Analog Verification Concepts: Industrial Deployment Case Studies

Frontiers in Analog CAD (FAC 2014)
July, 9-10, 2014, Grenoble, France
Peter Rotter, Infineon Technologies AG

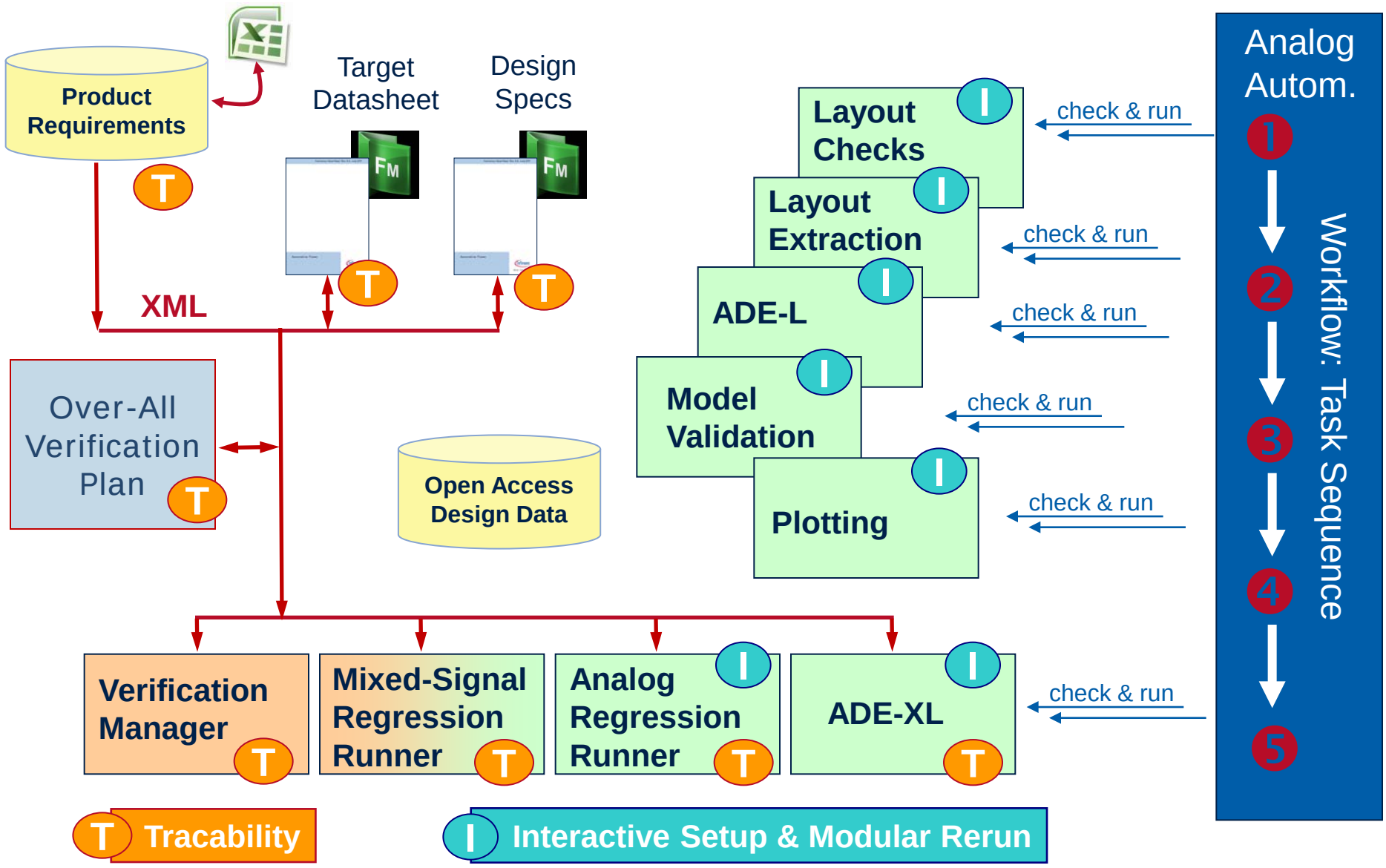
Agenda

- Analog Verification - Context/Components/Coordination
- (R)evolution of Industrial Verification Ecosystem
- Analog Verification Planning & Sign-Off
- Requirements on Verification Software & Integration
- Case Study 1: Safe Operating Area Checks
- Case Study 2: Analog Model Validation
- Summary & Final Thoughts

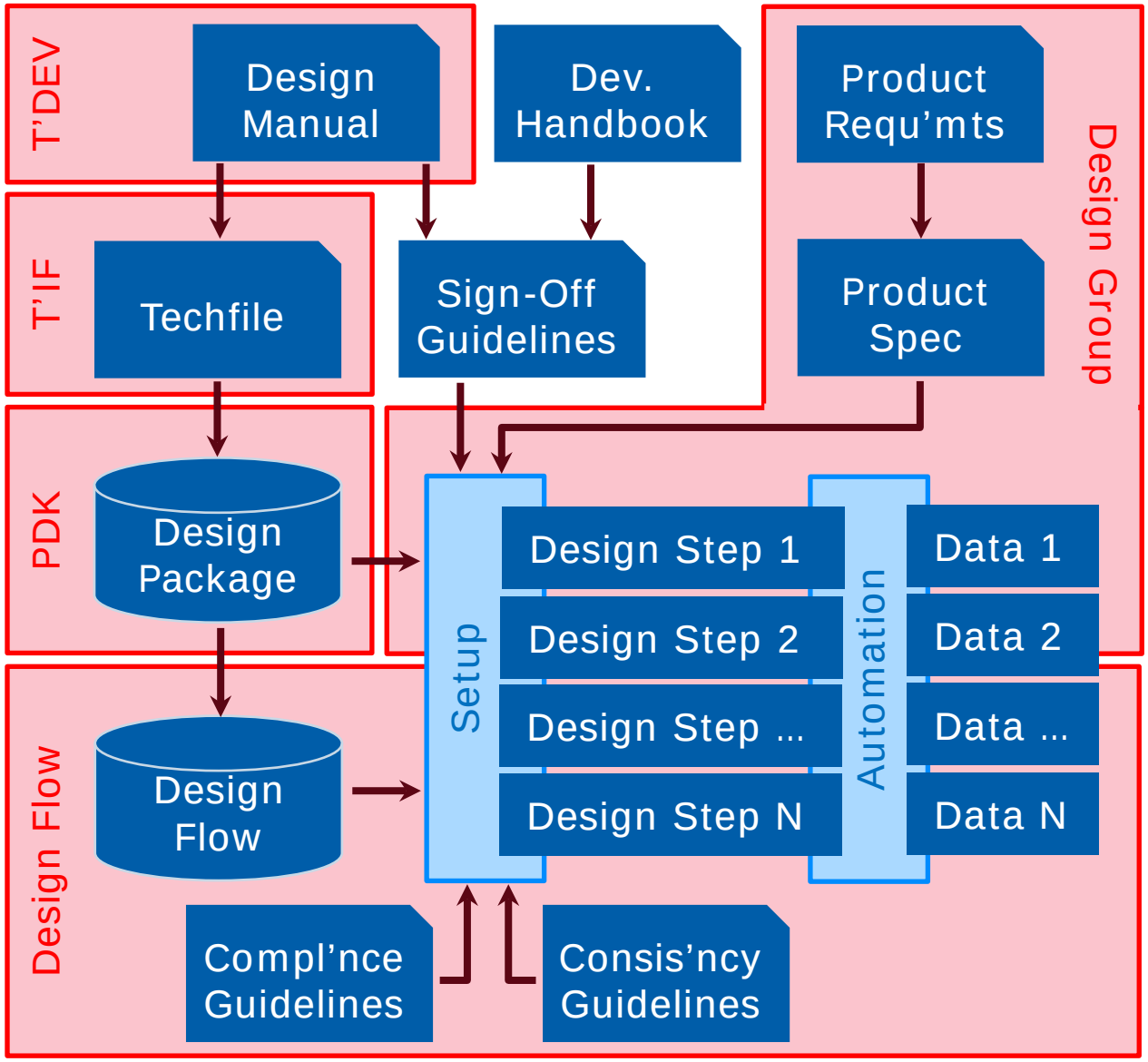
Agenda

- Analog Verification - Context/Components/Coordination
- (R)evolution of Industrial Verification Ecosystem
- Analog Verification Planning & Sign-Off
- Requirements on Verification Software & Integration
- Case Study 1: Safe Operating Area Checks
- Case Study 2: Analog Model Validation
- Summary & Final Thoughts

Mixed-Signal Sign-Off and Regression – Overview



Analog Verification - Context/Components/Coordination



Agenda

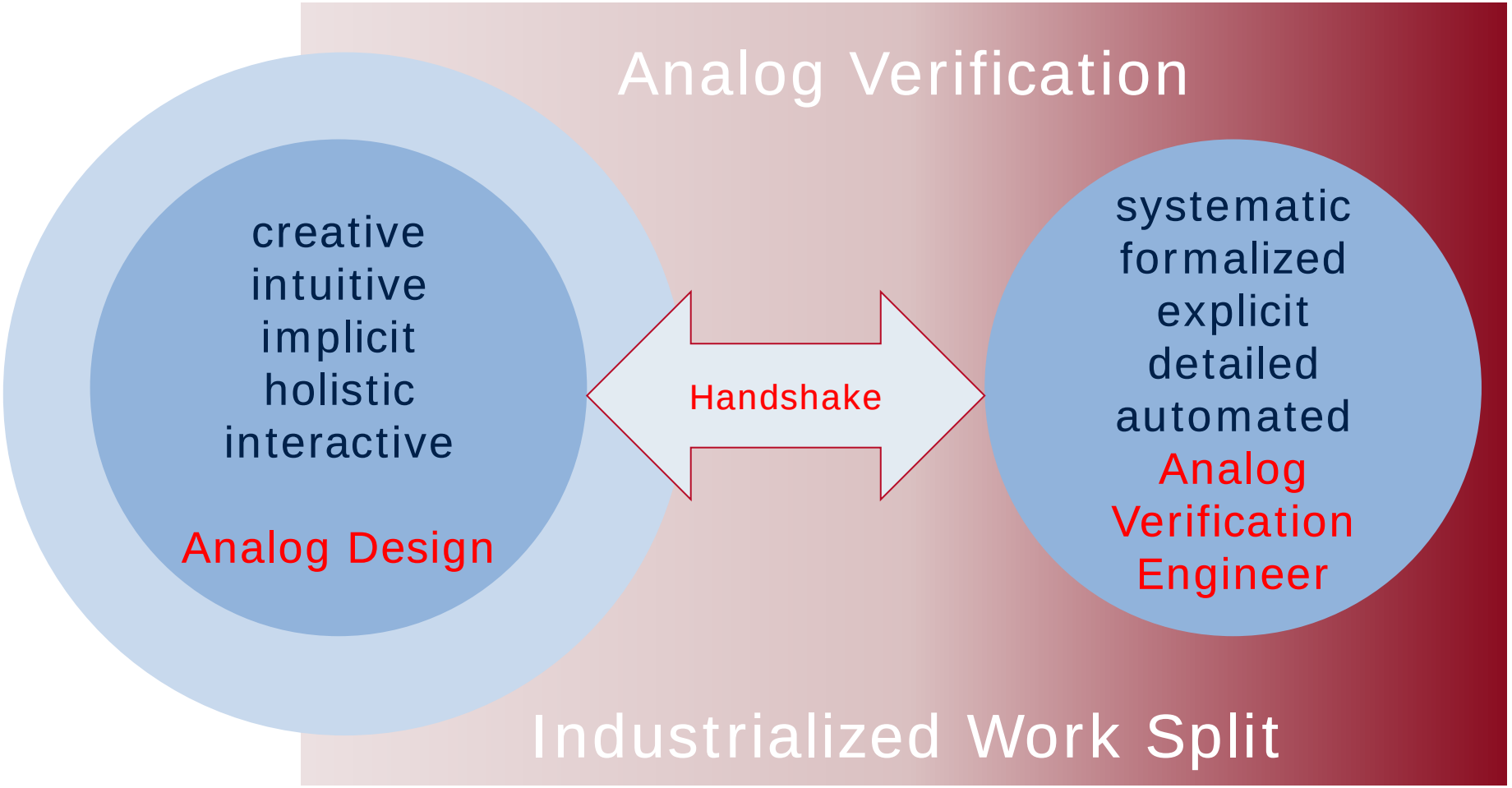
- Analog Verification - Context/Components/Coordination
- (R)evolution of Industrial Verification Ecosystem
- Analog Verification Planning & Sign-Off
- Requirements on Verification Software & Integration
- Case Study 1: Safe Operating Area Checks
- Case Study 2: Analog Model Validation
- Summary & Final Thoughts

Trends in Analog Design

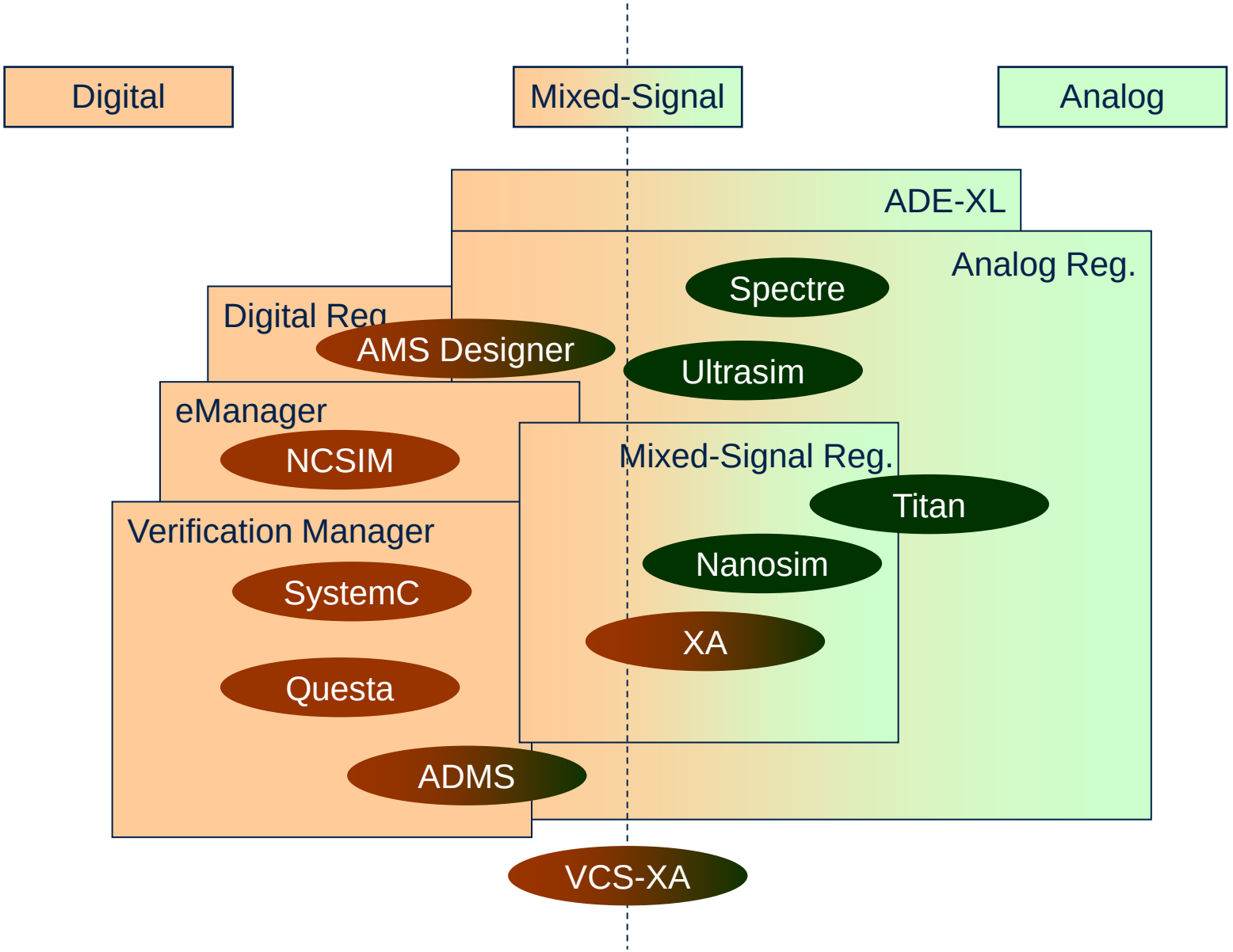
- Technology Rules – increasing number and complexity for
  - small technology nodes
  - technology variants for specific purposes (automotive/industrial)
- CMOS Transistor Model Complexity – increasing number of parameters (BSIM4, PSP, HiSIM-HV several hundred param's)
- Technology variations increasing for small technology nodes – statistical methods required
- Regression Tooling – increasing number and complexity of solutions
- Safety Requirements – formal proofs of compliance required
  - General (IEC-61508)
    Safety Integrity Level (SIL) Classification
  - Automotive (ISO 26262)
  - Aviation (DO-178/254)
  - Railway (CENELEC 50126/128/129)

Two Worlds collide …

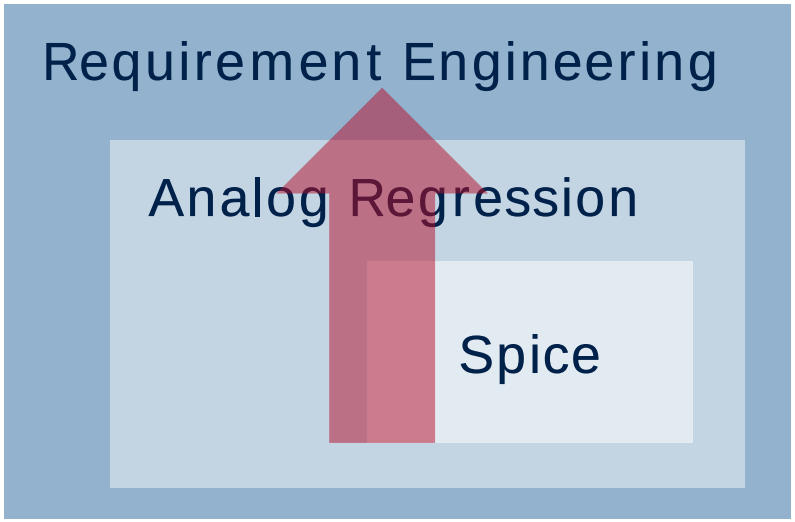
(R)evolution of Industrial Verification Ecosystem



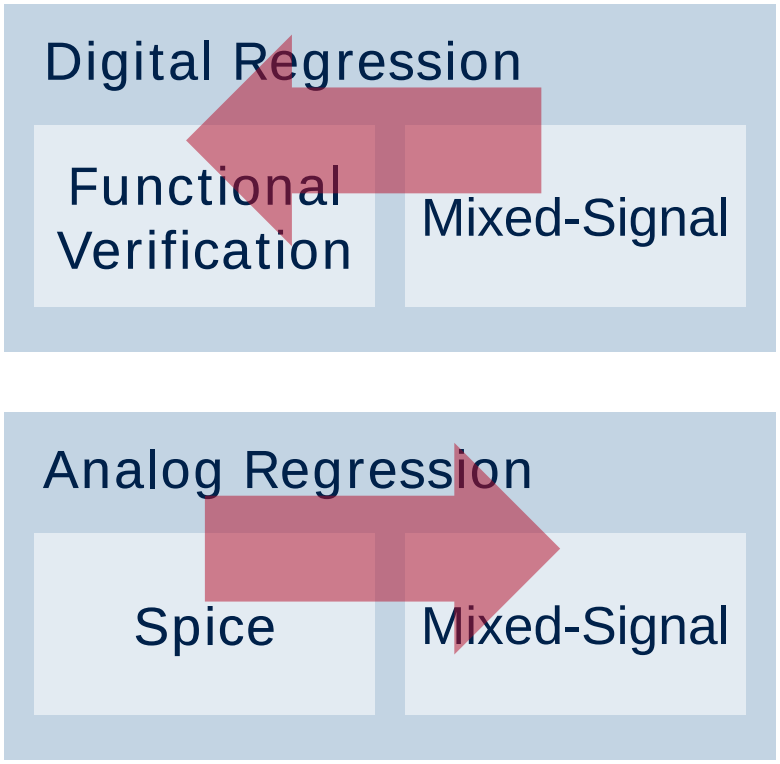
Digital vs. Analog Verification Management



Overall Verification – Crossing Design Domains



Vertical Verification
Domain-Crossing



Horizontal Verification
Domain-Crossing

Additional effort and complexity for design groups and CAD support

Some Open Issues …

- Design Flow
  - Automation support for transfer of Guidelines/Specifications to Design Steps Setup
  - Interface for automation support of Design Group specific Guidelines
  - Sign-Off Status Reporting
- General
  - Examples for Manual Design Steps:
    - Product Requirement to Block Level Specification break-down Process
    - Analog Block-level Verification Definition & Setup
  - Examples for Implicit Requirements:
    - Check for design critical aspects not covered by technology models
    - Implicit design constraints imposed by technology (topology selection)
  - Analog Design is a complex problem requiring to cover an immense number of currently non-formalized aspects.
    - Key for Successful Design: Designer Intuition
    - Hence: Water-proof formalization via guidelines questionable!

Agenda

- Analog Verification - Context/Components/Coordination
- (R)evolution of Industrial Verification Ecosystem
- **Analog Verification Planning & Sign-Off**
- Requirements on Verification Software & Integration
- Case Study 1: Safe Operating Area Checks
- Case Study 2: Analog Model Validation
- Summary & Final Thoughts

Analog Verification - Integration Objectives

- Systematic
  - Analog Sign-off Guidelines
  - tooling for hierarchical requirement break-down, verification planning and execution
  - completeness ensured via proper coverage measures
  - setup support via pre-defined templates (Analog Verification IP)
- Formalized
  - overview status reports
  - hierarchical status feed-back
  - proof of safety requirements compliance
- Explicit
  - measurements & checkers
  - specification ranges
  - result evaluation
- Detailed
  - break-down in list of individual tasks with dedicated objectives
- Automated
  - capture existing interactive setups and push to automation
  - (unified) regression runner for all verification domains
  - specification checks
  - automated generation of documentation

Analog Sign-Off – Motivation

- Status Quo
  - Systematic approach to ensure Analog Sign-Off Compliance generally requested
  - Complete supply chain for tool, technology and library data provision considered
- Overall Goals
  - Systematic, formalized approach to list, specify (short work instructions) and document Analog Sign-Off related Design Steps including Setups
  - Ensure Analog Sign-Off compliant Design System provision
  - Account for Technology Specific Flavors
- Deployment

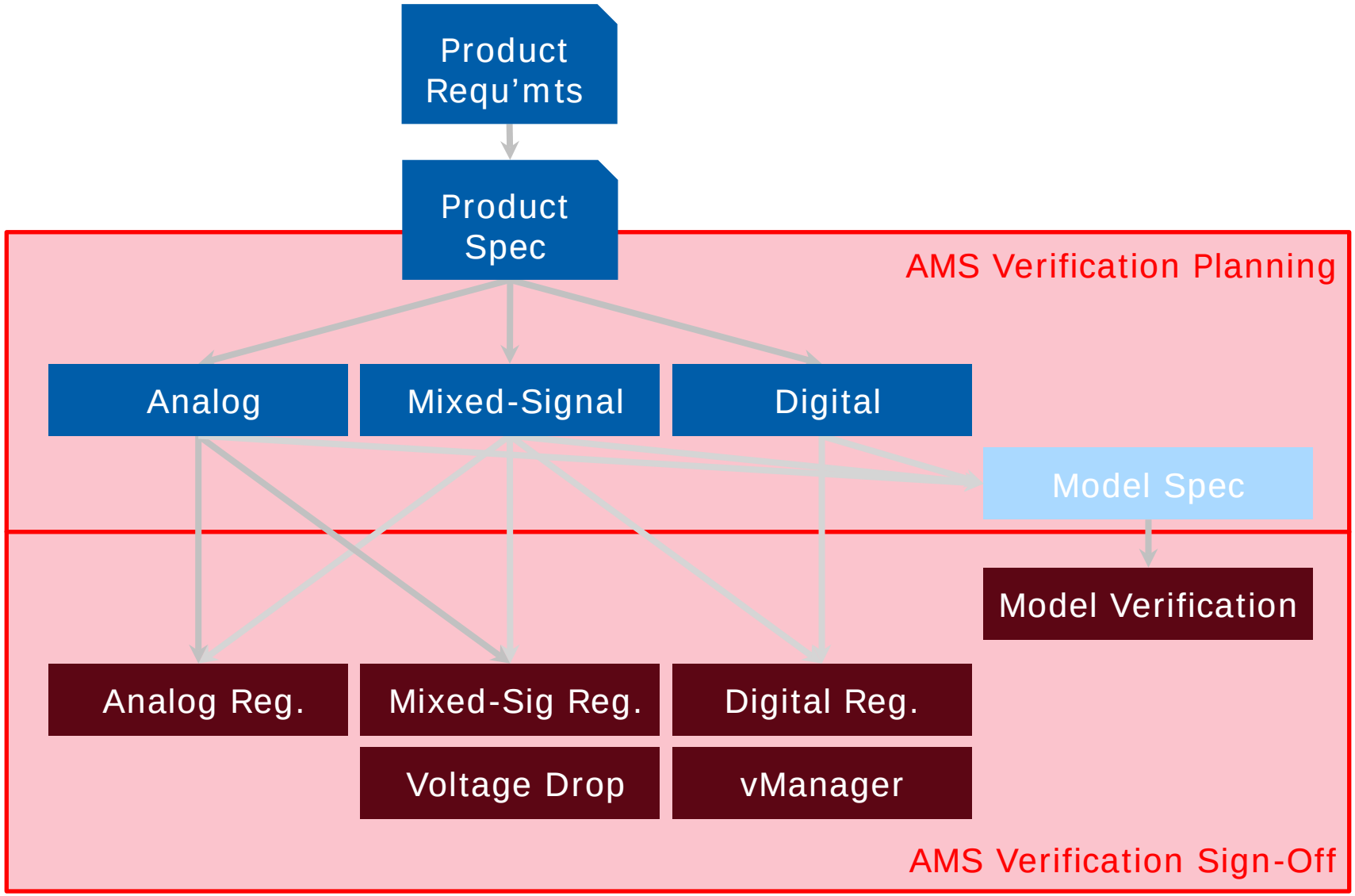
  Guidelines should be used for a systematic planning and execution of all defined steps and thus help to ensure Analog Sign-Off compliance for all Analog-/Mixed-Signal Design

Generic Scheme for Sign-Off Task Definition

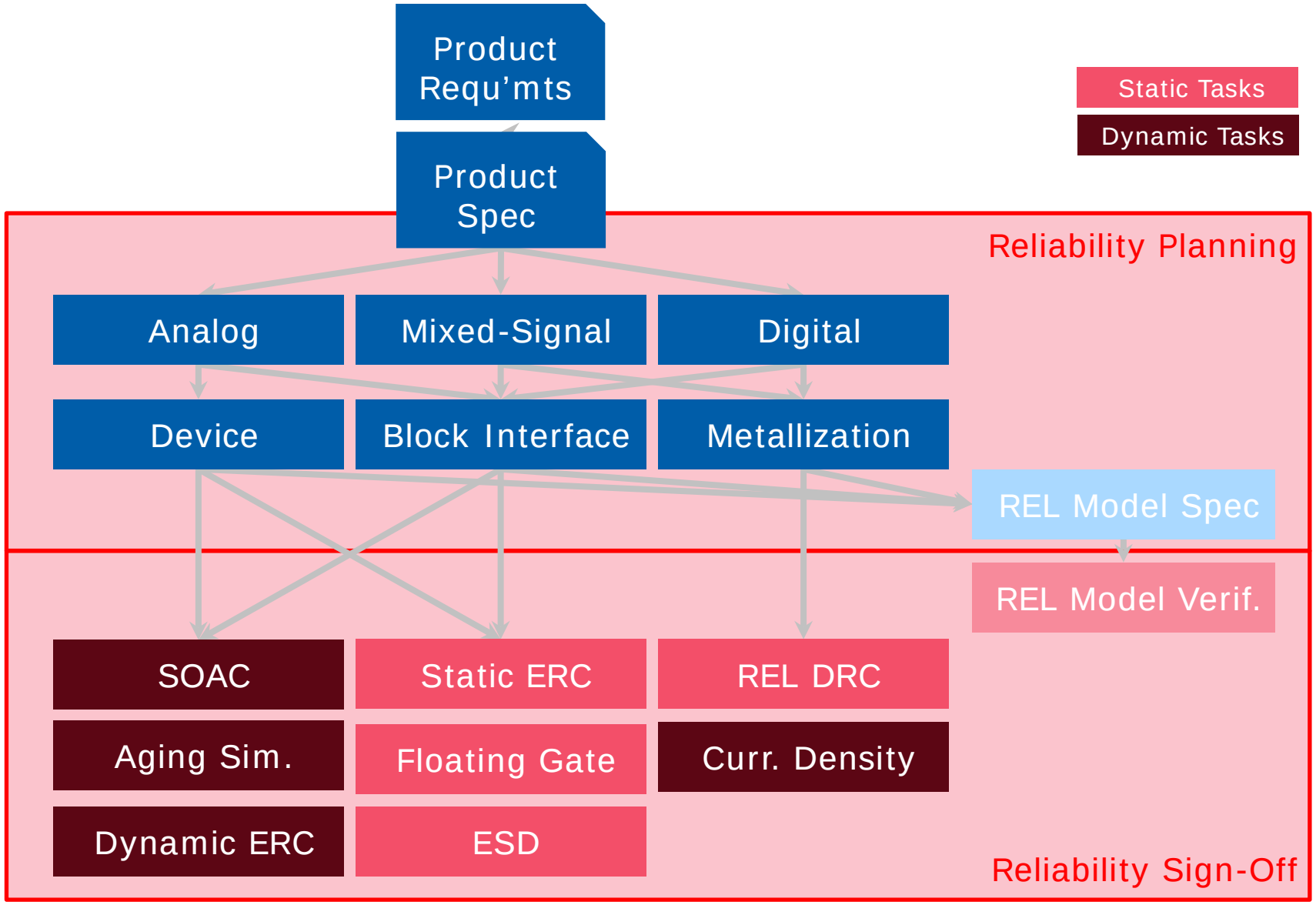
Concise scheme for descriptions of Analog Sign-Off Tasks:

- Sign-Off Task – unique keyword
- Category – ordering scheme
- Scope/Objective – concise definition of purpose
- Application Model – scenarios for check deployment
- Method/Tooling – description of tool chain
- Result Evaluation – guideline for result assessment

AMS Verification Planning & Sign-Off



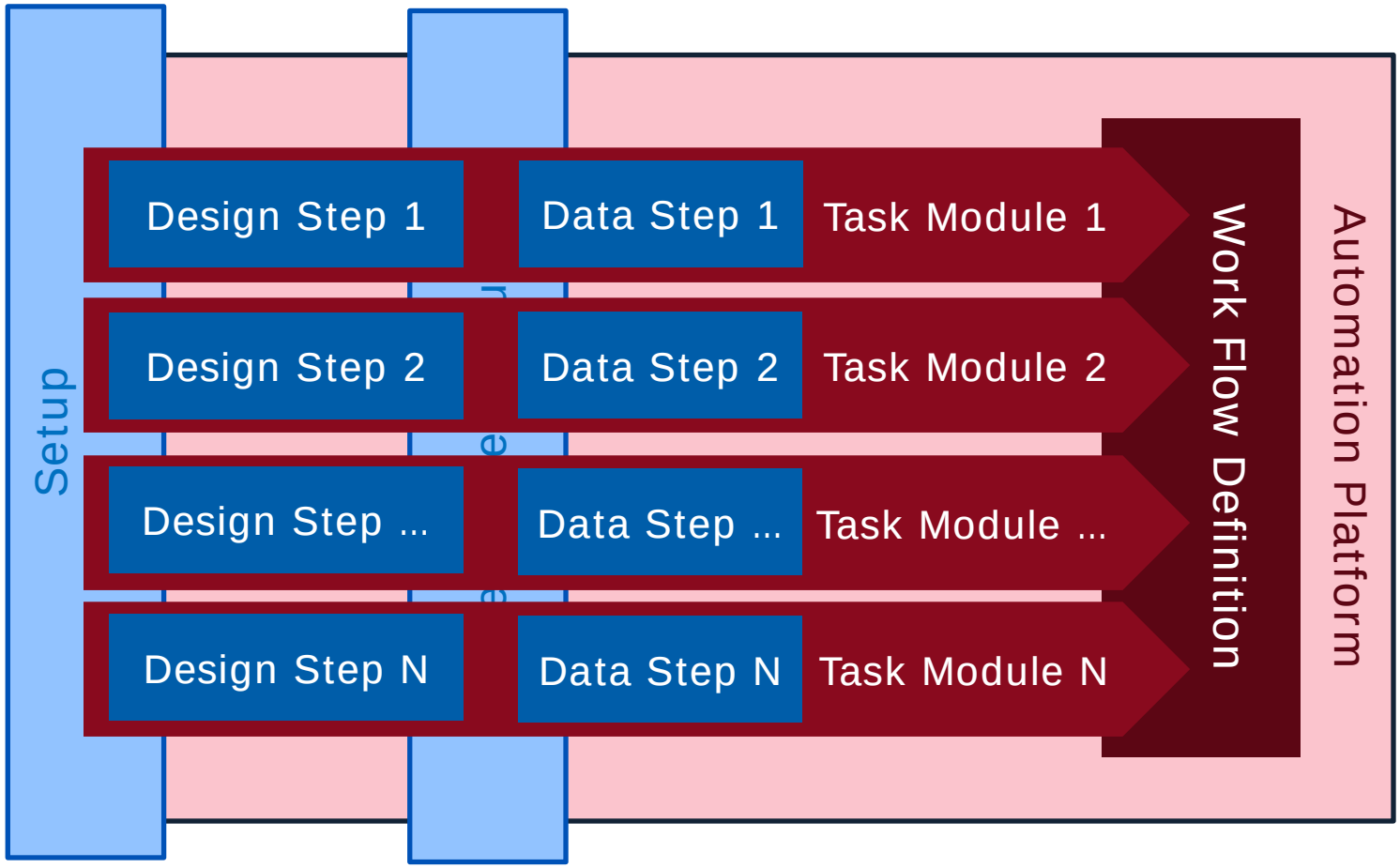
Reliability Planning & Sign-Off



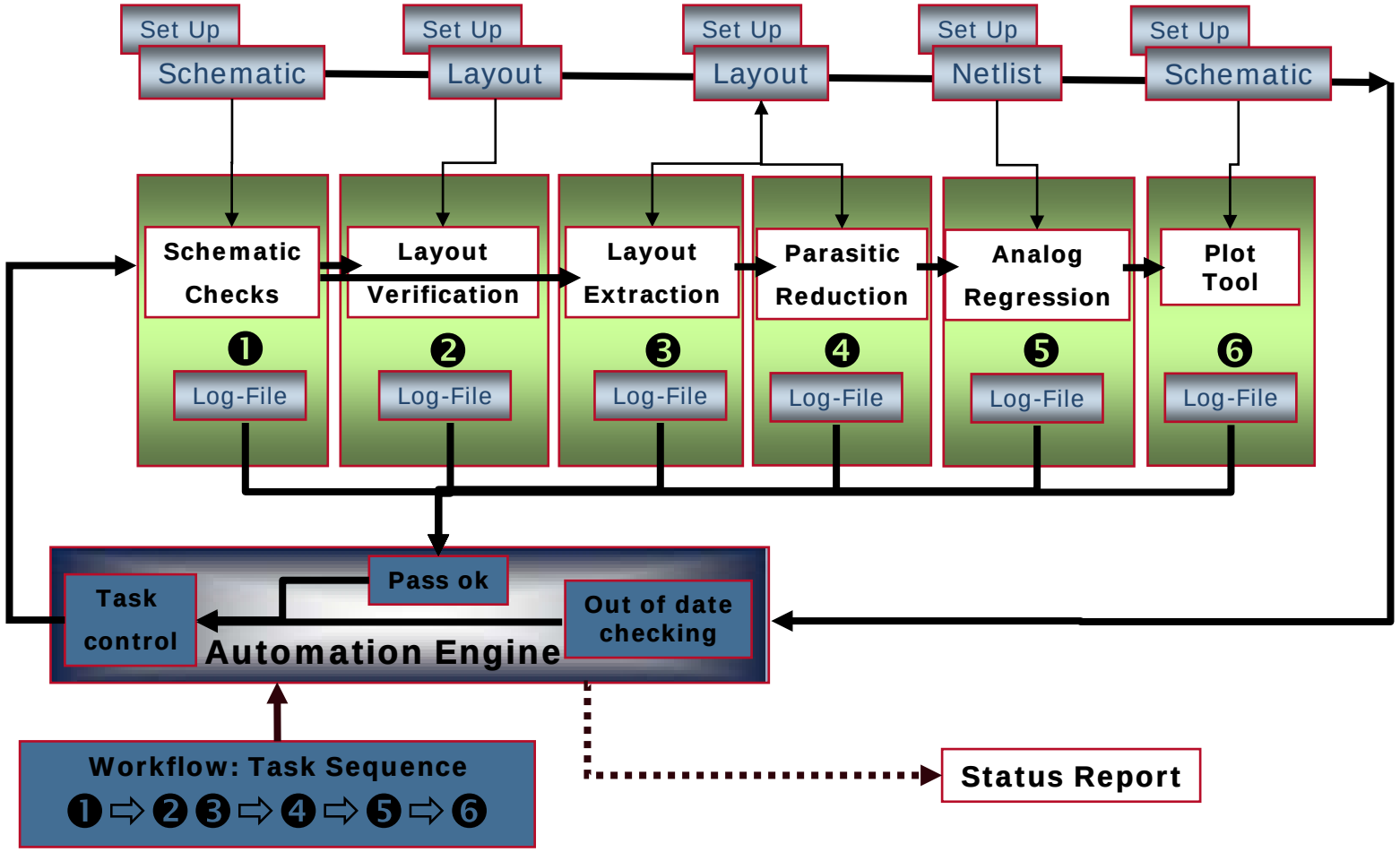
Agenda

- Analog Verification - Context/Components/Coordination
- (R)evolution of Industrial Verification Ecosystem
- Analog Verification Planning & Sign-Off
- **Requirements on Verification Software & Integration**
- Case Study 1: Safe Operating Area Checks
- Case Study 2: Analog Model Validation
- Summary & Final Thoughts

Basic Automation Concepts



Analog Automation Workflow in Action



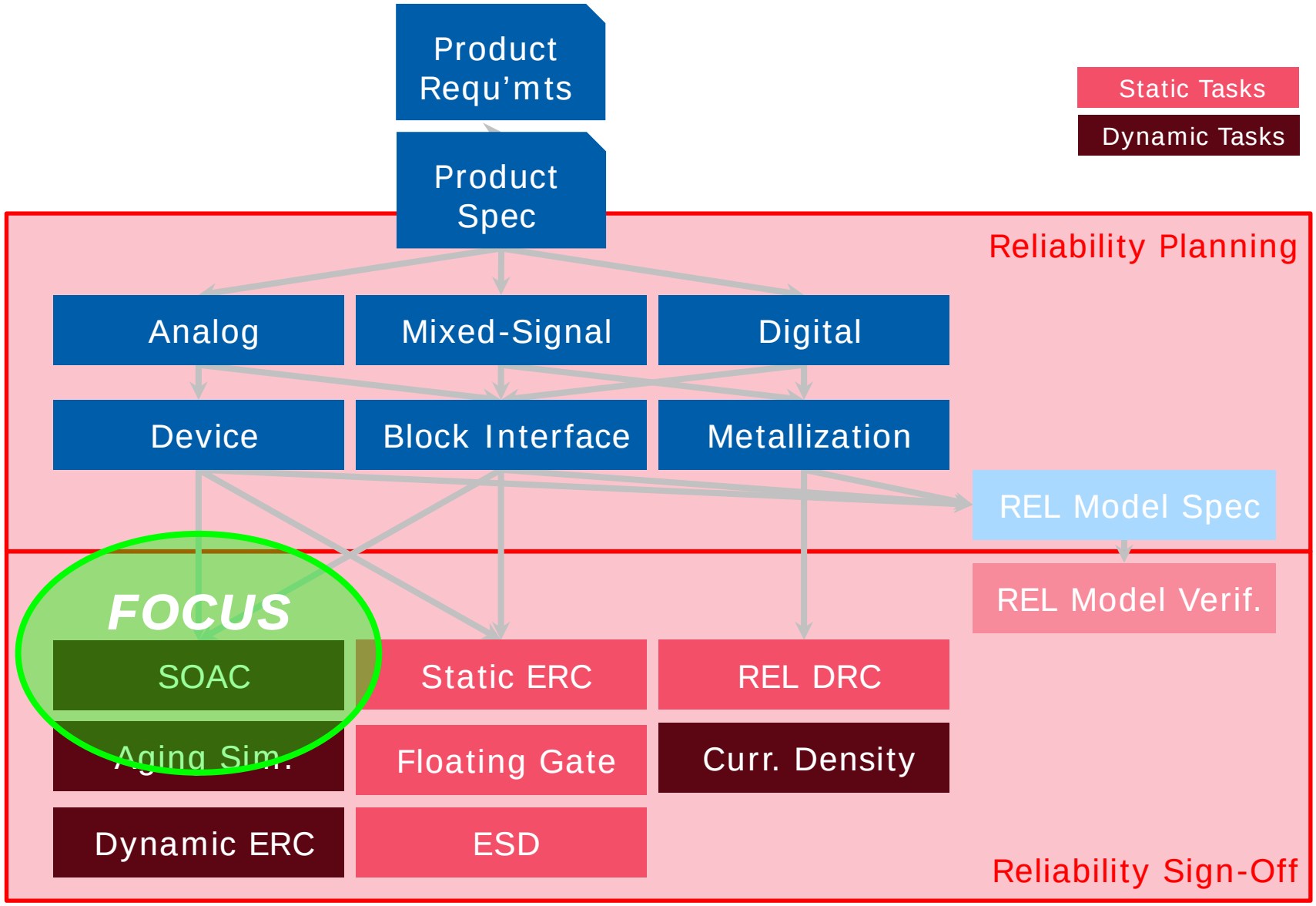
Mixed-Signal Sign-Off and Regression Strategy

- Pick-up Approach
  - ✓ avoid additional setup effort as much as possible
  - ✓ capture existing interactive setups and push to automation
  - ✓ identify and build up stand-alone base applications (Analog Regression Runner, Model Validation)
- Modularity
  - ✓ Task Modules as generic building blocks
  - ✓ Task Sequences provided as pre-defined sets for common use cases
- Flexibility
  - ✓ selection of individual Task Modules
  - ✓ local modification/definition of Task Sequences
  - ✓ address small sequence of tasks with highest automation benefit
- Transparency
  - ✓ native and stand-alone application usage should not be blocked out
  - ✓ strong focus on error checking and debugging

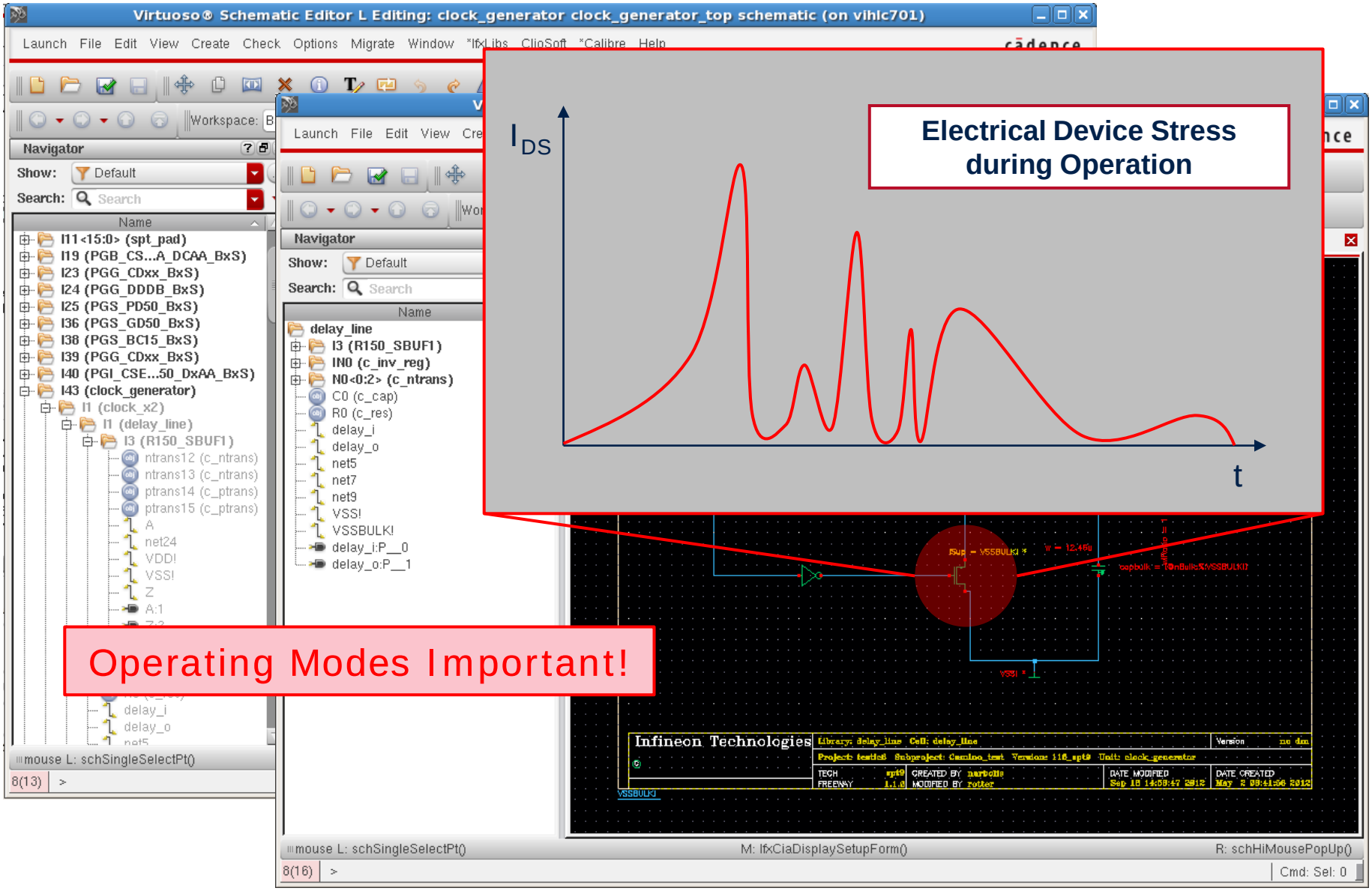
Agenda

- Analog Verification - Context/Components/Coordination
- (R)evolution of Industrial Verification Ecosystem
- Analog Verification Planning & Sign-Off
- Requirements on Verification Software & Integration
- **Case Study 1: Safe Operating Area Checks**
- Case Study 2: Analog Model Validation
- Summary & Final Thoughts

Reliability Planning & Sign-Off



Electrical Device Stress



Safe Operating Area Checks (SOAC) Introduction

- For semiconductor devices the Safe Operating Area is defined as the voltage and current conditions over which the device can be expected to operate without self-damage.
- Checks can be defined for Spice transistor level simulations using tool specific features (assertions, special checks). The allowed voltage/current ranges are dynamically monitored during simulation runtime and warning/error messages are issued in case of violations.
- SOAC setups are technology dependent and need to be defined and provided with each Technology Package.
- SOAC can be costly with respect to memory consumption, runtime and license usage. Therefore the usage needs to be transparent and controllable by the designer. A unified interface and used model concept for all supported tools is very desirable.

SOAC – Analog Sign-Off Definition

- **Check Scope**

  Device stress disaster check

- **Application Model**
  - Use standard test scenarios for specification verification over temperature, process corner, supply modes, …
  - Functional tests for power on/off, sleep, wake-up, …

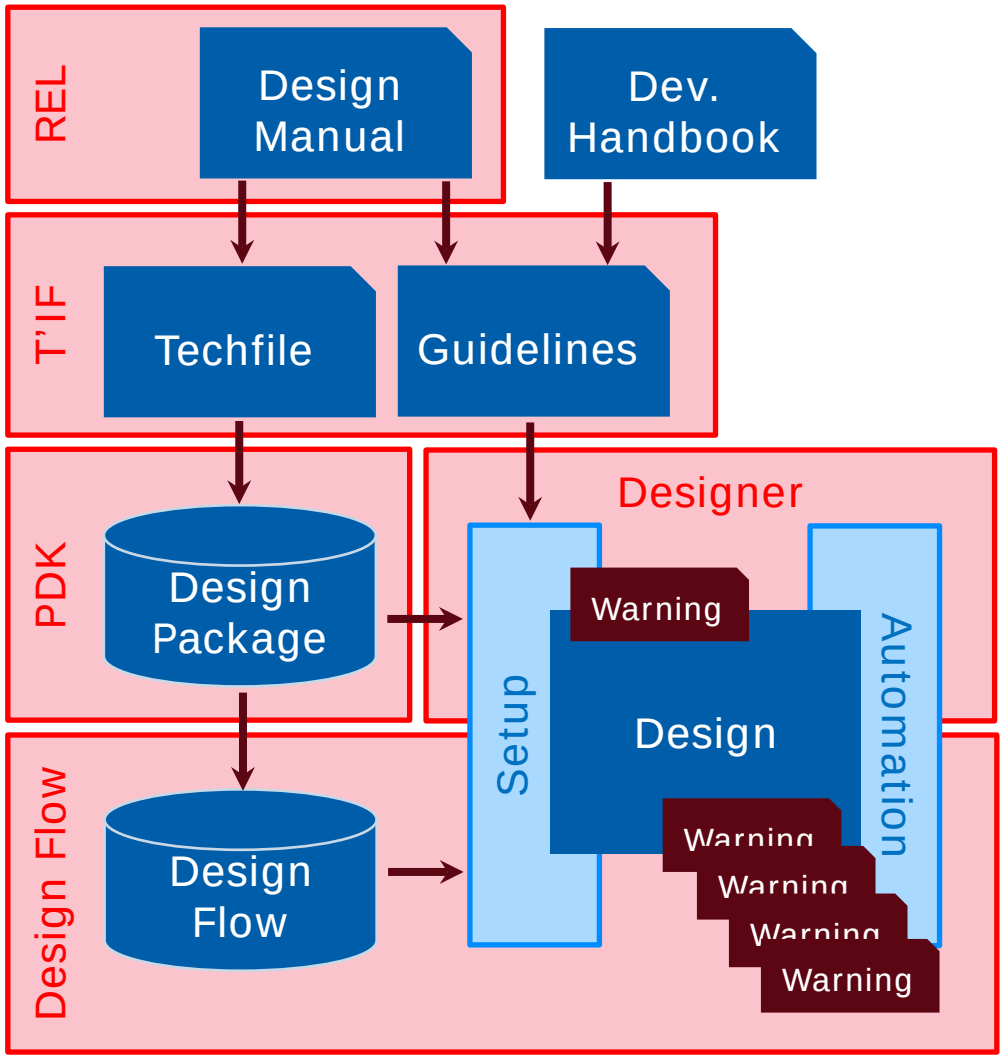
  Note: Stress-scenario definition via stimulus important for check coverage

- **Method/Tooling**
  - Include simulator specific Save Operating Area Checks (SOAC) files provided in Technology Package
  - Run Spice simulation (Spectre, Titan, Fastmos, Mixed-Signal)

- **Result Evaluation**

  Check all messages in violation files via Cadence ADE browser or text based in log file (Message Type: Warning)

SOAC – Context/Components/Coordination



SOAC – Design Flow: Overview & Open Issues

- **Primary Use Model – Spice Level Tools (Manual)**
  - Spice: Spectre, Titan (IFX)
  - FastSpice: XA, Nanosim, Ultrasim
  - Mixed-Signal: AdvanceMS, AMS Designer

  Problems:
  - non-standardized check definition: tool/vendor specific concepts
  - check scenario definitions with reasonable coverage
  - large number of dummy errors
- **Secondary Use Model – Regression Tooling (Automation)**
  - IFX Analog Regression Runner, ADE-XL
  - IFX Mixed-Signal Regression Runner
  - IFX Digital Regression Runner, vManager

  Problems:
  - non-standardized activation and result provision
  - transparent result propagation: documentation/summary report missing
  - waiver concept missing
- Design Flow Support
  - Design Package interface defined
  - Setup support for default SOAC settings where possible
  - Unique result evaluation script (IFX) for all tools supporting primary use model

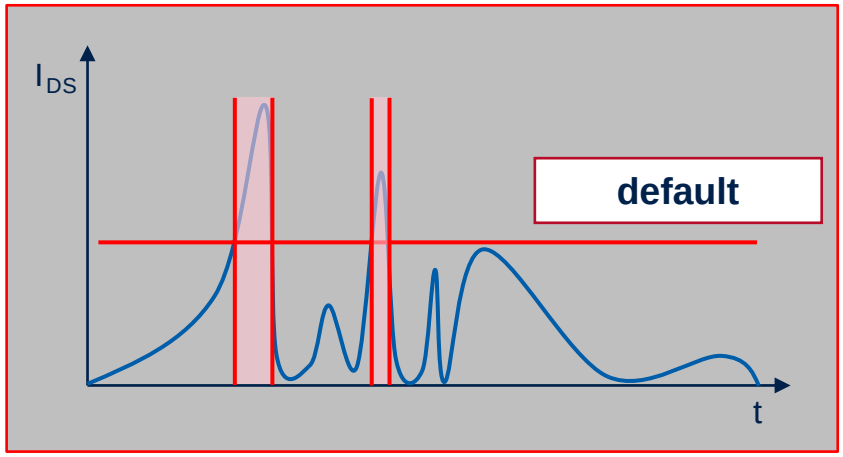
SOAC – Open Issues: Focus Topic Usability

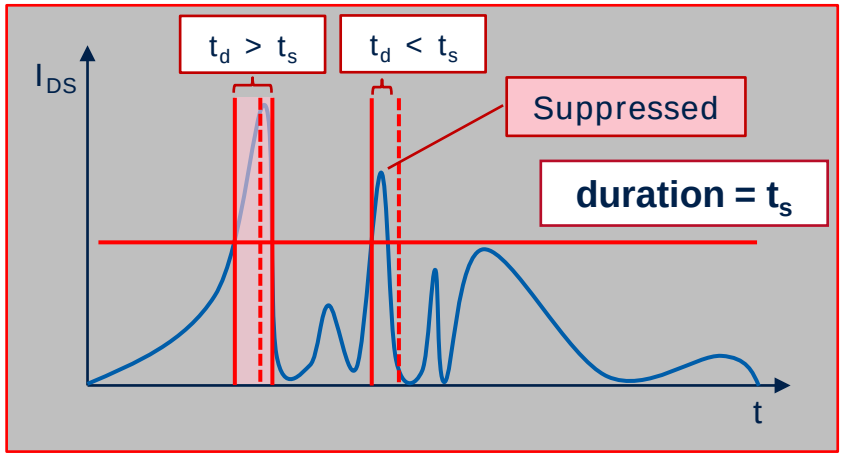
- Problem

  issue with large number of dummy errors

- Possible Solution Scenarios
  - define checks with improved selectivity:
    simple technology rules need to be extended
  - provide filtering/post-processing/sorting using some kind of severity criteria (e.g. duration factor: see next slide)
  - enhance tool functionality by more efficient assessment features (back-annotation, violations overlay to waveforms, waiving methodologies)

SOAC – Overshoot Suppression via ‘duration’





Case Study 1: Safe Operating Area Checks

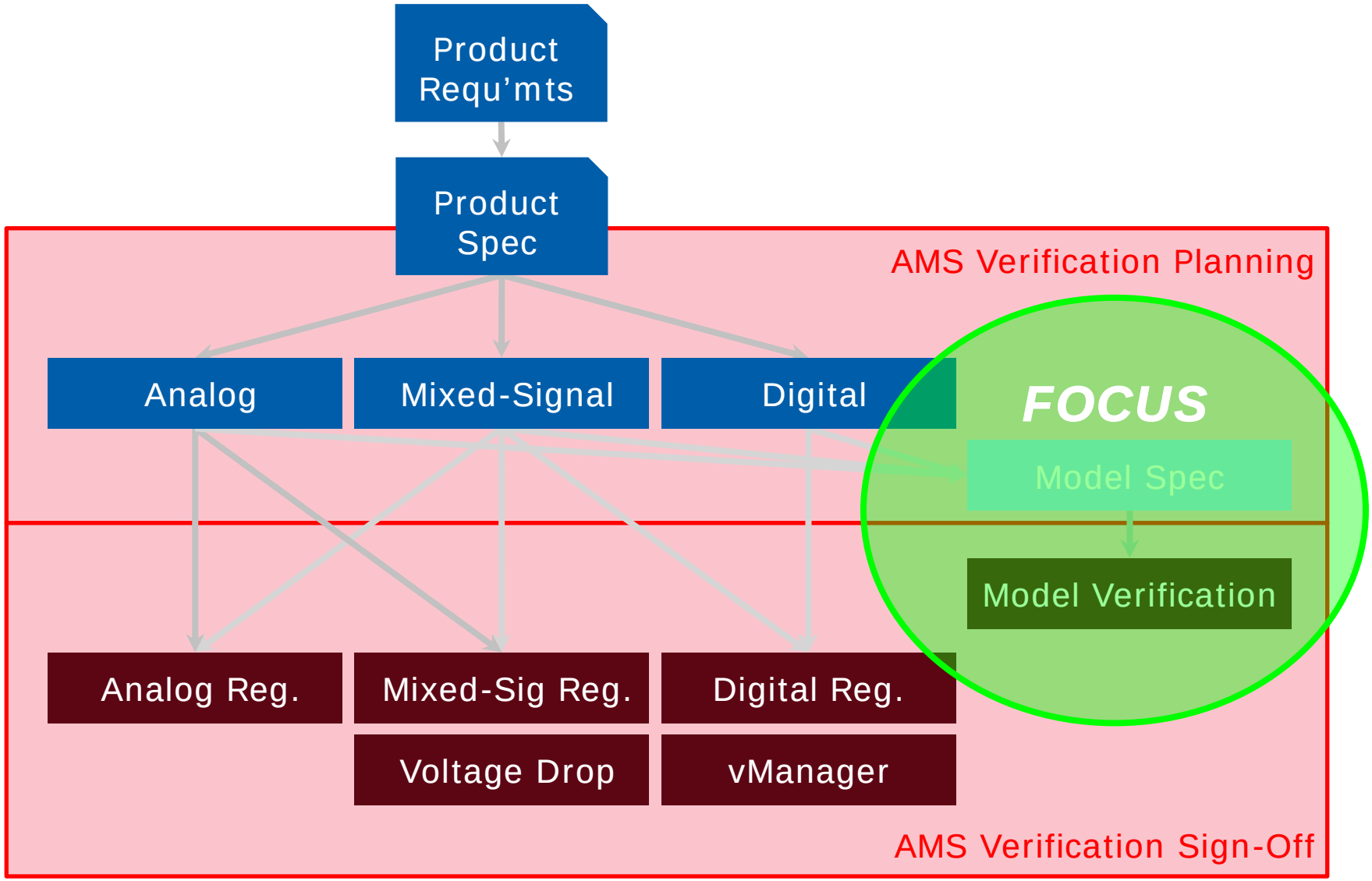
Deployment Issues:

- CAD Setup:
  Non-standardized SOAC check definition, activation and especially result provision leads to high effort for overall integration
- Vertical Verification Domain-Crossing:
  Effort for check integration to regression environments
- Horizontal Verification Domain-Crossing:
  Analog Verification / Reliability Checks require different application scenarios; improved check selectivity requires cross-domain cooperation
- Implicit/Explicit:
  Converting design know-how to check scenarios with reasonable coverage based on use modes specified at verification planning
- Interactive/Automation:
  Automatic result evaluation in case of dummy errors impossible; waiving concepts currently missing

Agenda

- Analog Verification - Context/Components/Coordination
- (R)evolution of Industrial Verification Ecosystem
- Analog Verification Planning & Sign-Off
- Requirements on Verification Software & Integration
- Case Study 1: Safe Operating Area Checks
- **Case Study 2: Analog Model Validation**
- Summary & Final Thoughts

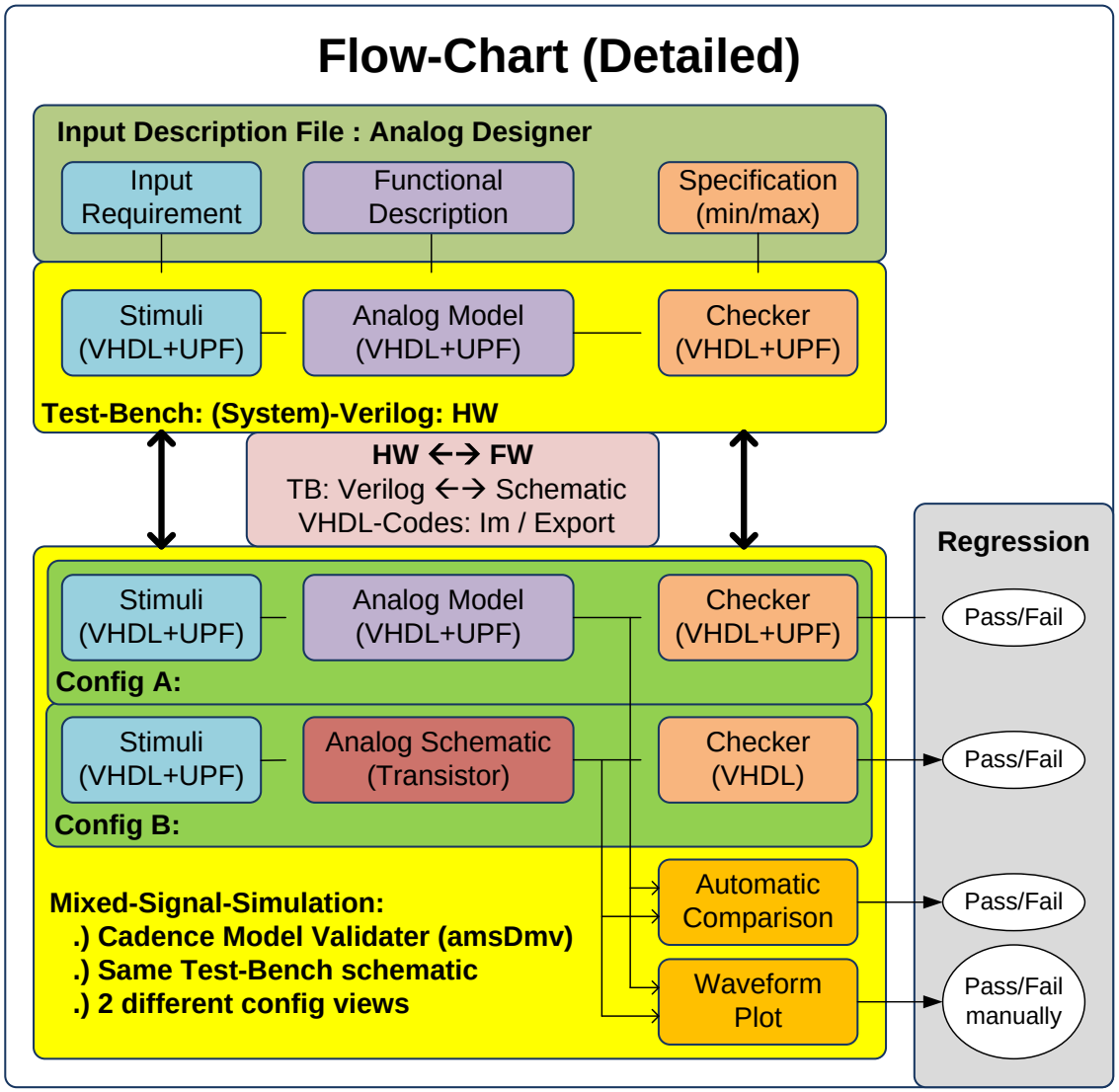
AMS Verification Planning & Sign-Off



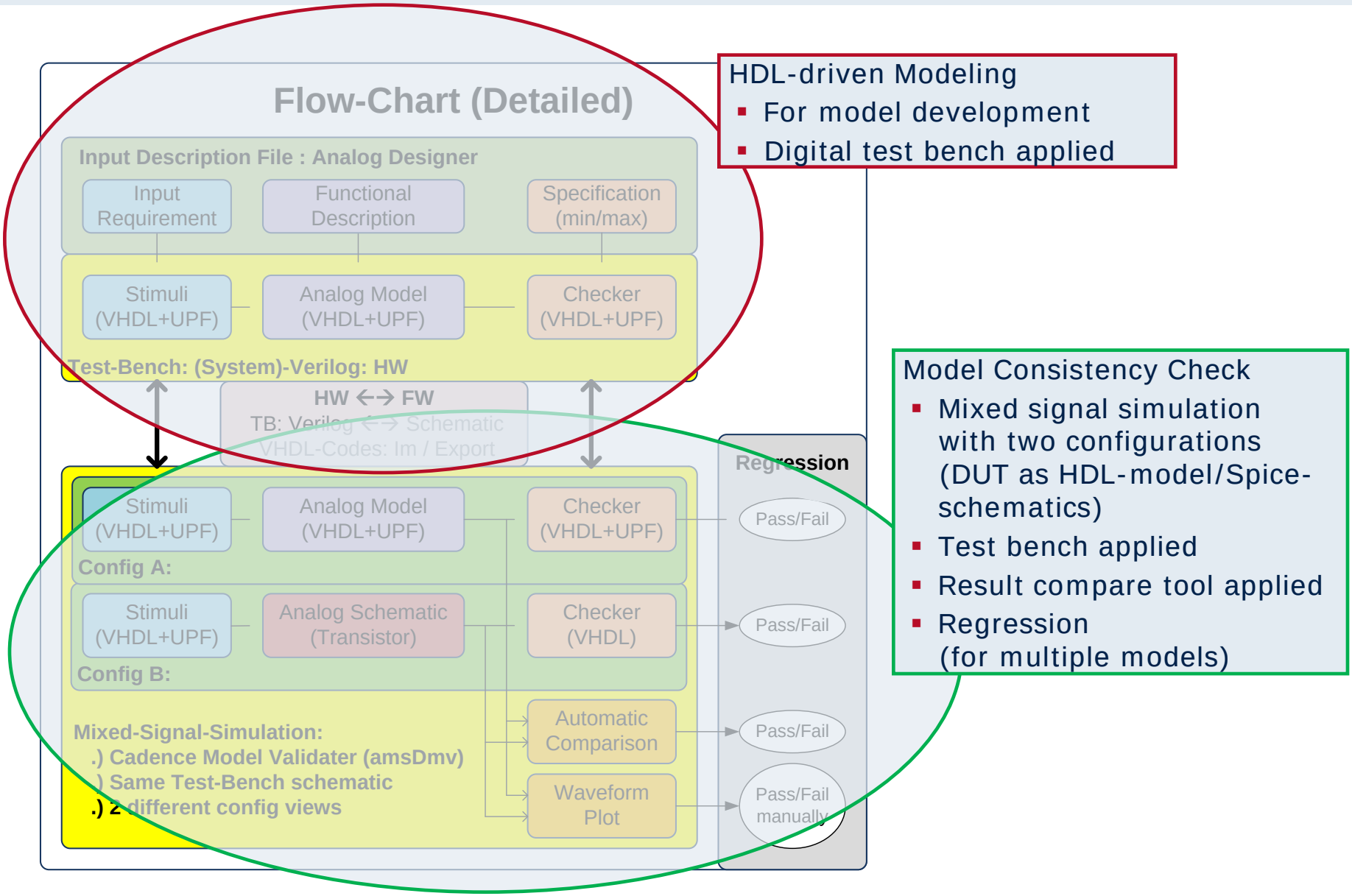
Analog Models – Specification, Features & Verification

- Analog Model Specification
  - interface definitions: list of terminals and power supply connections
  - list of functional features
  - list of limitations due to abstractions
  - validity/application ranges
- Analog Model Features
  - analog behavioral model
  - real-value models, i.e. data types: analog_t, analog_t_vector
  - UPF-support (IEEE.UPF VHDL-package application), `get_supply` for sink-power-connects, `set_supply` for supply sources (i.e. voltage regulator model) or equivalent (System-)Verilog library capability
  - validity-/out-of-range checks and reporting (top-level requirement)
- Test Bench Features – Two modes:
  - behavioral mode without out-of-range stimulus for model consistency verification (this talk)
  - behavioral mode incl. out-of-range stimulus for model feature verification

Model Consistency Concept – General Overview



Analog Model Development and Analog-centric Verification



HDL-driven Modeling
- For model development
- Digital test bench applied

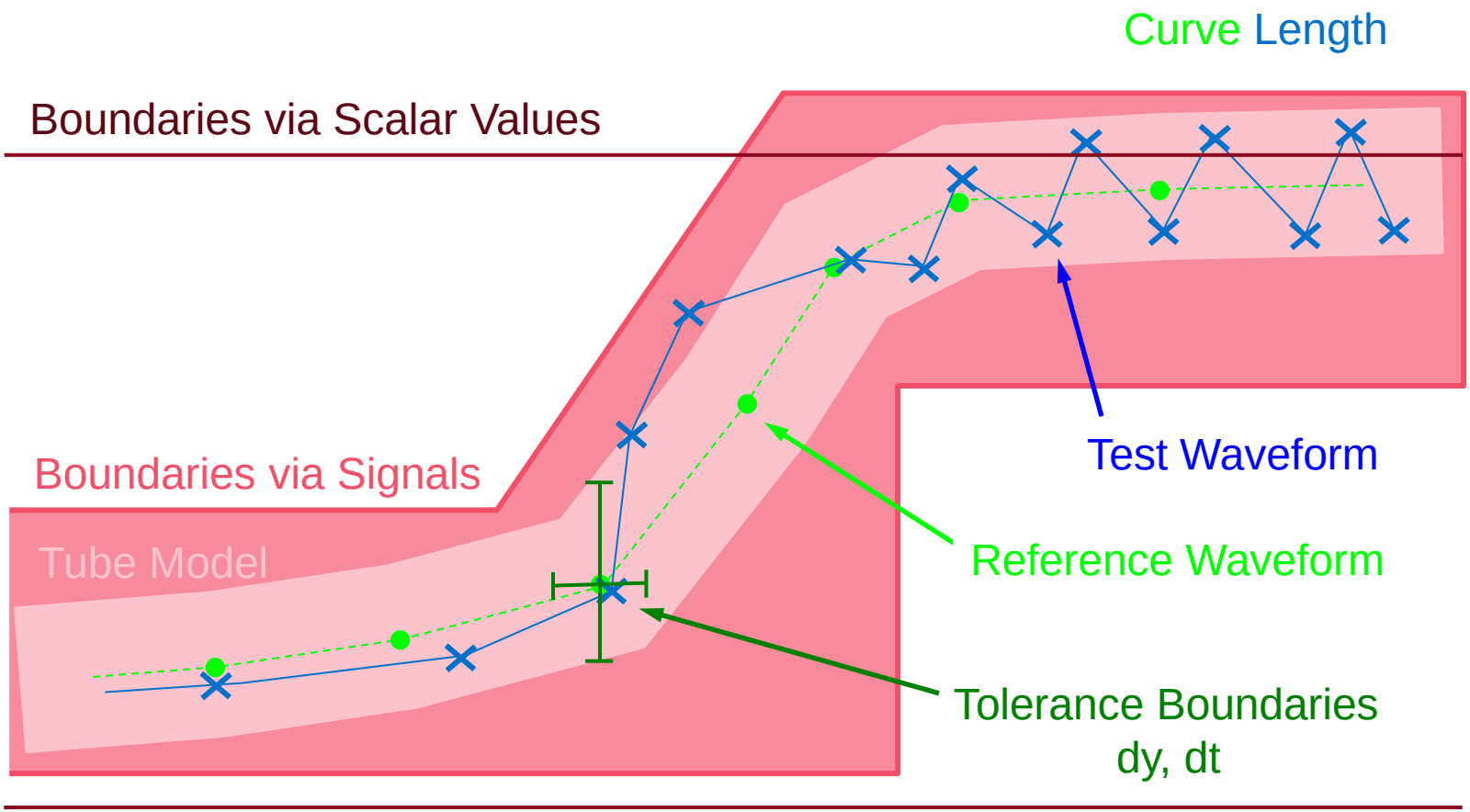
Model Consistency Check
- Mixed signal simulation with two configurations (DUT as HDL-model/Spice-schematics)
- Test bench applied
- Result compare tool applied
- Regression (for multiple models)

Waveform Comparison – General Approaches

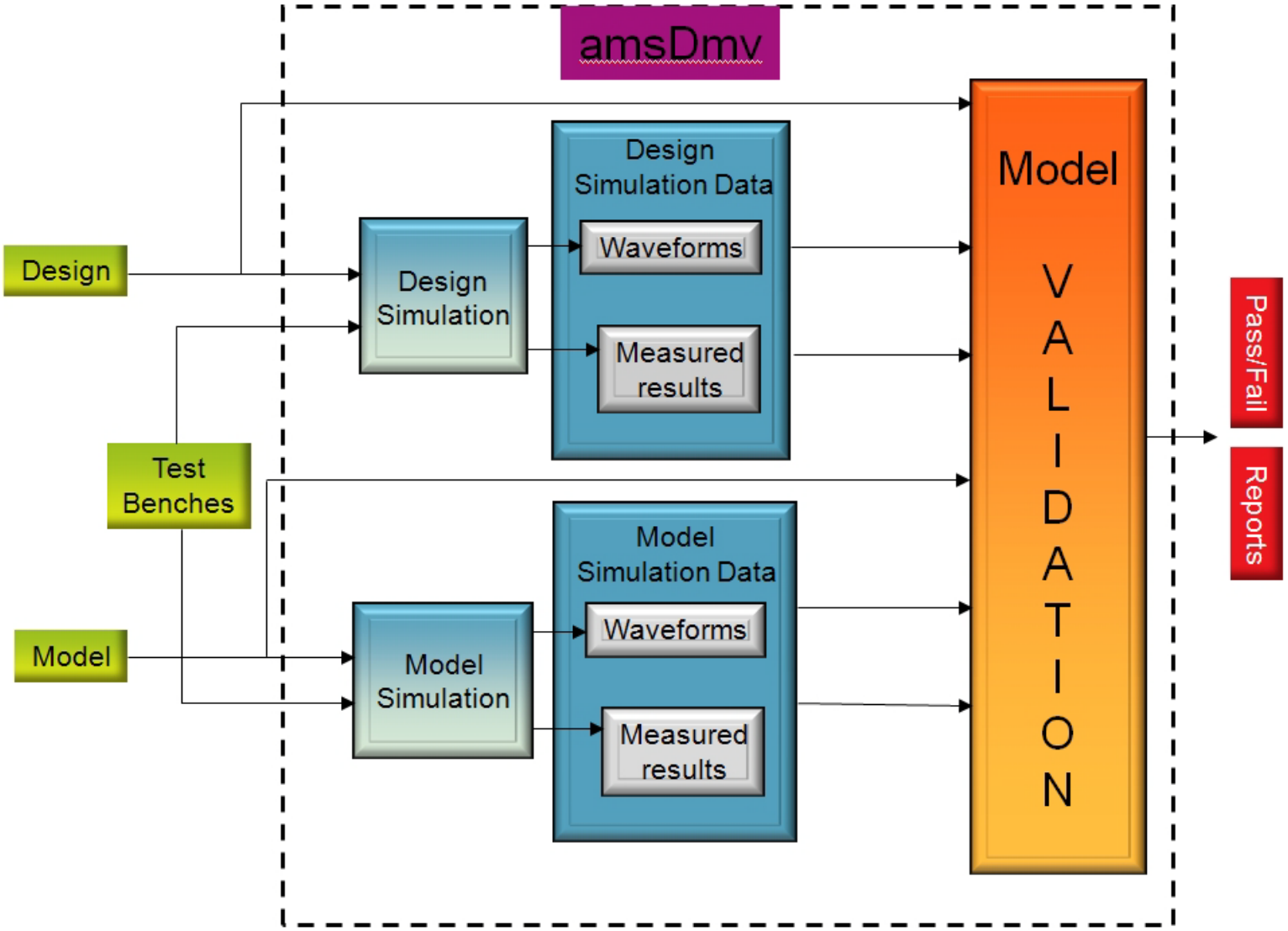
- Simulator internal - on-the-fly
  - Measurements with specification limits
  - Assertions (block-level self-checking)
  - Vector files with expected outputs (VEC, EVCD format)
- Simulator external - post-processing
  - Specifications (Analog Regression Runner, ADE-XL)
  - Vector-valued Measurements (series of time points)
  - Waveform (result data points) post-processing
- Signal Types
  - DC, AC, TRAN, HB, Noise
  - analog, digital

➔ Focus:
time-domain (transient) simulations with varying time steps

Transient Comparison Algorithm – Basic Ideas



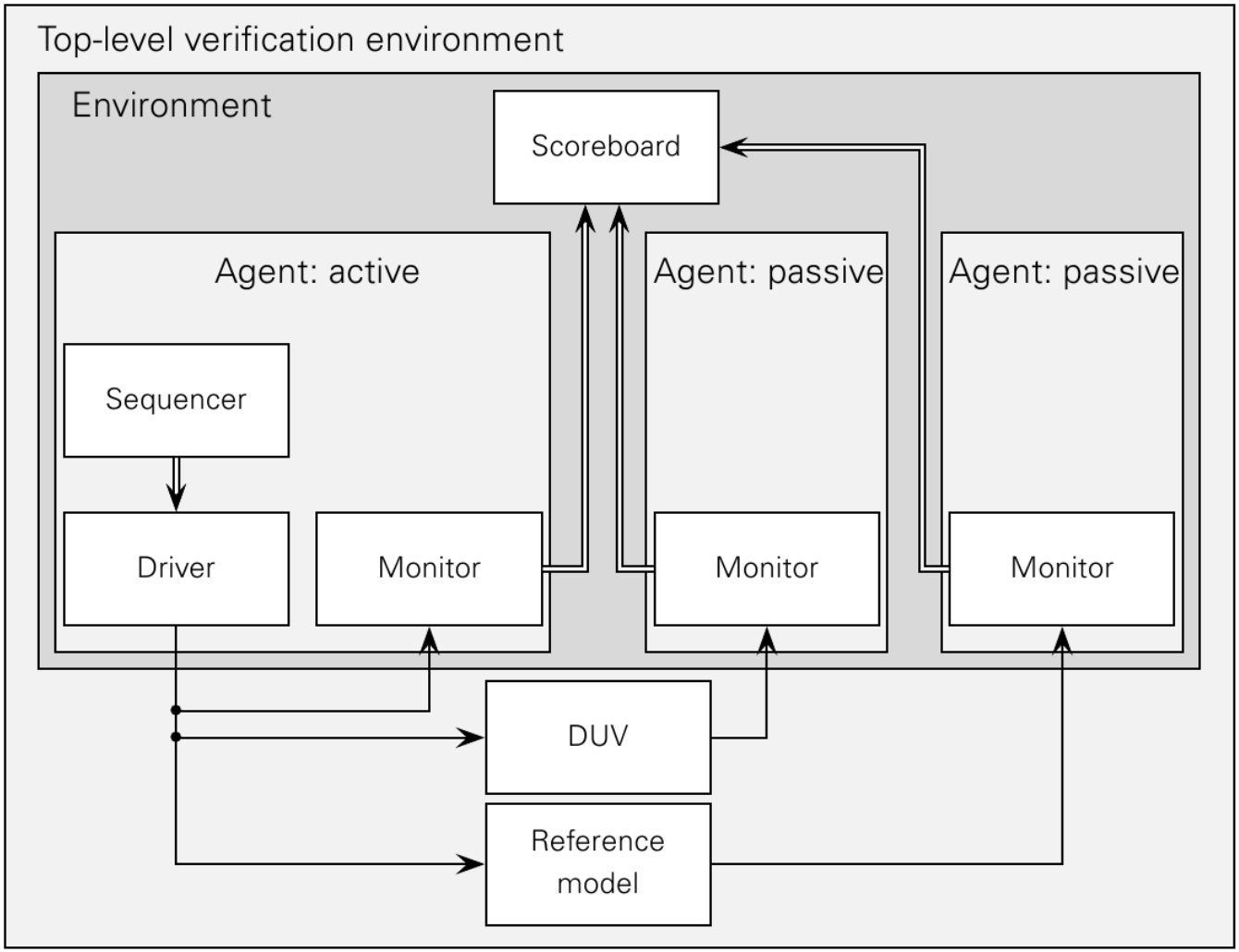
Automation of Analog Simulation and Waveform Compare Flow – Cadence Solution amsDMV



Digital-centric Model Verification

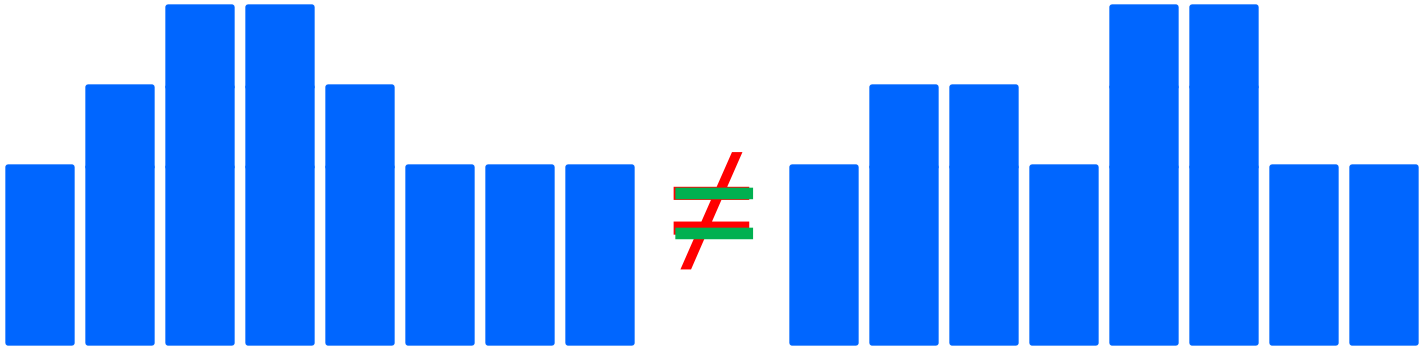
- CAD Framework for Digital-centric Model Verification
  - UVM (Standard Universal Verification Methodology) as basis
  - IFX analog verification extensions **A-UVM**
  - Design and Verification Conference, DVCON 2014
    Automated Comparison of Analog Behavior in a UVM Environment,
    Sebastian Simon, Alexander E. Rath, Volkan Esen, and Wolfgang Ecker, Infineon Technologies AG
- A-UVM Features
  - ✓ Analog transactions
  - ✓ Constrained-random analog stimulus
  - ✓ Monitoring of analog behavior
  - ✓ Basic checks for comparing analog transactions

Automation of Digital Simulation and Waveform Compare Flow – UVM Example



Digital Comparison Algorithm – Basic Idea
Metric: Earth Mover's Distance

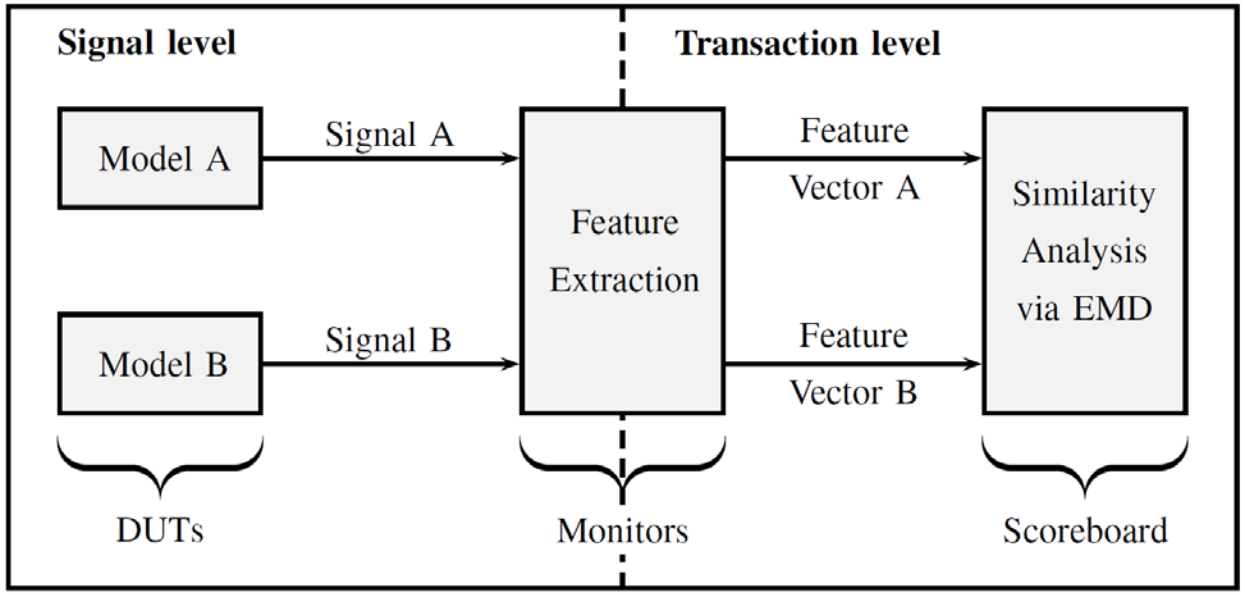
- Approach to measure the distance between two distributions
- Visualization: transportation of soil from one pile to another



- Work += distance x amount: $A = \sum_{i=1}^{n} \sum_{j=1}^{n} d_{ij} f_{ij}$
- Find minimum flow which equalizes distributions

→ optimization problem has to be solved

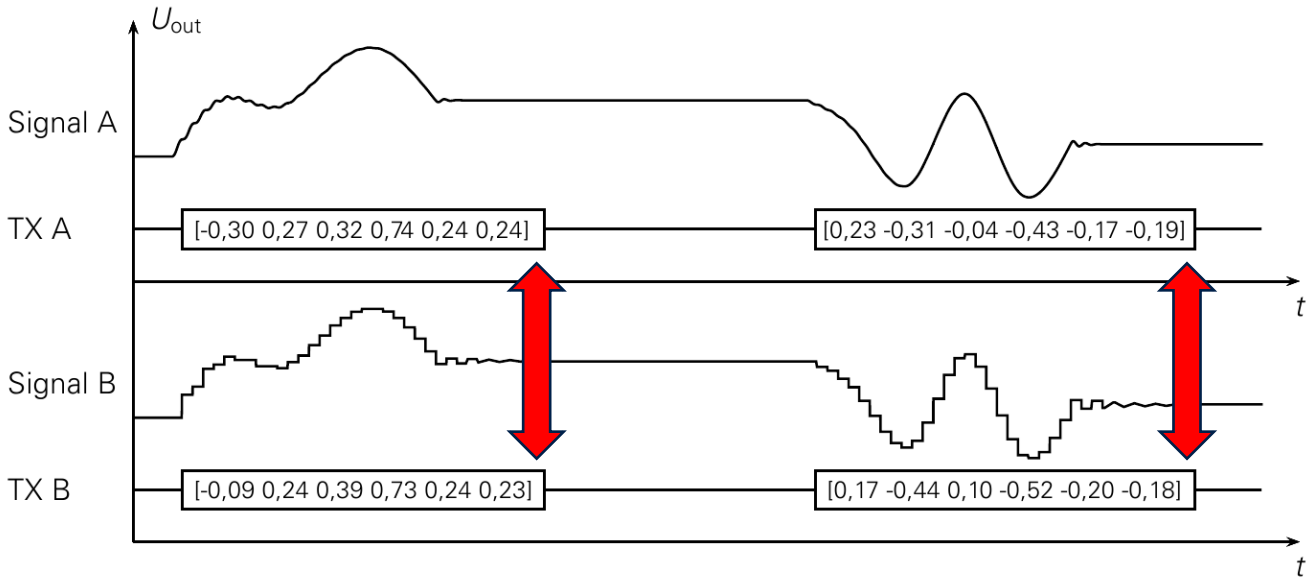
Automation of Digital Simulation and Waveform Compare Flow – Integration Concept I



- Generic monitors extract analog transactions = feature vectors
- Extraction types: sampling, FOURIER, sine, ramp, jump
- Similarity analysis implemented with SystemVerilog DPI-C

Automation of Digital Simulation and Waveform Compare Flow – Integration Concept II

- Analyzing one pair of transactions results in exactly one value s_{EM} for the Earth Mover's Distance



- Range: $0 \leq s_{EM} \leq 1$ (where 1 implies a full match)
- Basic idea for regression: defining a lower bound for s_{EM}
- Once s_{EM} falls below this bound, the regression test fails and the regarding transactions can be examined

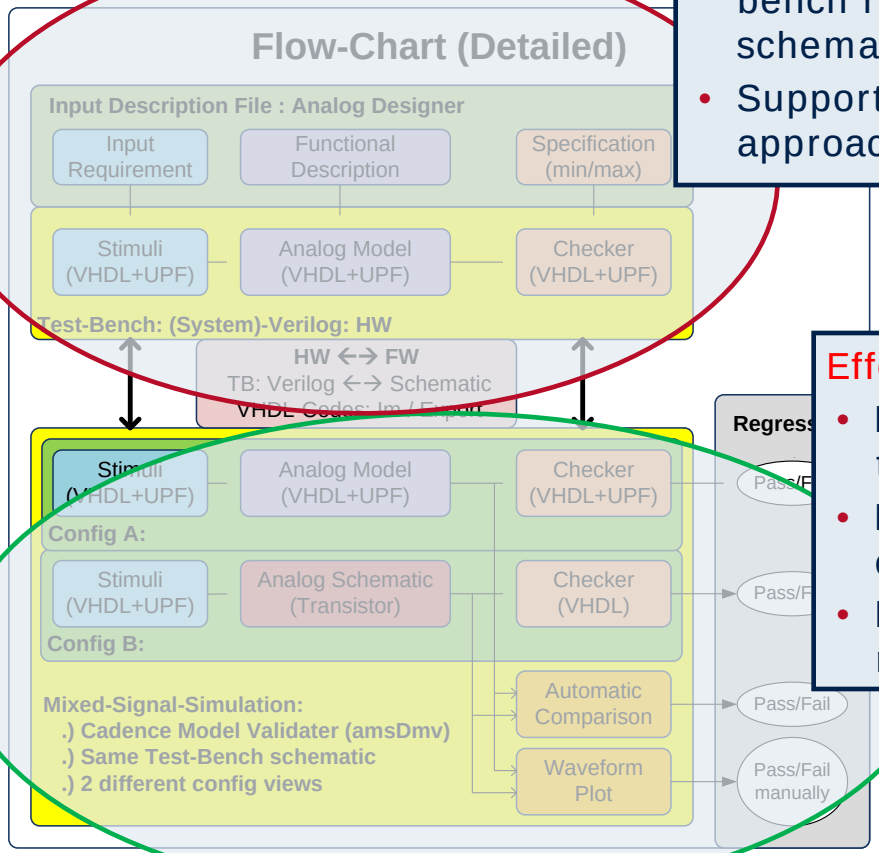
Analog-centric vs. Digital-centric Comparison of Model Verification Features

Features	Analog-centric	Digital-centric
Test Bench	schematic result data access manual stimulus generation load conditions	A-UVM (driver, monitor, score board) constrained random stimulus generation reuse from digital functional verification
Compare Setup	waveform pre- and post-processing A/D, D/A conversion GUI based tolerance definitions	extraction type (sampling, Fourier, ...) extraction parameter (sampling rates) lower bound for metric
Debugging	messages in log file analog waveform viewer: • automatic signal pair selection • violation area reports and marker • (visualization of tolerance boundaries)	messages in log file digital waveform viewer: • manual signal pair selection • digital signal check via SimVision
Regression	amsDmv Analog Regression Runner (each in Analog Automation Env.)	UVM in Digital Regression Runner vManager
Documentation	automated via Regression detailed reporting of violations	automated via UVM/Regression

Analog-centric vs. Digital-centric Comparison of Use Models

Efforts for Digital-centric Model Verification:

- Full reuse potential of existing digital test bench
- No transfer to full-custom Design Flow required
- No additional analog test bench setup (digital test bench reuse and substitution of HDL by Spice-schematics for mixed-signal configuration)
- Support for UPF enabled by default in HDL-driven approach



Efforts for Analog-centric Model Verification:

- Requires semi-custom to full-custom Design Flow transfer of all HDL-elements
- Requires to setup a second simulation in analog environment (analog test bench schematics)
- No UPF support (manual and expert-level effort required for UPF-enabling)

Case Study 2: Analog Model Validation

Deployment Issues:

- Vertical Verification Domain-Crossing:
  Integration of model validation into different regression environments lead to different use models
- Horizontal Verification Domain-Crossing:
  Analog-centric / Digital-centric Model Verification requires different domain know-how
- Implicit/Explicit:
  Converting design block know-how to model specification and check scenarios with reasonable coverage based on validity/application ranges specified in verification planning
- Interactive/Automation:
  Validation acceptance criteria need to be defined explicitly; additional effort for out-of-range checks

Agenda

- Analog Verification - Context/Components/Coordination
- (R)evolution of Industrial Verification Ecosystem
- Analog Verification Planning & Sign-Off
- Requirements on Verification Software & Integration
- Case Study 1: Safe Operating Area Checks
- Case Study 2: Analog Model Validation
- Summary & Final Thoughts

Summary & Final Thoughts

- Analog Verification Coverage
  - definitions/measures currently missing
  - design/topology/check specific
- Formalization
  - binary good/bad decisions not compliant with analog designer reasoning
  - trade-offs and margins usually not covered (but: compare Worst-case Distance Measure for Yield Estimation)
  - converting implicit to explicit knowledge often tricky (compare topology recognition for automatic constraint setup in tool Wicked, MunEDA)

ENERGY EFFICIENCY
MOBILITY
SECURITY

Innovative semiconductor solutions for energy efficiency, mobility and security.