MATLAB EXPO 2016
KOREA

4월 28일 (목)

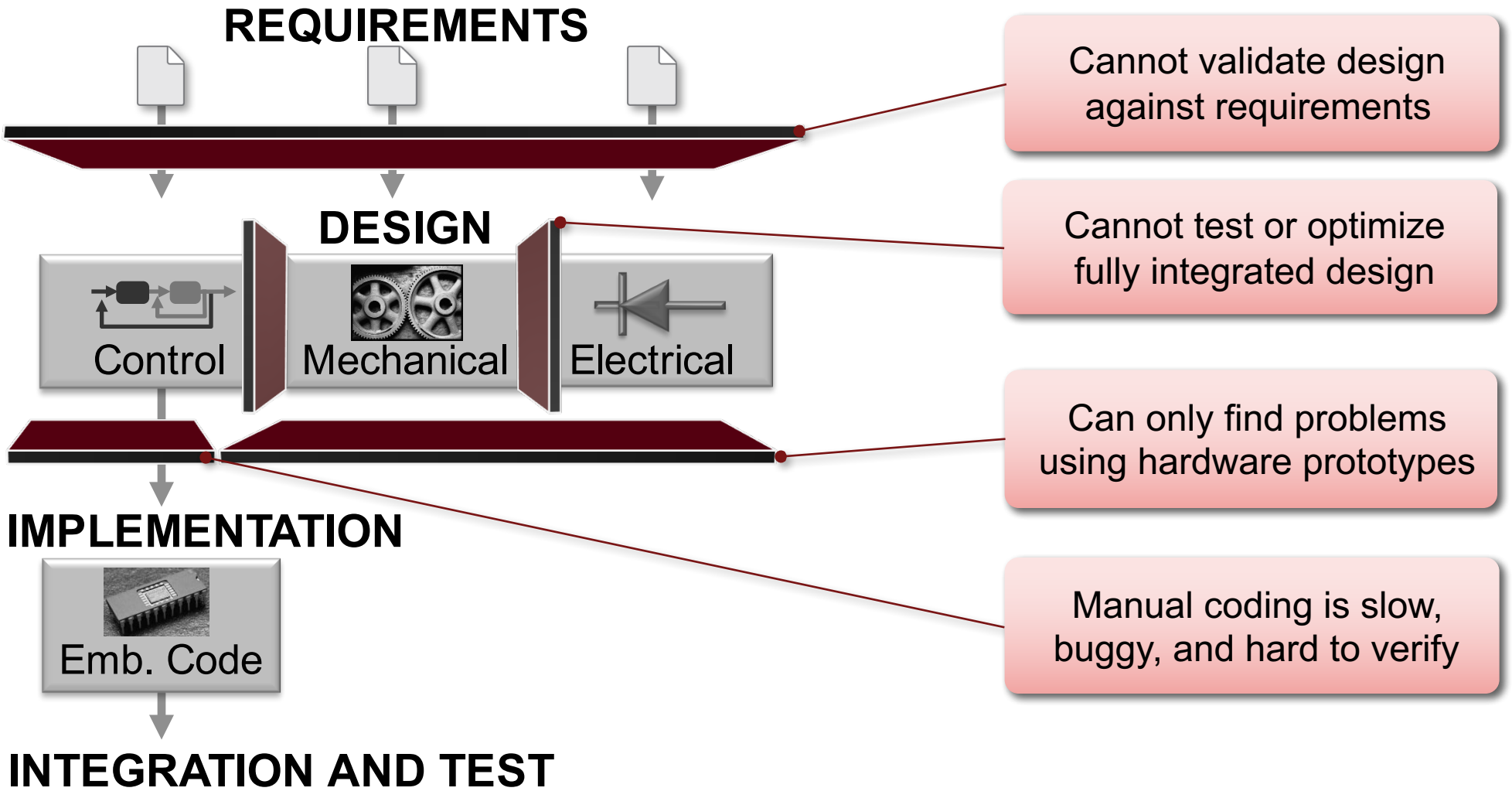
등록 하기 matlabexpo.co.kr

임베디드 시스템 개발을 위한 모델링 및 시뮬레이션

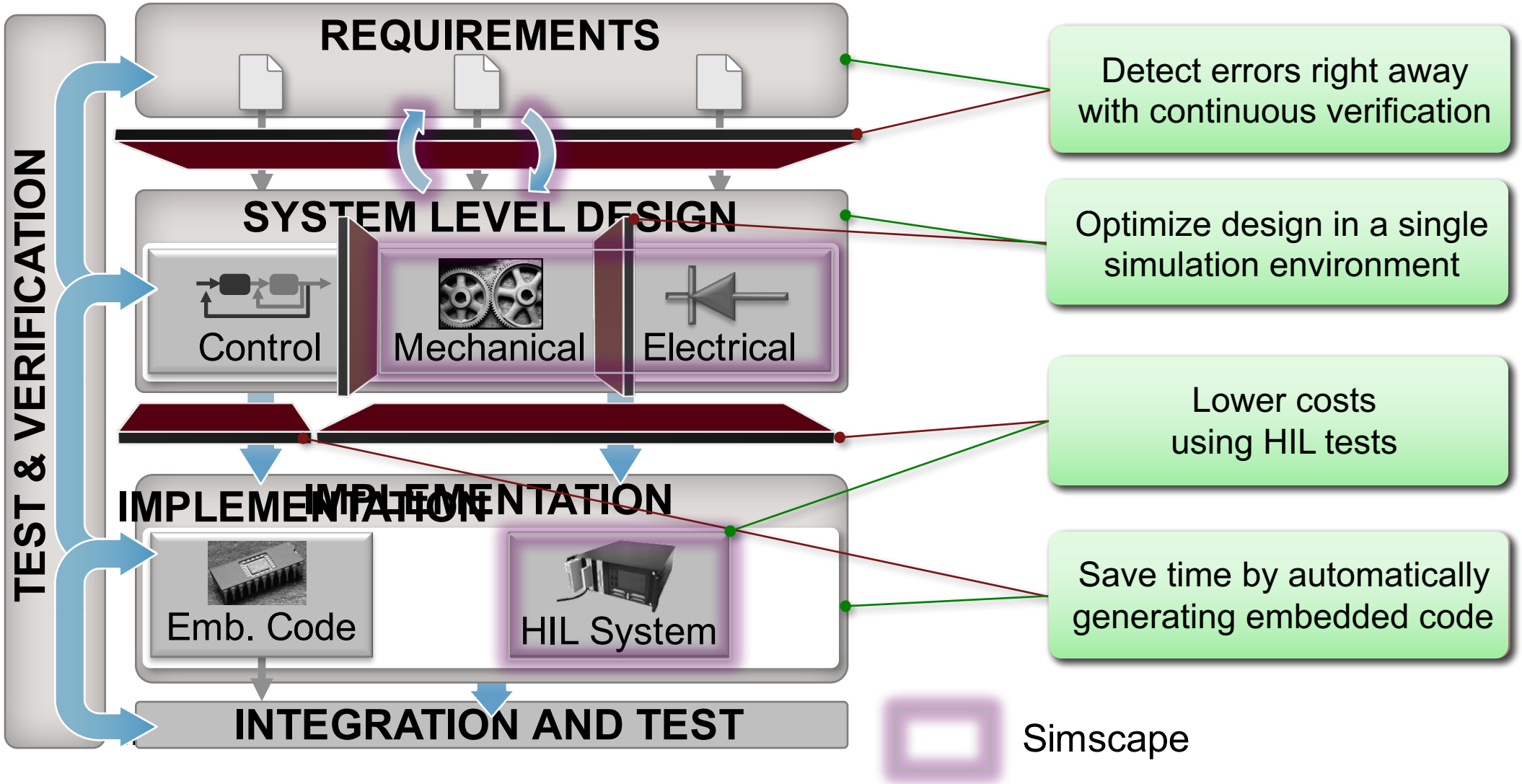
Young Joon Lee
Principal Application Engineer

Model-Based Design

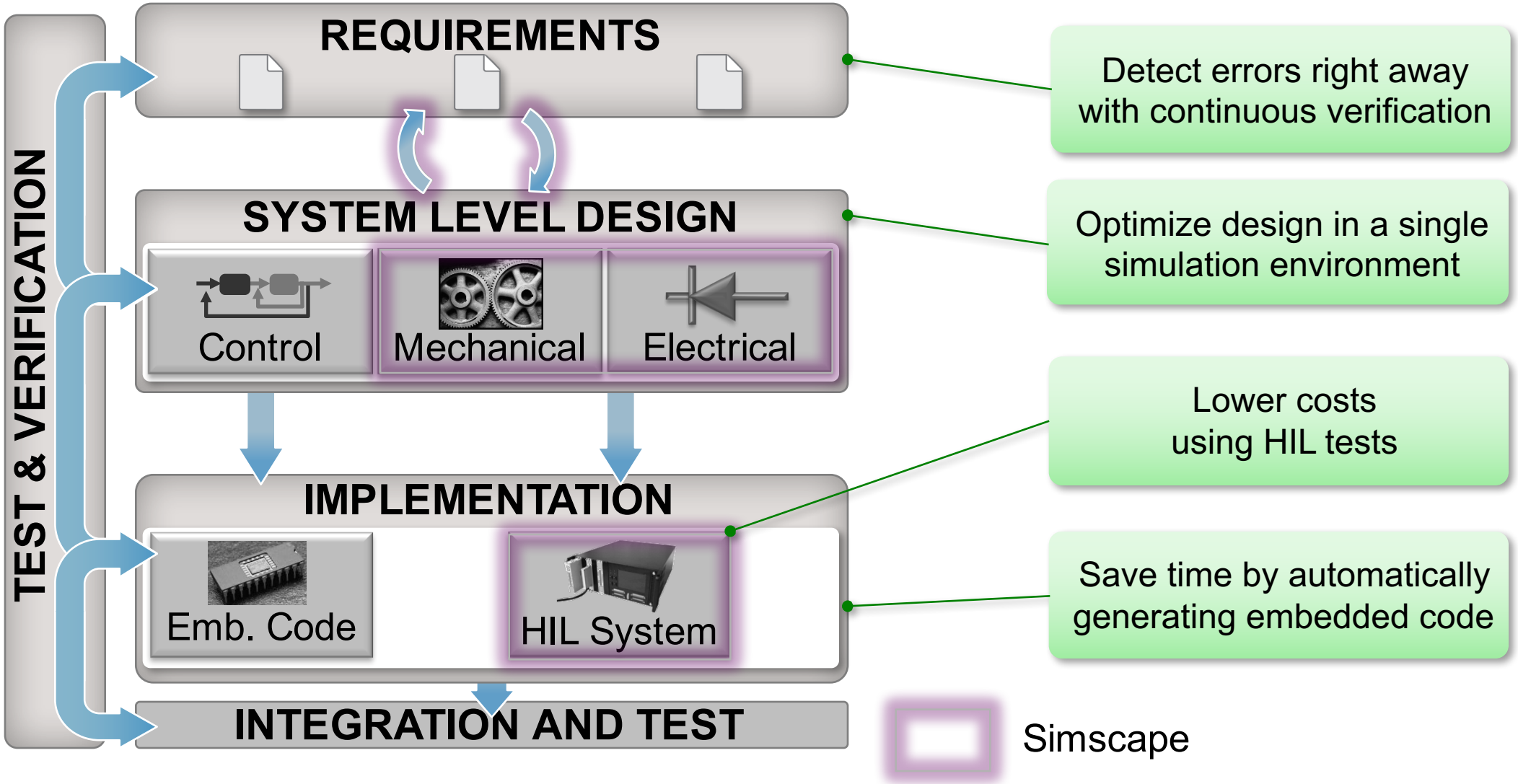
Traditional Design Process



Model-Based Design

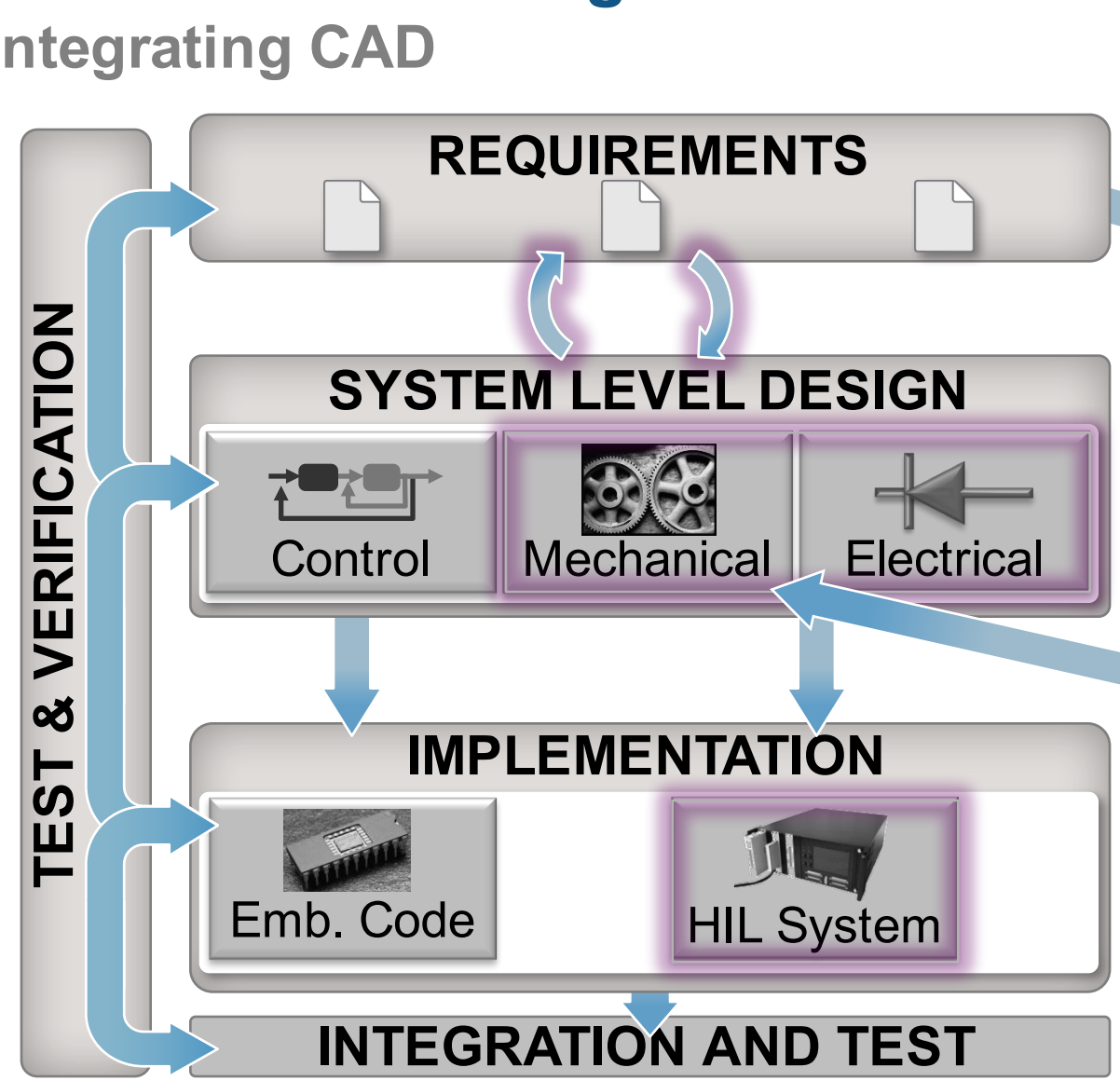


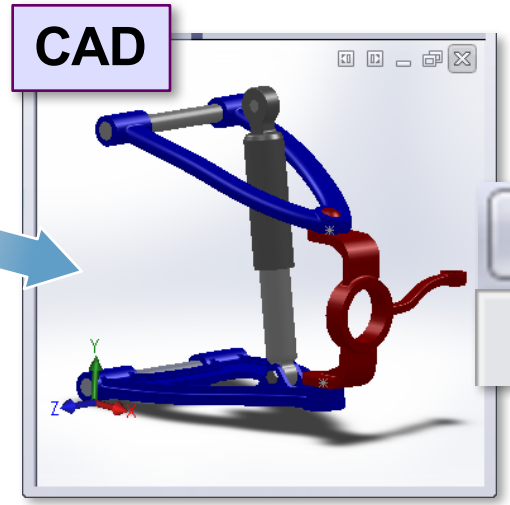
Model-Based Design



Model-Based Design

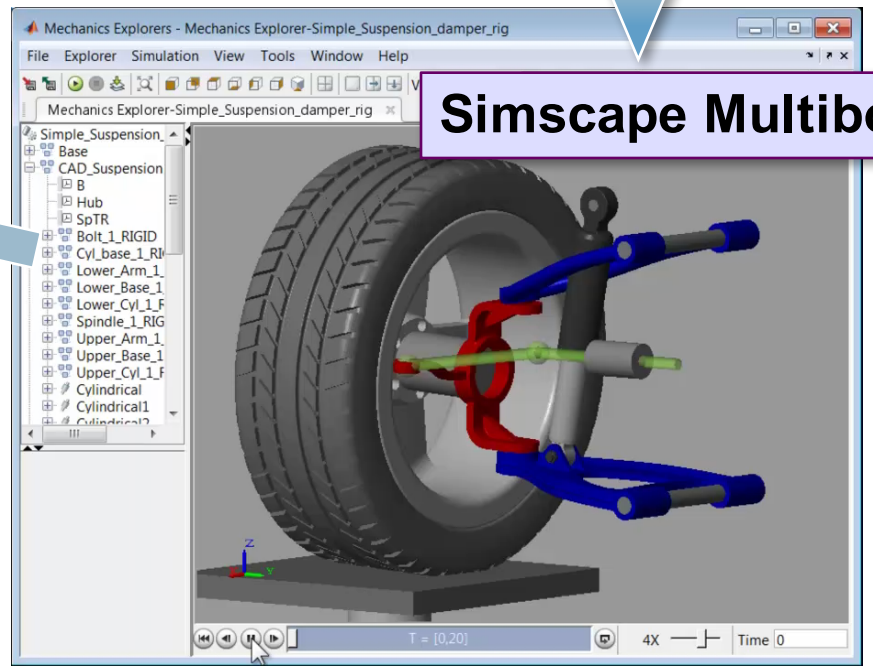
Integrating CAD





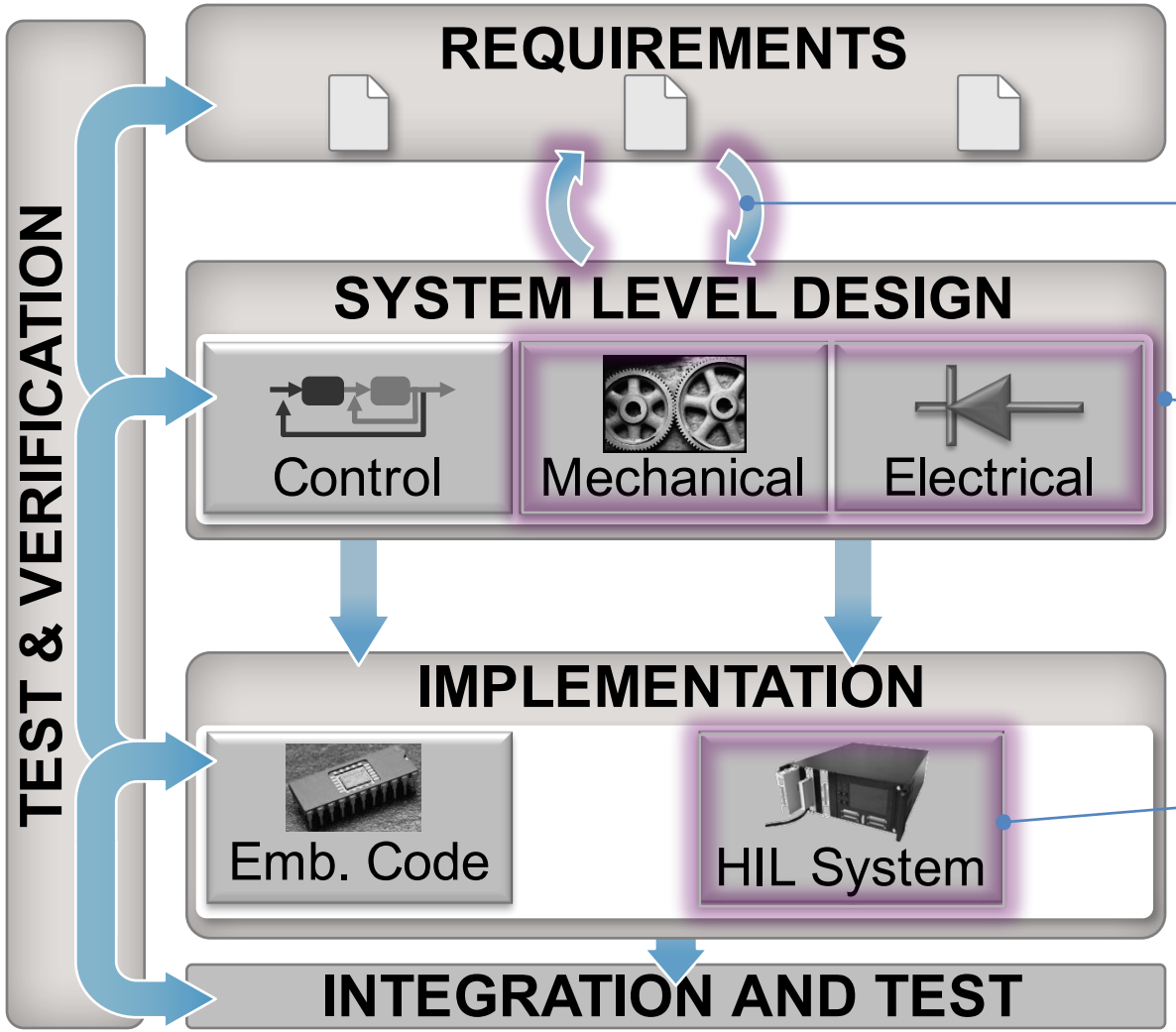
Simscape Multibody Link

Export



Model-Based Design

Key Strength: Integration in MATLAB and Simulink



Simscape Key Strengths

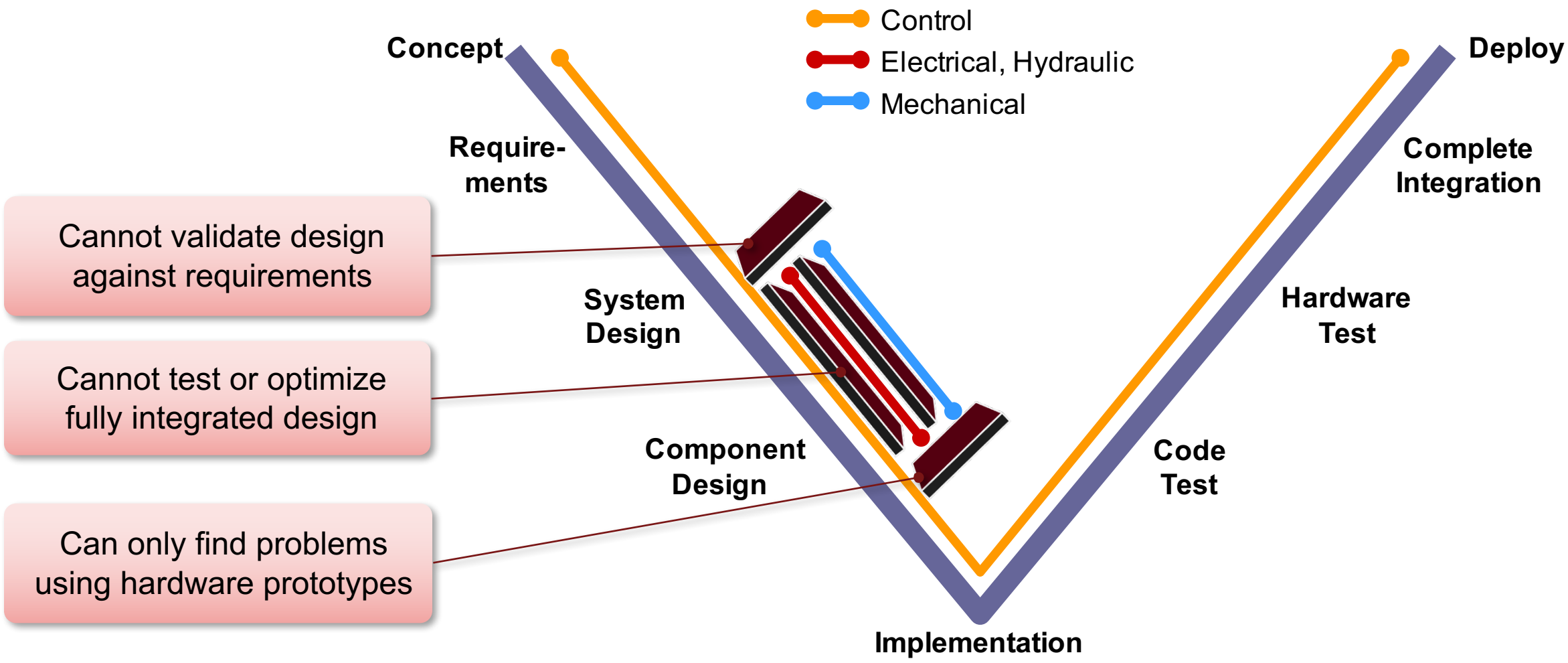
Nearly any task can be automated
Rapid testing via parallel computing

Integration in Simulink enables powerful control design capabilities

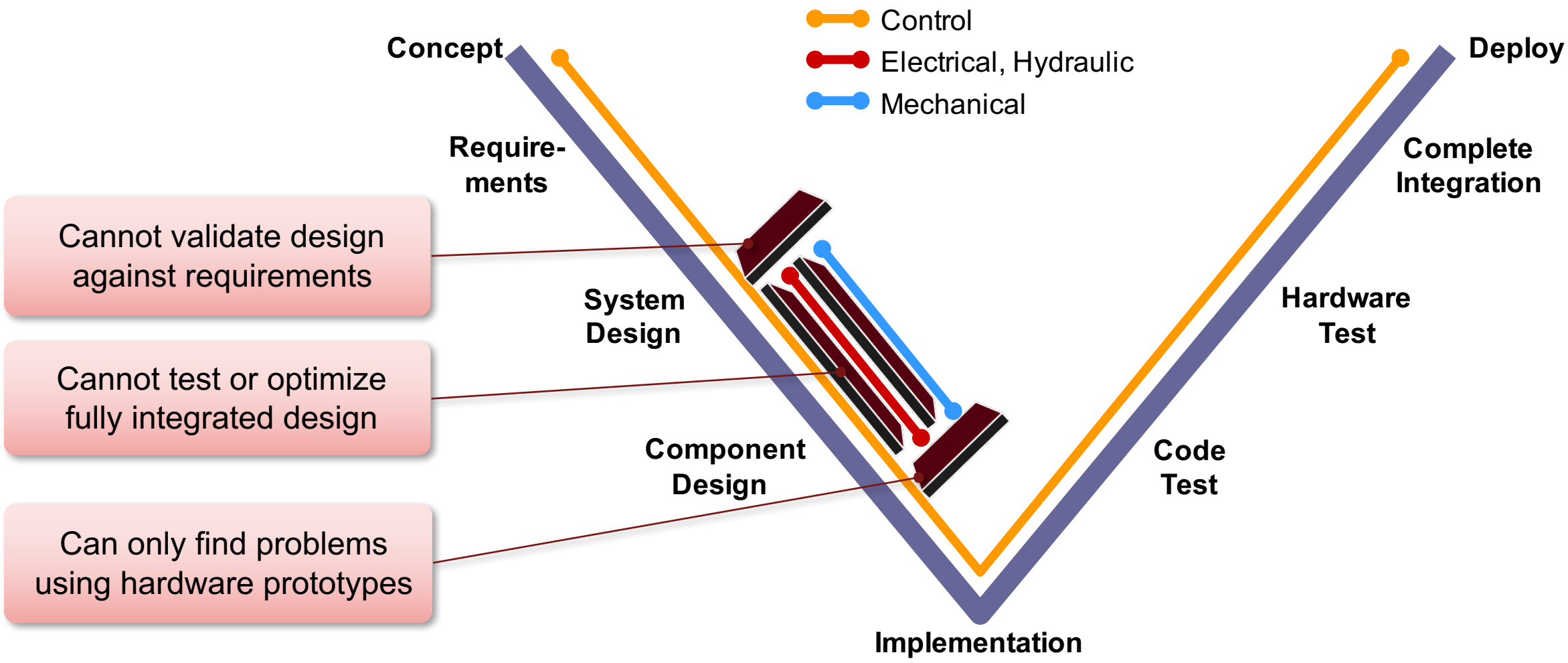
C code generation capabilities

Model-Based Design V-Diagram

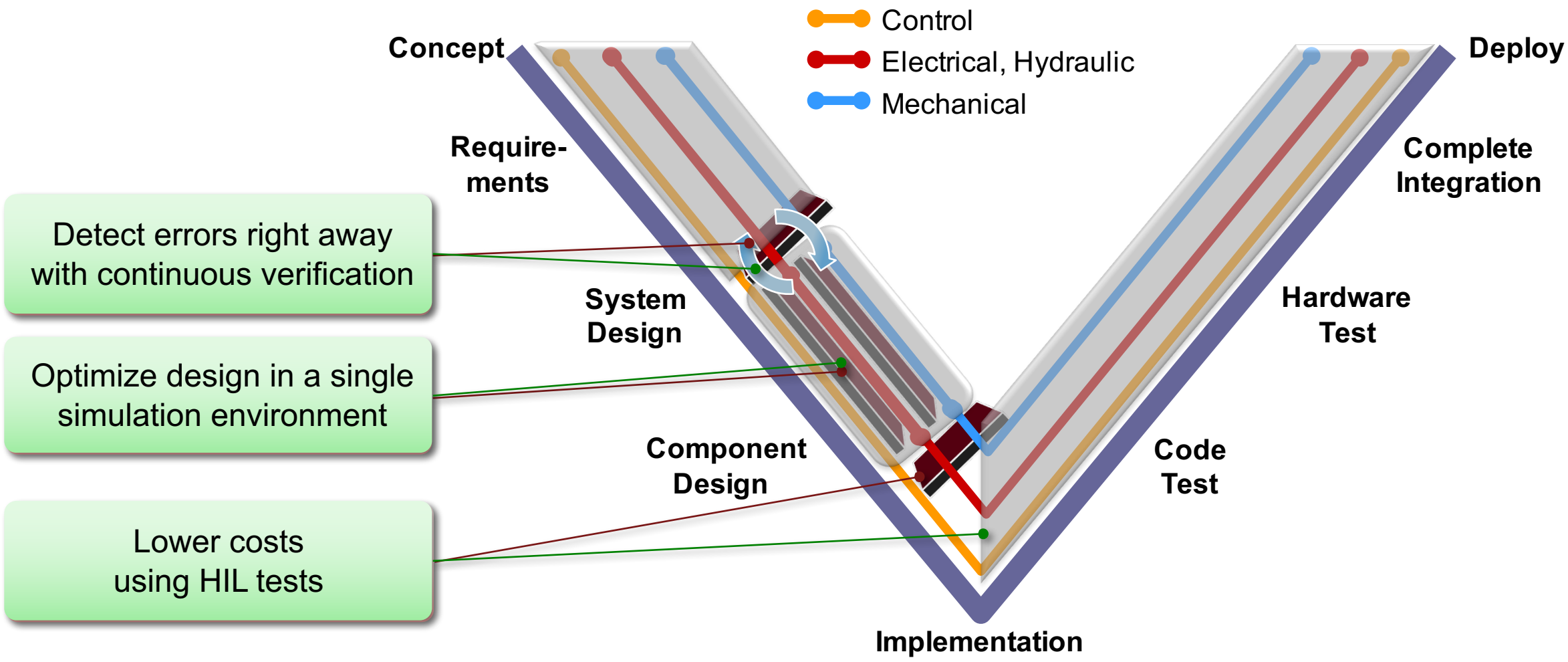
Traditional Development Process



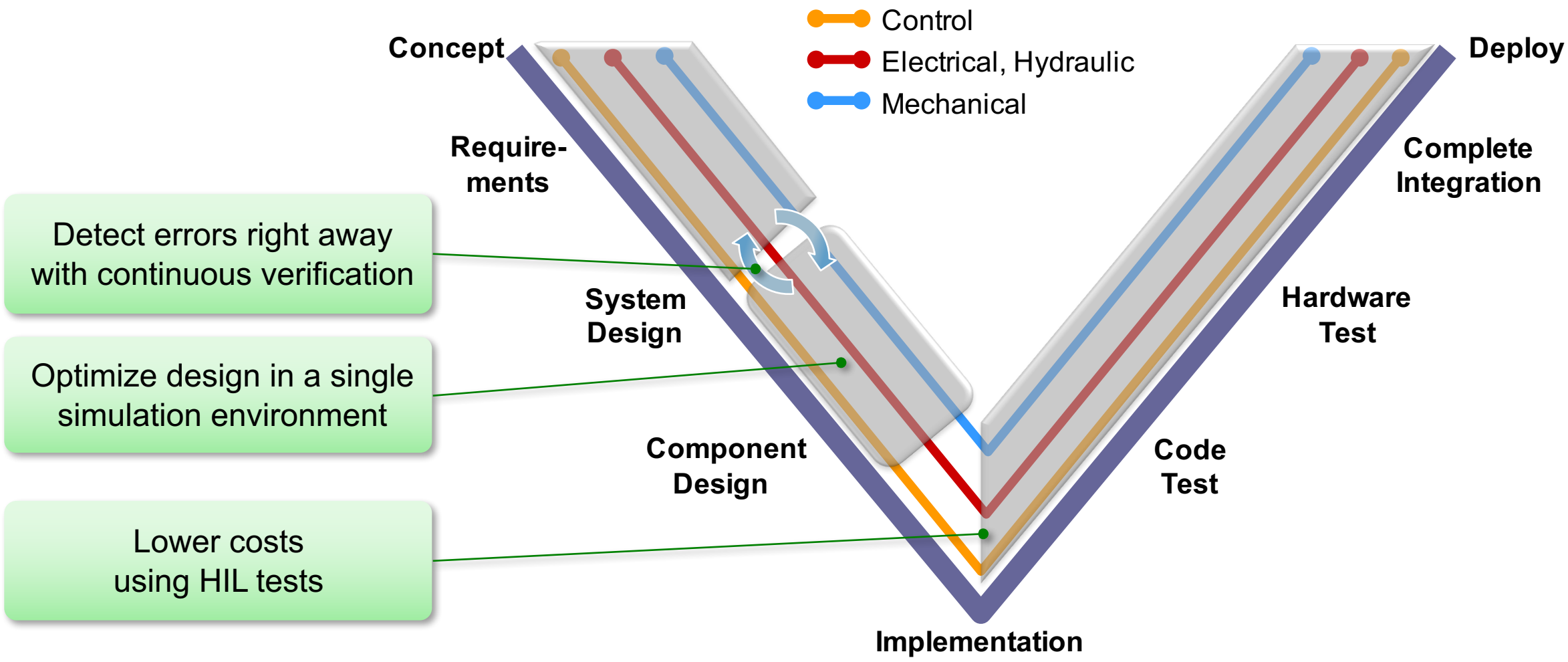
Traditional Development Process



Model-Based Design Process

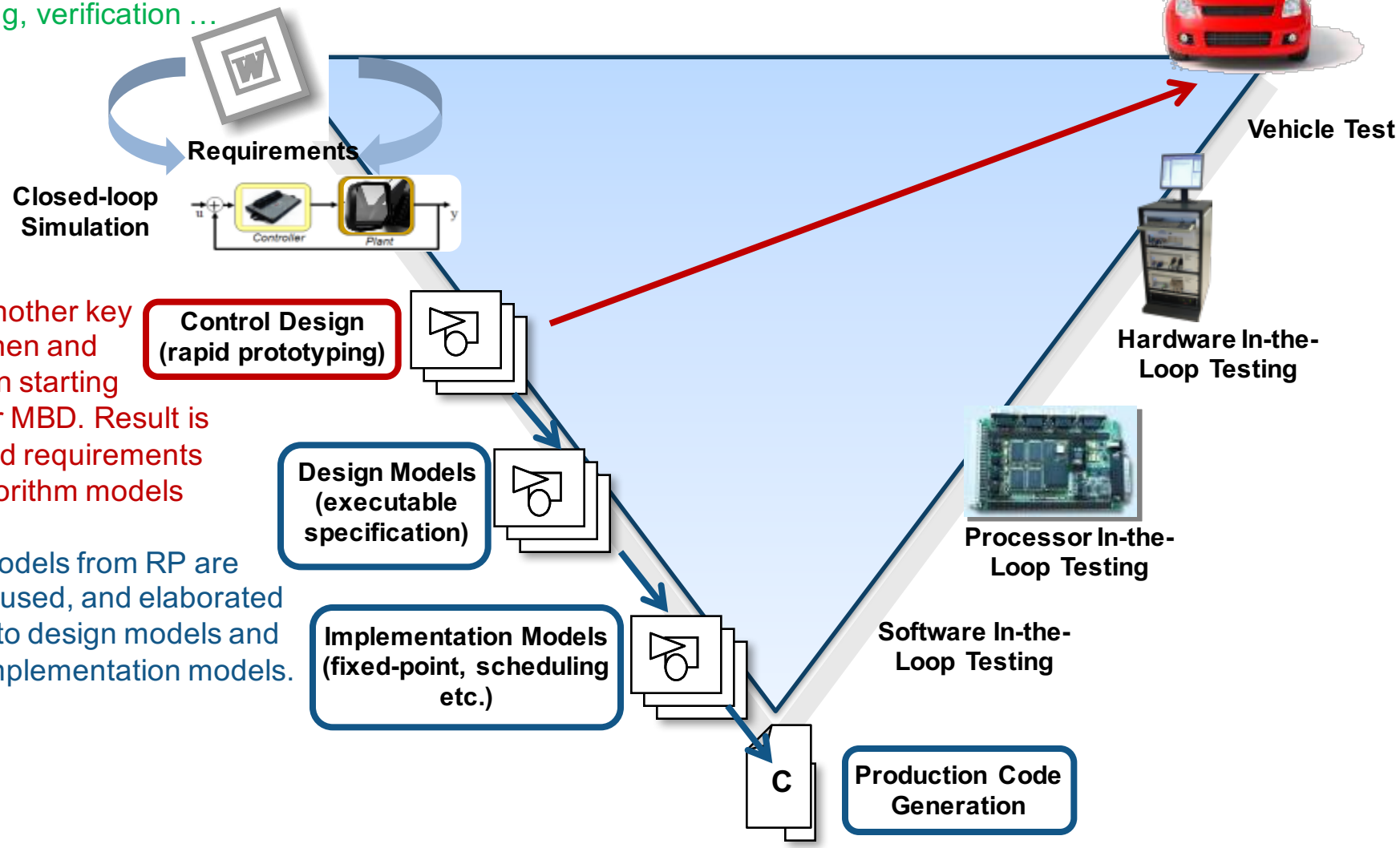


Model-Based Design Process



Model-Based Design Workflow

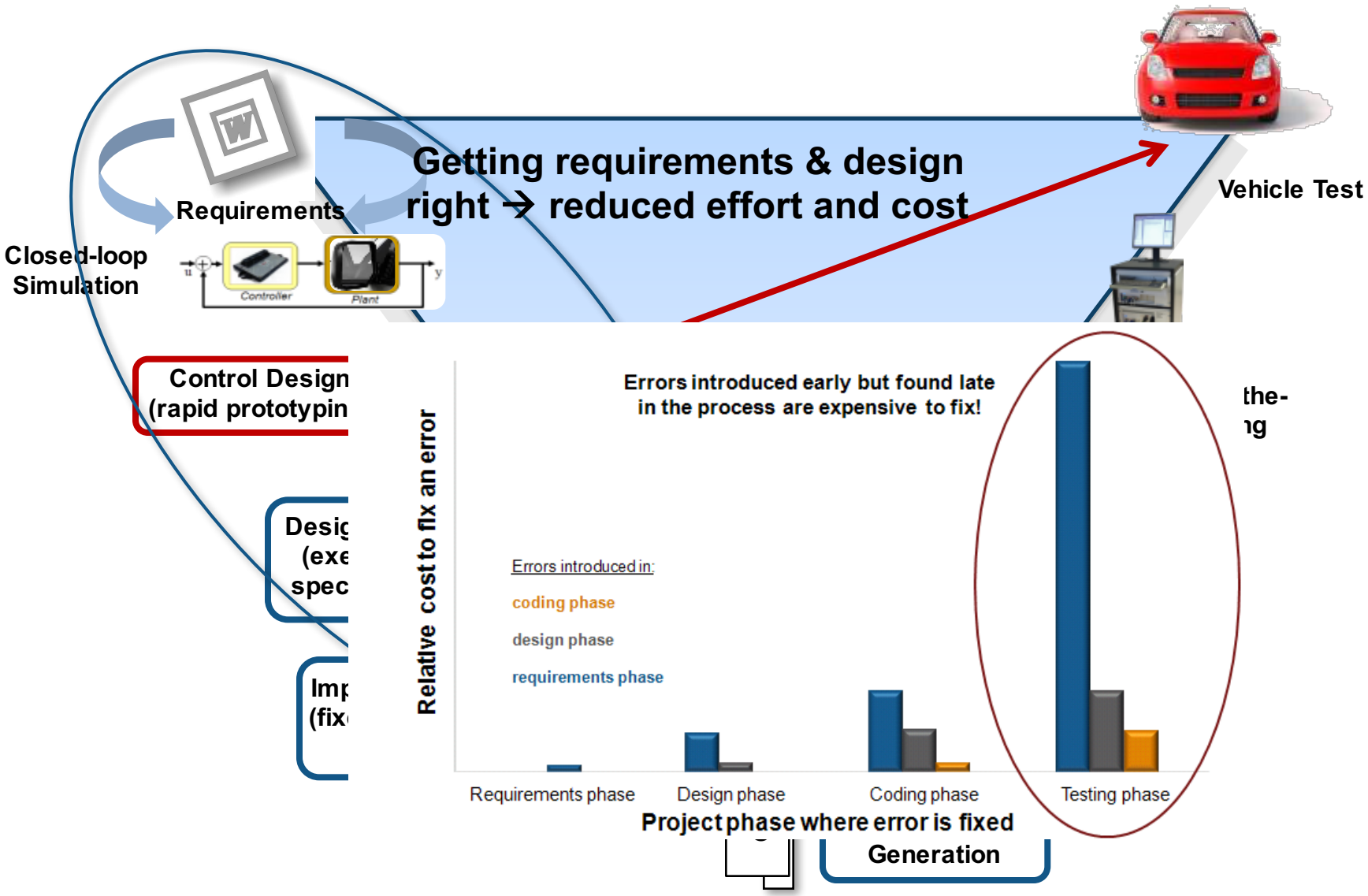
Closed loop simulation is a key strength of MBD: design trade-off, performance analysis, component sizing, verification …



RP is another key strengthen and common starting point for MBD. Result is validated requirements and algorithm models

Models from RP are reused, and elaborated into design models and Implementation models.

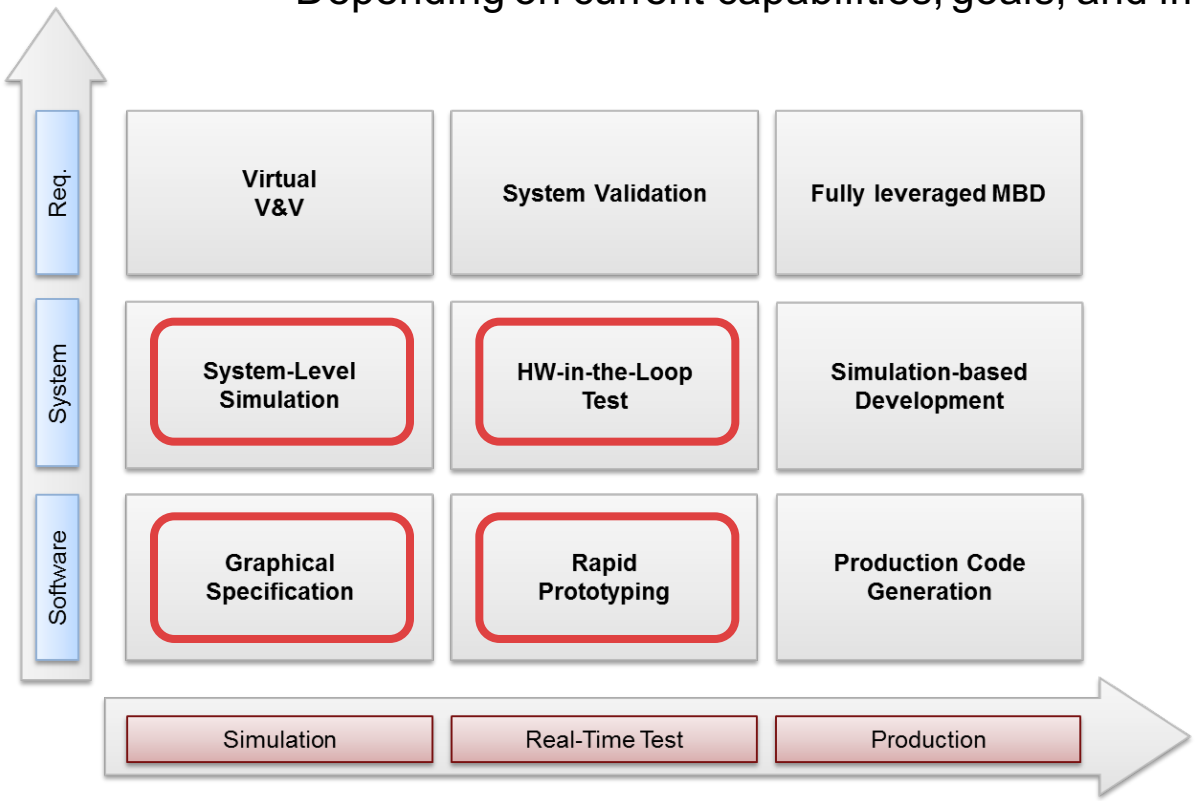
Model-Based Design Makes Early Verification Possible



Model-Based Design Adoption Starting Points and Paths

Model-Based Design Adoption Grid

- Adoption of Model-Based Design occurs in phases.
- Most companies move from lower left towards upper right
- Four common starting points are illustrated below
- The exact starting point and paths vary by company
  - Depending on current capabilities, goals, and immediate opportunities.



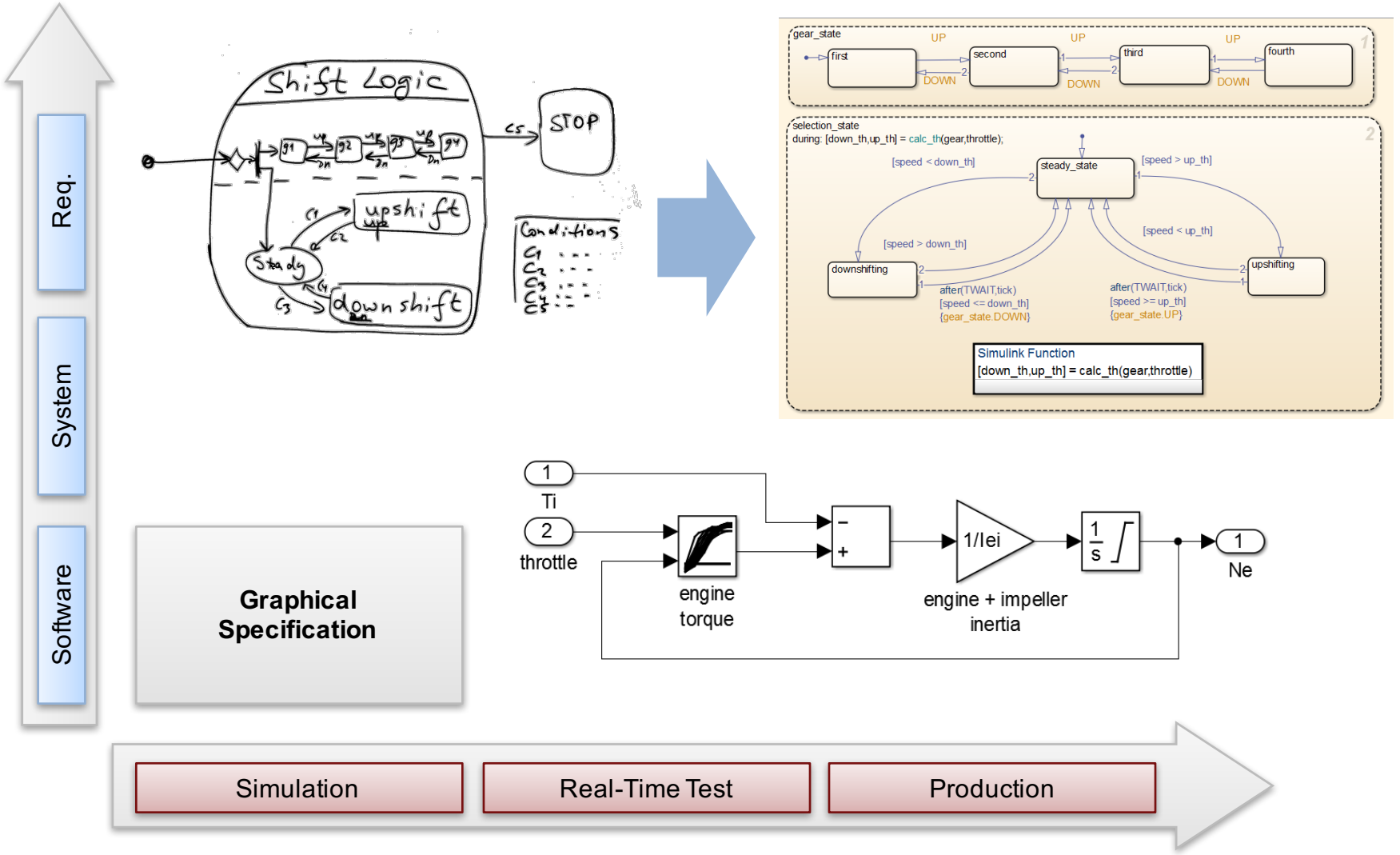
Best Practice: Use models for at least two things – "Rule of Two"

- Overcome startup costs and resistance to change
- ROI increases with multi-use models

Example 1: Validate requirements through simulation and add new functionality through rapid prototyping

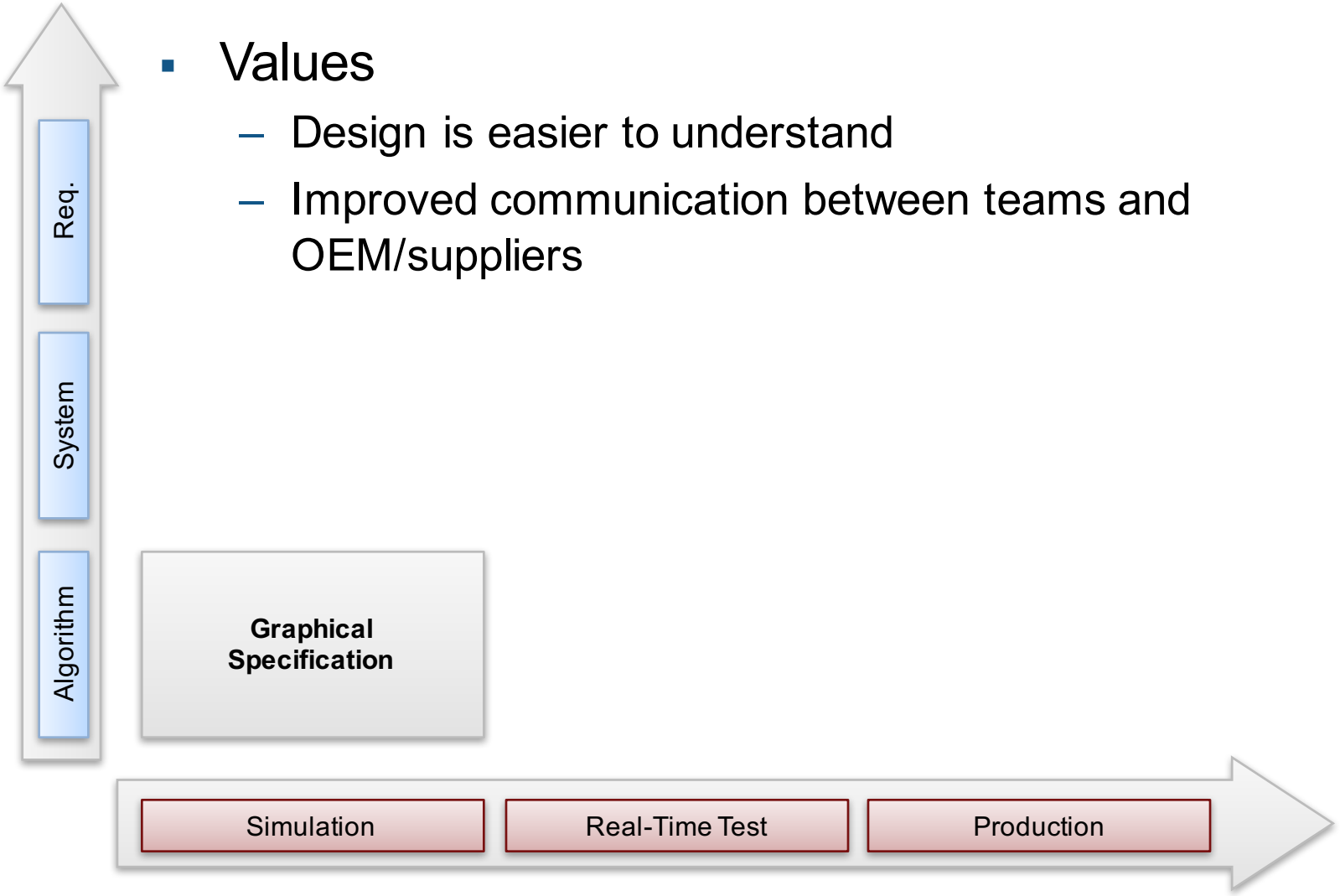
Example 2: System specification and automatic code generation

Model-Based Design Adoption Grid

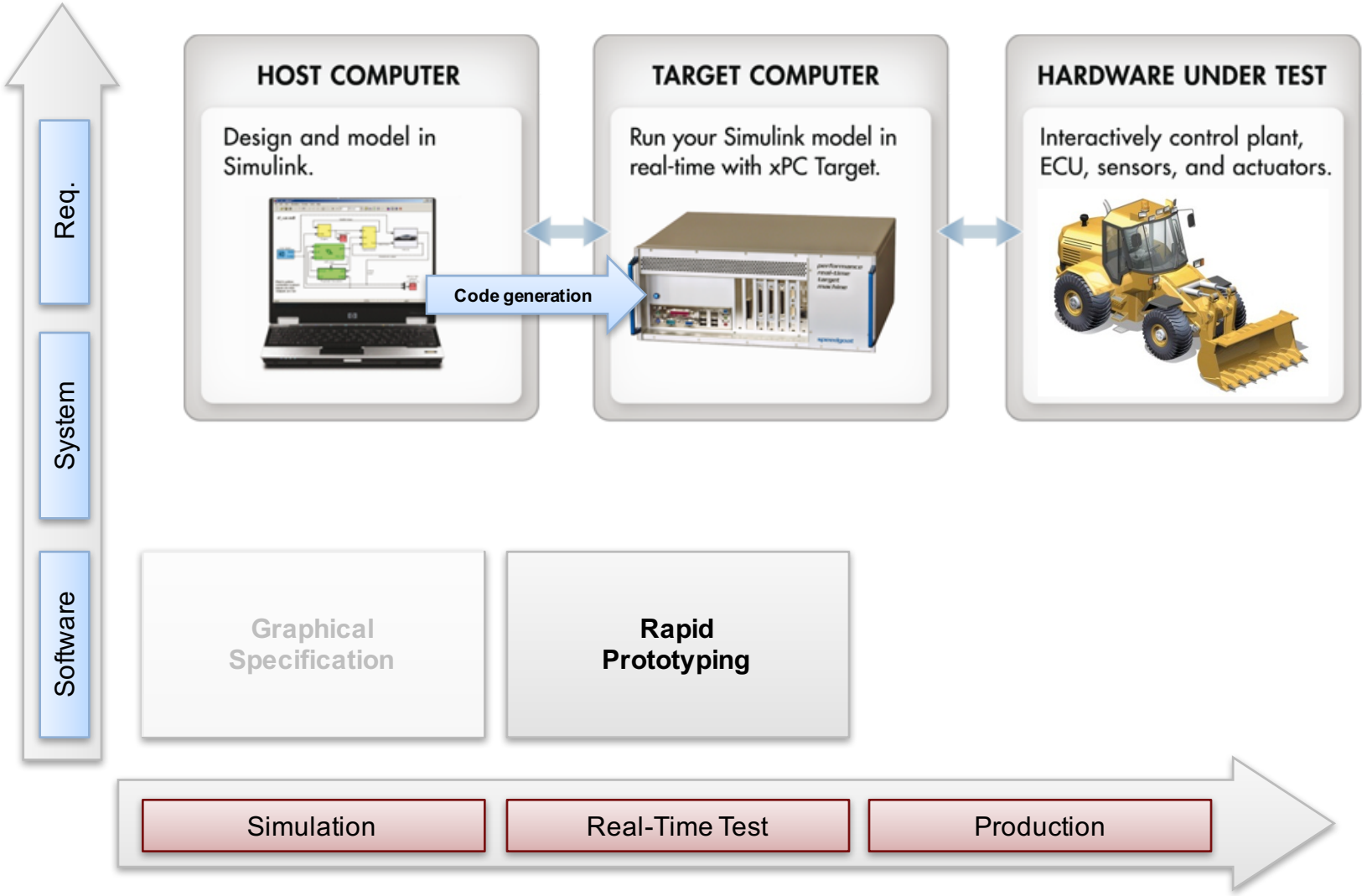


Model-Based Design Adoption Grid

- Values
  - Design is easier to understand
  - Improved communication between teams and OEM/suppliers

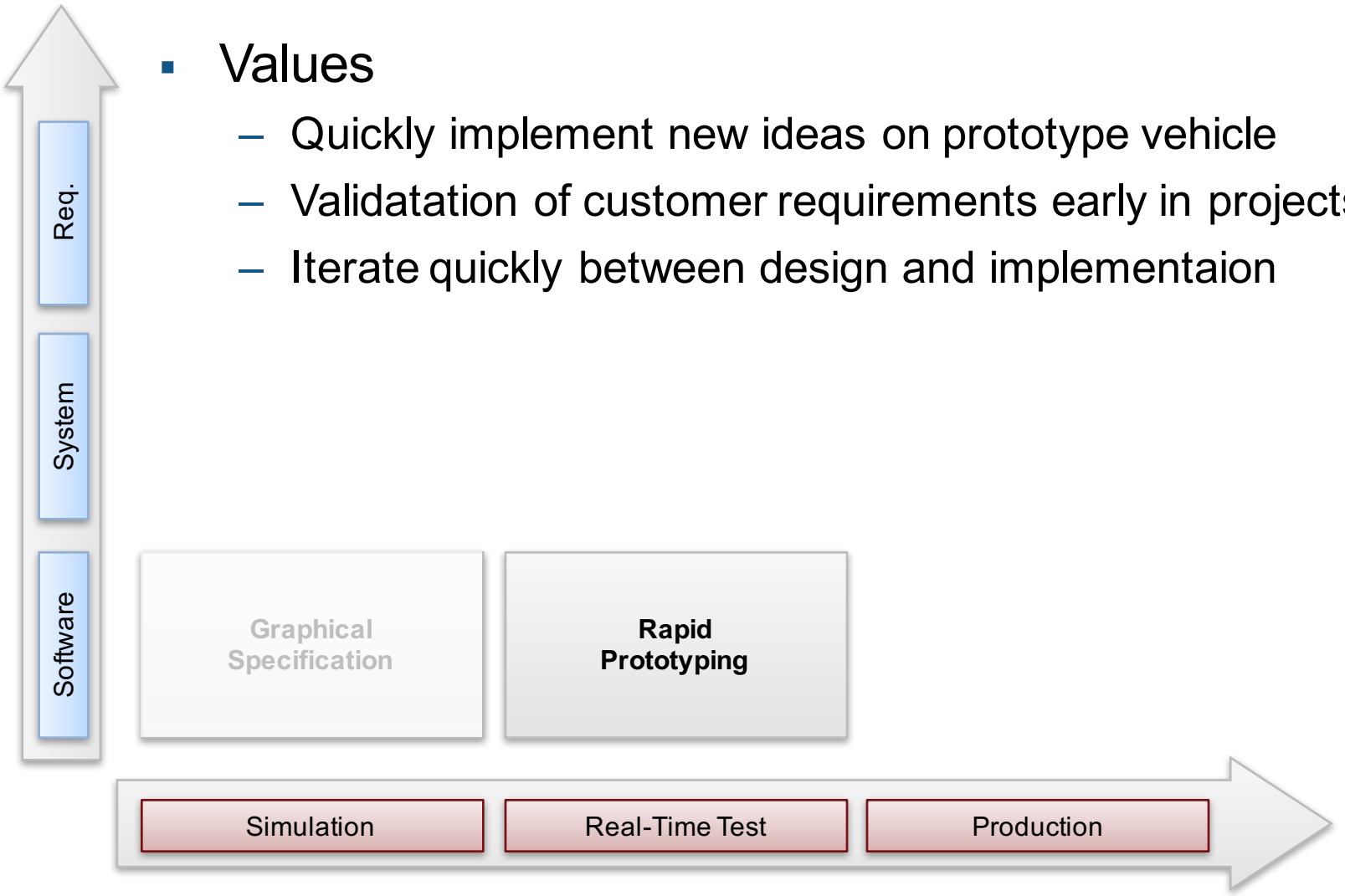


Model-Based Design Adoption Grid



Model-Based Design Adoption Grid

- Values
  - Quickly implement new ideas on prototype vehicle
  - Validatation of customer requirements early in projects
  - Iterate quickly between design and implementaion

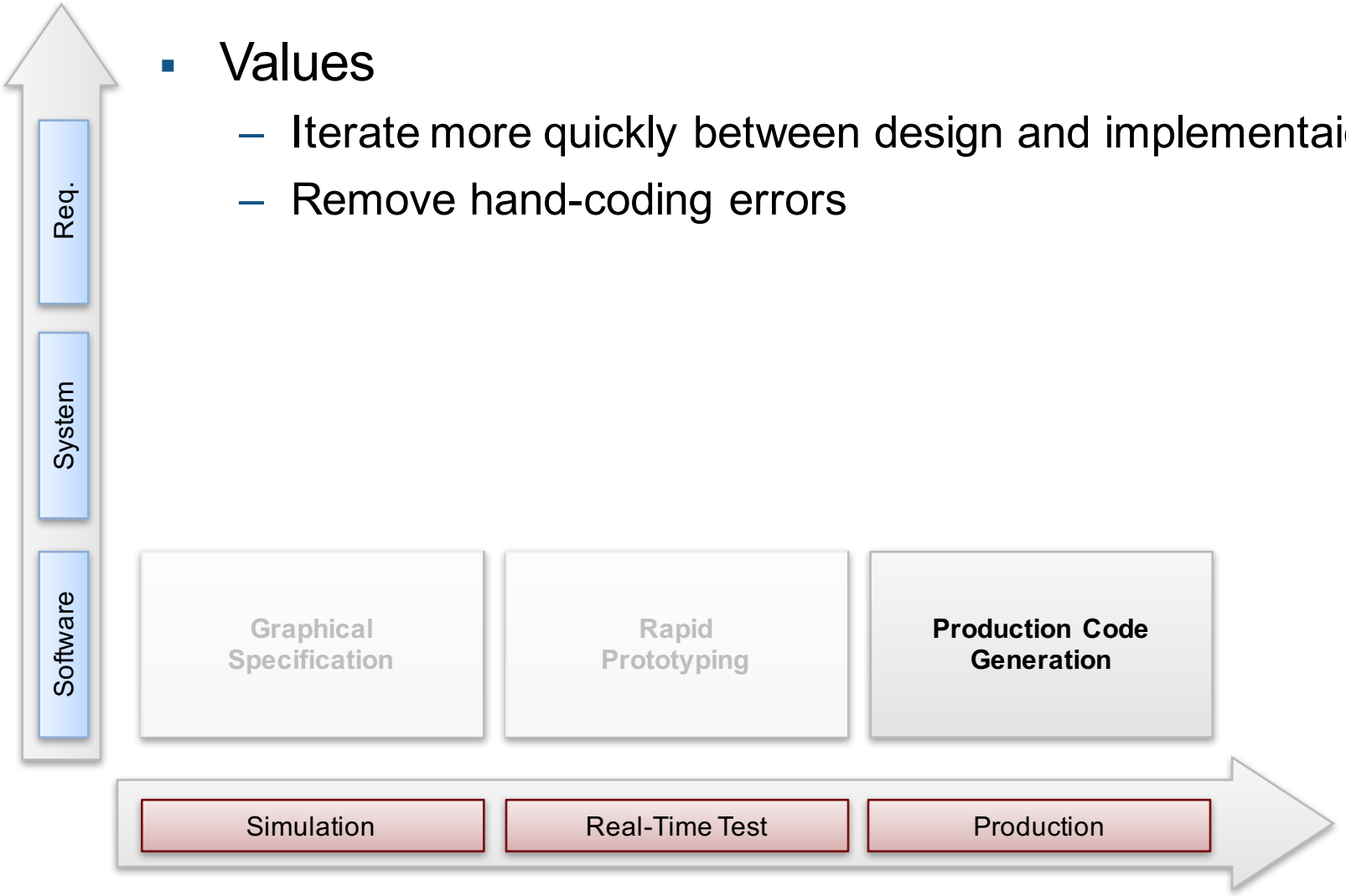


Best Practice: Use the models for production code generation

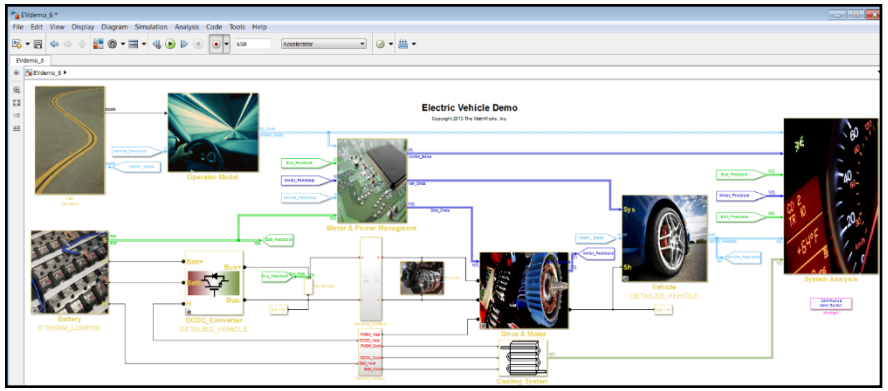
- To ensure success you must connect models to real system
- Enable a culture of modeling by removing temptation and option to write code
- Executable code is what makes machines move and generates profits

Model-Based Design Adoption Grid

- Values
  - Iterate more quickly between design and implementaion
  - Remove hand-coding errors



Model-Based Design Adoption Grid



Req.

System

Software

System Level Simulation

Graphical Specification

Design Prototyping

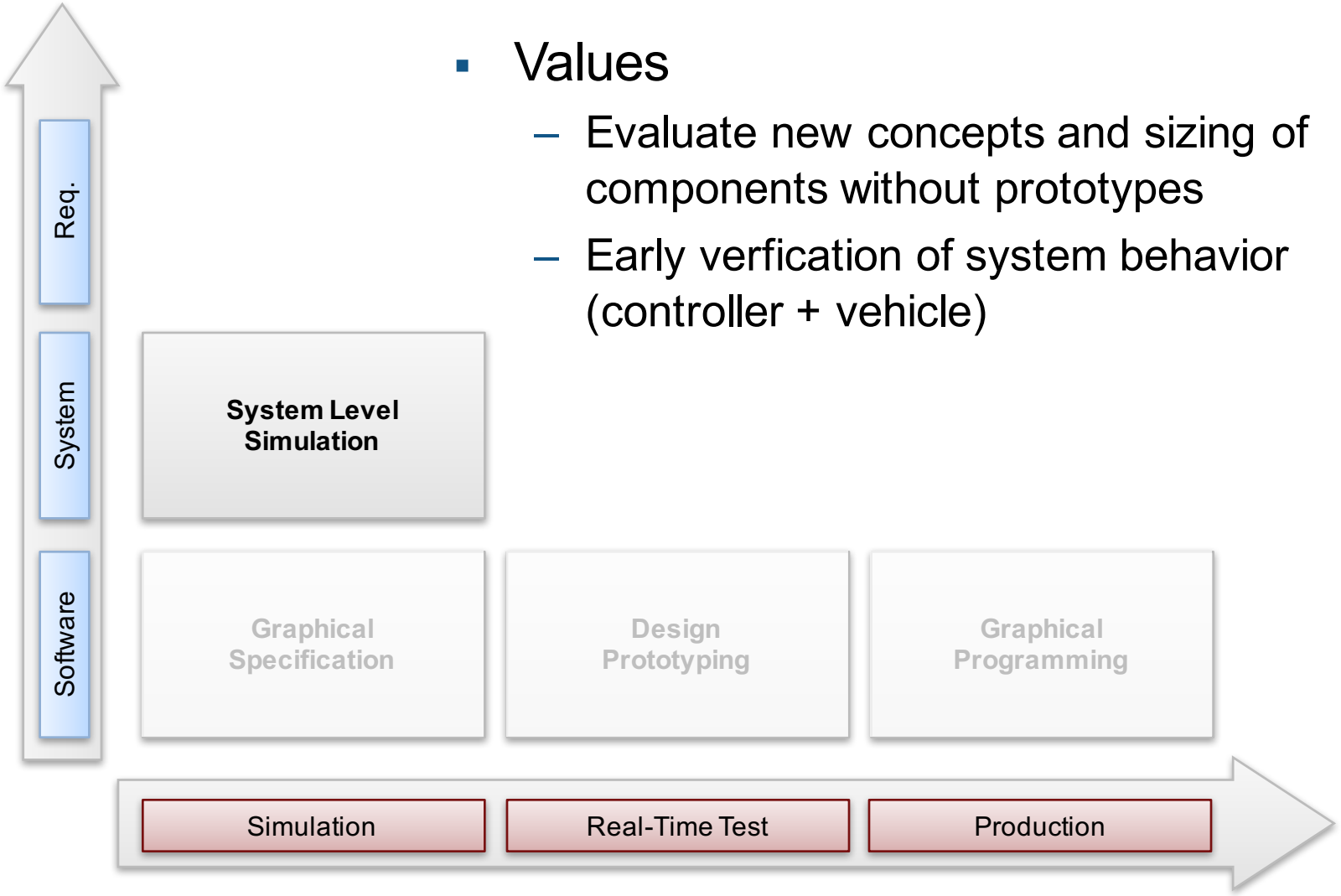
Production Code Generation

Simulation

Real-Time Test

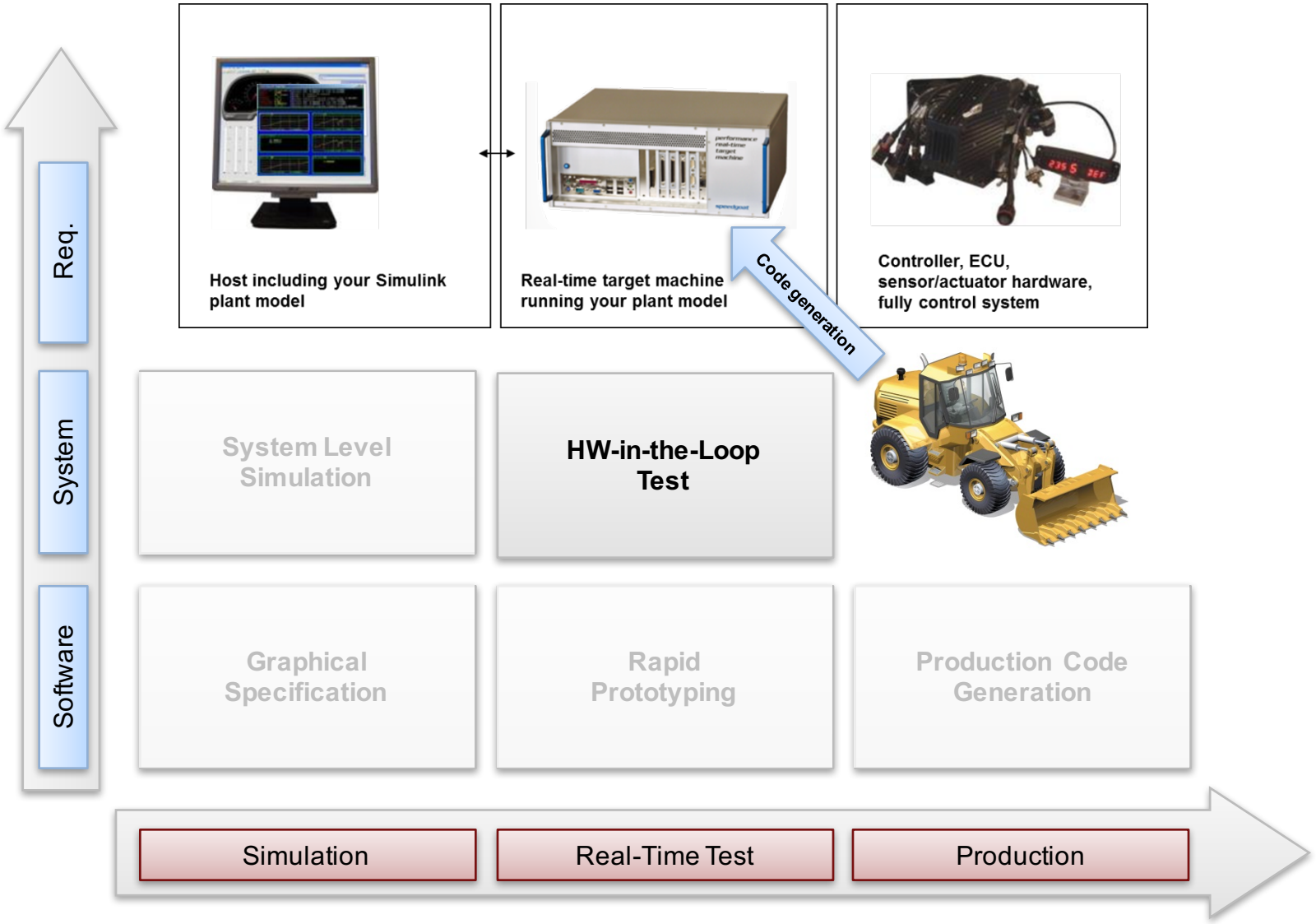
Production

Model-Based Design Adoption Grid



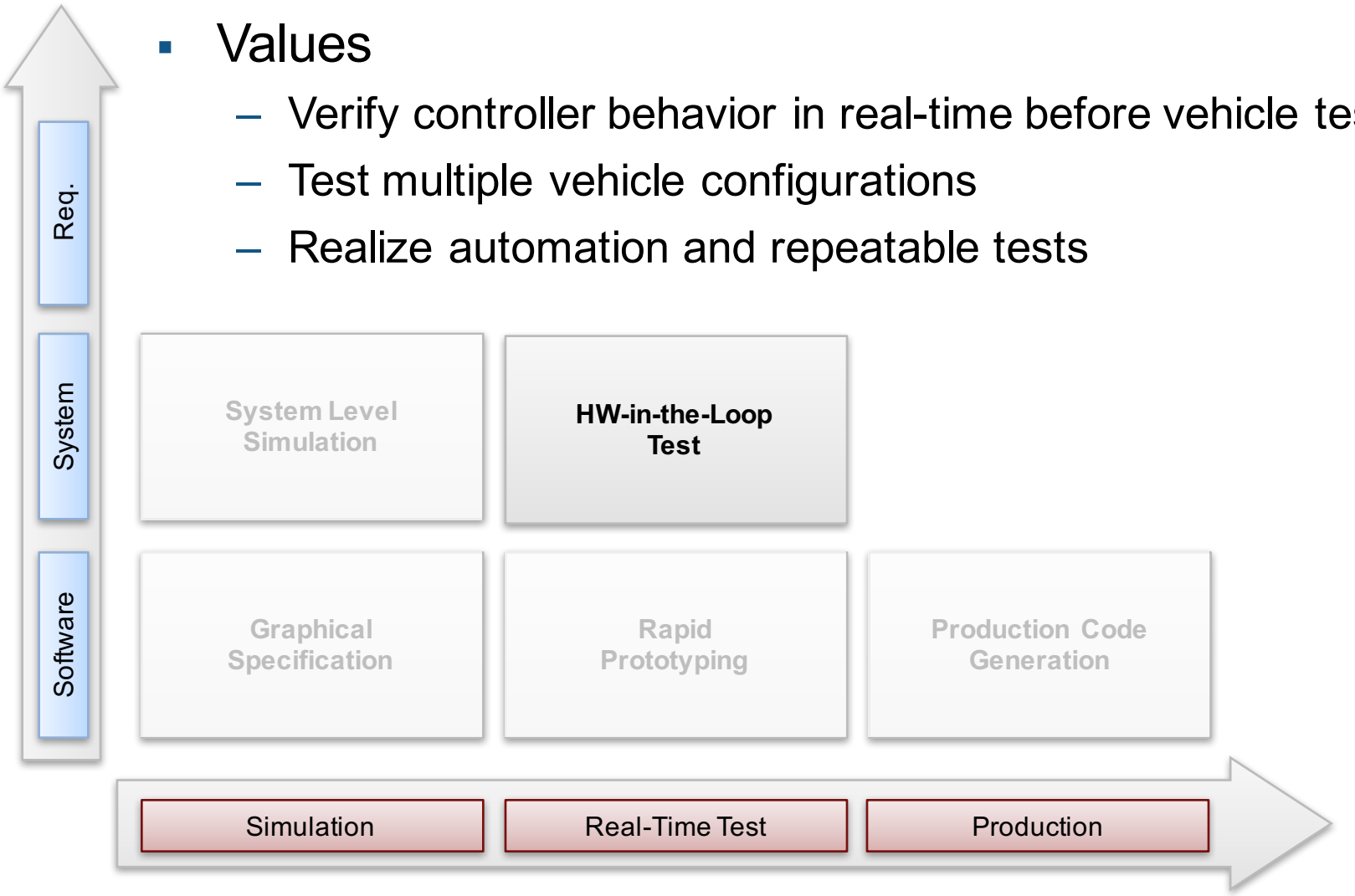
- Values
  - Evaluate new concepts and sizing of components without prototypes
  - Early verfication of system behavior (controller + vehicle)

Model-Based Design Adoption Grid

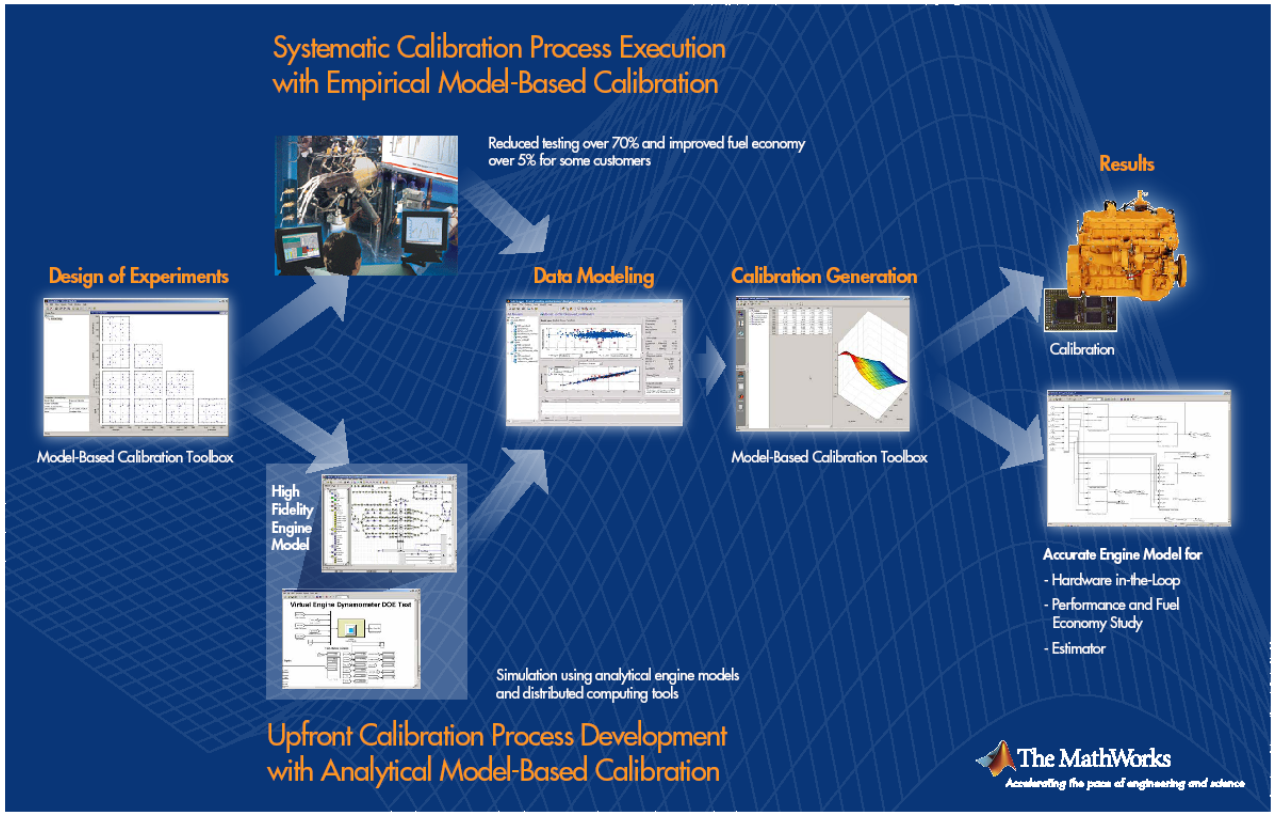


Model-Based Design Adoption Grid

- Values
  - Verify controller behavior in real-time before vehicle test
  - Test multiple vehicle configurations
  - Realize automation and repeatable tests



Model-Based Design Adoption Grid



Req.

System

Software

Closed-L
Simulati

Graphical Specification

Rapid Prototyping

Production Code Generation

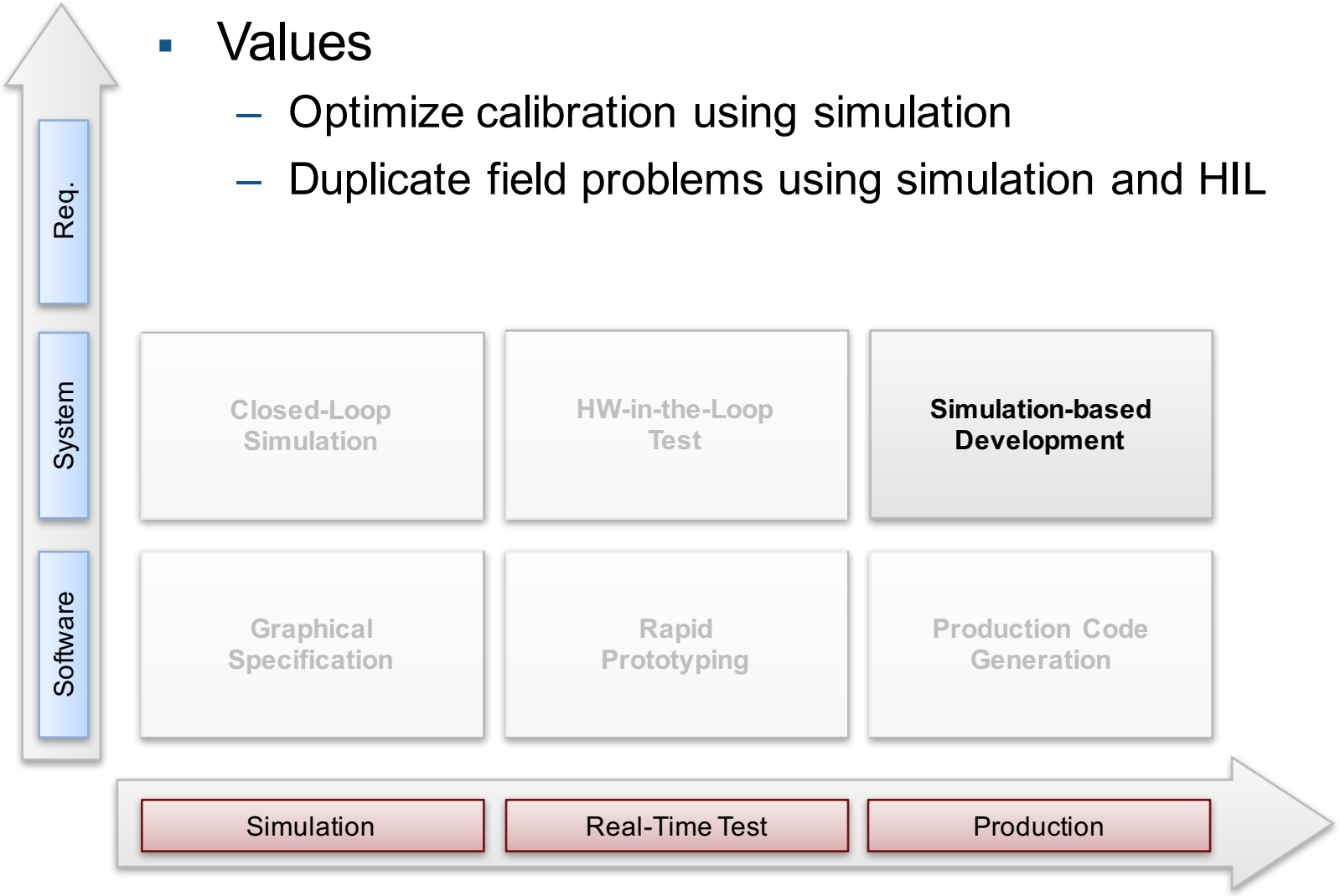
Simulation

Real-Time Test

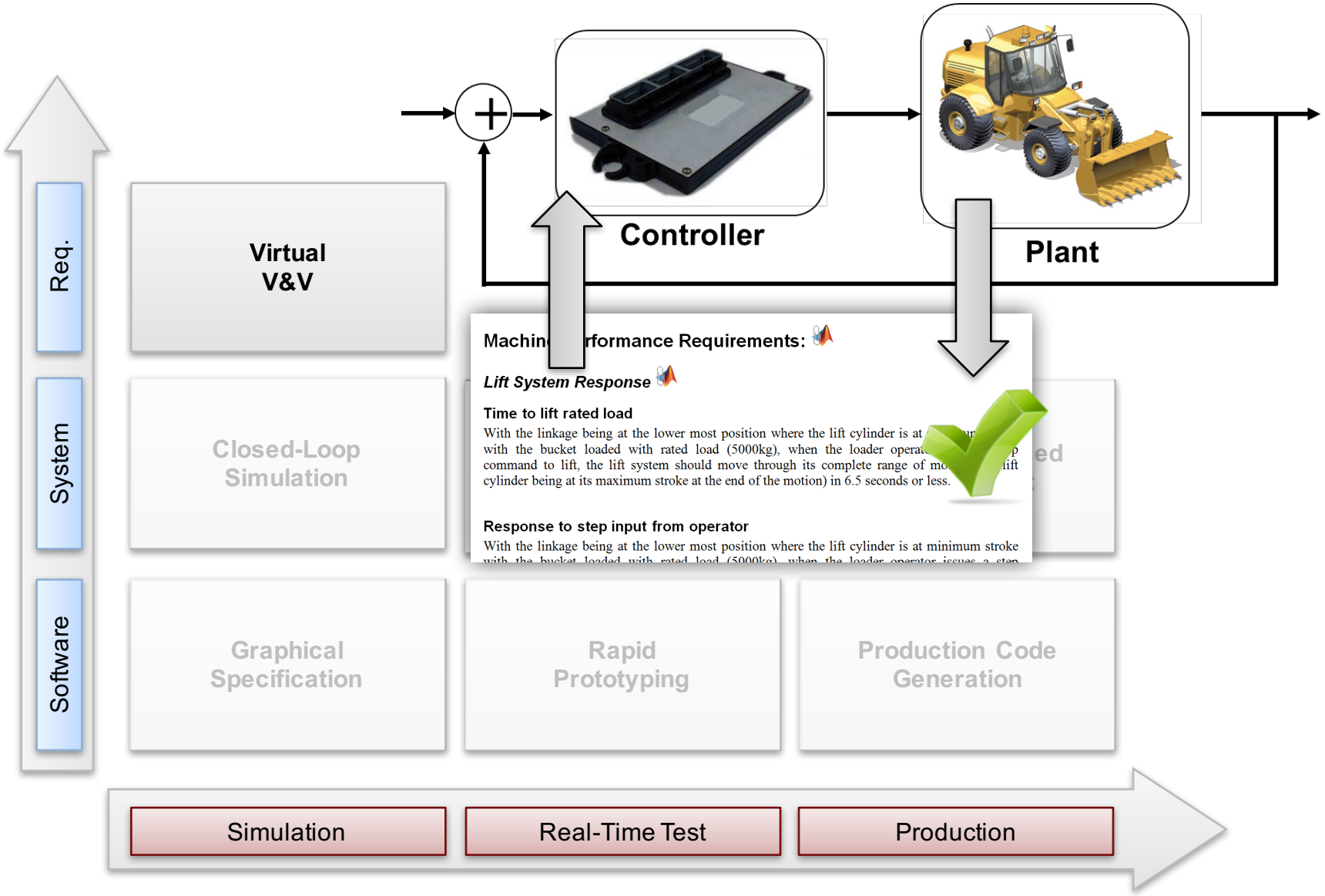
Production

Model-Based Design Adoption Grid

- Values
  - Optimize calibration using simulation
  - Duplicate field problems using simulation and HIL

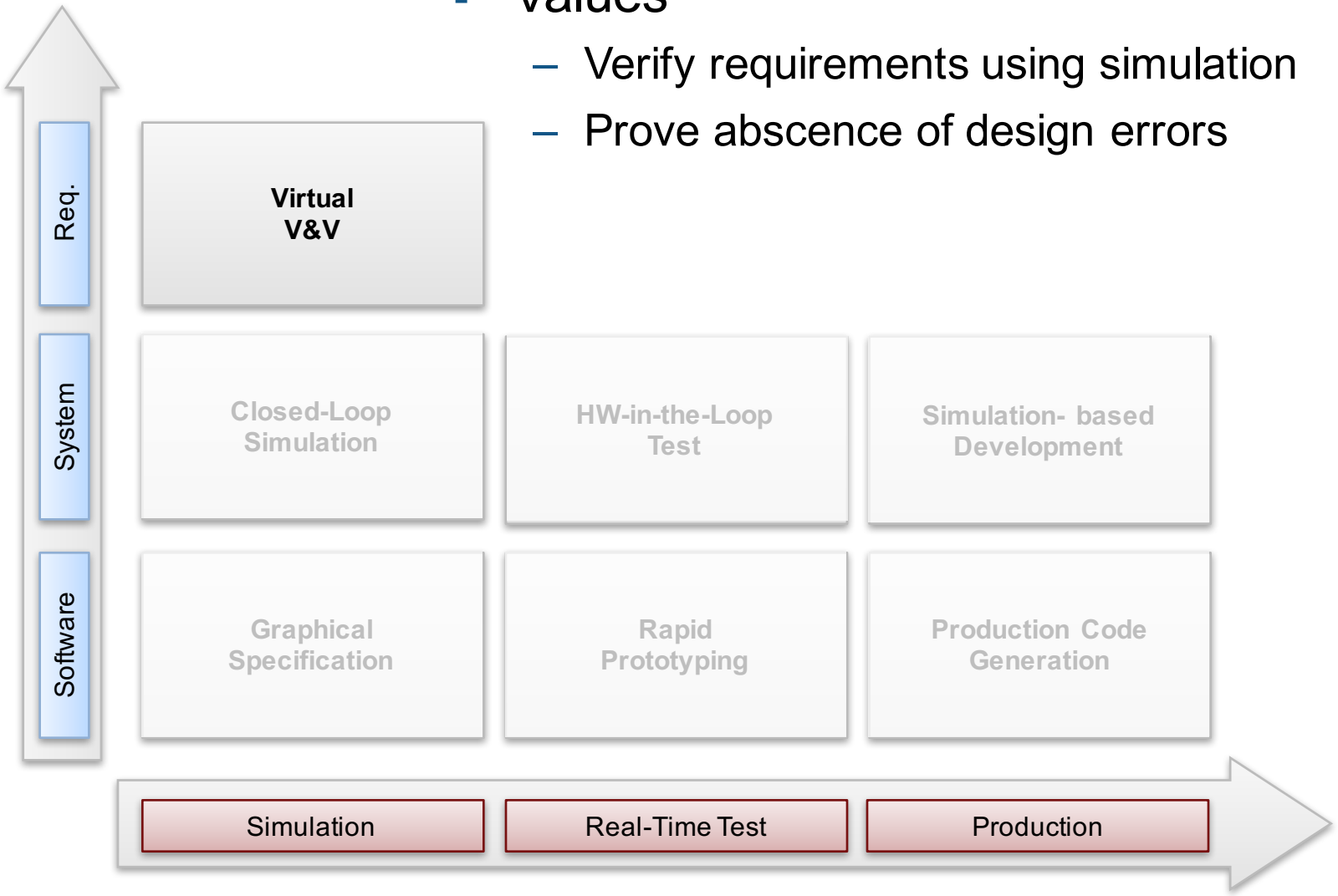


Model-Based Design Adoption Grid

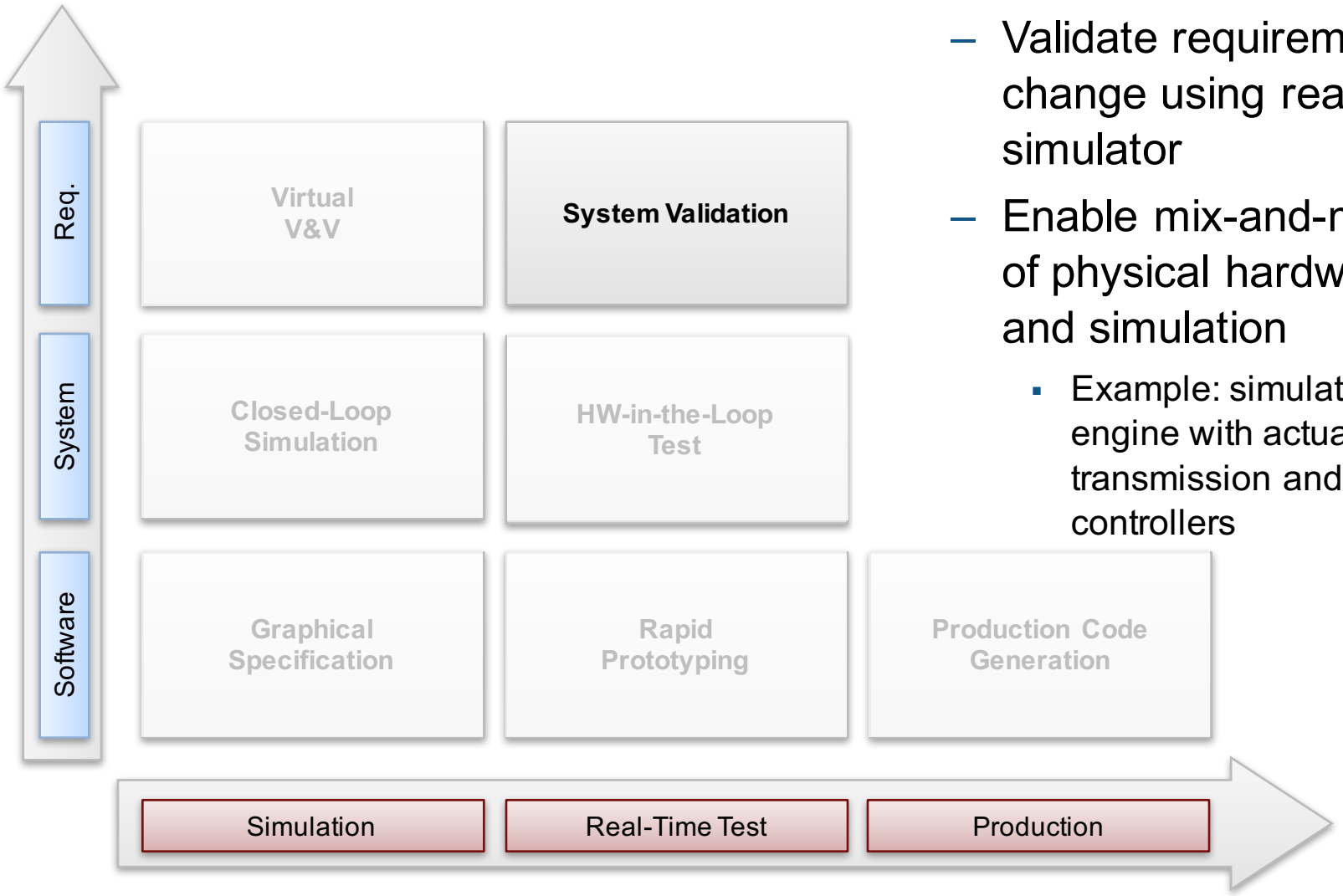


Model-Based Design Adoption Grid

- Values
  - Verify requirements using simulation
  - Prove abscence of design errors



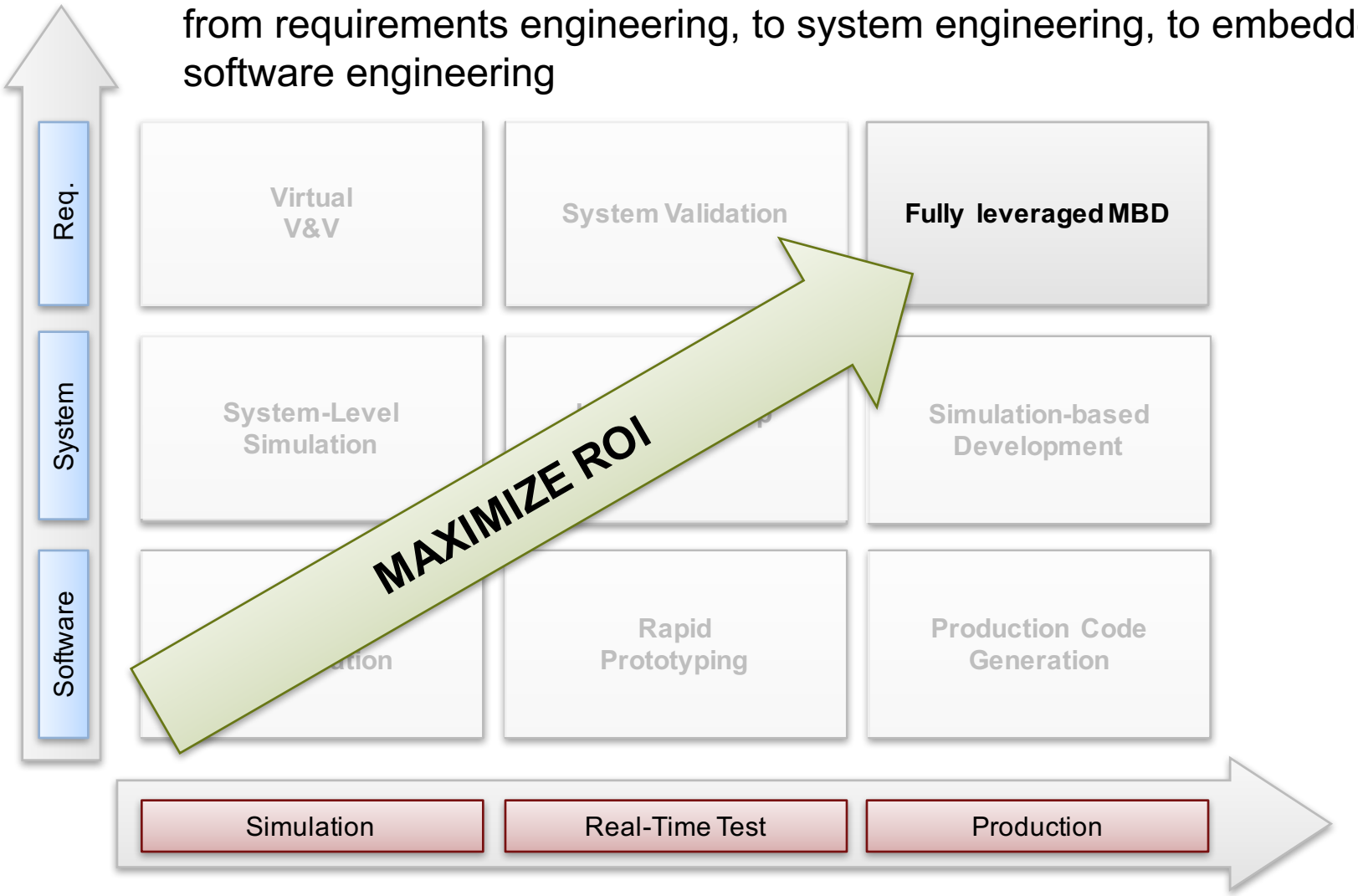
Model-Based Design Adoption Grid



- Values
  - Validate requirement change using real-time simulator
  - Enable mix-and-match of physical hardware and simulation
    - Example: simulated engine with actual transmission and controllers

Model-Based Design Adoption Grid

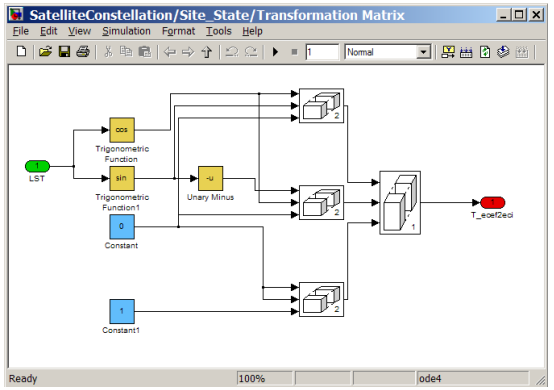
- Extensive deployment of models throughout development process, from requirements engineering, to system engineering, to embedded software engineering

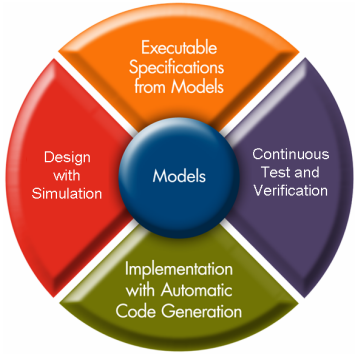


Simulink & Stateflow

What is Simulink?

1. Simulink is **Time and Event Domain Simulation**
2. Simulink is **Graphical Modeling**
3. Simulink is **Model-Based Design**



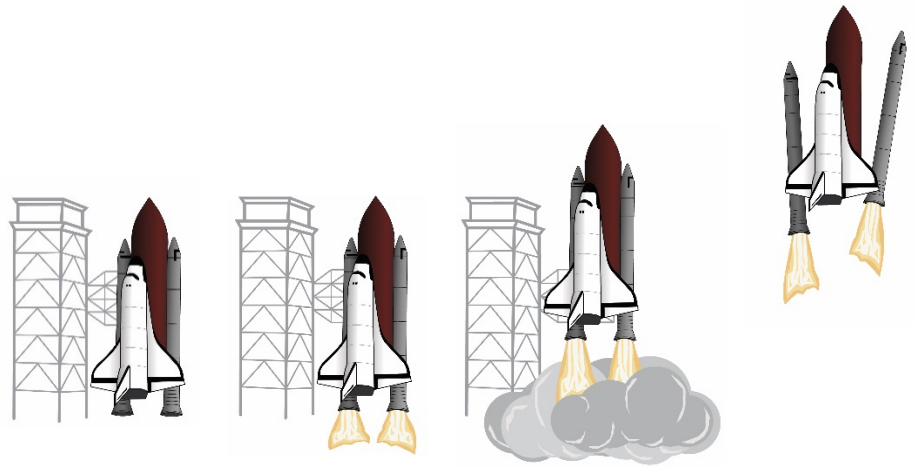


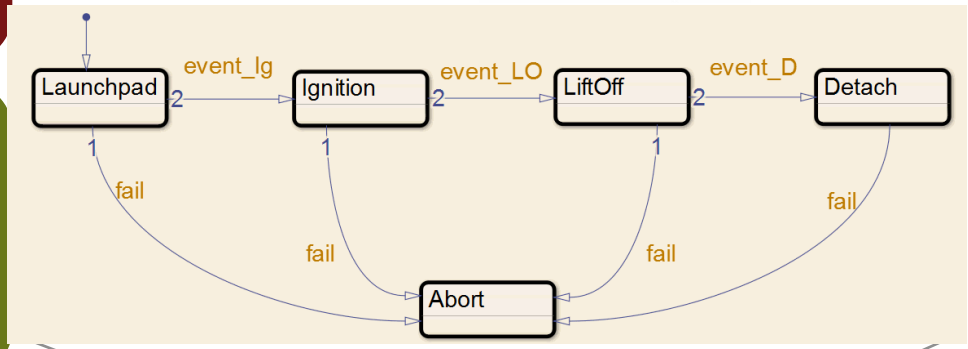
What is Stateflow?

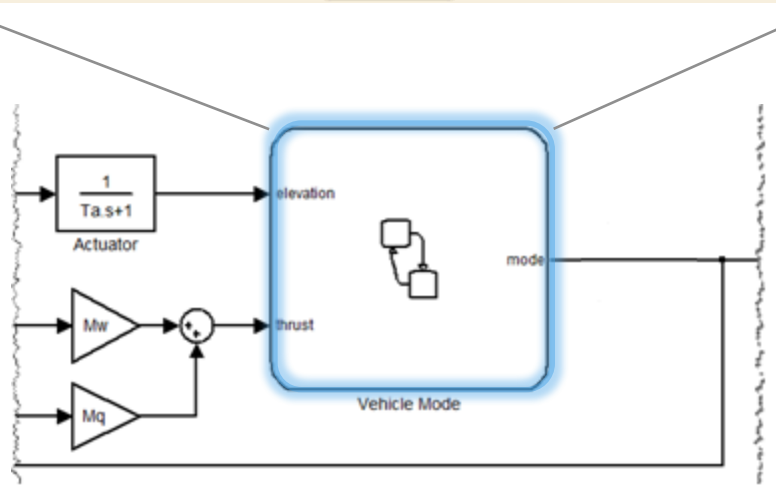
Extend Simulink with state charts and flow graphs

Design supervisory control, scheduling, and mode logic

Model state discontinuities and instantaneous events





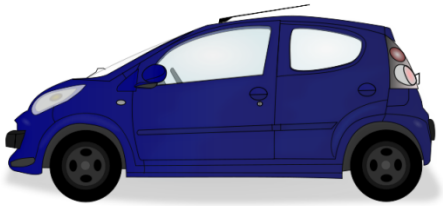


How Does Stateflow Work with Simulink?

Simulink models **continuous** changes in dynamic systems.

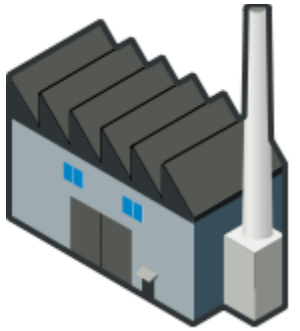
Stateflow models **instantaneous** changes in dynamic systems.

Real-world systems have to respond to both continuous and instantaneous changes.



suspension dynamics
gear changes

propulsion system
liftoff stages

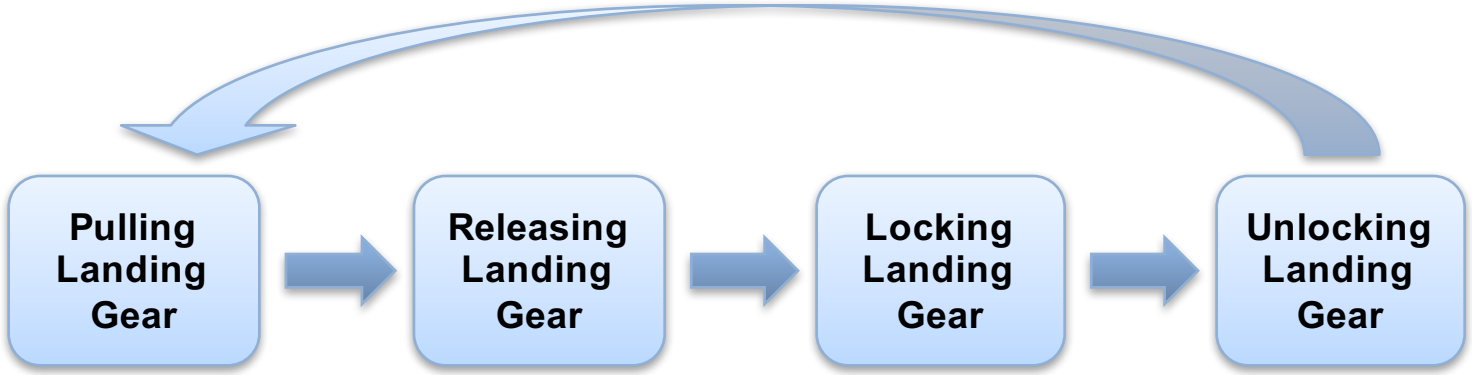


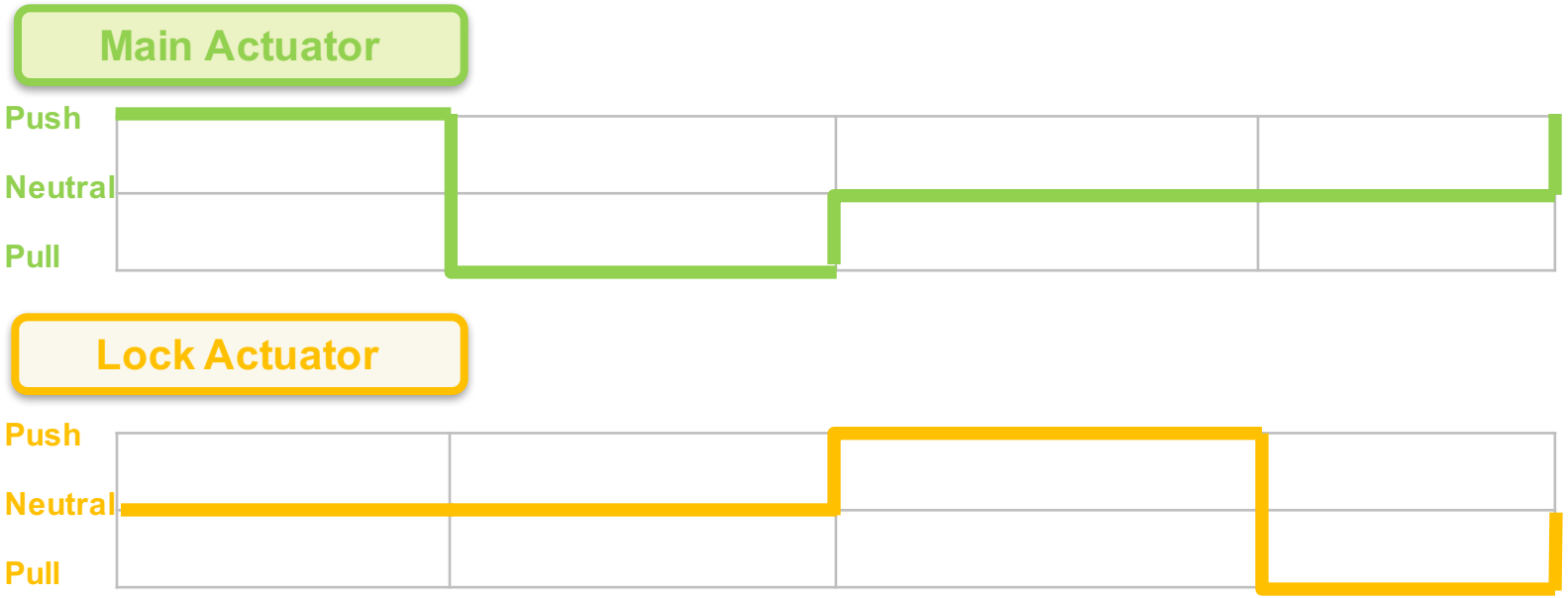
manufacturing robot
operation modes

Use both Simulink and Stateflow so that you can use the right tool for the right job.

Track Overview

Landing Gear System





Model-Based Design

What You Can Hear Today