Jump start your digital transformation with model based systems engineering and IBM Engineering Lifecycle Management

Kim Cobb
IBM Engineering Lifecycle Management
WW Industry Leader

Watson IoT

Agenda

- Engineering challenges driving the need for change
- Systems Engineering industry trends - document to digital
- Introducing Model Based Systems Engineering
- Deeper Dive – IBM Engineering Lifecycle Management for MBSE
- Customer References
- Conclusion and Next Steps

Innovation is driving change

Today’s smart products offer an increasingly feature rich functionality and autonomous action dominated by software components…

- Need to deliver more function meeting same quality and schedule
- Meeting growing industry regulatory demands can be also time and resource consumer
- Globalization drive competition requiring fast response for RFP/RFQs
- Complexity and time to market challenge effective supply chain management
- Business environment requires effective support of multi-variants programs

Design and engineering teams required to accelerate time to market; without compromising quality, safety and compliance.

89%

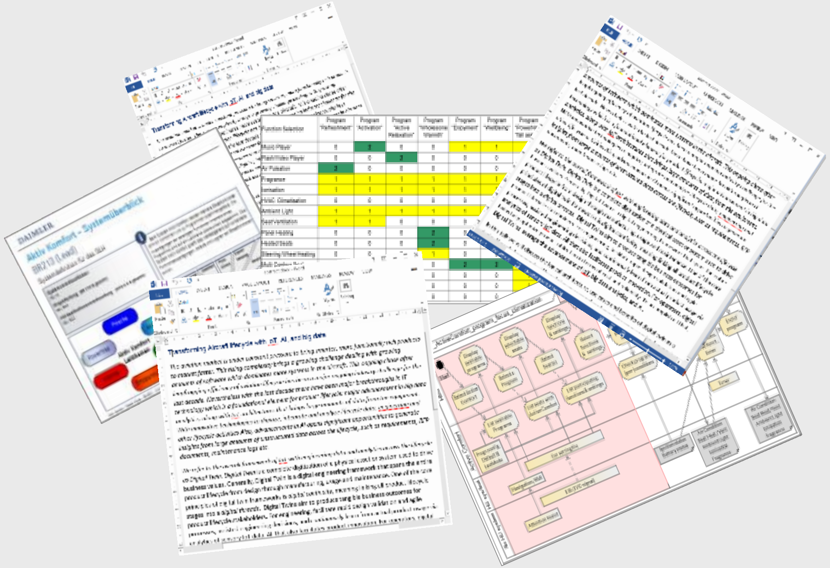
of CIO’s state their teams need to release software even faster




“Document centric” engineering practices are challenged to keep up

- Lack traceability, version & variance management, governance and testability

- Expensive rework due to late discovery of issues
- Slow response to bids and change requests
- Inefficient supplier collaboration
- Non-optimal designs due to early design lock
- High costs of regulatory compliance activities



The Science of engineering has evolved to empower innovation, have you?

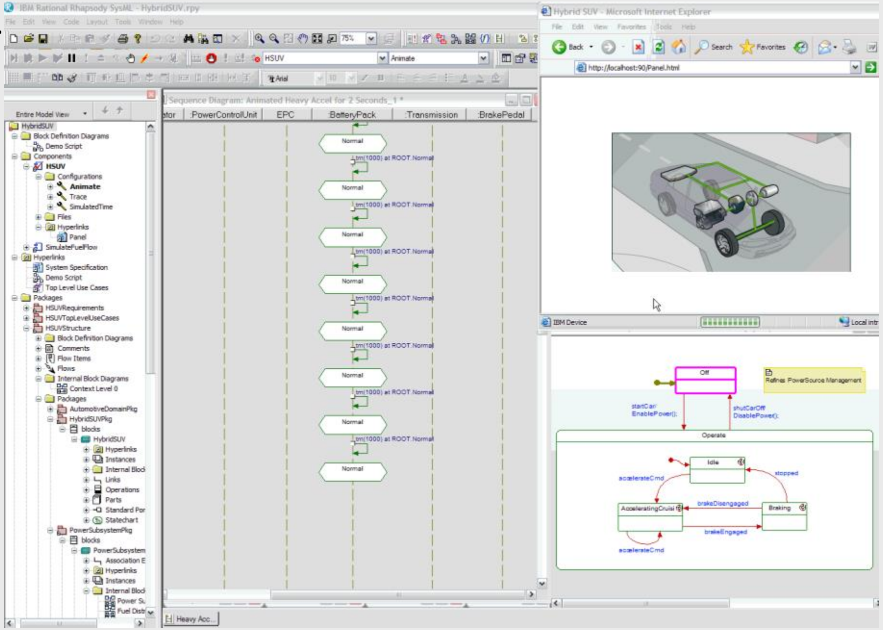
Opportunity: Digital representation of the system
Model Based Systems Engineering (MBSE)

– Precise: with standards based modeling language for systems engineering

– Visual: “a picture’s worth thousand words”, simplifies communication

– Verifiable: models are formal artifacts that can be verified against the requirements

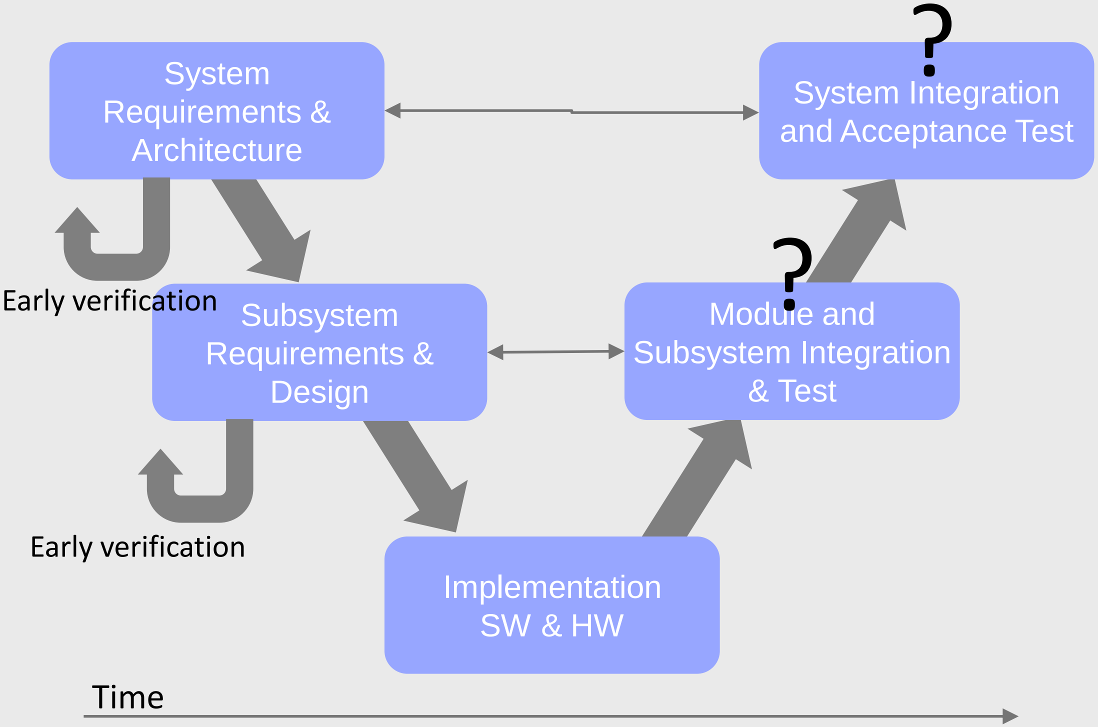
– Traceable: models are based on fine grain digital artifacts traced to other essential lifecycle artifacts such as requirements
and plans



…MBSE proliferates fully traceable, self documenting and verifiable system

Models enable early verification of complex systems

- Finding design errors during physical system integration is costly and inflicts significant delays
- System design models are testable virtual representations that identify design issues before anything physical is built
- Designs are elaborated and verified iteratively enabling adoption of agile principles to systems engineering

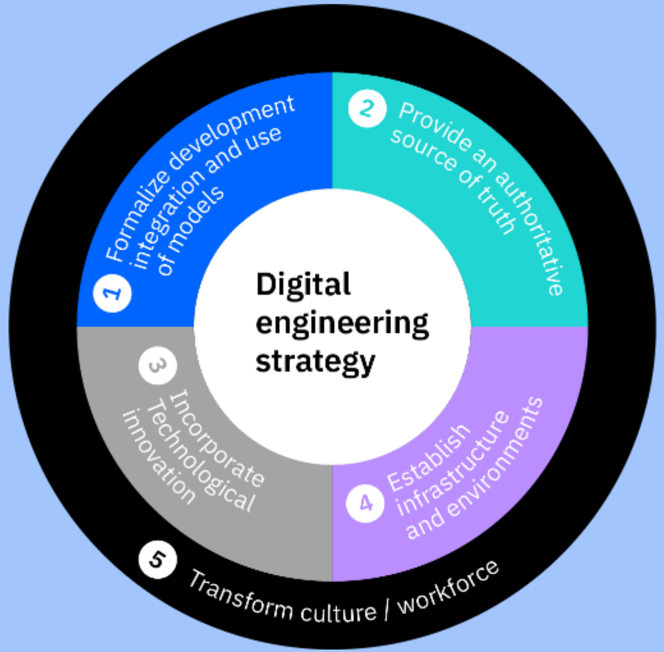


Cost of correcting an error found in integration may be two orders of magnitude (100x) more expensive than identifying it during specification!

Industry MBSE Vision: Digital Engineering advocated by US Department of Defense

*"**conduct engineering in more integrated virtual environments** to increase customer and vendor engagement, improve threat response timelines, [..], reduce cost of documentation and impact sustainment affordability.*

Such engineering environments will allow DoD and industry partners to evolve designs at conceptual phase, reducing the need for expensive mockups, premature design lock, and physical testing." [1]



"Systems Engineering focuses on ensuring the pieces work together to achieve the objective of the whole"

System Engineering body of knowledge (SEBoX)

2018, the Office of the Under Secretary of Defense for Research and Engineering (USD(R&E)) released a Digital Engineering Strategy built on five foundational elements necessary **for a Digital Engineering Ecosystem to thrive.**

1. DoD Digital Engineering Initiative: https://www.acq.osd.mil/se/initiatives/init_de.html

MBSE: Referenced by INCOSE[1] 2025 vision[2]

Systems of the future

- Growing and diverse spectrum of needs
- Harness growing body of technology innovation
- Engineered by an evolving diverse workforce which, with increasingly capable tools, can innovate and respond to competitive pressures

"Modelling, Simulation and visualization will become more integrated and powerful to cope with the systems challenges in 2025"

Simulation and Visualization

Modeling, simulation, and visualization enable complex system understanding that help us anticipate and verify solutions and their cost before building them. As systems become more complex, understanding their emergent behavior due to increasingly complex software, extreme physical environments, net-centricity, and human interactions becomes essential for successful systems development.

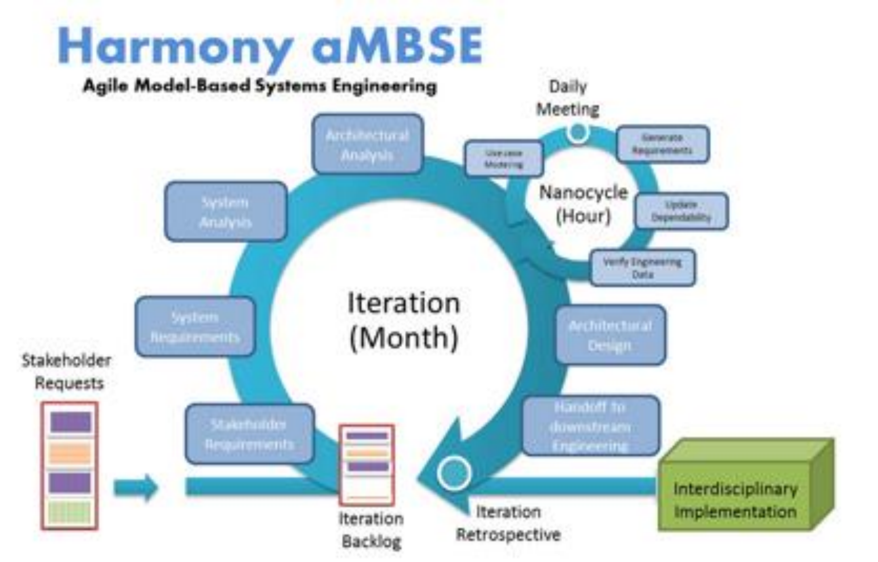
Integrated Model-based Approaches

Model-based Systems Engineering will become the "norm" for systems engineering execution, with specific focus placed on integrated modeling environments. These systems models go "beyond the boxes", incorporating geometric, production and operational views. Integrated models reduce inconsistencies, enable automation and support early and continual verification by analysis.

1 – International Council for Systems Engineering
2 - https://www.incose.org/docs/default-source/aboutse/se-vision-2025.pdf

MBSE: Essential to Agile systems development

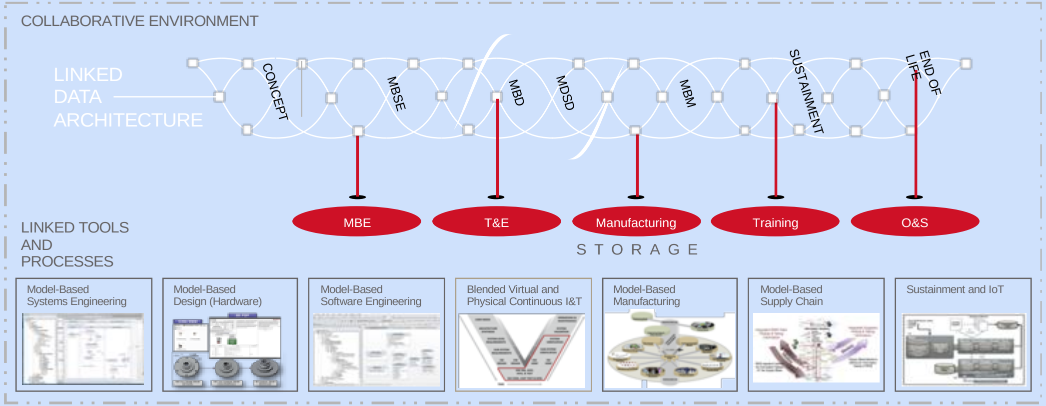
- Agile principles are today's building block for engineering practices that deliver the right function with controlled risk and high quality
- MBSE is part of the Scaled Agile Framework (SAFe) practice for large solutions
- Leverage of verifiable models and collaborative incremental cycles (sprints)
- Enables test driven development of the design
- Proven adoption practices with IBM industry expertise



Harmony Agile Model Based Systems Engineering (aMBSE) is a documented IBM practice

MBSE: Integral to Raytheon's MBE digital engineering roadmap

- Engineering knowledge represented as a set of models across lifecycle
- Models are integrated with digital threads establishing the necessary traceability
- Supports virtual reasoning across lifecycle:
  - Early verification of system integrity and KPIs
  - Impact analysis when a requirement changes
  - Coverage of requirements down to hardware and software parts

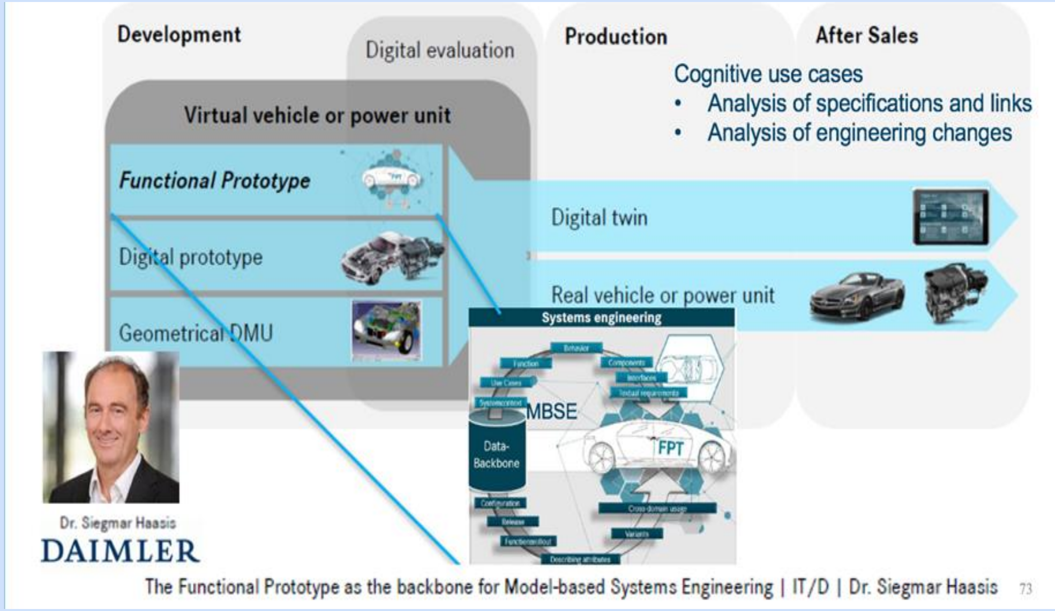


MBSE = Model-Based Systems Engineering
MBD = Model-Based Definition
MDSD = Model-Driven SW Development
MBM = Model-Based Manufacturing

As presented at IBM Agile Engineering Conferences

MBSE: Building block for Daimler's vehicle digital twin

- Pace of innovation challenges traditional mechanically centered engineering practices
- The key decisions for connected vehicles take place during early systems engineering phases
- MBSE integrate systems engineering with physical implementation models enabling end to end reasoning
- Establishing a multi-facet digital twin for connected cars



As presented at IBM IoT Genius of Things

MBSE: A key common platform for MBDA

Business problem: Needed to improve complex engineering design collaboration and cycle time across geographical borders

Solution: A model-based systems engineering platform across widely distributed design and engineering teams to analyze and communicate software, mechanical and electrical requirements

“We’re now able to define and model all the requirements of a complex missile system very early in the development process. This differentiates our delivery capabilities in a highly competitive global marketplace.”

— *MBDA executive*

modeling, testing and sharing highly complex designs and design protocols

Reduced typical system design time from

60+ months

to

18 months

while enabling design team collaboration across geographies

Integrated automotive control systems

Continental Automotive Body & Security Group

Business Problem

- Passive start and entry systems, remote keyless entry, and more - in one integrated system
- Enhanced driver experience with intelligent safety and convenience features

Solution

- Cost-optimized flexible system solution
- Reduced development costs based on use of standardized hardware and software components

How IBM empowers

- Requirements management across development teams and with vehicle manufacturers
- Streamlined development environment with model-driven systems and software development supporting **AUTOSAR**

"IBM Rational DOORS and Rhapsody are essentially helping us prevent fragmentation of our development environment and enabling us to better manage the complex architectures of our products."

Why Aerospace and Defense companies adopting MBSE

A&D projects are typically complex system of systems:
Aircrafts, UAVs, Warfare command and control systems –
They leverage all the general benefits of MBSE

MBSE supports A&D specifically:

- Effectively Implement INCOSE and ISO Systems engineering practices
- Support collaboration with government agencies like DoD requiring frameworks like DoDAF, NAF, and MoDAF as primary means of stakeholder communication
- Accelerate compliance with A&D industry standards such as DO-178C and ARP 4754
- Streamline collaboration with system suppliers
- Improve interworking among multi-disciplinary design teams

Why automotive OEM and suppliers adopting MBSE

– Establish an effective systems engineering practice to manage rapidly increasing complexity of car functions

– Accelerate introduction of new features and changes in a highly Competitive market

– Realize an end to end “digital twin” in the connected car era

– Accelerate compliance with key industry standard ASPICE and ISO26262 which strongly endorse MBSE as a core practice

As a result, more and more Automotive OEMs and suppliers adopt MBSE as a mainstream practice!

Why medical device manufacturers adopt MBSE

Medical device manufacturers benefit MBSE to:

– Enable highly effective engineering providing early verification and validation of system specification and design choices

– Comply with functional safety to the standards like ISO 15288, IEC 62366 and/or ISO 14971 that require safety analysis performed on systems design and detailed documentation of the system design

– Managing complexity, amplified by regulations, integrations, and advances in technology, without compomising quality and time

– Address FDA’s finding that lack of design controls as one of the major causes of device recalls

Why railway companies adopt MBSE

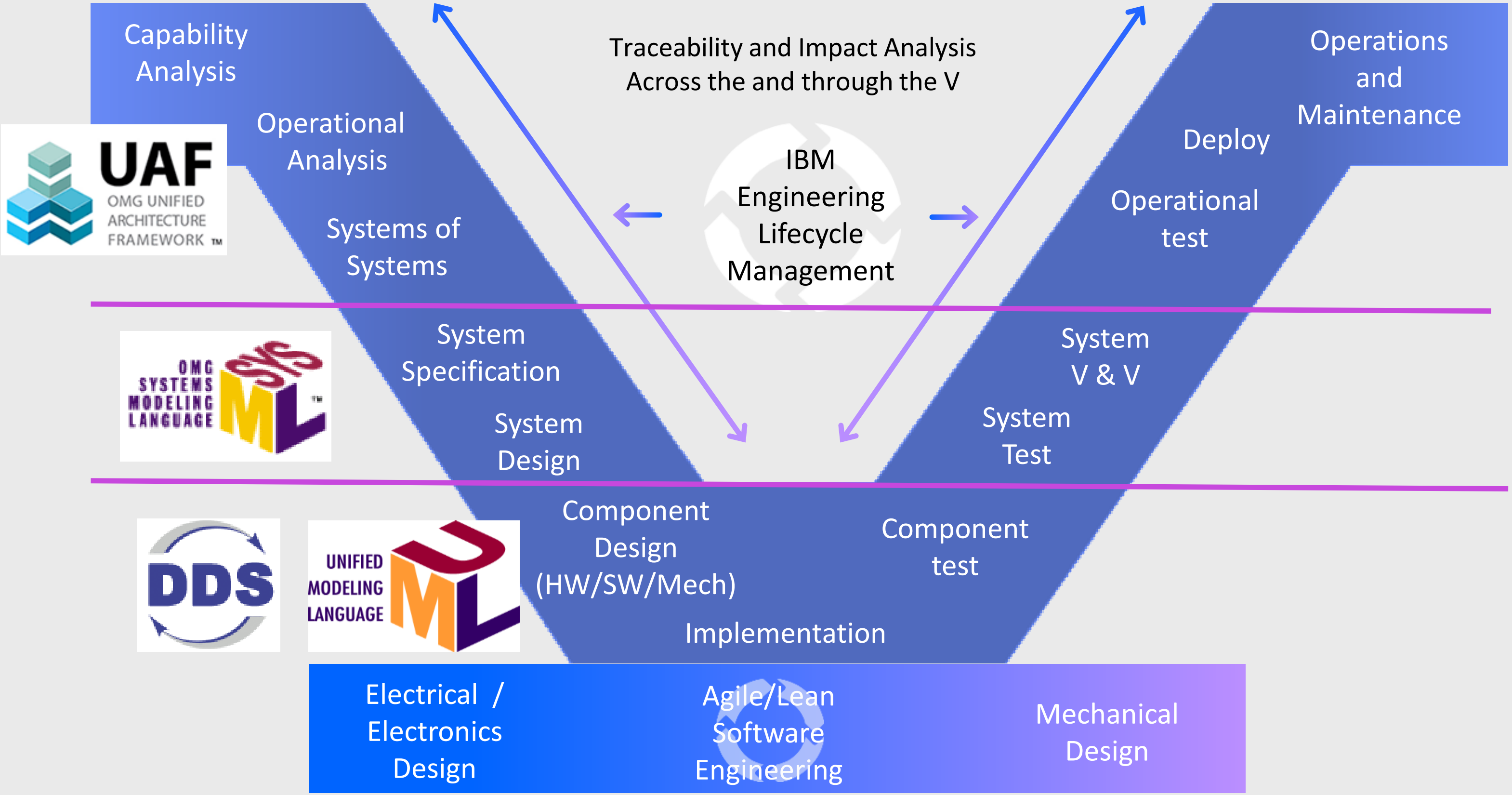
– Compliance to regulations like EN5012x are becoming increasingly crucial in the Railway landmark. Model based approach significantly reduces the manual effort of compliance

– Early verification and validation, needed for CENELEC safety requirements, can be met by the left shift characteristic of MBSE via simulation and testing of the system model.

– Traceability, assessing the impact of a design decision or change effectively through interconnected systems and subsystems, especially important for long term projects like railway.

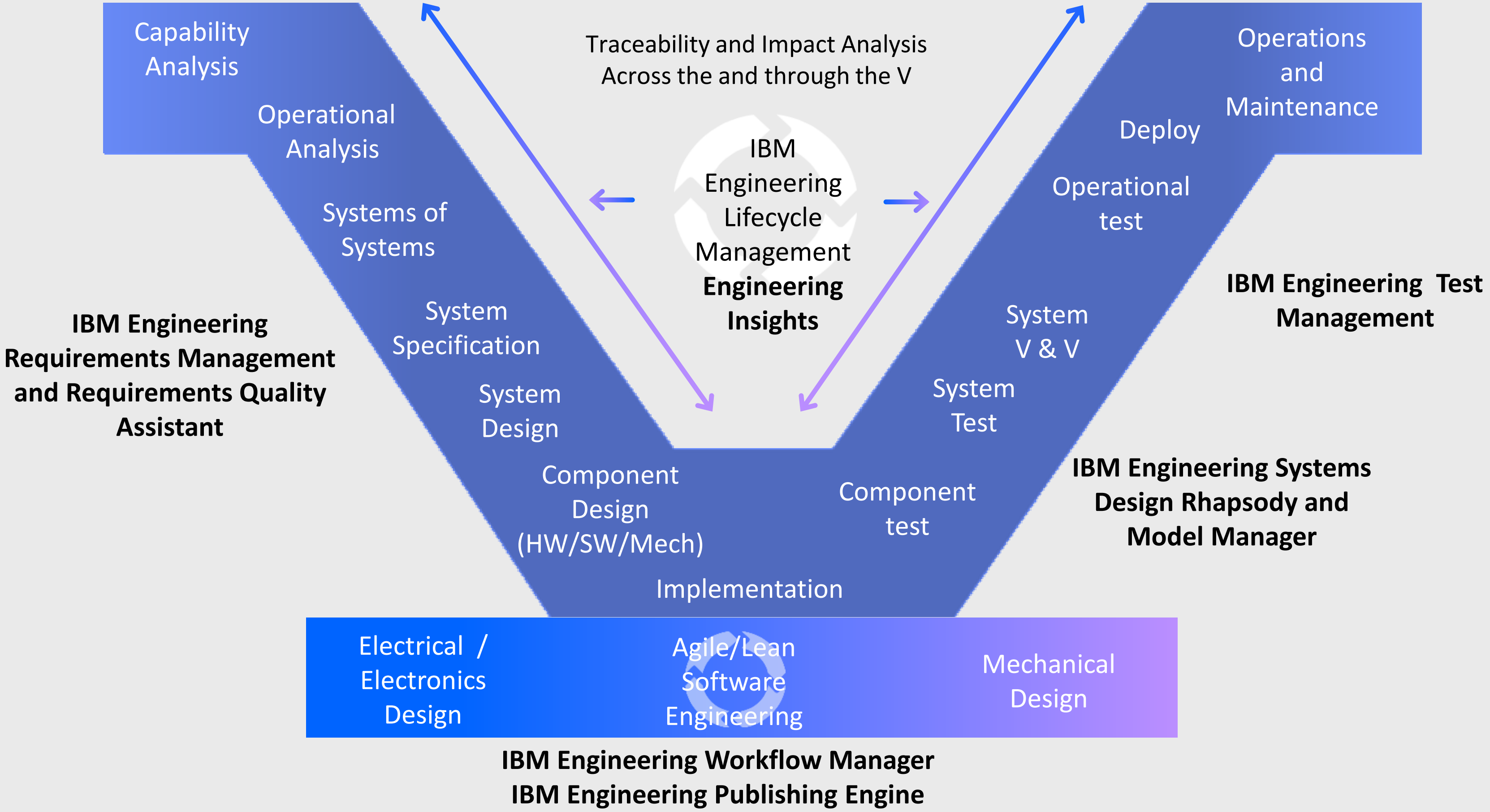
Realizing MBSE with IBM Engineering Lifecycle Management - Capabilities

An integrated set of model centric capabilities with Industry-Leading expertise



Realizing MBSE with IBM Engineering Lifecycle Management - Products

An integrated set of model centric capabilities with Industry-Leading expertise



Jump start your digital transformation with IBM MBSE

- Meet aggressive timelines with quality through early validation
- Enable agile systems engineering practices
- Avoid rework by with clear visual stakeholder communication and collaboration
- Streamline integrity and industry standards compliance with design traceability
- Automate manual error prone tasks such as documents and code generation
- Get started with confidence with IBM MBSE proven practices

A recommended practice by industrial engineering practice standards – provides credit for industry standards compliance such as ASPICE, SAE 4754, DO-178C, ISO-26262 and more.

IBM

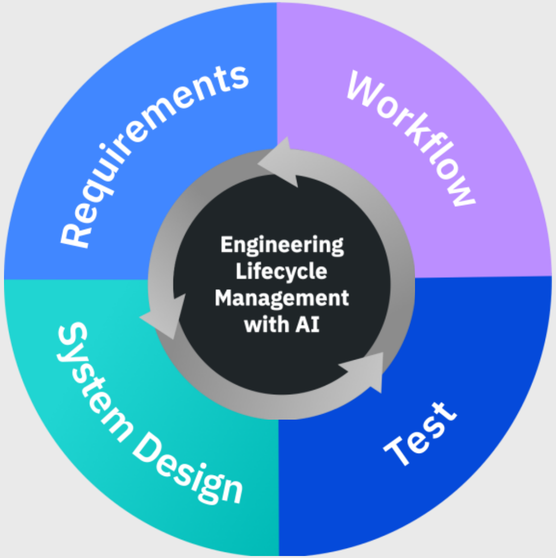
Back up

Technical Implementation

MBSE enabled by IBM Engineering Lifecycle Management

Model Based Systems Engineering

- **MBSE is today's key practice** to address the systems engineering complexities with a digital representations of the systems under development
- **IBM Engineering lifecycle management** MBSE integrates modeling with essential lifecycle applications to enable
  - Early Validation and elaboration of the system requirements
  - Exploration and validation of an effective system architecture and function allocations
  - Design integrity with system design traceability to requirements, test, and change management
  - High quality collaboration among all stakeholders
  - Engineering agility via rapid validation of changes
  - Automating production of key work products such as reports and source code






IBM MBSE solution core capabilities that deliver the value…

Create

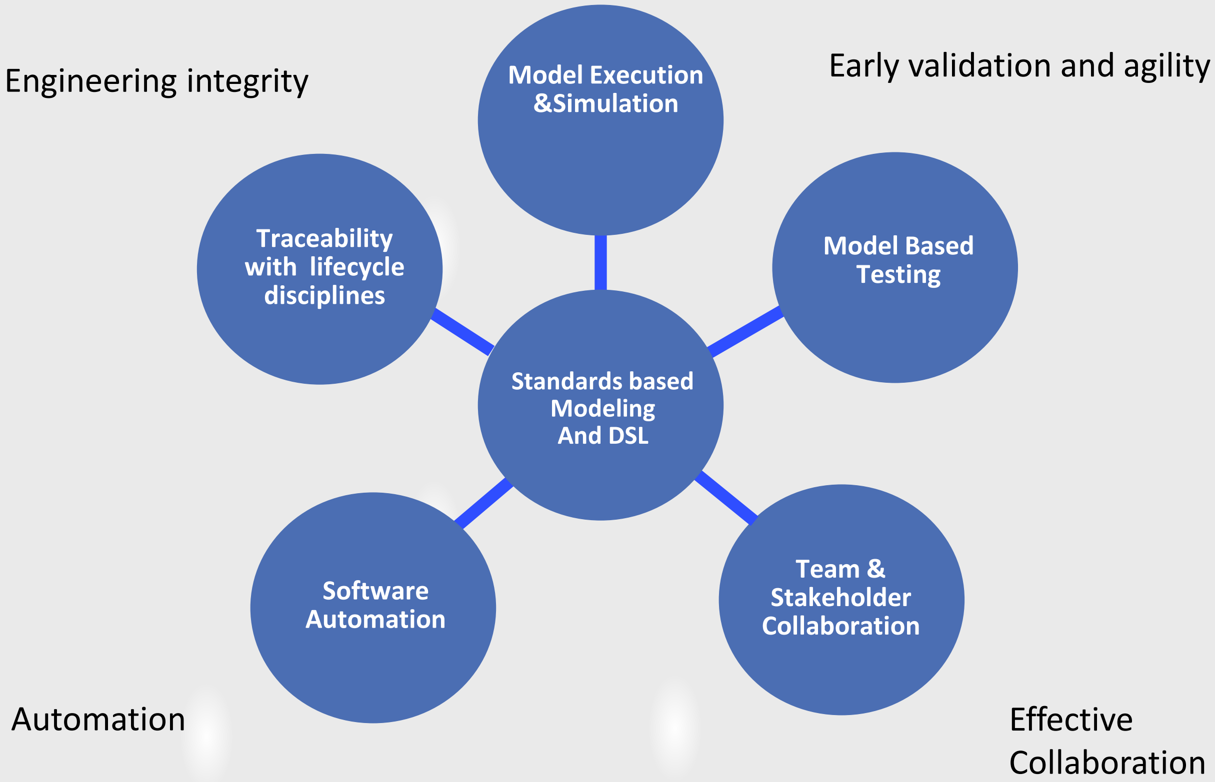
Trace

Analyze

Validate

Automate

Deliver



Standards based industry domain modeling frameworks

Standards based Modeling And DSL

Enterprise/Architectural Modeling

- Mission Planning/Acquisition support
- Impact/Change Analysis
- Usually authored in a architectural framework

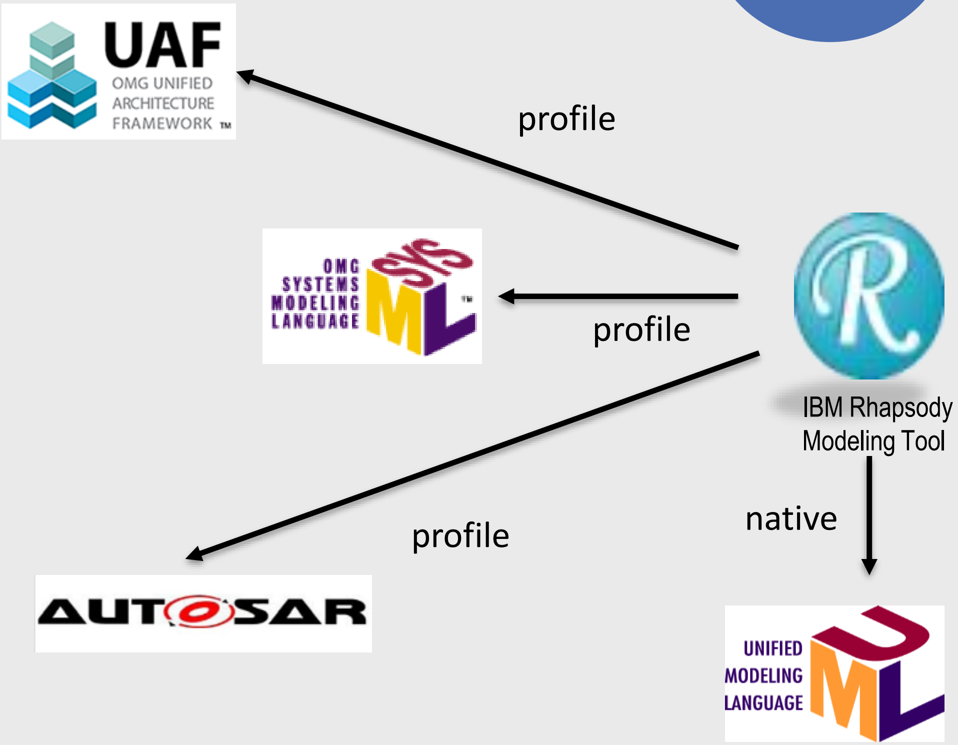
System Modeling

- Descriptive – Functional specification, Interface development
- Analysis – Impact, Trade Studies, Safety and Security
- Usually authored in SysML

Software Modeling

- Specification/Implementation of code
- Reverse Engineering/Documentation
- Usually authored in UML

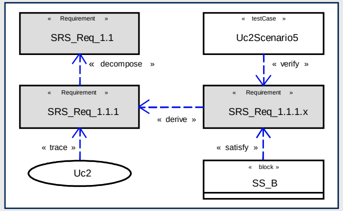
Domain specific modeling: Automotive software architectures



SysML - Diagrams for systems engineers

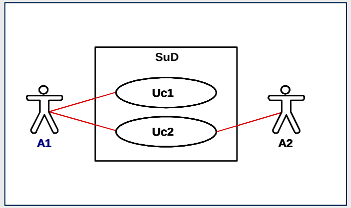
Standards based Modeling And DSL

Requirements Diagram



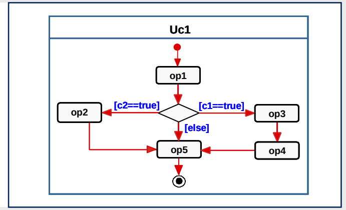
- *Taxonomy of Requirements*
- *Relationship between Model Elements and Requirements*

Use Case Diagram



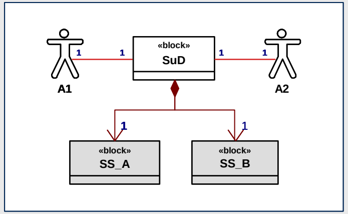
- *Definition of System Scope*
- *Grouping of Requirements into Use Cases*

Activity Diagram

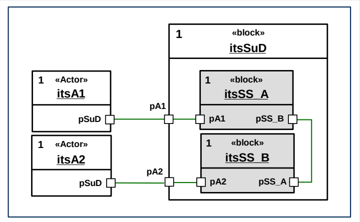


Functional Flow in Use Case / Block(s)

Structure Diagrams

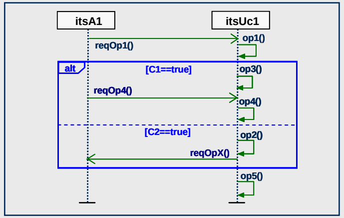


Block Definition Diagram: Structural Elements (Blocks) and their Relationship



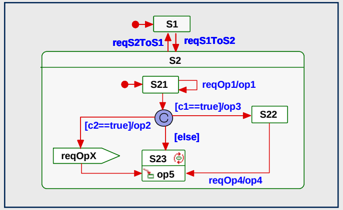
Internal Block Diagram: Realization of System Structure

Sequence Diagram



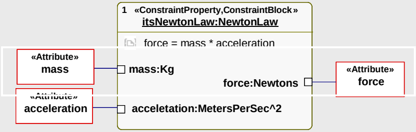
Message Interactions between Nodes

Statechart Diagram



State-based Behavior of Block

Parametric Diagram



Parametric Relationship between System Properties.

Streamlining MBSE with lifecycle traceability

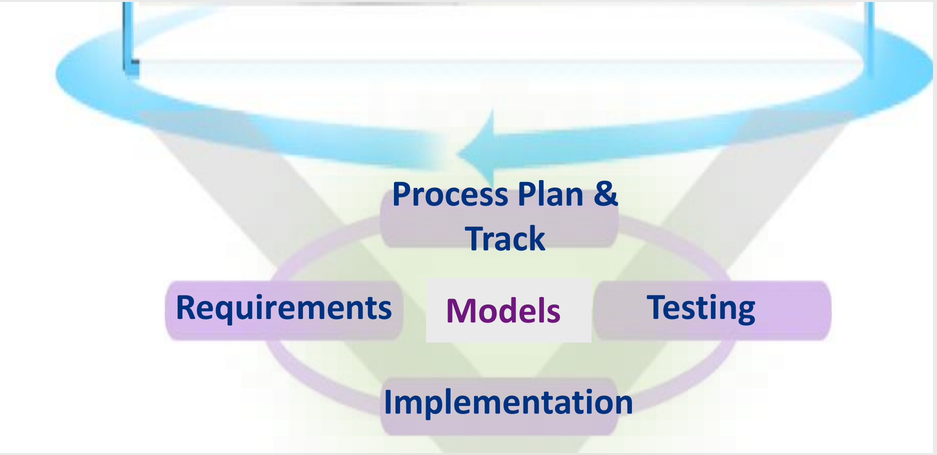
Traceability with lifecycle disciplines

– Requirements traceability enables requirements verification and alignment with design

– Liniking Modeling activities with track & plan (work items) to provide visibility and track progress

– Traceability with implementation: SW modules, HW blocks, Mechanical BOMs to ensure system integrity

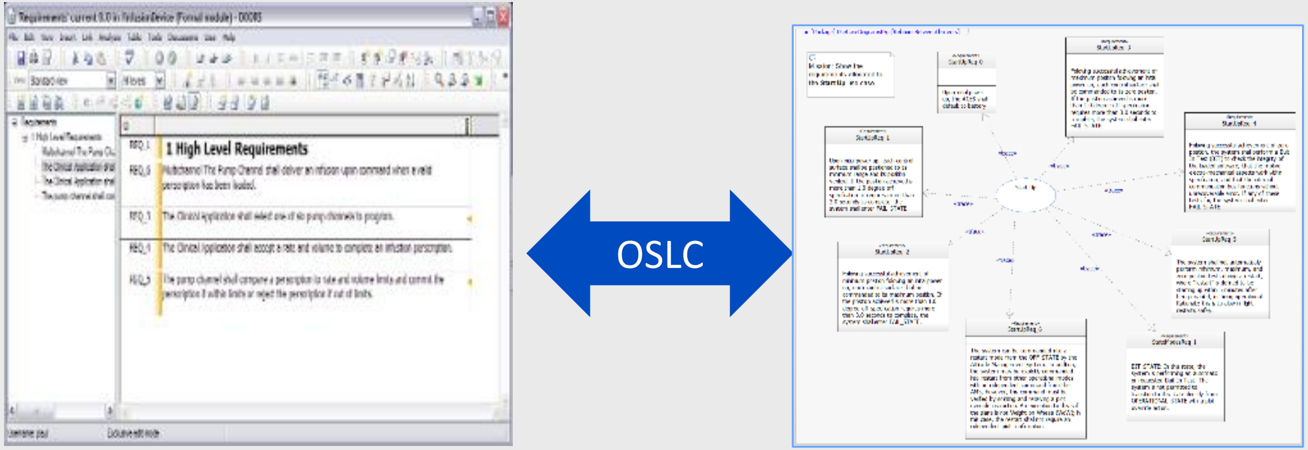
– Traceability with Test management: coverage of architectural components by verification plans



Advanced requirements and MBSE traceability

Traceability with lifecycle disciplines

- Today’s requirements are captured in a requirements management tool (IBM DOORS/DNG)
- Requirements are linked across tools to MBSE elements via OSLC
- Create transient requirements “shadows”
  - No need to manage copies and synchronizations
- Enables full traceability between requirement levels and their realizations
  - Impact Analysis
  - Coverage Analysis



Early design validation with model execution and simulation

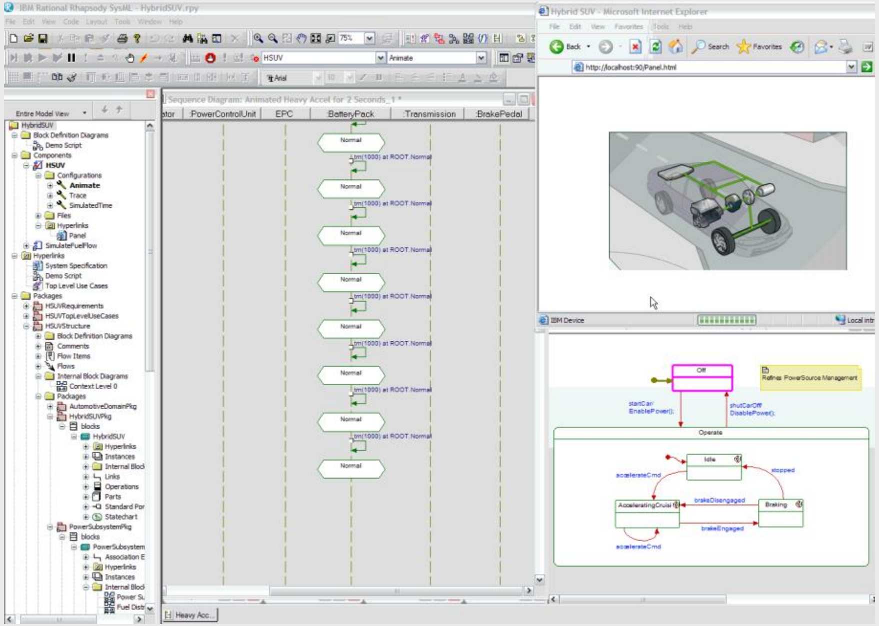
Model Execution &Simulation

Model execution validates the system behavioral specification

- System black box behavior
- System components interaction (white box)

Throughout execution various aspects are highlighted

- Emergent system interaction depicted as a dynamic sequence diagrams
- System states depicted as animated Statecharts
- Sequencing of activities
- Exchanges of events and data

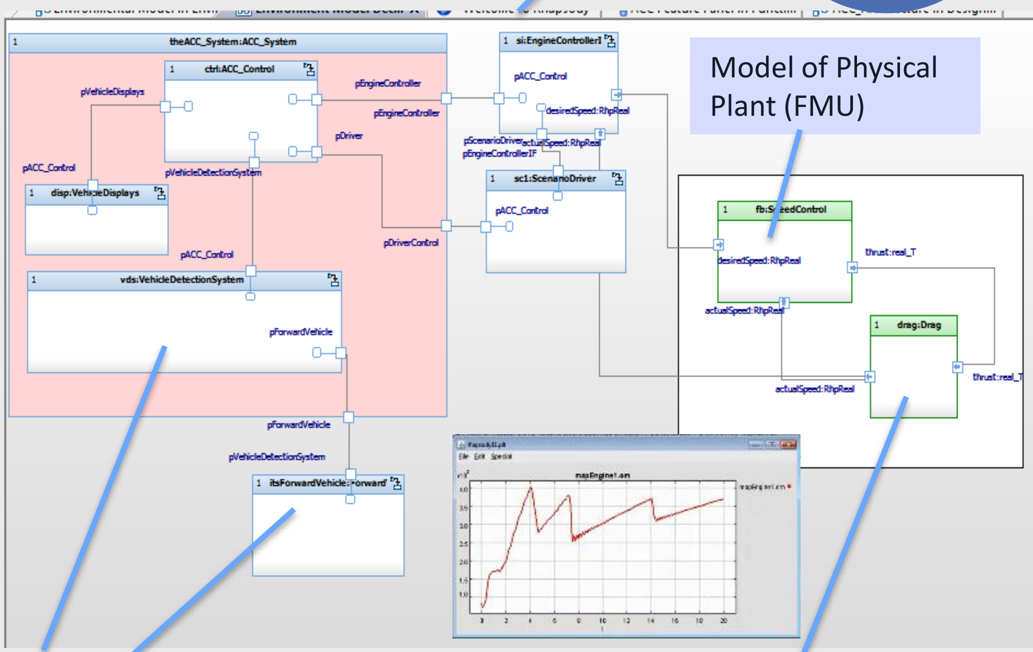


High Fidelity validation with joint Simulation

Simulation architecture (SysML)

Model Execution &Simulation

- Leverage domain specific simulation tools to simulate physical systems and control algorithms
- Integrate FMI compliant simulation tools as part of the SysML system design environment
- Leverage Modelica solvers (or Simulink) to orchestrate multi-domain simulation
- Specialized integration with the Mathworks Simulink enables integration of Simulink models as part of the SysML model.



Model of Physical Plant (FMU)

Discrete functional behavior models in SysML

FUNCTIONAL MOCK-UP INTERFACE

Simulink Model of Control Algorithm

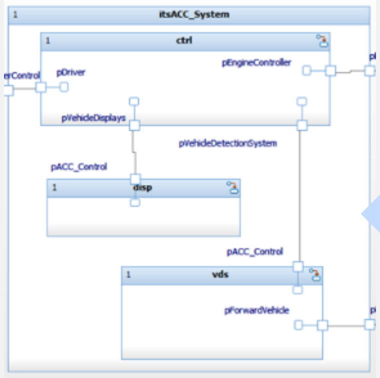
MODEL Based Testing

Model Based Testing

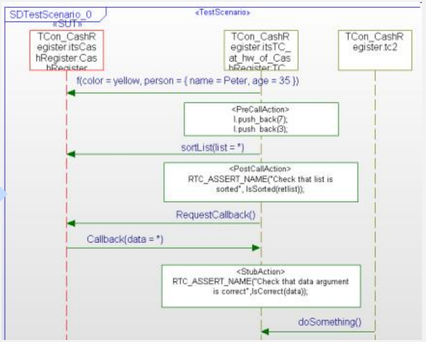
- Use modeled scenarios to verify system design
- Test how the elaborated black and white box behavior align with the requirements
- System level testing
- Component level testing
- Map test scenarios to requirements and calculate requirements coverage

Test Architecture

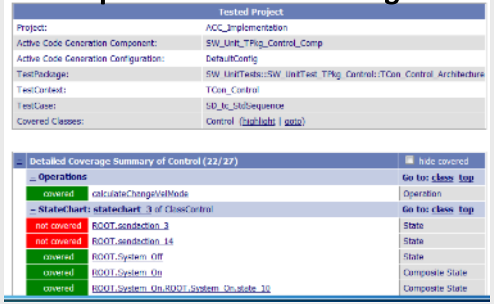
System Under Test



System test scenarios



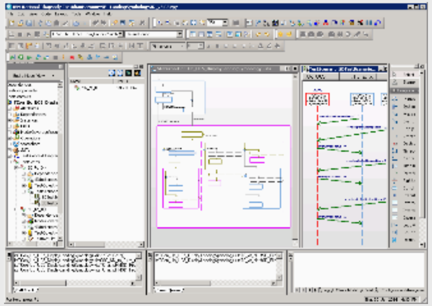
Report results & coverage



Tested Project	
Project:	ACC_Implementation
Active Code Generation Component:	SW_Unit_TPkg_Control_Comp
Active Code Generation Configuration:	DefaultConfig
TestPackage:	SW_UnitTests::SW_UnitTest_TPkg_Control::TCon_Control_Architecture
TestContext:	TCon_Control
TestCase:	SD_tc_StdSequence
Covered Classes:	Control (highlight | goto)

Detailed Coverage Summary of Control (22/27)		hide covered
Operations		Go to: class top
covered	calculateChangeVelMode	Operation
StateChart: statechart_3 of ClassControl		Go to: class top
not covered	ROOT.sendaction_3	State
not covered	ROOT.sendaction_14	State
covered	ROOT.System_Off	State
covered	ROOT.System_On	Composite State
covered	ROOT.System_On.ROOT.System_On.state_10	Composite State

Execute tests



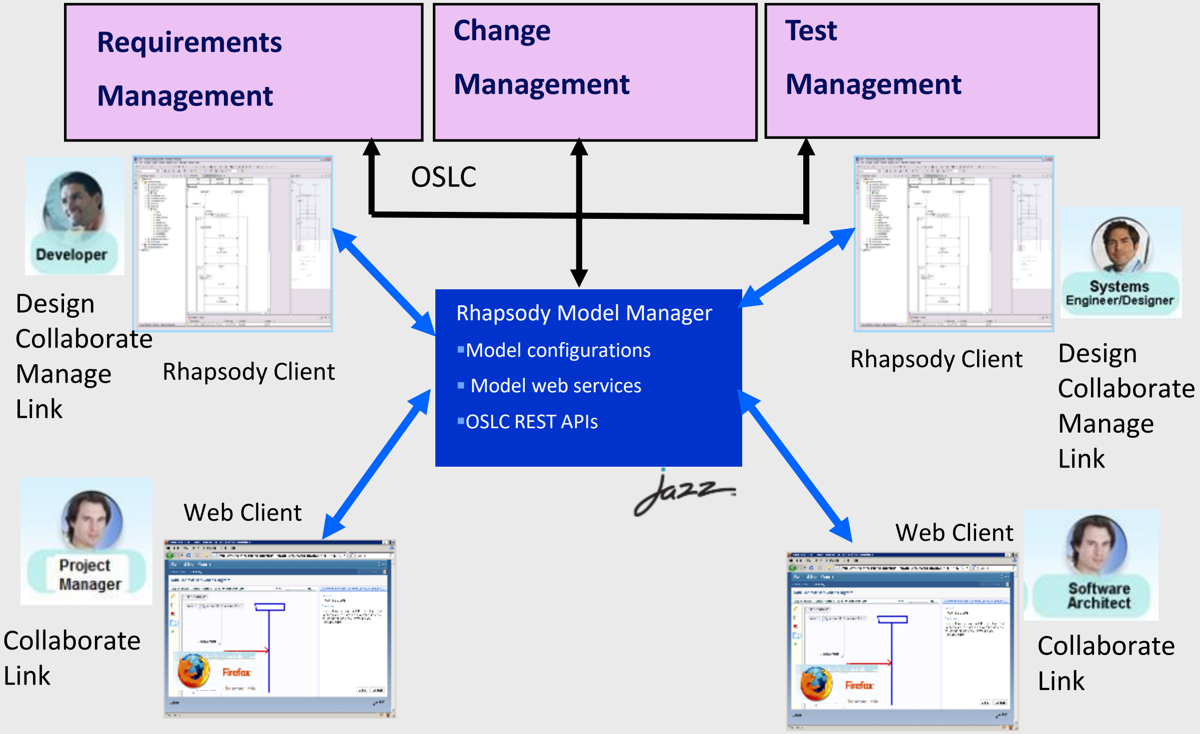
Fostering collaboration across the value chain

- Models are uniquely collaborative artifacts, to share creation, ideation, communication and solicitate feedback
- Peer co-creation – how models can be jointly created by a team of engineers
- Model reviews – how various stakeholders can review and provide feedback on models
- Document generation – how models can be automatically transformed to a set of necessary documents for stakeholders and customers

Model Management and Collaboration

Team & Stakeholder Collaboration

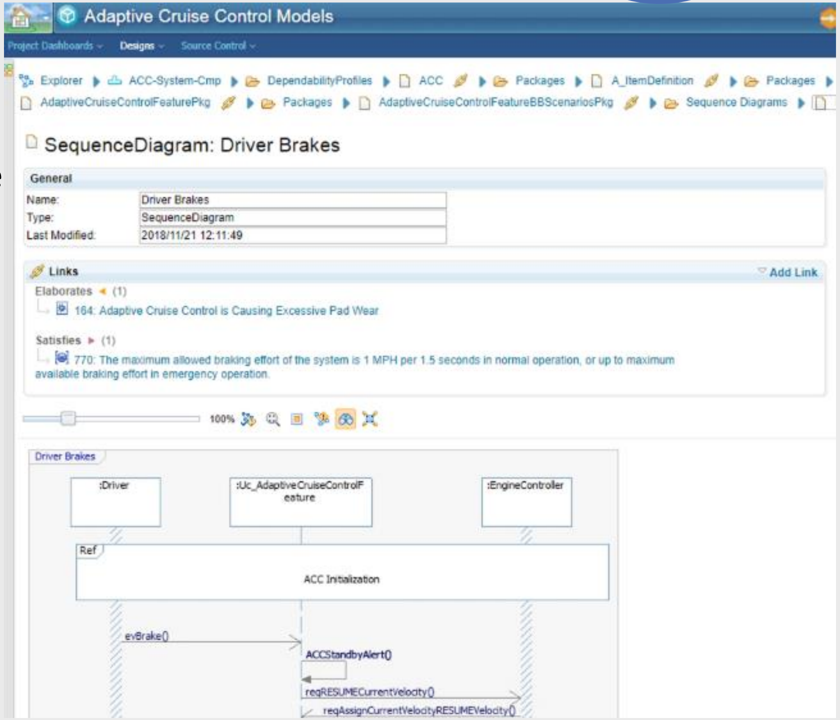
- Model management and lifecycle integration and traceability to other disciplines
  - OSLC service
- Collaborative working on a model by multiple stakeholders
- Parallel development with multiple streams
- Model baselining and version control
- Baselining models together with all other lifecycle artifacts
- Distributed model development across teams and geographies!



Model Reviews

Team & Stakeholder Collaboration

- Models are ideal for clear communication among stakeholders
- Orchestrate a digital model review where stakeholders can access modeling artifacts and provide feedback
- System models can be part of a review similar to other lifecycle artifacts such as requirements or test cases
- Reviewing models can be done via a standard web client
- Reviews can be orchestrated leveraging specific review tasks



Automate delivery of engineering documents

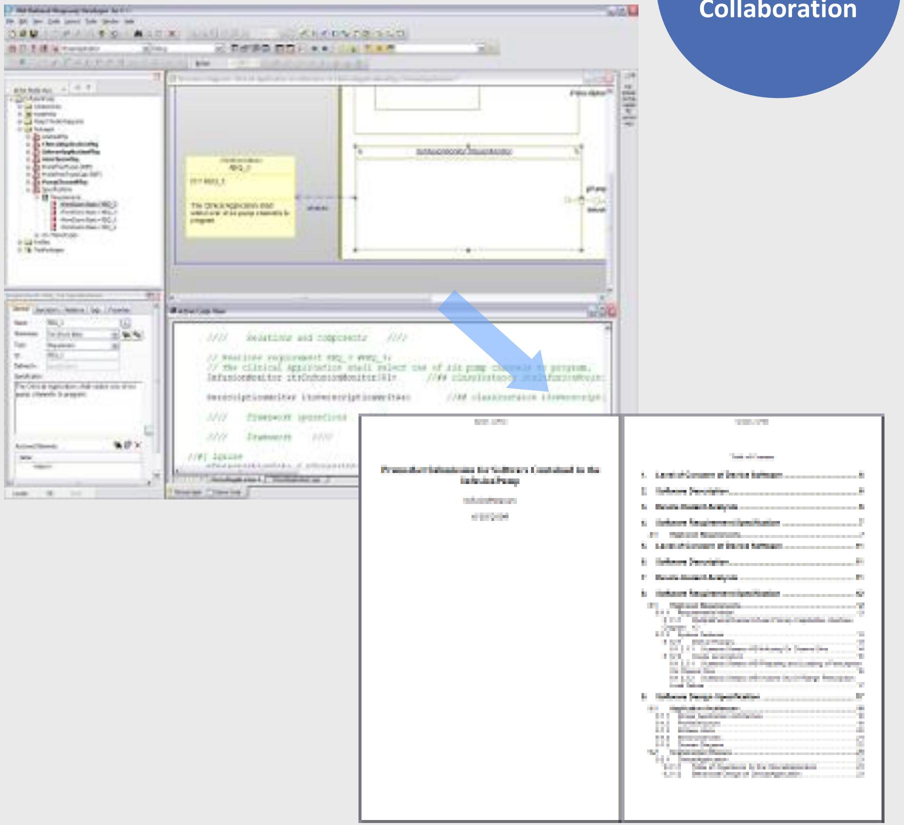
Team & Stakeholder Collaboration

– Various systems engineering stakeholders consume documents to review and approve specifications and designs

– Producing quality documents for specification, design, ant system and subsystem level is an important work product of systems engineering

– IBM MBSE automates the production of the various engineering documents

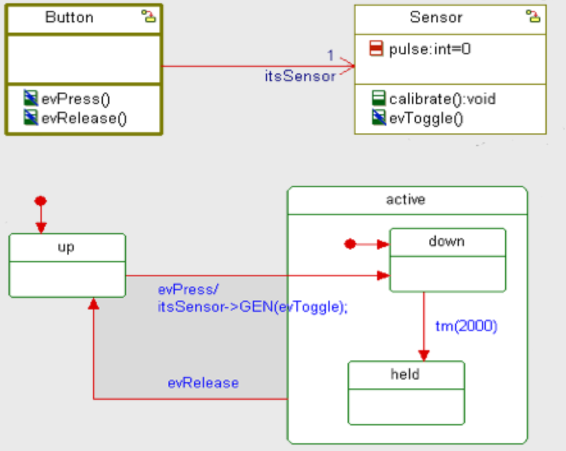
– Automatic production of engineering documents results in higher quality documents which are self consistent and it saves significant time & costs spent today by traditional approaches

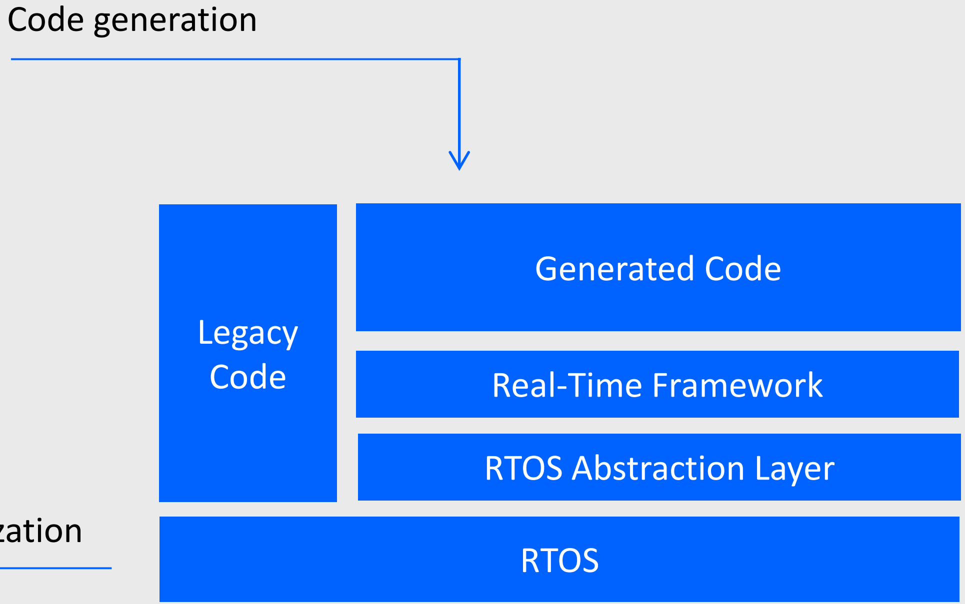


Model Driven Software Development (MDD)

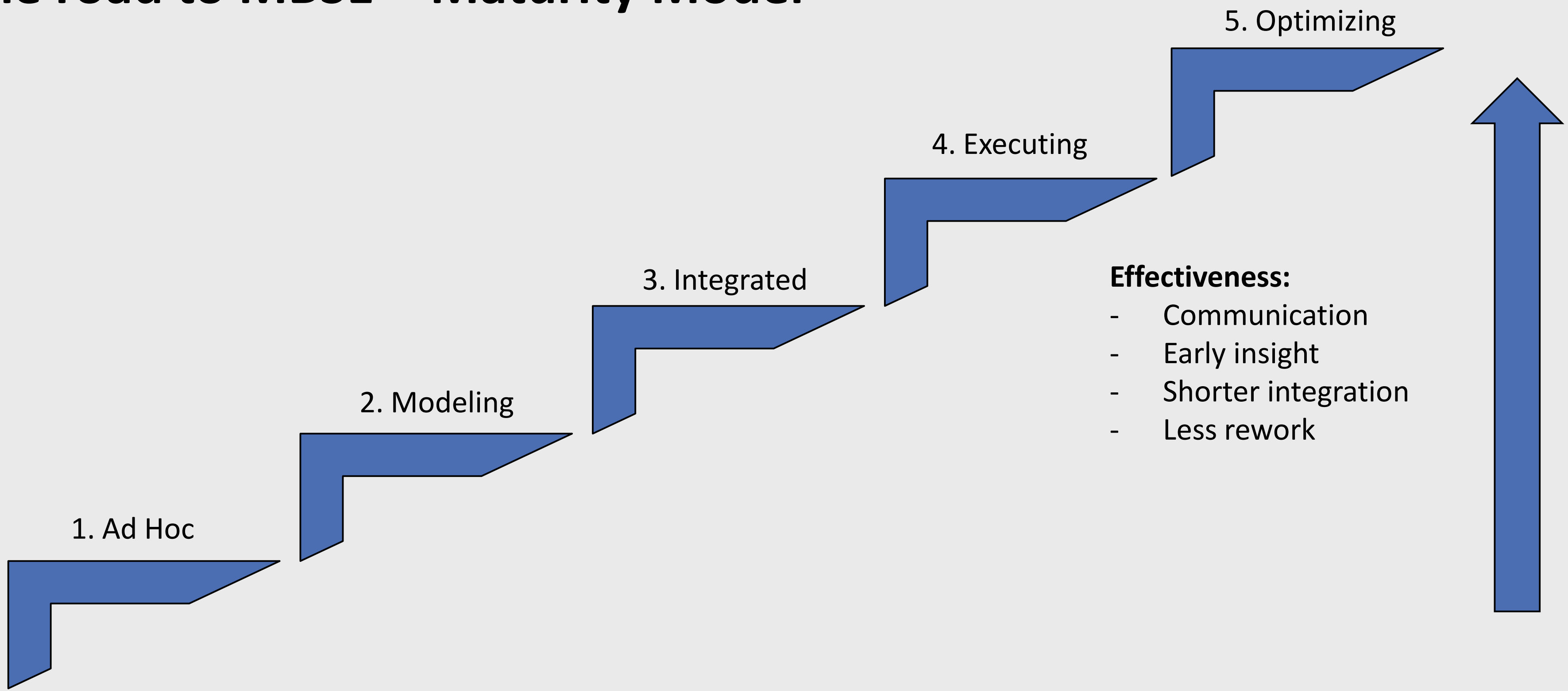
Software Automation

- MDD enables creation and visualization software implementation code from/by models
- Models capture the architectural and behavioral aspects of the code using graphical abstractions that can be rapidly created, and effectively shared and communicated across developers and stakeholders





The road to MBSE - Maturity Model



Modeling is informal, not a deliverable of the process

Modeling are standards and practice based. Models are formal deliverables of the process

Models are integrated into lifecycle artifacts and disciplines: requirements, workflow, test, and implementation

Models are verified using Execution and use for logical verification of the design

Models are used to optimize system architecture and parameters based on high fidelity simulation and parametric solving