Virtual Platforms for Automotive: Use Cases, Benefits and Challenges

Angela Kramer, Martin Vaupel
Corporate Research and Advance Engineering Robert Bosch GmbH

BOSCH

accellera SYSTEMS INITIATIVE

2014 DESIGN AND VERIFICATION DVCON CONFERENCE AND EXHIBITION EUROPE

Virtual Platforms for Automotive: Use Cases, Benefits and Challenges

Angela Kramer, Martin Vaupel
Corporate Research and Advance Engineering
Robert Bosch GmbH

BOSCH

The importance of software in a modern car

- An estimated 40% of a vehicle‘s cost is determined by electronics and software[1]
- An estimated 90% of all innovations in automotive are driven by electronics and software[1]

Efficient system development is a competitive advantage!

- More lines of code in a modern car than in a jumbo jet
- Increasing cross-linking of functions
- Increasing number of safety-critical applications

Efficient system development is becoming increasingly difficult!

[1] *taken from De Schutter, T. et al. “Better Software. Faster! Best Practices in Virtual Prototyping”. Mountain View: Synopsys, 2014.*

BOSCH

Why and where use virtual platforms?

Issues

- Ambiguous specification
- Late SW/HW integration
- Late feedback of SW developers
- Limited number of HW systems
- Limited system visibility and control
- Hard to set up regressions and automated tests

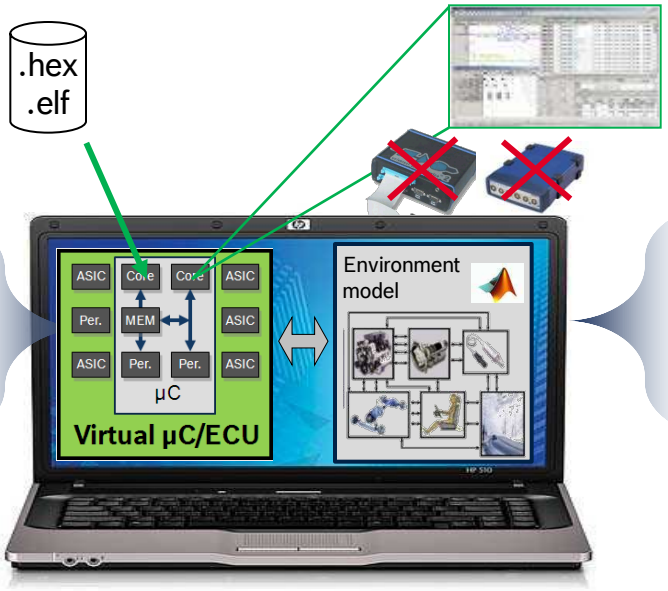
→ ***Real HW is the bottleneck***

BOSCH

Executable models of hardware serve as virtual platform for SW and system development

Execution of **same binary** as on target platform

Simulation of hardware **behavior and timing**



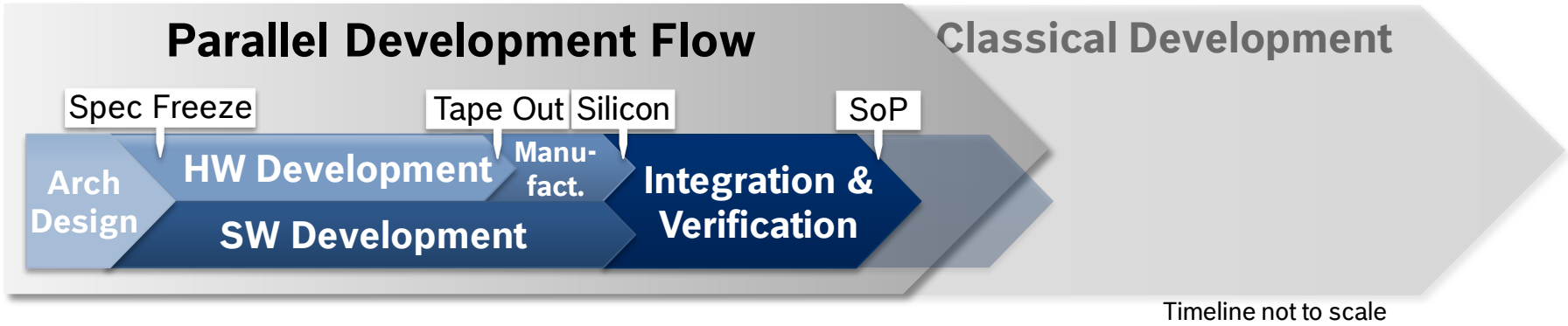
Same tool chain
compiler, debugger, calibration

Embedded in simulated environment (testbench)

Like real HW, but
- Earlier
- Observable & Controllable, Distributable & Scale-able, Repeatable, ...

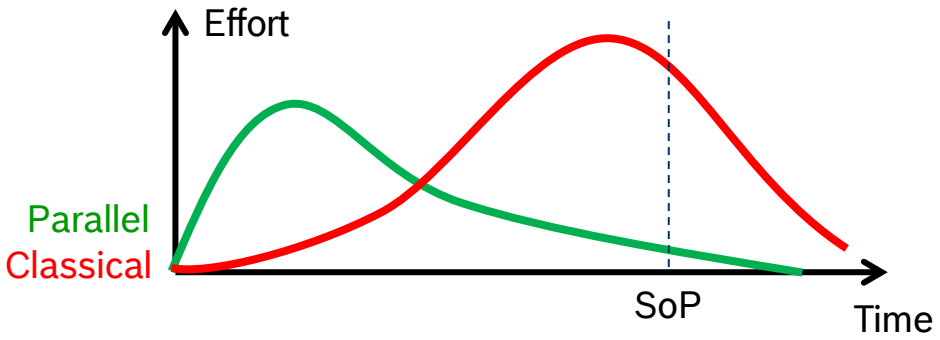
BOSCH

HW/SW Co-Design



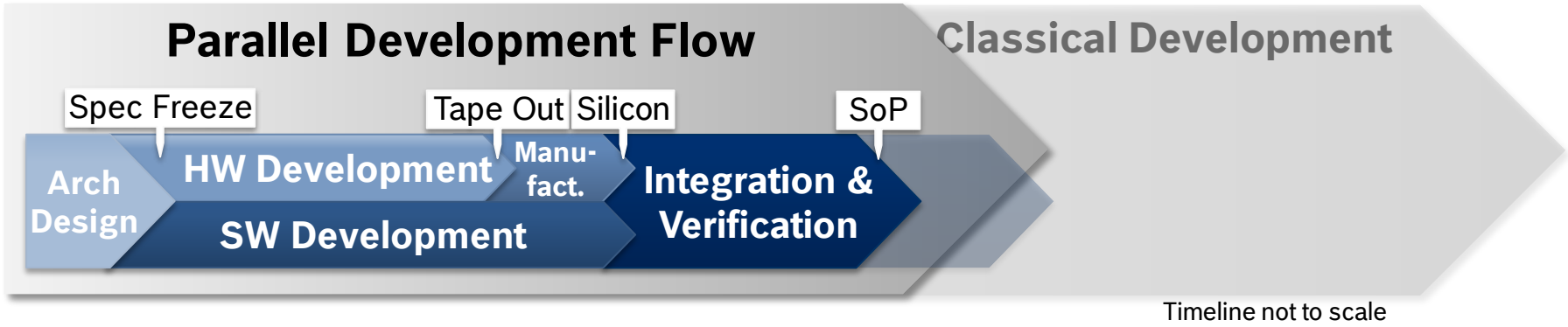
Minimize Risks
- Re-Spins
- Ensure matching HW/SW

Maximize Performance
- MultiCore
- New IP / ASICs



BOSCH

HW/SW Co-Design



Timeline not to scale

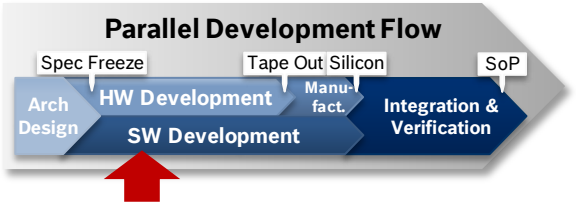
Minimize Risks
- Re-Spins
- Ensure matching HW/SW

Maximize Performance
- MultiCore
- New IP / ASICs

Time-to-Market
- Compiler
- Debugger
- Bring-up 1st silicon
- AUTOSAR base SW
- ComplexDriver

BOSCH

Selected use case: HW-dependent SW



Examples:

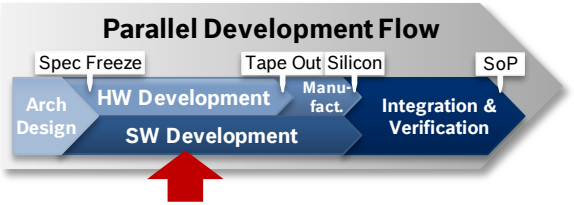
- ➔ Microcontroller Abstraction Layer (MCAL)
- ➔ Complex Drivers

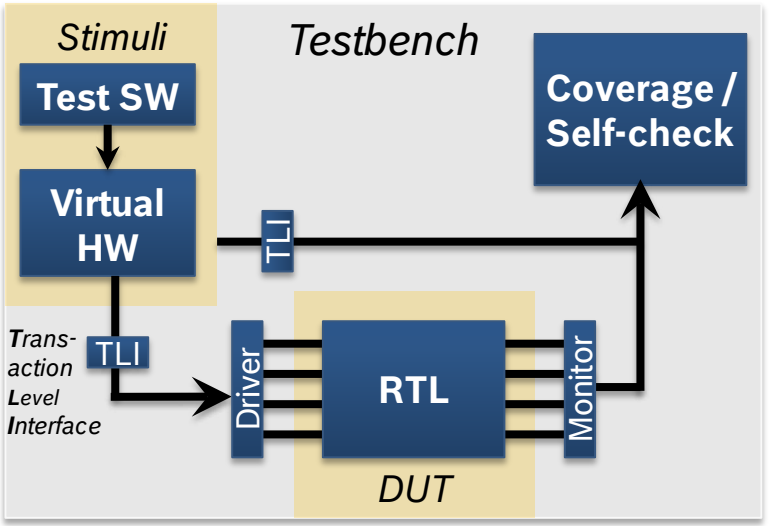
Advantages of virtual platforms:

- \+ Visibility and controllability
  - ➔ Correlate instructions executed in SW with peripheral register accesses & values
- \+ No special equipment required

BOSCH

Selected use case: Virtual HW as stimuli generator





as in Leupers, R. et al. "Virtual platforms: Breaking new grounds". Design, Automation & Test in Europe Conference & Exhibition (DATE), 2012 , pp.685-690, 12-16 March 2012

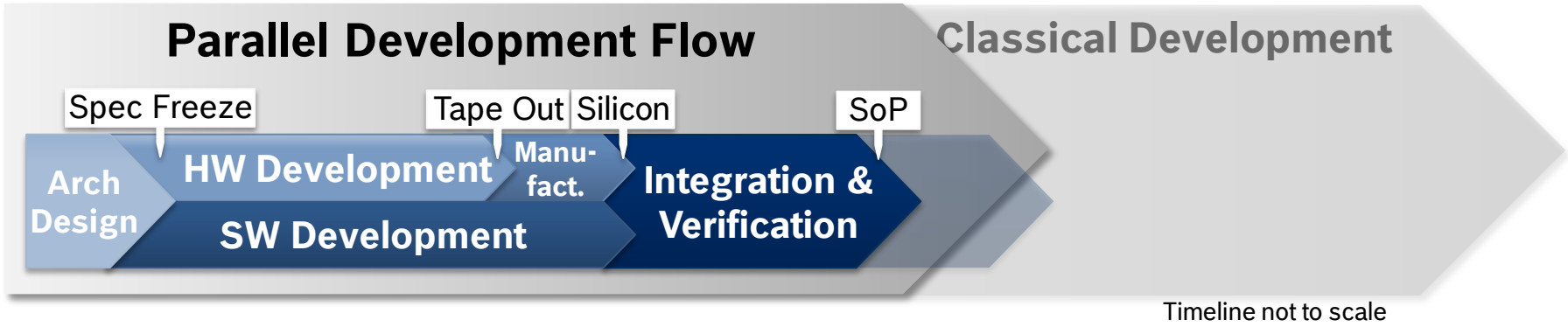
Advantages of virtual platforms:

- \+ Realistic test cases
- \+ Increased test coverage
- \+ Verify design before HW Tape Out
- \+ Less re-spins

➔ Virtual HW can also serve as a golden reference

BOSCH

HW/SW Co-Design



Timeline not to scale

Minimize Risks
- Re-Spins
- Ensure matching HW/SW

Maximize Performance
- MultiCore
- New IP / ASICs

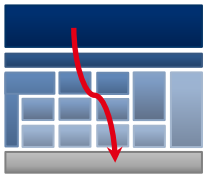
Time-to-Market
- Compiler
- Debugger
- Bring-up 1st silicon
- AUTOSAR base SW
- ComplexDriver

Reduce Cost
- ECU prototypes & lab equipment
- Global development

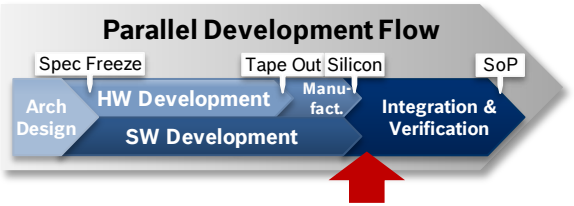
Quality
- Regressions; Fault Injection; Coverage
- (MultiCore) Debugging

BOSCH

Selected use case: SW stack integration & bring-up

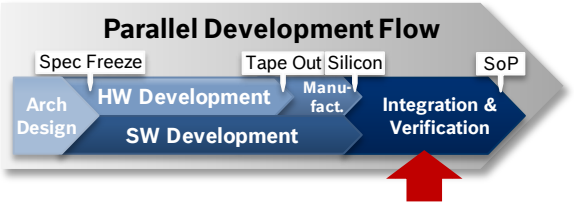


- + MultiCore Debugging
- + Effects are repeatable
- + Check scheduling with function call graph
- + Performance analysis



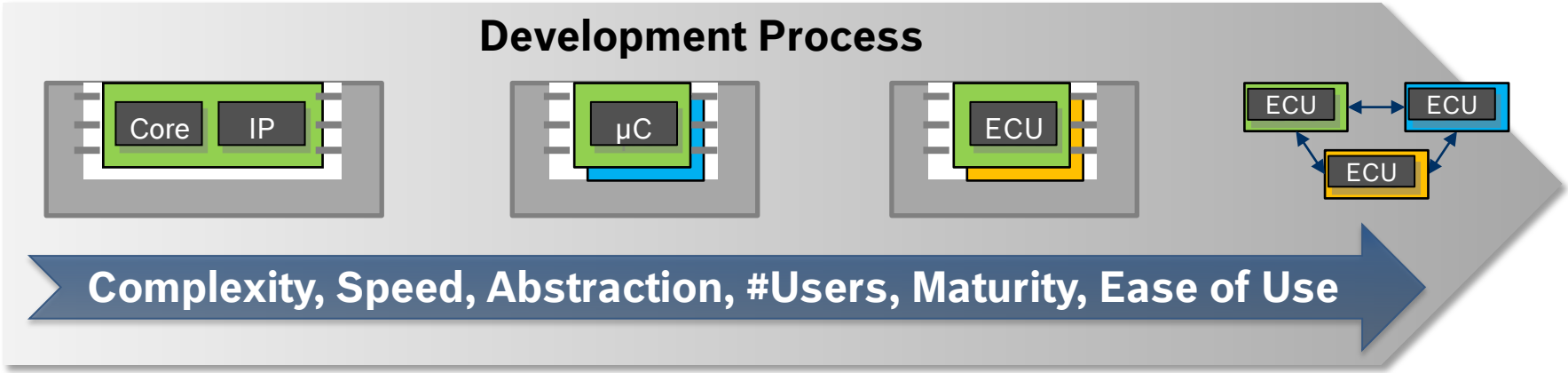
Selected use case: Embedded SW Testing

- + Non-intrusive code coverage
- + Fault injection, corner cases
- + Parallel regression testing



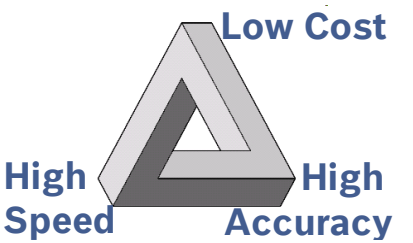
BOSCH

Challenge: Changing Model Requirements



One Model fits all?

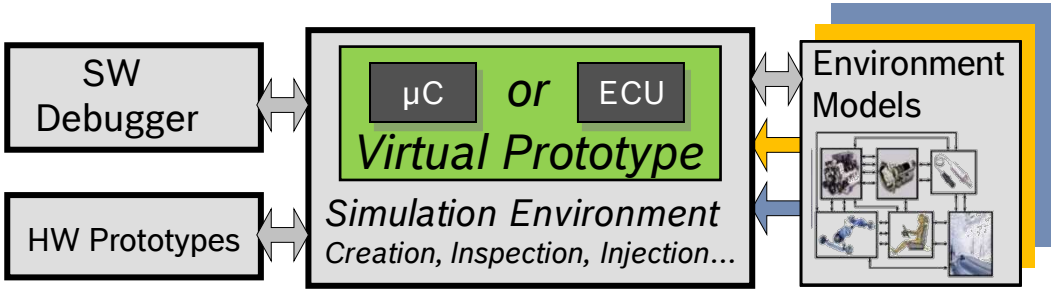
- ***Expand usage range → higher ROI***
  - Reuse of components, configurable subsystems
  - Reduce modeling costs



Albrecht Mayer
Infineon

BOSCH

Challenge: Model applicability



Simulation Environment:

- ***Models have to be tool-independent*** (create here, use there)

SW tool environment and development processes:

- Reuse of test cases and scripts; for real HW, too
- Connections to SW tools and existing simulations
- ***Standards for automotive tool interfaces needed***

Quality:

- Successful abstraction depends on application (expected behaviour)
- Very close interaction with HW provider needed for verification

Challenge: Model Cost

Missing modelling standards hinder cost-efficient usage!

Existing standards help (SystemC, TLM2.0) but not sufficient

➔ ***Standard for automotive model interfaces needed***

- *SPI, CAN, LIN*, FlexRay, Ethernet, ...

BOSCH

Further Challenges

Hundreds of variants

- Robust variant and version handling
- Automated model assembly technology

Long lifetime of product generation

- Providing stable tool environment
- Complex legacy: HW, SW, tools and processes

BOSCH

Summary

Benefits of virtual platforms:

+ Develop SW before HW is available
+ Efficient debugging
+ More thoroughly test functional safety aspects

Driving incentives:

➔ Keep costs of SW development and testing low
➔ TTM is becoming increasingly important
➔ Maintaining high quality

Challenges:

➔ Simulation speed vs. accuracy
➔ Heterogeneity of use cases and collaboration models
➔ Match existing development processes and tools
➔ Standards on automotive interfaces of models and tools needed

Questions?