34th Annual INCOSE international symposium

hybrid event

Dublin, Ireland
July 2 - 6, 2024

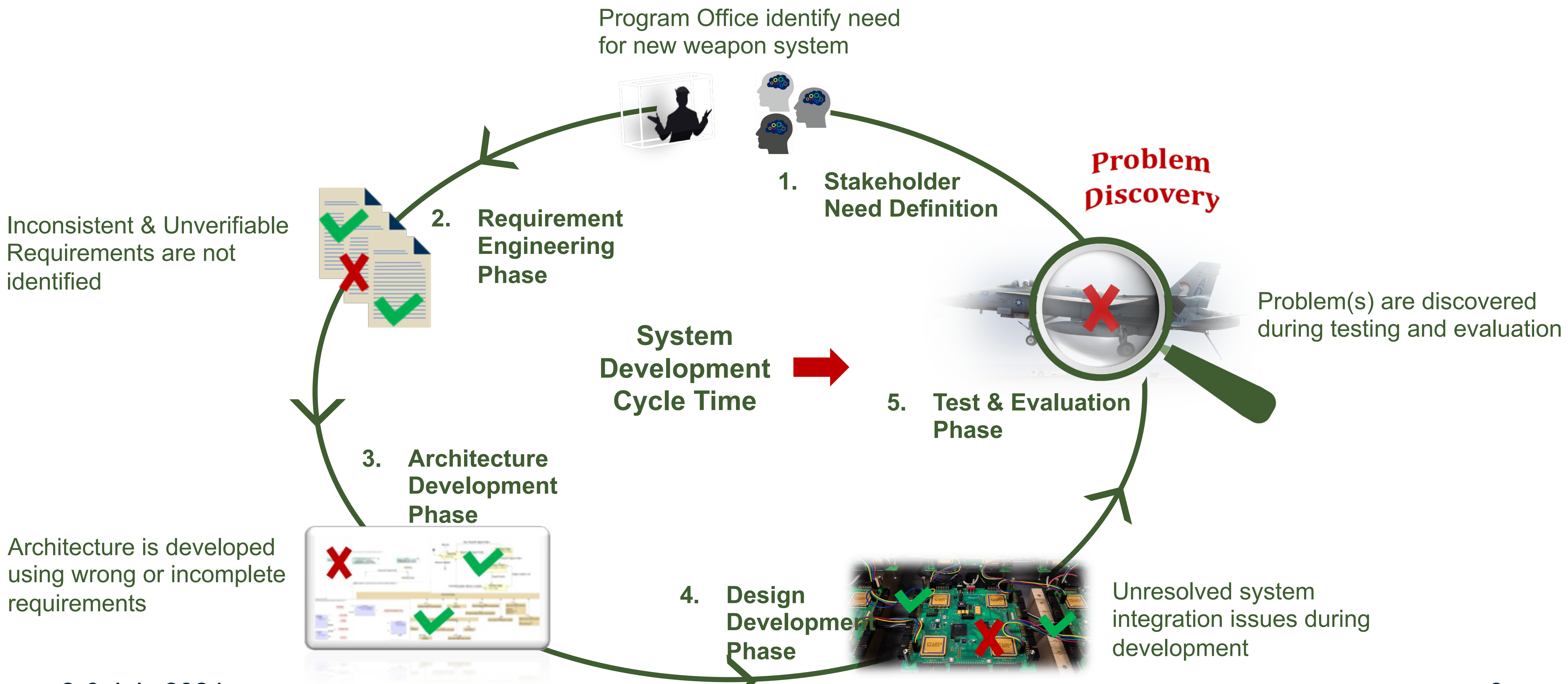
Shifting Left: A Test-Integrated System Architecture Approach to Reduce System Development Cycle Time

Georgia Tech Research Institute

Dr. Awele Anyanhun, CSEP
Senior Research Engineer
Georgia Tech Research Institute (GTRI)
awele.anyanhun@gtri.gatech.edu

Dr. Craig Arndt, ESEP
Principal Research Engineer
Georgia Tech Research Institute (GTRI)
craig.arndt@gtri.gatech.edu

Problem Discovery during test and evaluation (T&E) impacts on-time delivery of operationally effective systems.



Decision makers need to be equipped with crucial T&E data to inform decisions pertinent to system feasibility.

What's the right decision to take?

How reliable is this data?

Can this acquisition shift left???

How do we quantify the risk associated with this acquisition?

How soon can testing begin?

What test capabilities are available to test this system?

The end goal is to implement robust T&E modeling to generate data and insights for decision making.

A test-integrated architecture minimizes project risk by linking test capability with system artifacts.

1

REQUIREMENTS VIEWPOINT

Specify Testing, Test Facility & System Requirements

2

TEST FACILITY & SYSTEM VIEWPOINT

Architect a Test-Integrated Federated Model

3

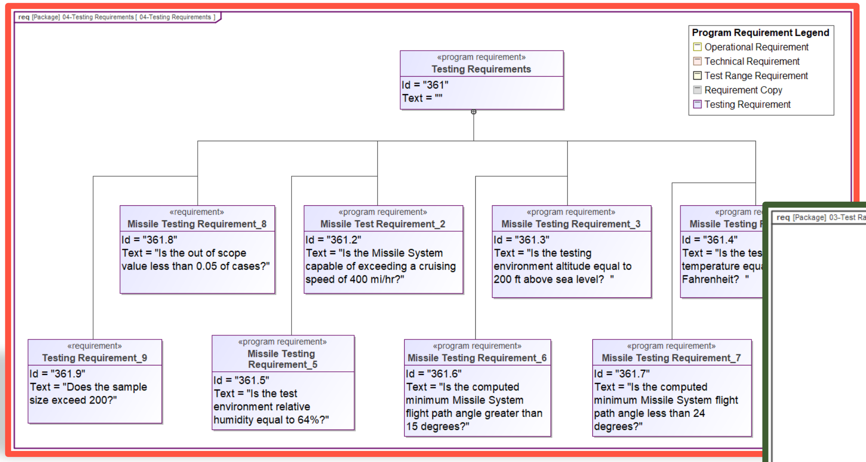
TEST CONFIGURATION VIEWPOINT

Configure & Execute model-Based Test

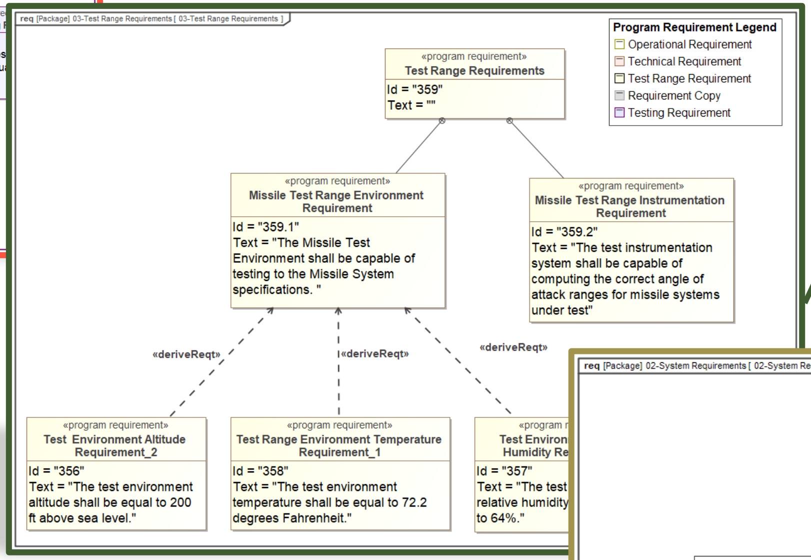
Requirements Model Viewpoint

Develop a requirements model to link together system, test facility, and testing requirements

Capture the system, test facility, and testing requirements in a requirements model to enable traceability & analysis.

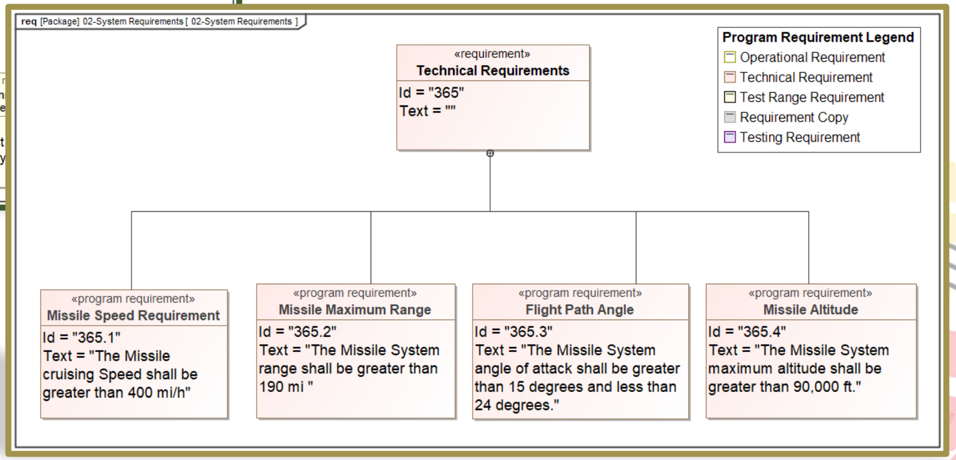


Testing Requirements



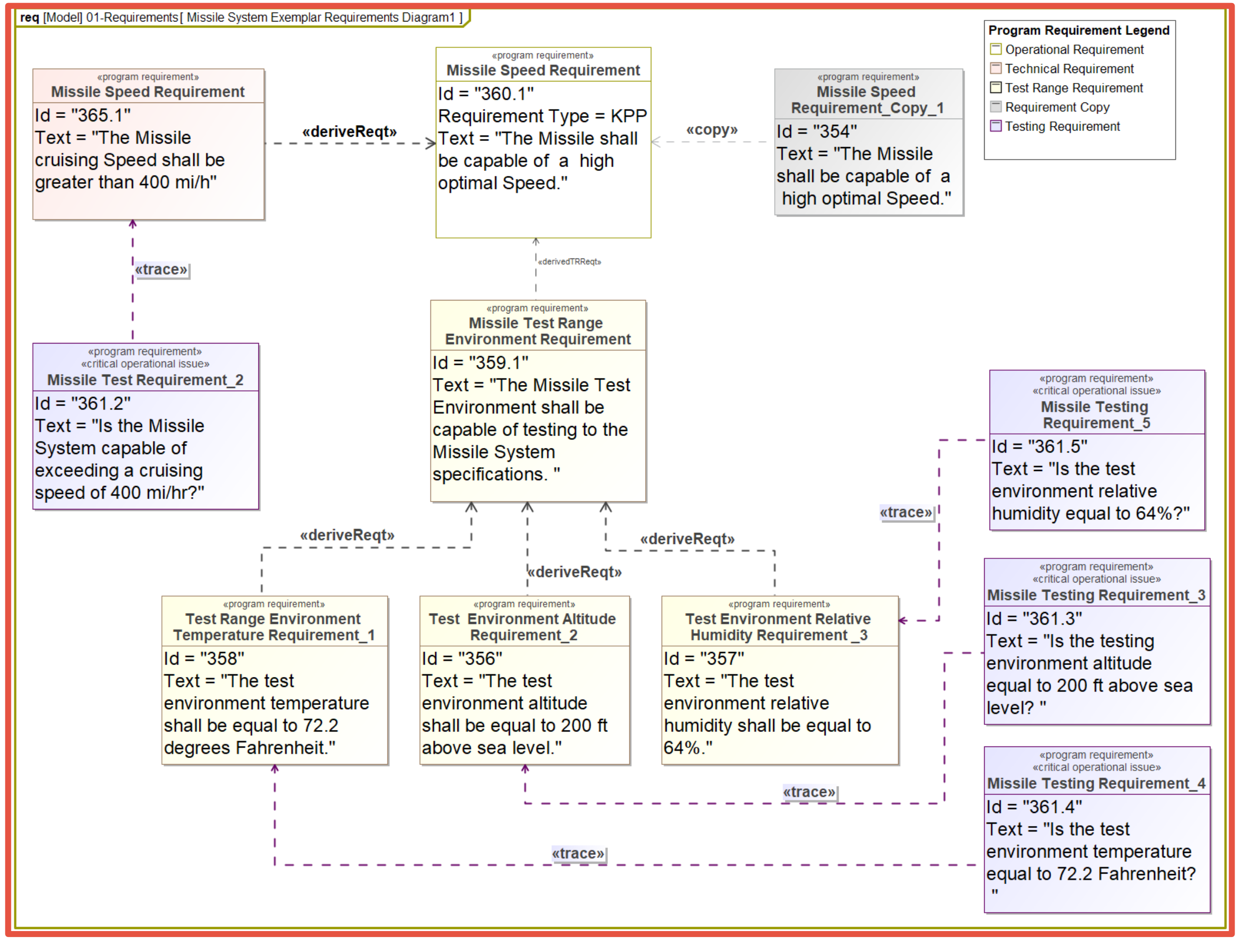
Test Facility Requirements

Test range requirements allow for real-time gap and impact analysis of a test range's ability to test a given system capability



System Requirements

Traceability relationships are created between the various requirement types.

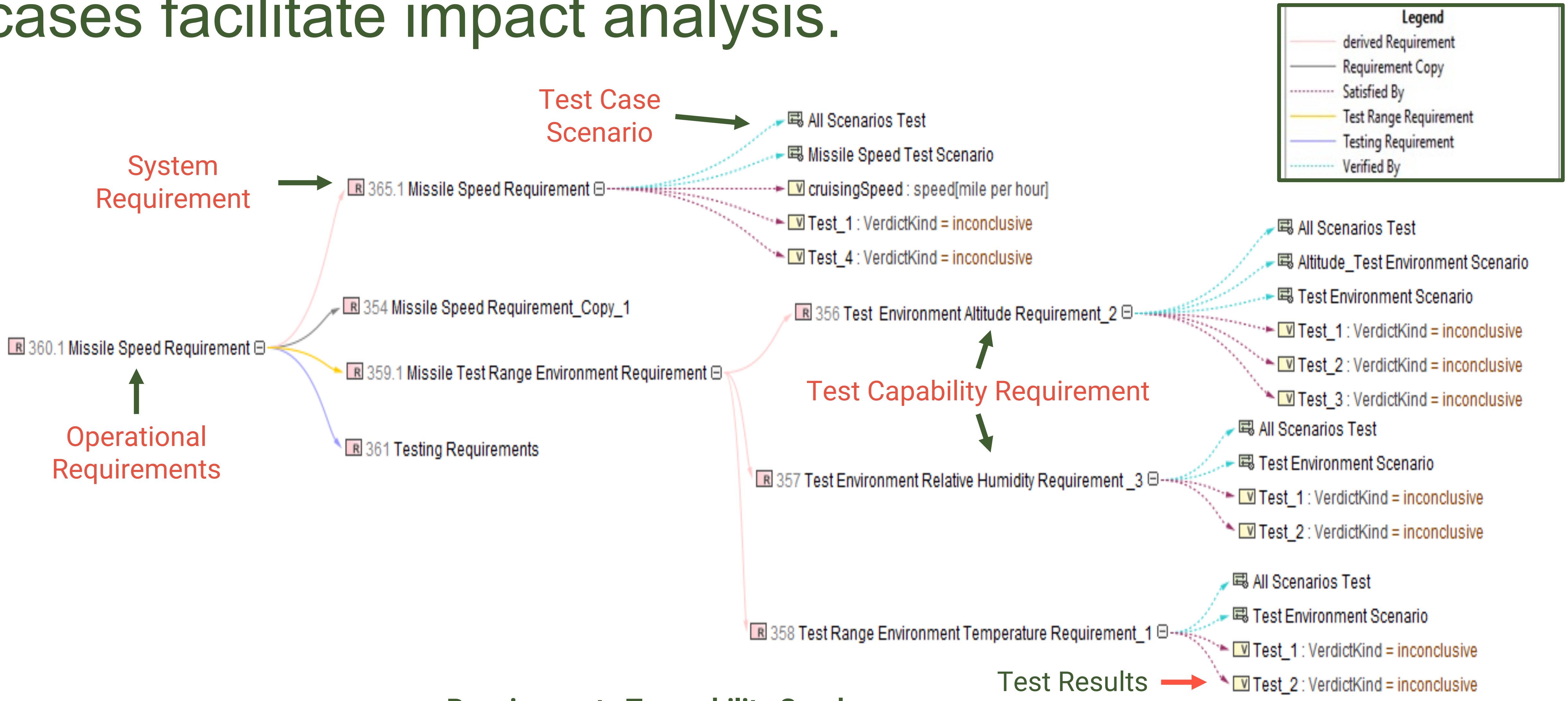


The Requirement Traceability Pattern:

- Establishes traceability between requirement types.
- Reduces time needed to assess if a given test range can meet testing requirements.
- Enables more efficient test planning by organizations.

Requirements Model Traceability pattern

Traceability relationships between requirements and test cases facilitate impact analysis.



Requirements Traceability Graph

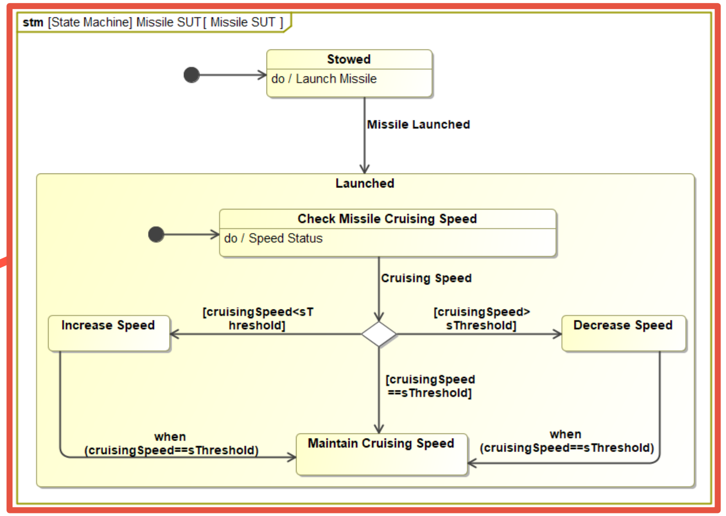
Test-Integrated Digital Model Viewpoint

Develop a set of models that specify and define the system of interest and test capability data.

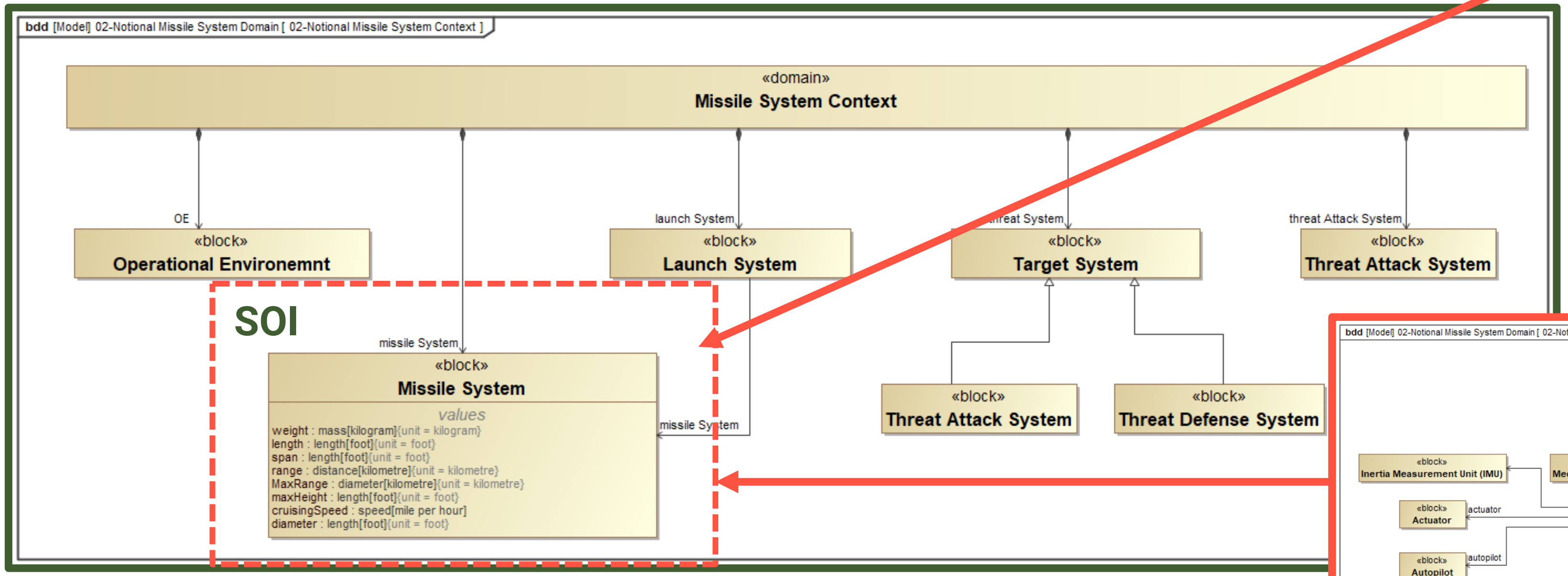
Architect the System-of-Interest

Create a system model to capture system-level properties and capabilities of the weapon system.

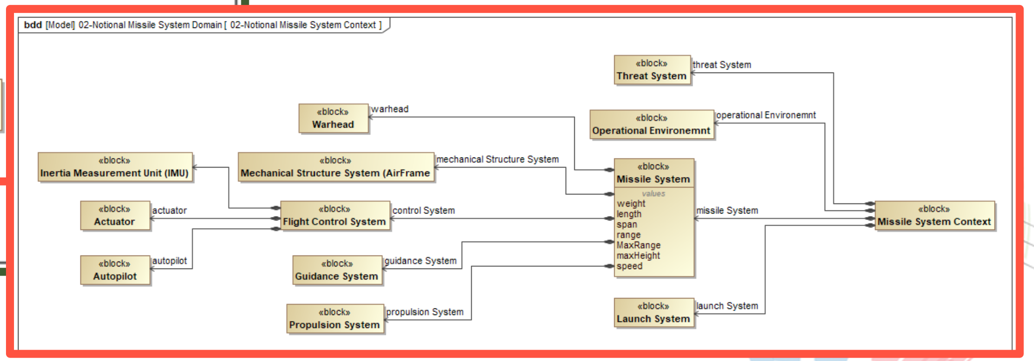
- The Missile System Model captures relevant system properties and capabilities needed to inform quick decision making at the program level.
- Abstractions, simplifications and intended limitations of system properties during model development based on project requirements.



Monitor Missile System Use Case Behavior View



Missile System Contextual View

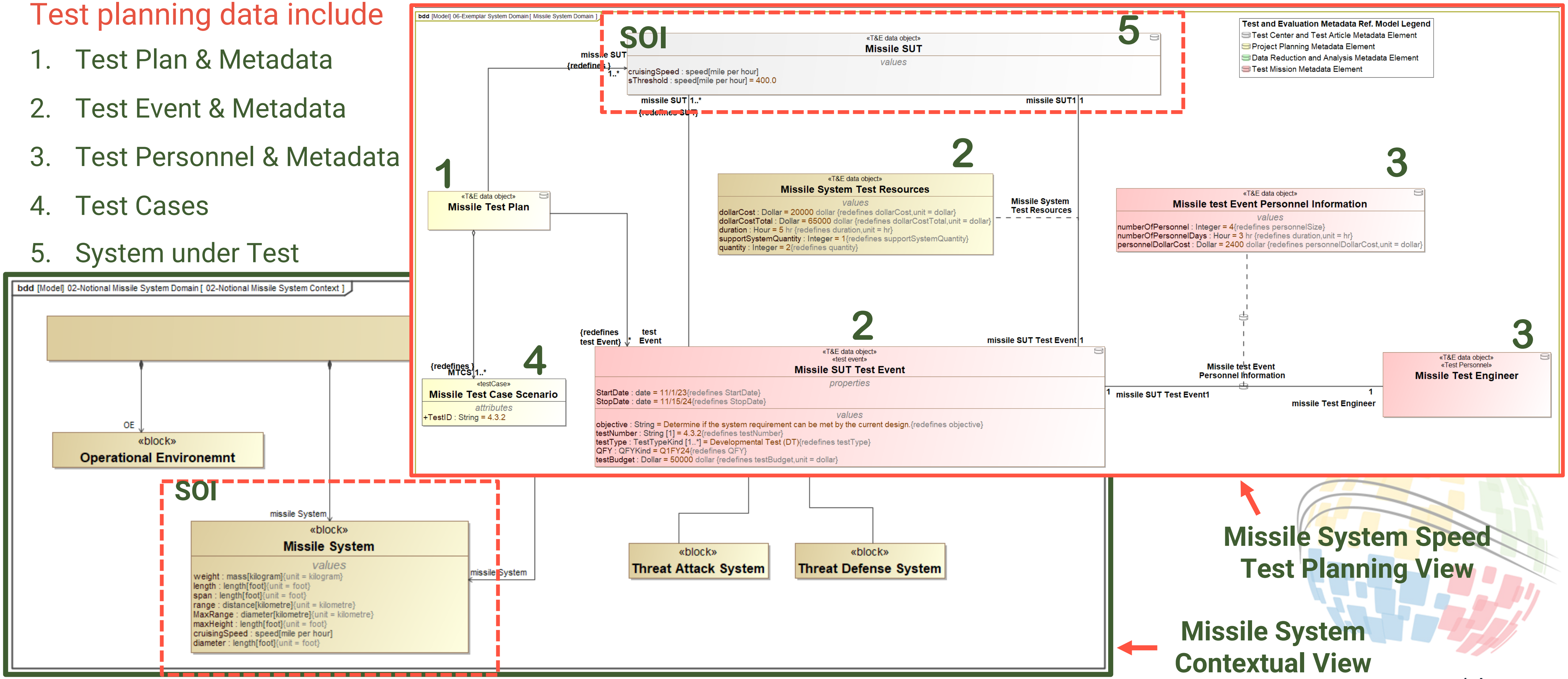


Missile System Structural View

Create test planning views to capture pertinent test planning data required to test the system of interest.

Test planning data include

1. Test Plan & Metadata
2. Test Event & Metadata
3. Test Personnel & Metadata
4. Test Cases
5. System under Test



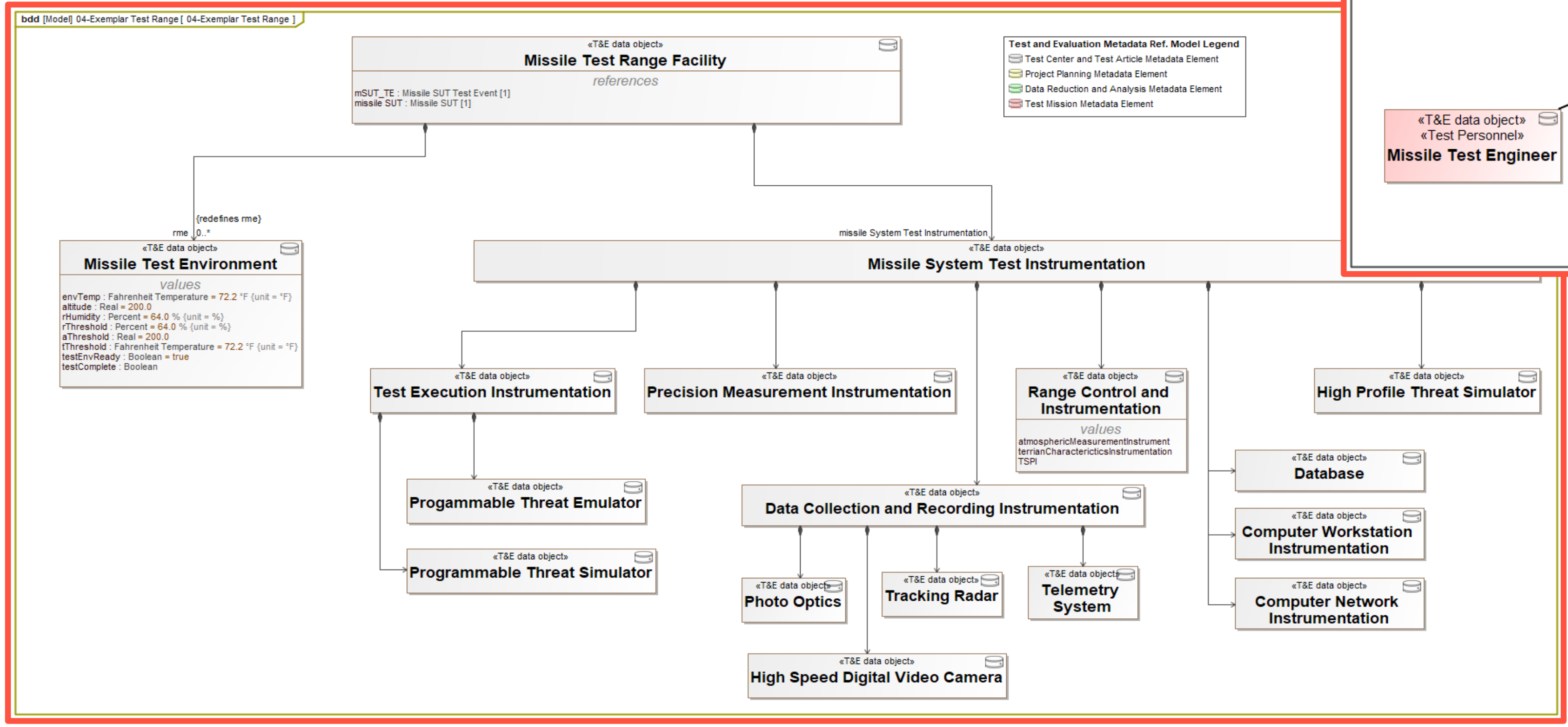
Missile System Speed Test Planning View

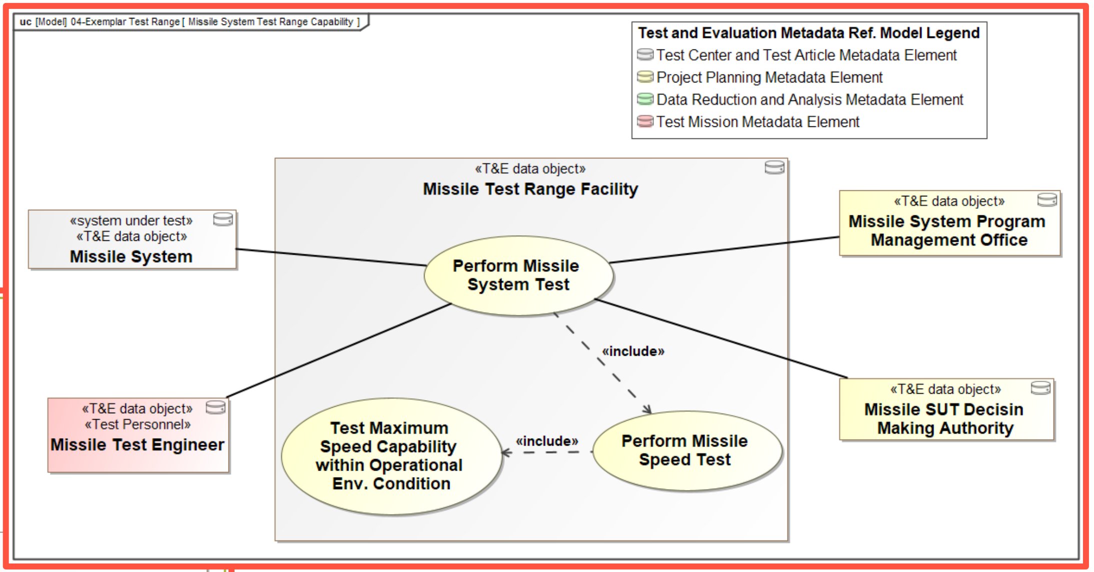
Missile System Contextual View

Develop Model-based Representations of Testing Capabilities

Develop a model-based representation of the testing capabilities required to test the weapon system.[1]

- Test range capabilities include the types/forms of tests a test facility is capable of executing.
- Defining test range resources & their properties enable construction of holistic and integrated test case configurations.



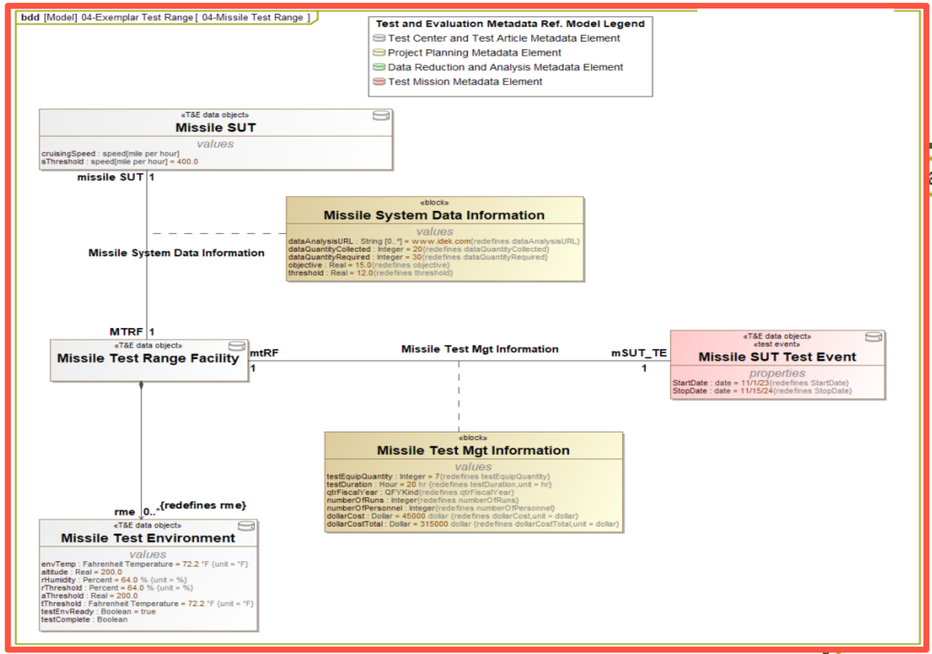


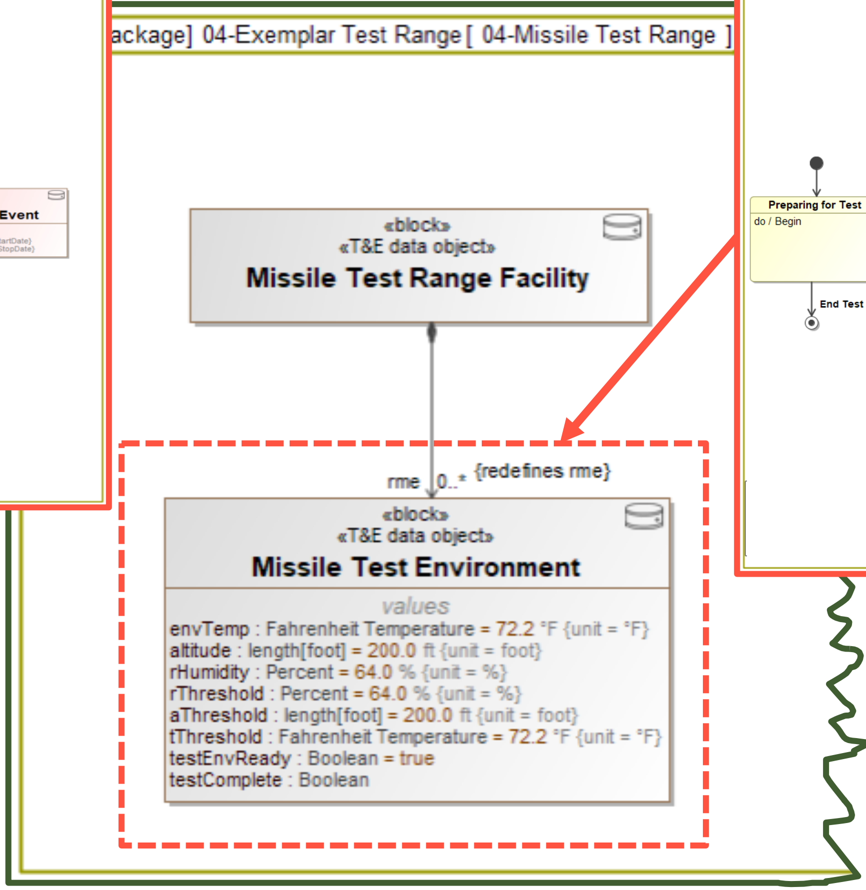
Use case diagram captures the "Perform Missile Speed Test Capability"

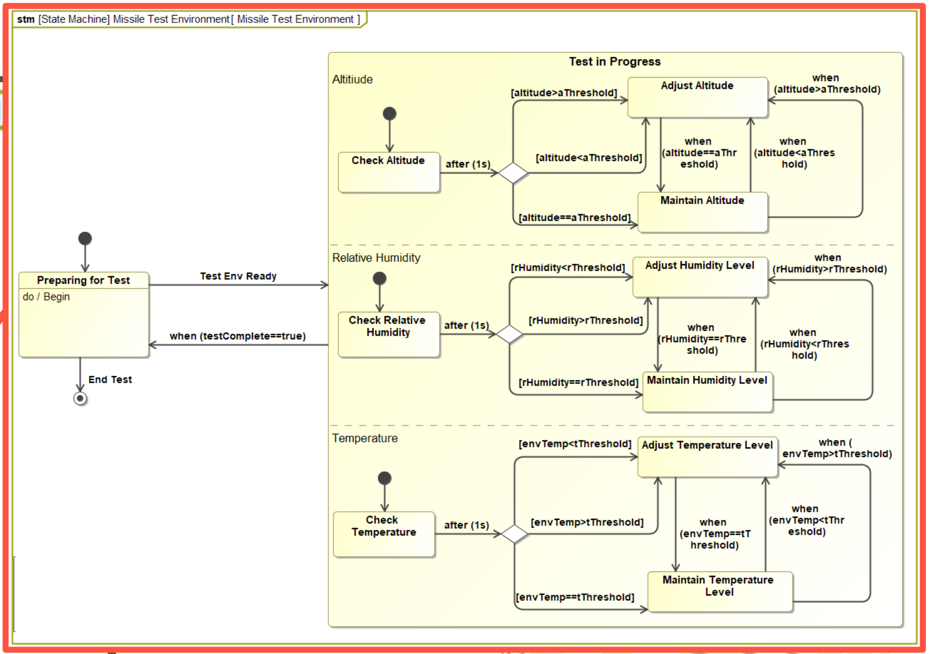
High-Level Missile Test Facility Structural View"

Develop a model-based representation of the testing capabilities required to test the weapon system.[2]

Model view captures the diagram used in simulating the behavior of the missile test environment characteristics.







Missile test planning data required for adequate testing of the system is specified in the test range model.

Model view captures the temperature, altitude and humidity of the operational environment"

Test Configuration Viewpoint

Perform Model-based Testing of system and testing capabilities.

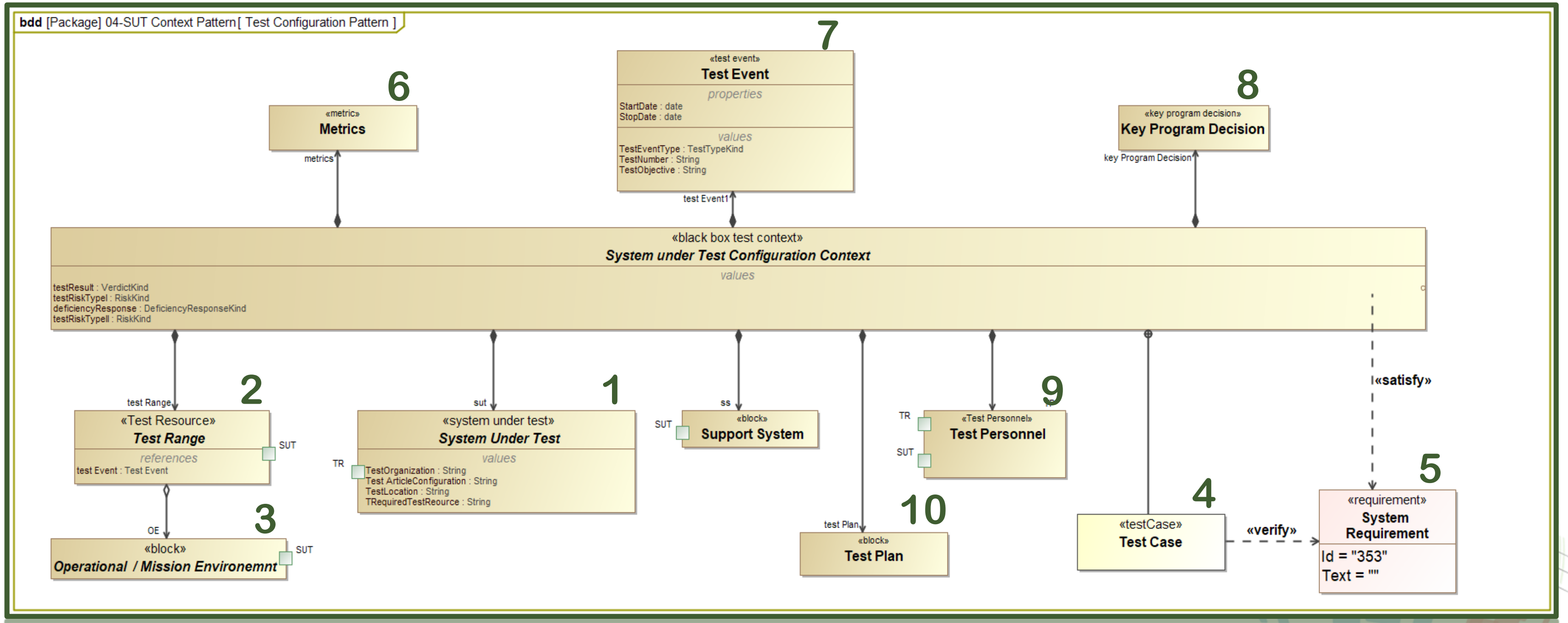
Setup the Model-Based Test Configuration

The tailorable model-based test configuration pattern shown describes the testing context for the SUT.

- The Model-Based Testing Context enables the verification and validation of system-level requirements while providing visibility into test range capabilities.

MB-Test Pattern links together model representations of:

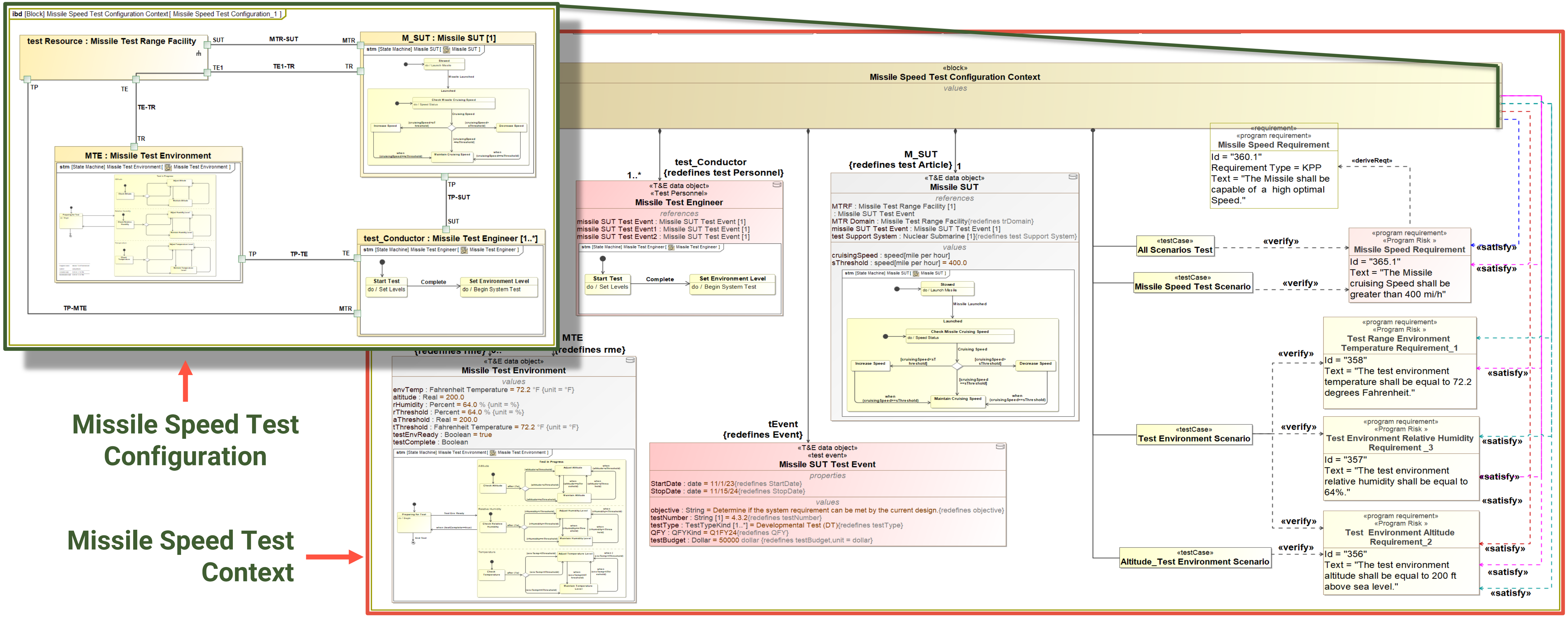
- System under Test (SUT)
- Test Facility Capabilities / Resource
- Operational Environment
- Test Case / Test Scenario
- Requirements
- Metrics
- Test Event
- Key Decisions
- Test Personnel
- Test Plan



Model-Based Test Configuration Pattern

The missile speed test context highlights key elements participating in each test run instance.

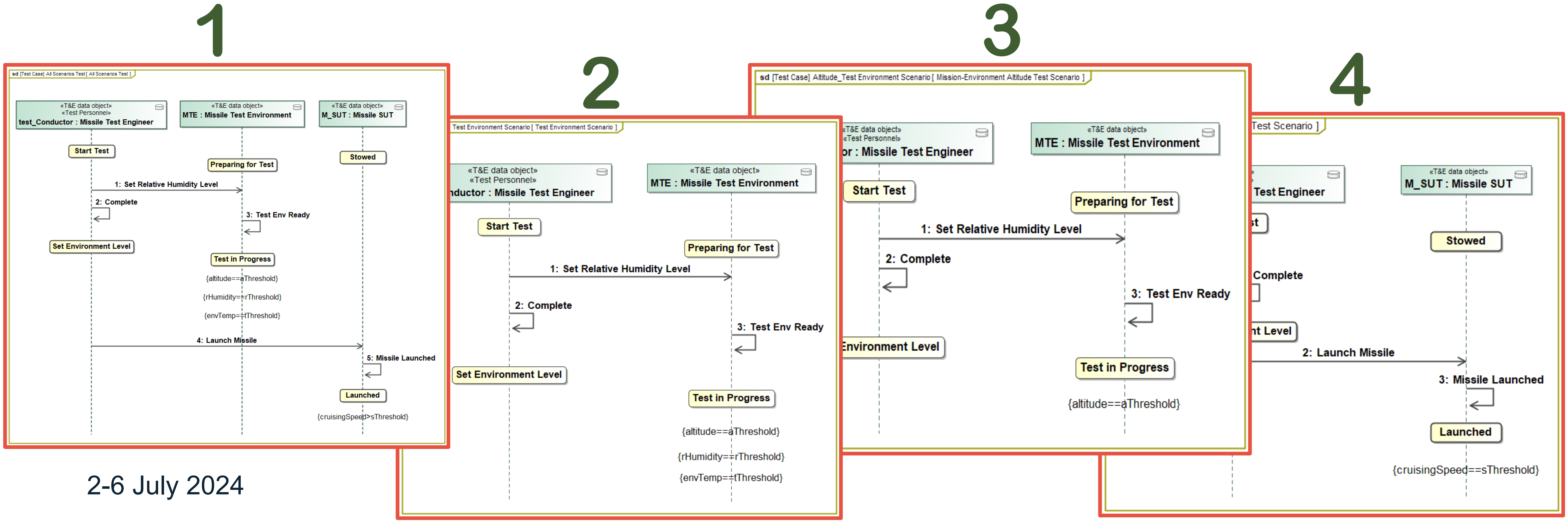
Utilizing the model-based test configuration facilitates quick analysis of required capabilities using virtual test range resources.



Create Test Cases and Test Case Scenarios

Develop model-based test cases and test scenarios for the system capability being tested.[1]

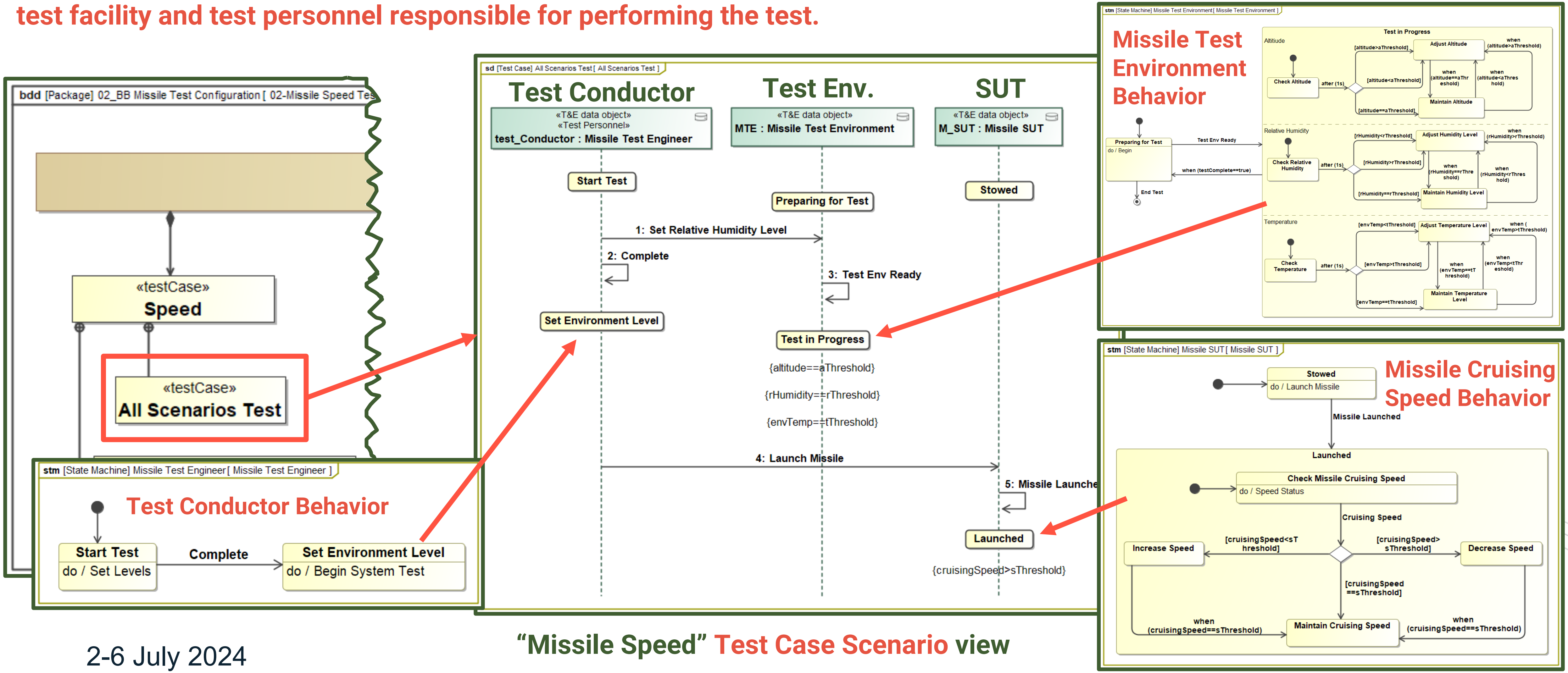
1. Test Environment & Test System Test Case Scenario
2. Missile Test Environment Test Case Scenario
3. Test Environment Altitude Test Case Scenario
4. Missile Cruising Speed Test Case Scenario



2-6 July 2024

Develop model-based test cases and test scenarios for the system capability being tested.[2]

Test Cases simulate not only the behavior of the SUT but also the behaviors of the test facility and test personnel responsible for performing the test.

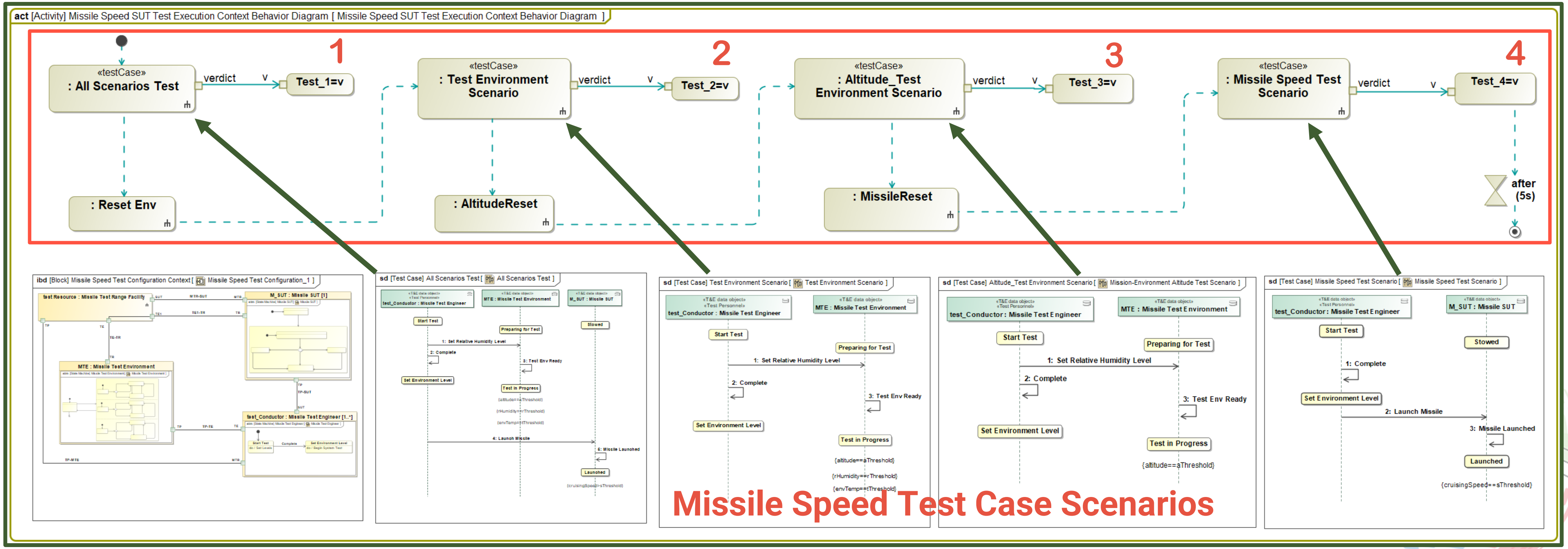


"Missile Speed" Test Case Scenario view

Execute Tests & Evaluate Test Results

Executing the test configuration generates four test results per test run instance.

Execution of the test context behavior (Activity Diagram) generates four test results per test run instance.

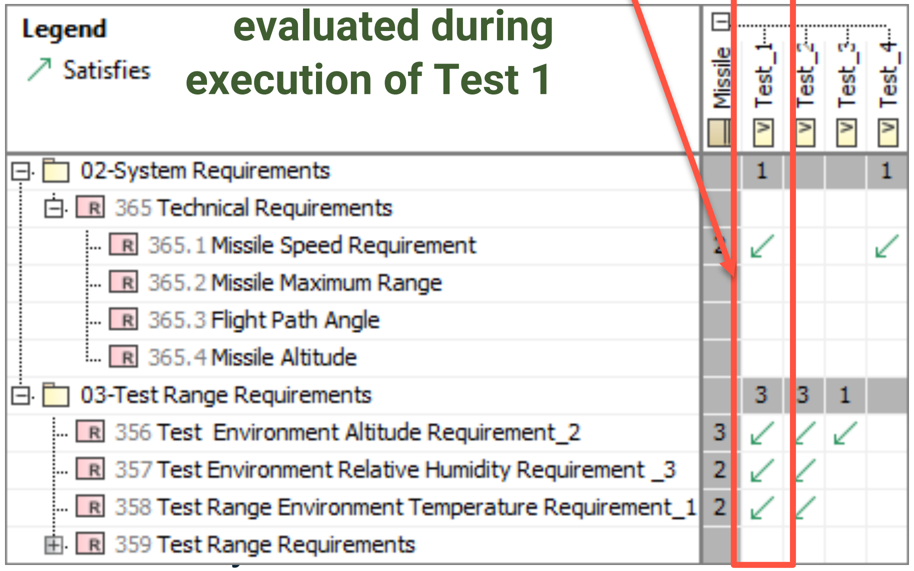


Execute the test configuration instance for each test run and evaluate test results.[1]

Time-stamped results of each test run are captured in the test configuration results table, and a pass or fail verdict highlights whether the requirement is satisfied.

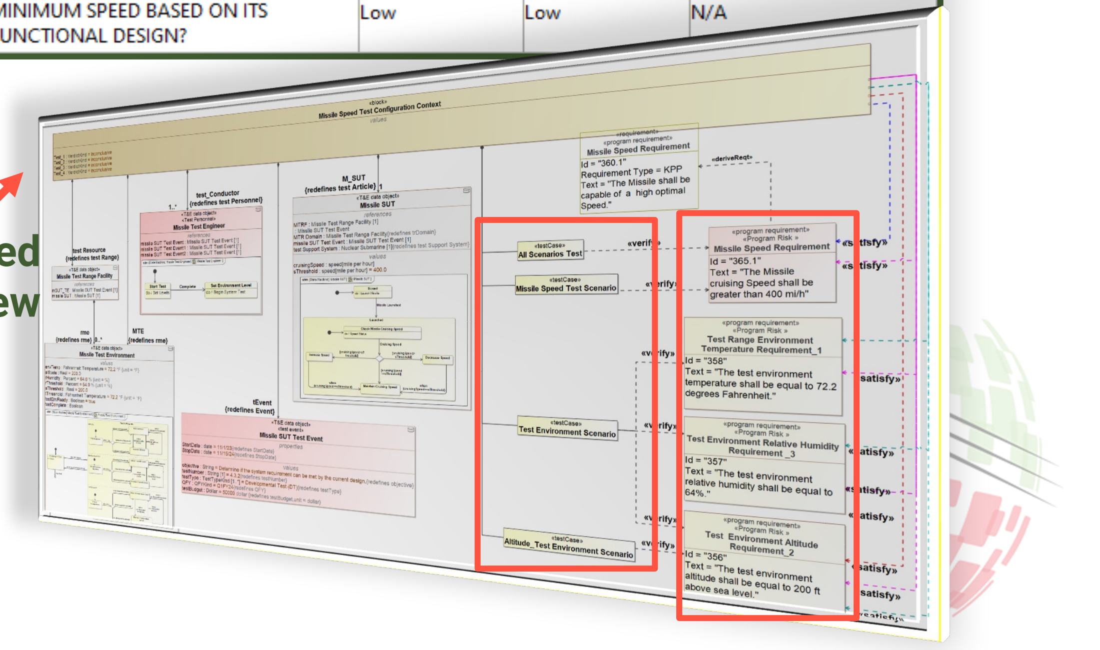
#	Name	Test_1 : VerdictKind	Test_2 : VerdictKind	Test_3 : VerdictKind	Test_4 : VerdictKind	ptr.decision : String	ATTRisk : RiskKind	CITRisk : RiskKind	DeficiencyResponse : DeficiencyResponse...
1	MIssile Speed Test Run	pass	fail	pass	fail	IS THE MISSILE ABLE TO MAINTAIN MINIMUM SPEED BASED ON ITS FUNCTIONAL DESIGN?	High	High	Mandatory M&S
2	Missile Speed Test Run 2	pass	pass	pass	fail	IS THE MISSILE ABLE TO MAINTAIN MINIMUM SPEED BASED ON ITS FUNCTIONAL DESIGN?	Low	Medium	Mandatory M&S
3	Missile Speed Test Run 3	pass	pass	pass	pass	IS THE MISSILE ABLE TO MAINTAIN MINIMUM SPEED BASED ON ITS FUNCTIONAL DESIGN?	Low	Low	N/A

Four requirements are evaluated during execution of Test 1



Requirements to Test Traceability View

Missile Speed Testing Context View

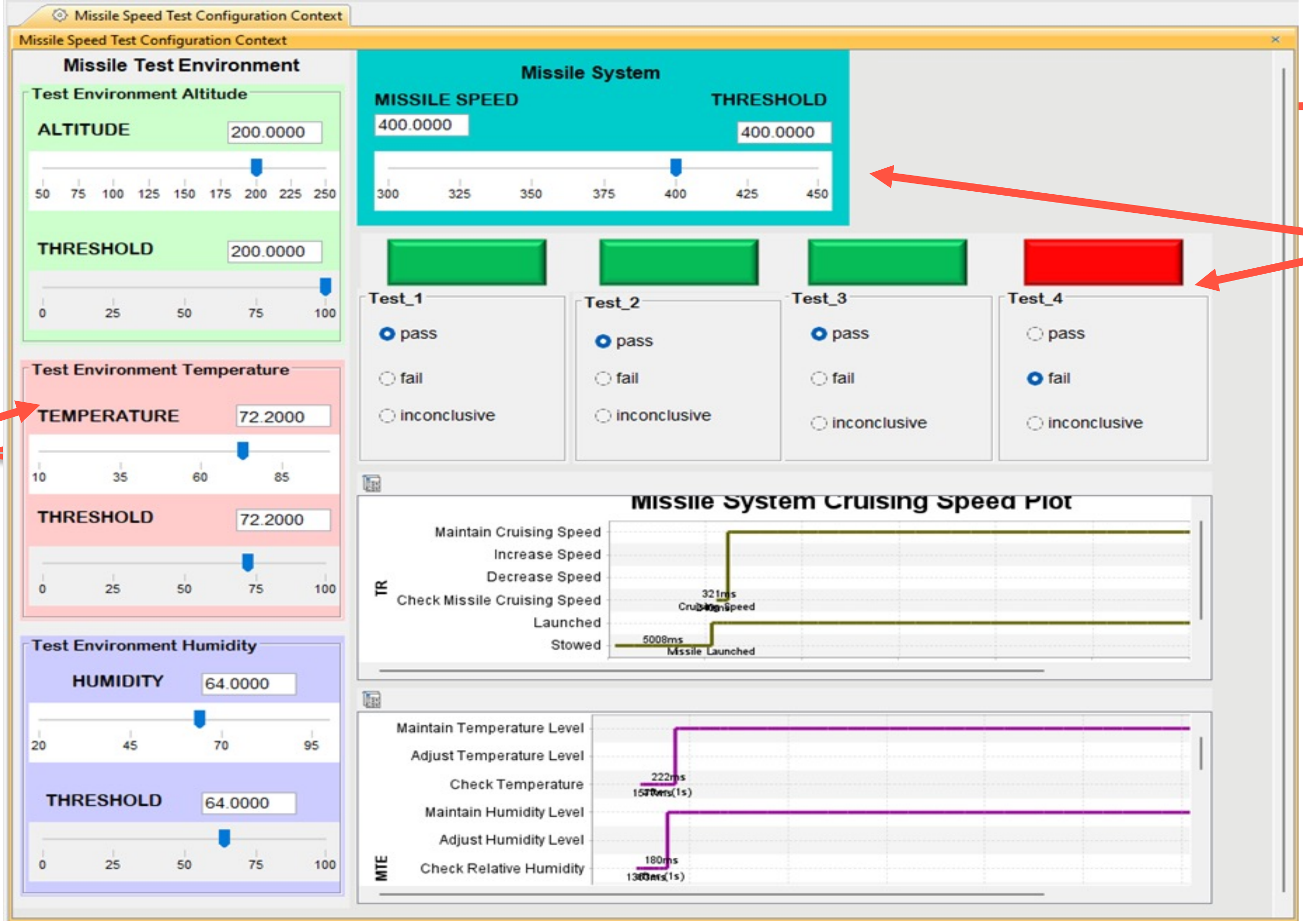


Execute the test configuration instance for each test run and evaluate test results.[2]

The Test Context Dashboard offers an efficient way to execute and visualize model-based tests.

«program requirement»
Test Range Environment Temperature Requirement_1

Id = "358"
Text = "The test environment temperature shall be equal to 72.2 degrees Fahrenheit ."



System
Re

«program requirement»
Missile Speed Requirement

Id = "365.1"
Text = "The Missile cruising Speed shall be greater than 400 mi/h"

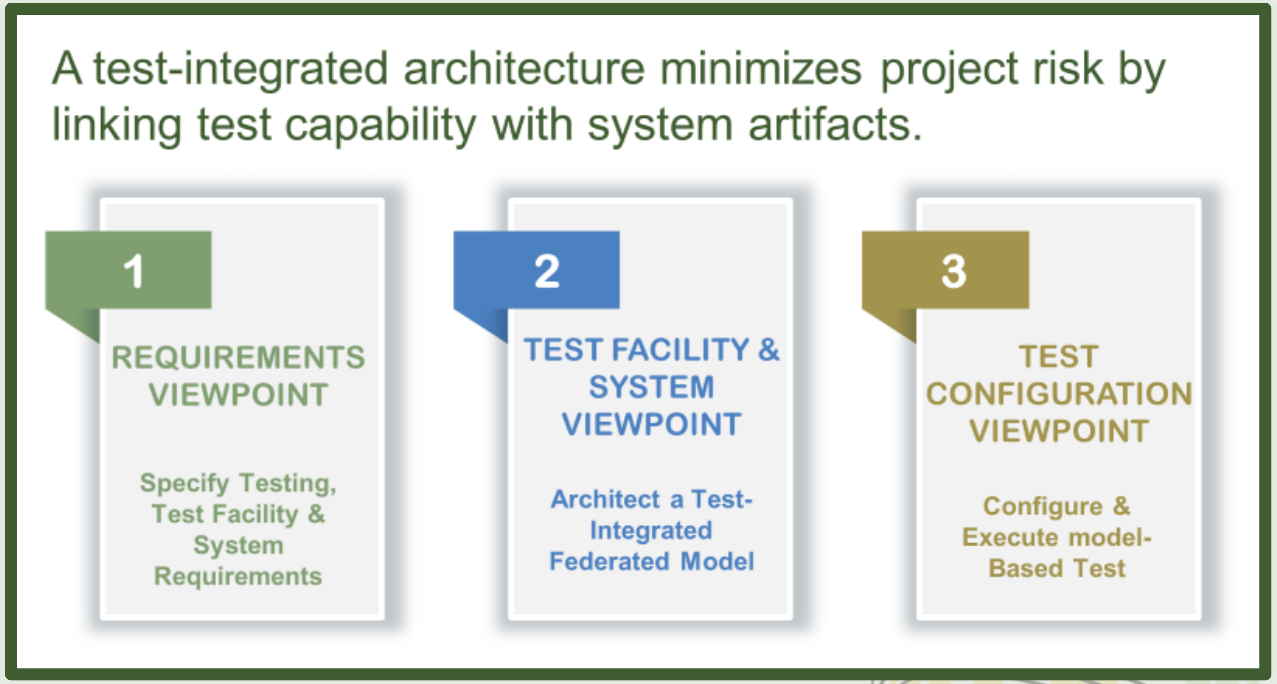
Conclusion

Shifting Left: A Test-Integrated System Architecture Approach

Expanding traditional MBSE to include T&E modeling enables a shift-left in system development cycle time.

Integrated Test planning & Capability Modeling:

- *Allows* organizations effectively plan and manage test programs.
- Accelerates product development and fielding through improved visibility and management of program performance.
- Directly links system requirements and architecture to test planning and testing capabilities.



Shifting Left: A Test-Integrated System Architecture Approach

Questions

References

1. DOT&E. (2022). Office of the Director, Operational Test and Evaluation Strategy Update - Strategic Pillars, viewed 30 March, 2023, FINAL DOTE 2022 Strategy Update 20220613.pdf (osd.mil)

2. Arndt, C. B., Anyanhun, A., & Werner, J. S. (2023). *Shifting Left: Opportunities to Reduce Defense Acquisition Cycle Time by Fully Integrating Test and Evaluation in Model Based Systems Engineering.* Acquisition Research Program.

3. Werner, J. S., & Arndt, C. (2023). Development of Digital Engineering Artifacts in Support of MBSE-based Test Planning, Execution, and Acquisition Decision Making. *Acquisition Research Program*.

4. US Department of Defense (2018). Digital Engineering Strategy, Office of the Deputy Assistant Secretary of Defense for Systems Engineering, Washington, DC (US).

Abstract

Problem Discovery during operational test and evaluation (T&E) has a significant impact on the acquisition cycle time of systems. A loss in competitiveness in the marketplace due to delays in fielding systems or products can prove detrimental to the survivability of organizations, and is a critical risk that must be actively monitored and mitigated. To achieve a reduction in the acquisition cycle time, i.e., a Shift Left, for a given system, traditional methods of testing need to be modified to accommodate more systemic and disciplined approaches that consider T&E more as a continuum, beginning at the conceptual phase of system design and development. Consequently, a systems engineering approach that introduces test capability modeling as an integral part of the model-based systems modeling methodology would result in a fully test-integrated MBSE approach to descriptive system modeling that emphasizes alongside the system-of-interest's (SOI) architecture, a test capability architecture, i.e., the necessary test resource artifacts required for verification and validation.

This proposed approach differs from most Model-based T&E approaches reviewed in literature due to the specification of testing capabilities as an inherent part of a model-based test context and configuration. Most traditional model-based approaches usually capture a tester model element within the test context as the model element initiating a given test case. However, our approach specifies within a test capability model, the test facility and test resources required to perform testing of the system and/or specific capabilities as part of the test context. This approach accomplishes several T&E goals: firstly, it alerts the system architects and decision makers to the suitability of a specific test facility's ability to test to the system requirements and/or operational capabilities due to the availability of either adequate or inadequate test resources.

Secondly, the approach utilizes a model-based test environment which could significantly reduce the overall T&E costs for the actual system by enabling the exploration of multiple test case specifications using testing techniques such as simulation via model execution, and analysis in order to identify the right tests and methods that could be deployed in testing the actual system. Performing verification and validation of a system using system models is a cost-effective way to show theoretical compliance of the system architecture prior development. Accordingly, in this work we demonstrate how a model-based test integrated configuration can improve a system's acquisition cycle time with a specific use case. A notional Missile System constitutes the SOI/SUT for our exemplar and is modeled using SysML. The notional test facility, i.e., the test capability model is defined as a model-based representation of required test resources — including representations of the system's operational environment — required to enable testing of a given set of system capabilities.

Author Bios

Dr. Awele Anyanhun currently serves as a senior research engineer on the faculty of the George Tech Research Institute (GTRI) in the Enterprise & Open Architecture branch within the System Engineering Research division. Dr. Anyanhun is an INCOSE Certified Systems Engineering Professional (CSEP), OMG certified Systems Modeling Professional (SysML-MBA), IREB certified Requirements Modeling Professional (CPRE) and a Senior Member of IEEE. She has extensive professional research and engineering experience in the space, automotive, and defense industry. Dr. Anyanhun is widely published in the areas of Model-based Systems Engineering, and Systems Architecture and serves as a reviewer for multiple international peer-reviewed journals and conferences. Dr. Anyanhun holds a PhD in electrical engineering with a concentration in model-based systems engineering.

Dr. Craig Arndt has extensive experience as a senior executive and technology leader in research, education, engineering and defense, homeland security and intelligence technologies, with extensive experience as an innovative leader in industry, academia, and government. Dr. Arndt currently serves as a principal research engineer on the faculty of the George Tech Research Institute (GTRI) in the System Engineering Research division of the Electronic Systems Lab. Dr. Arndt is a licensed Professional Engineer (PE), a Certified Human Factors Professional (CHFP), a Expert Systems Engineering Professional (ESEP), and he has over 40 years of professional engineering experience though the defense and government engineering community. He is widely published in the areas of Electrical, Systems and Human Factors engineering, and serves on the boards of several technical organizations. Dr. Arndt holds engineering degrees in electrical engineering, systems engineering and human factors engineering and a Masters of Arts in strategic studies from the US Naval War college.

34th Annual INCOSE international symposium

hybrid event

Dublin, Ireland
July 2 - 6, 2024

www.incose.org/symp2024
#INCOSEIS