAB Model Generation



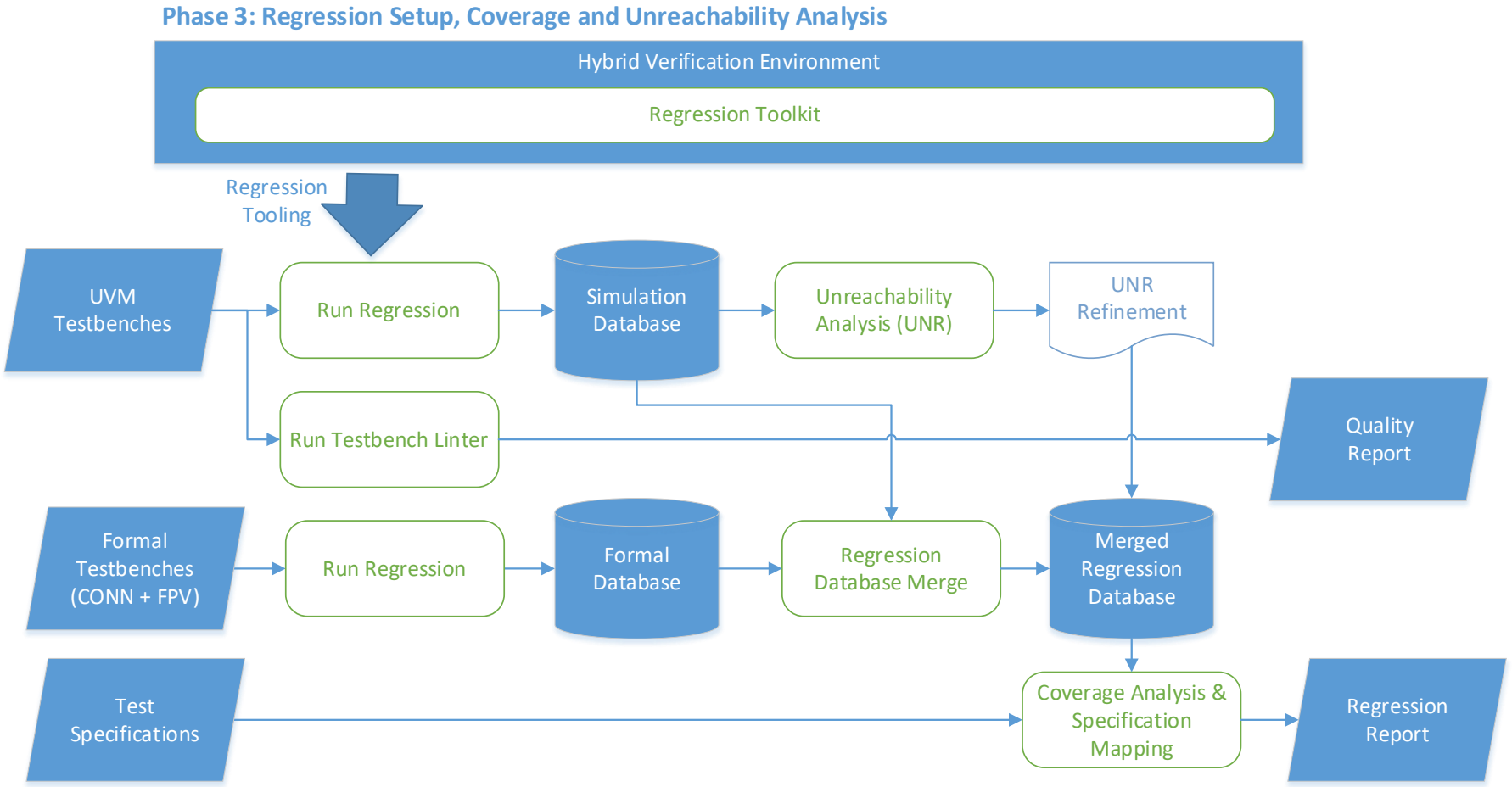
# Phase 2.1: Functional Verification



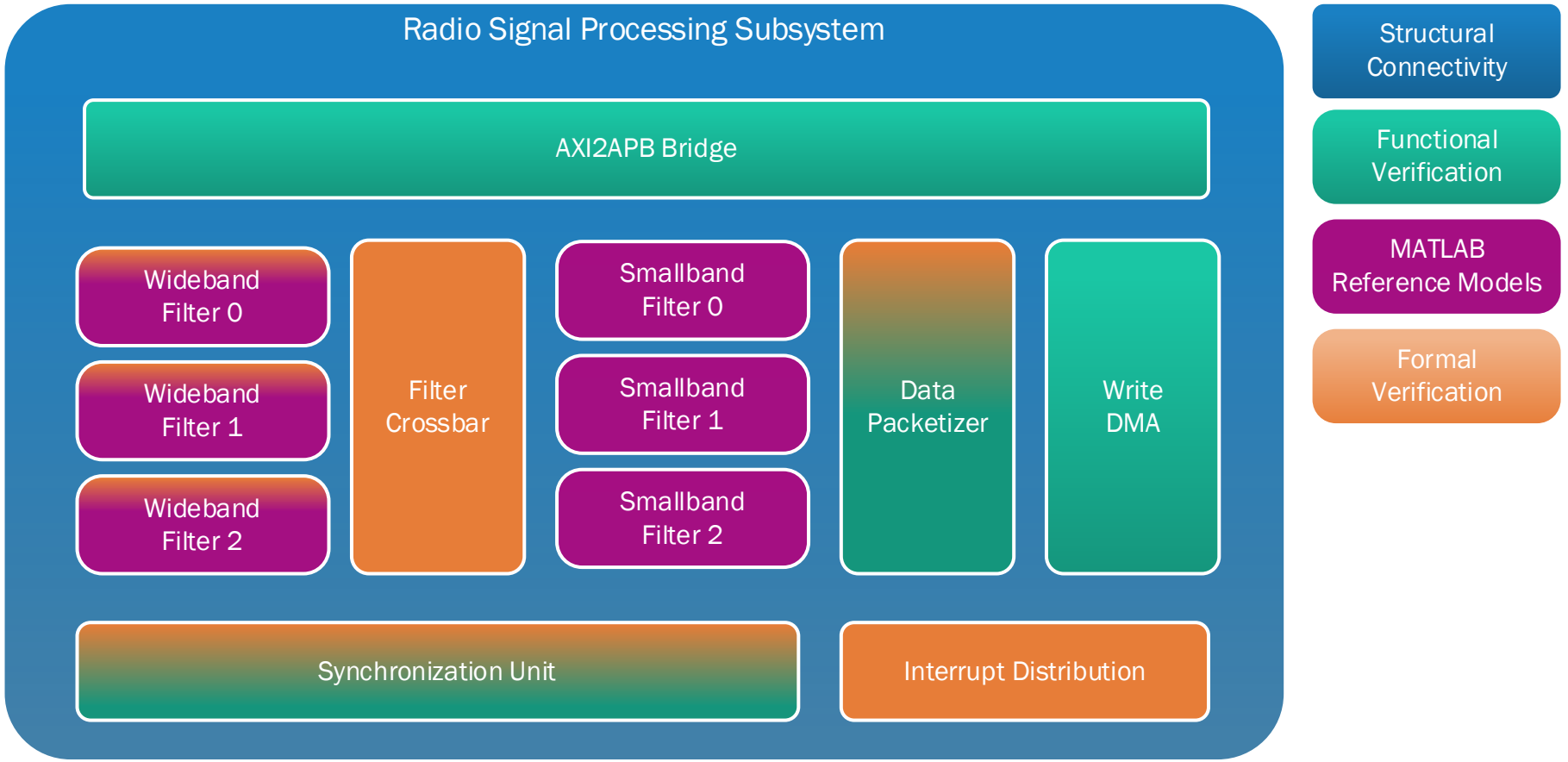
# Phase 2.2: Formal Property Verification



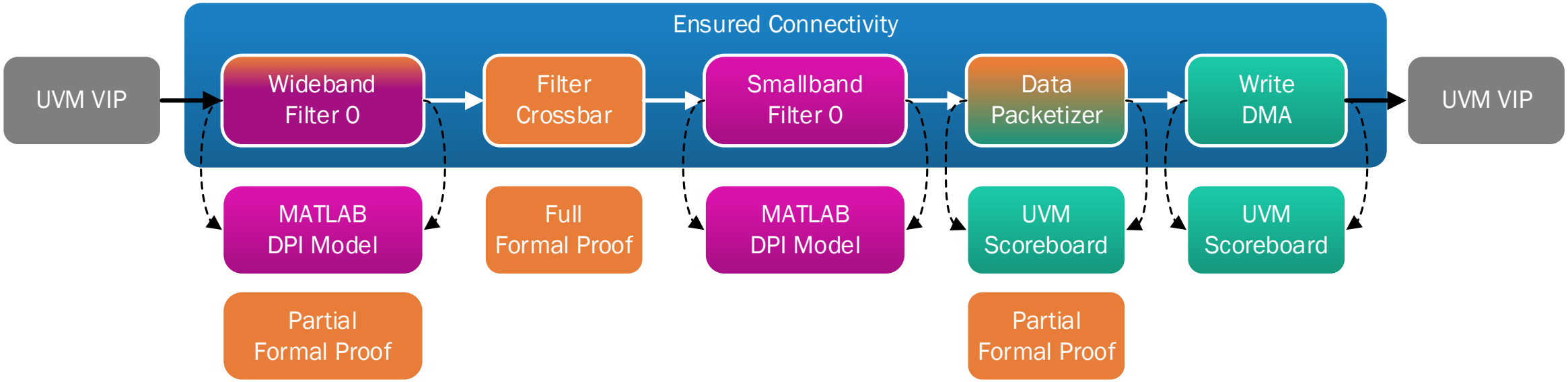
# Phase 3: Regression, Coverage and UNR



# Applying Hybrid Verification

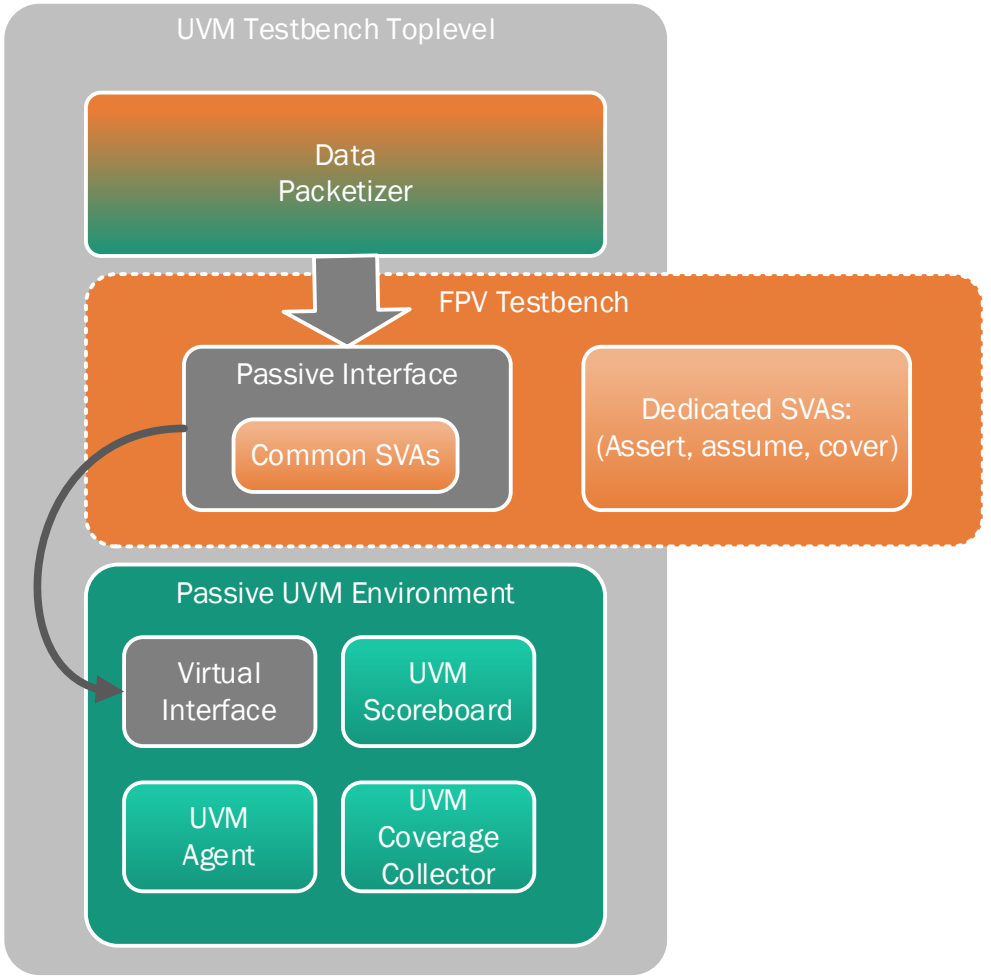


# Coherent Data Check Flow



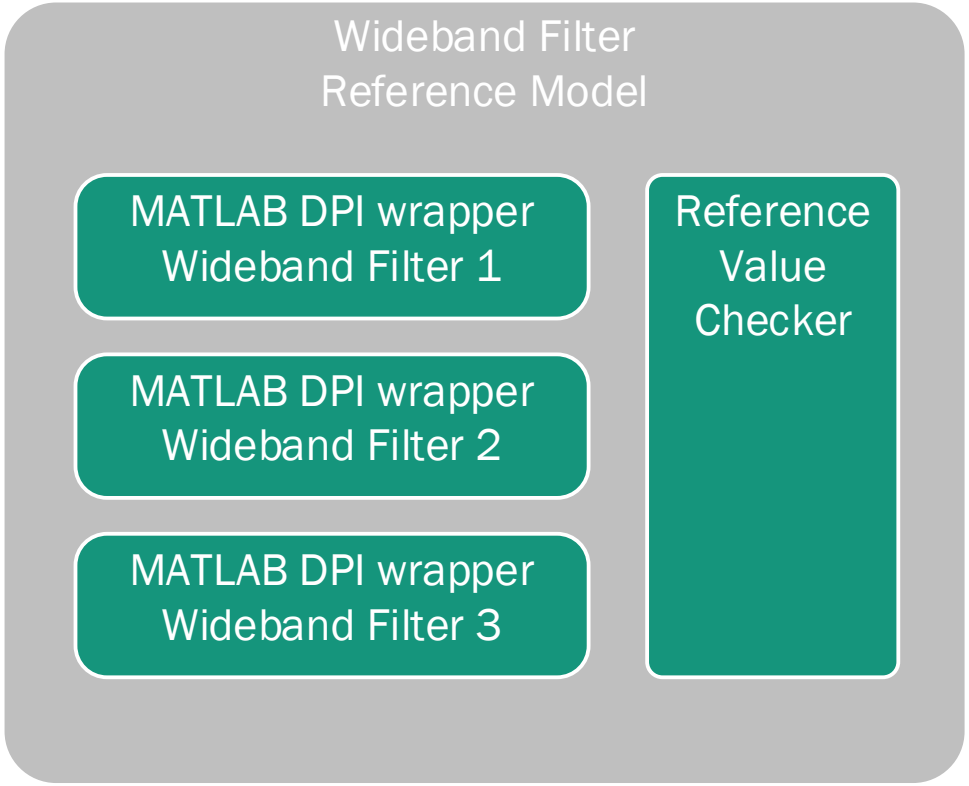
# Sharing Properties

- Sharing properties between UVM and FPV via System Verilog interfaces
- Limited to DUT interface signals
- Rules
  - No liveness properties
  - No usage of free variables
  - Limit number of cover properties



# MATLAB DPI Wrapper

- Required accuracy provided by test specifications
- Reference value check every clock cycle
- Standardized DPI wrapper interface
  - DPI_<name>_initialize()
  - DPI_<name>_terminate()
  - DPI_<name>_reset()
  - DPI_<name>_output()
  - DPI_<name>_update()



# Results – Coverage Merge

| Module | Properties | Sim Coverage in % (*1) | Formal Coverage in % (*2) | Merged Coverage in % |
|---|---|---|---|---|
| Wideband Filter | 2 | 75 | 1.1 | 75 |
| Filter Crossbar | 106 | 94 | 100 | 100 |
| Data Packetizer | 150 | 65 | 75 | 88 |
| Interrupt Distribution | 66 | 95 | 100 | 100 |
| Synchronization Unit | 50 | 73 | 90 | 95 |

*1 Branch, Expression, Statement, Toggle on interfaces
*2 Normal Precision, Reset excluded

# Results – Unreachability Analysis (UNR)

| Module | Coverage Pre-UNR in % | Coverage Post-UNR in % (*1) | Total UNR in % |
|---|---|---|---|
| DUT top level | 76 | 82 | 6 |
| Wideband Filter | 75 | 82 | 7 |
| Smallband Filter | 84 | 87 | 3 |
| Data Packetizer | 88 | 88 | 0 |
| Write DMA | 93 | 95 | 2 |

*1 Running UNR with uninitalized start state

# Summary

- We created a powerful hybrid verification setup which combines a traditional UVM based approach with powerful extensions like formal property verification, structural connectivity checks, unreachability analysis paired with integrated MATLAB reference models
- Hybrid verification approach provides free choice of methodology for verifications teams based on verifications tasks and team experience
- Outlook
  - Extend the usage of formal property verification (FPV)
  - Evaluate more automated formal verification apps

Questions?