Building a Requirements Digital Thread from Concept to Testing using Model-Based Structured Requirements Applied to Thrust Reverser Actuation System Development

HERBER Daniel R.¹ and EFTEKHARI-SHAHROUDI Kamran¹²

¹Colorado State University — Department of Systems Engineering
²Woodward, Inc.

{daniel.herber, kamran.eftekhari_shahroudi}@colostate.edu

Recent Advances in Aerospace Actuation Systems and Components
September 27, 2023
Toulouse, France
Introduction
Motivation for Creating the Aero-Actuation Systems Engineering Test (ASET) Lab for Studying Electric TRAS Systems

- **Reduced carbon and emissions strategy**
  - Turbine Bypass Ratios are continuing to increase

- **Airbus, Boeing Gulfstream et. AI MEA Strategy**

- **Reduced weight and cost**

- **Increasing Complexity of Aircraft Systems**
  - More sensors, CPU’s, Actuators, networks etc.

- **Classical BiTs are becoming less effective/reliable**

**Circle Diagram:**
- Lighter more flexible transcowl
- Reduced hydraulics
- Geared elec. Act. sensitive to jamming
- Increased # & freq. complex anomalies

**Need for “real-world” Complex Anomaly Detection and Tolerant Control**
Aero-actuation Systems Engineering Test (ASET) Lab for Anomaly Detection and Anomaly Tolerant Control
### Faults and Anomalies Simulated

<table>
<thead>
<tr>
<th>Fault Modes</th>
<th>Electronic power unit</th>
<th>Reducer</th>
<th>Ball screw</th>
<th>Rail</th>
<th>Transcowl</th>
</tr>
</thead>
<tbody>
<tr>
<td>Current Leakage</td>
<td>S</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Short circuit</td>
<td>S</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Unbalanced Shaft</td>
<td>S S S</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Damaged Bearings</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Misalignment</td>
<td>S S S</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Winding open circuit</td>
<td>S</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Winding default connexion</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Eccentricity</td>
<td>S</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Insulation melting</td>
<td>S</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Tooth breaking</td>
<td>S</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Backlash</td>
<td>S S</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Dilatation</td>
<td>S S S</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Damaged balls</td>
<td>S</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Failure</td>
<td>S S S S</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Jamming</td>
<td>S S S S</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>External fault</td>
<td>S S S S</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Changing temperature</td>
<td>S S S</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>High temperature</td>
<td>E</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Low temperature</td>
<td>E</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

| Test coverage        | 93%                    |
| Simulation percentage | 95%                    |

**Legend**
- S: The fault mode will be simulated by the testbed
- E: The fault mode will be emulated by the testbed
- S/E: The fault mode can either be simulated or emulated by the testbed

1. Imbalance
2. Misalignment
3. Jamming
4. Flexibility
5. Backlash
6. Sideload
Model-Based Systems Thinking & Engineering is Adapting to Handle the Increasing Complexity

- Model-Based Performance, Reliability & Safety (Amesim, Simulink, etc.)
- Model-Based Architectural and V&V Development (MBSE Cameo Systems Modeler, Systems Composer, etc.)
- Model-Based (automated) Structured Requirements *(this paper)*
  - **Goal:** apply model-based systems thinking/architecting/automation to Needs/Requirements Development and Management
Motivation (1)

- A requirement is a statement that “translates or expresses a need and its associated constraints and conditions”\(^1\) and is essential throughout a product’s lifecycle
- Guidelines have been established\(^2\), providing general characteristics of well-defined requirements
  - Necessary, appropriate, unambiguous, complete, singular, feasible, verifiable, correct, and conforming
- However, creating requirements that adhere to these characteristics takes time and effort

\(^1\) IEEE 2018  \(^2\) Pohl and Rupp 2015; IEEE 2018; Wheatcraft et al. 2022
Motivation (2)

- This heavy cost and time delay burden causes unproductive tension between program management and systems engineering functions on both sides, causing late discovery of requirements, compliance, and critical product issues.
- This risk to development is due to ever-changing nature of system understanding.
- Additionally, the sheer volume of interrelated shall statements across different groups.
- Modern approaches to requirements development and management should help an engineer.
  - Produce systematically well-defined requirements with lower effort.
  - Create practical digital threads to R&D and testing results.
  - Reuse relevant aspects of previous efforts and organization best practices.
Model-based Approaches (1): Document vs. Model-based Approaches

(a) Traditional document-centric SE approach
(b) Model-based SE approach
Central to this work is the usage of modern model-based perspectives of developing and understanding systems, often falling under the general term **model-based systems engineering (MBSE)**\(^1\).

The model here is an abstract representation of a system and its development captured through **structure, behavior, and rules and their relationships**.

Such a comprehensive model is often associated with the notion of **system architecture**.

The “model-centric” perspective is perhaps best characterized by the principle: *For every concept, there is a singular, unique element in the model that is used as much as necessary.*

This perspective contrasts with more “document-centric” practices with static (potentially incorrect, incomplete, or convoluted) artifacts.

---

\(^1\) Mounting formal and informal evidence suggests that MBSE has the potential to improve efficiency, rigor, and quality during system development (Huldt and Stenius 2018; Carroll and Malins 2016; Madni and Purohit 2019)
The benefits of model-based approaches are properly realized when they effectively provide support for various activities during a product lifecycle, including testing.

Now, there is codified and disciplined support for MBSE in the form of methodologies, formal modeling languages, and tools.

Additionally, MBSE concepts and techniques are extensible to meet the needs of particular communities and organizations.

---

1 Borky and Bradley 2019; Friedenthal, Moore, and Steiner 2015
The Systems Modeling Language (SysML) is a general-purpose modeling language for creating system architecture models. The SysML standard prescribes a set of constructs and allowable relationships, thereby supporting the creation of an integrated system model. Software tools are the key enablers of this form of system modeling and MBSE, providing means to interact with the model through diagrams and other forms. One popular tool (and the one used in this work) is Cameo Systems Modeler (CSM), which implements SysML along with other features to support MBSE activities.

1 OMG 2019  2 No Magic 2020
Systems Modeling Language or SysML (2): Four Pillars

- SysML is often described at a high level through the **four pillars of structure, behavior, requirements, and parametrics**

---

1 Friedenthal, Moore, and Steiner 2015
Methodology
Structured Requirement (1): Definition

- A *structured requirement (SR)*\(^1\) or template requirement\(^2\) defines an *orderly structure with specified attribute placeholders* (i.e., a sentence blueprint)
- Helps capture the precise meaning and communicate the required information to define a complete requirement
- The particular SR statement template based on Carson 2015 considered here is:

\[
\text{[Who] shall [What] [How Well] under [Condition].} \quad (1)
\]

\(^{1}\) Carson 2015 \(^{2}\) Pohl and Rupp 2015
Structured Requirement (2): Who

[Who] shall [What] [How Well] under [Condition]. (1)

- [Who]: The element (product, service, user, activity, etc.) which is subject to the remainder of the statement
- If the requirement is satisfied, this element would be said to have the desired characteristic or is capable of performing the intended function

As an example, the following is a SR written in natural language:

[The testbed] shall [transition to safe mode] [within 2 sec] under [enclosure door open event]. (R1)
Structured Requirement (3): What

[Who] shall [What] [How Well] under [Condition]. \hspace{1cm} (1)

- [What]: Refers to an element, either as an observable characteristic or function, that will now have a measurable legal obligation in relation to [Who]

Example again:

[The testbed] shall [transition to safe mode] [within 2 sec] under [enclosure door open event]. \hspace{1cm} (R1)
Structured Requirement (4): How Well

\[ \text{Who} \text{ shall } \text{What} \text{ [How Well] under [Condition].} \quad (1) \]

- **[How Well]**: A measurable condition to assess how well the legal obligation for the [What] attribute is met

- Example again:

\[ \text{The testbed} \text{ shall [transition to safe mode] [within 2 sec] under [enclosure door open event].} \quad \text{(R1)} \]
Structured Requirement (5): Condition

[Who] shall [What] [How Well] under [Condition]. (1)

- **[Condition]**: Describes the stipulated complete set of operational scenarios, states, and environmental conditions when the legal obligation must be met.
- This set can include triggering or initiating events\(^1\)

---

- Example again:

  [The testbed] shall [transition to safe mode] [within 2 sec] under [enclosure door open event]. (R1)

---

\(^1\) Carson 2015
Structured Requirement (6): Comments

• Equation (1) is not the only SR form
  • Different general templates, such as ones in Pohl and Rupp 2015 or based on the 29148-2018 standard\(^1\)
  • Templates with structure and attributes tailored to different requirement types and organizational needs\(^2\) (e.g., functional, non-functional, interface, design constraint, and operational)

• Overall, this approach to requirements creation helps improve their quality by helping meet the general characteristics of well-defined requirements from Slide 6

• However, as currently presented, these techniques offer only a refinement to the classical textual requirements
  • Have currently no relationship to a system (SysML) model and only an imprecise relationship to the broader system context

\(^1\) IEEE 2018 \(^2\) Carson 2015; Pohl 2010
Model-based Structured Requirement or MBSR (1): Definition

Model-based structured requirements (MBSRs) provide a mechanism for the usage of SysML modeling elements to define a structured requirement. A new «Structured Requirement» stereotype is defined that inherits all the properties of the original SysML «Requirement». Then four attributes from Eq. (1) are added to «Structured Requirement».

1 Herber, Narsinghani, and Eftekhari-Shahroudi 2022
Model-based Structured Requirement or MBSR (2): Total Context

- Selected **model elements will have much more than their name** (as is the case in textual SRs) defining what it is and where it fits within the system.

- These attributes for a given MBSR can be linked to other model elements that represent aspects of the physical system, precise mathematical conditions, test cases, etc.

- This capture of meaning can also include **documentation and links to external artifacts, input data, and testing results**.
Model-based Structured Requirement or MBSR (3): Changes

<table>
<thead>
<tr>
<th>«Structured Requirement»</th>
<th>Door Open Shutdown</th>
</tr>
</thead>
</table>

«AbstractRequirement»

Text = "The testbed shall transition to safe mode within 2 sec when the enclosure door is opened."

1. **Who** = Testbed
2. **What** = Transition to Safe Mode
3. **How Well** = Shutdown 2 s
4. **Condition** = Enclosure Door Open

- Changes to SysML MBSR «Structured Requirement» stereotype from (Herber, Narsinghani, and Eftekhari-Shahroudi 2022) are described in the paper
Results
ASET Lab Testbed Case Study

- What is shown here is only a simplified version of a more complete SysML model in CSM used to support the development and running of the testbed.
- Several illustrative requirements were created as MBSRs.
Door Open Shutdown MBSR Example

Introduction

Methodology

Results

Conclusions

References

Door Open Shutdown MBSR Example

[Structured Requirement]

Text = "The testbed shall transition to safe mode within 2 sec when the enclosure door is opened."

Who = Testbed

What = Transition to Safe Mode

How Well = Door Open Shutdown 2 s

Condition = Enclosure Door Open

«AbstractRequirement»

Door Open Shutdown

«Structured Requirement»

Text = "The testbed shall remain safe under test when an unusual situation is detected."

«requirement»

time_limit = 2.0 s

Shutdown 2 s : Time Limit

time_limit = 2.0 s

Activity details actions like "disconnecting power supply to actuation system". This is modeled in the linked diagram.

«testCase»

Enclosure Door Open

Safety Tests

The relations of these elements to the requirement are modeled within the requirement block on the left.

[Artifact]

2023-07-01 Test 25 Safety Test Report

Documentation = "Contains screenshots of measurement software with zero power to all motors following various safety tests. All tests passed."

File = Test 25 - Safety test.docx

Hyperlinked test report

To guard against the hazard of entanglement in high speed rotating parts and the hazard of parts flying out of enclosure due to operator error or a part failure.

The parent requirement (not updated as MBSR). Traceability between requirements.

More specific for enclosure door open event.

To guard against the hazard of entanglement in high speed rotating parts and the hazard of parts flying out of enclosure due to operator error or a part failure.

Traceability between requirements.MBSR requirement with original text.

Bottom left icon linked to an activity diagram with steps of the test.
## Requirements Table

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Induction Motors</td>
<td>All four (4) testbed lines shall include one induction motor.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>Starting Capacitors</td>
<td>The output of the power supply shall connect to six 24 mF and two 16 mF capacitors rated for 350 V in parallel to provide the requisite power to start up the induction machines.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>Gearboxes</td>
<td>The gearbox on each actuation line shall be rated to 15 Nm at 2000 rpm.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Motor Sustained High Load Thermal Test</td>
<td>With all four lines running simultaneously, all BLDC motors shall remain below 75 degC when operated at rated speed of 1800 RPM at high torque of 4 Nm.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>Motor Open Circuit</td>
<td>The rms value of motor coil current shall be greater than 250 mA when the rms value of coil current demand is greater than 1 A to simulate an open circuit in the motor winding.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>Brush Inductance</td>
<td>The microcontroller shall ramp up the voltage from the power supply to the testbed via a slew rate of 30 V/s to charge the starting capacitors.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>Microcontroller Digital Inputs</td>
<td>The microcontroller shall have digital (61) input ports.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>Microcontroller Outputs</td>
<td>The microcontroller shall broadcast system status, error conditions, and RMS system parameters over UART every 1 second using 10 digital PWM outputs.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>Power Converter Max Weight</td>
<td>The power converter assembly shall not exceed the weight limit of 1300 lbs.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>Power Converter Output</td>
<td>The power converter shall deliver 10KW power in total to the four (4) actuation lines using an AC supply.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>Door Open Shutdown</td>
<td>The testbed shall transition to safe mode within 2 sec when the enclosure door is opened.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>Safety Signal Shutdown</td>
<td>The testbed shall transition to safe mode within 2 sec when the safety signal is transmitted.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
### Summarizing MBSRs in the Model with a Matrix

<table>
<thead>
<tr>
<th>Legend</th>
<th>Requirements</th>
</tr>
</thead>
<tbody>
<tr>
<td>Condition</td>
<td>Testbed</td>
</tr>
<tr>
<td>How Well</td>
<td>Behavior</td>
</tr>
<tr>
<td>Who</td>
<td>Enclosure Door Open</td>
</tr>
<tr>
<td>What</td>
<td>Four Lines Running</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Testbed</th>
<th>Behavior</th>
</tr>
</thead>
<tbody>
<tr>
<td>Door Open Shutdown</td>
<td>1</td>
</tr>
<tr>
<td>Safety Signal Shutdown</td>
<td>1</td>
</tr>
<tr>
<td>Gearboxes</td>
<td>1</td>
</tr>
<tr>
<td>Motor Open Circuit</td>
<td>1</td>
</tr>
<tr>
<td>Motor Sustained High Load</td>
<td>1</td>
</tr>
<tr>
<td>Power Converter Max Weight</td>
<td>1</td>
</tr>
<tr>
<td>Microcontroller Digital Inputs</td>
<td>1</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Constraints</th>
<th>Testbed</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 A : Current Demand</td>
<td>1</td>
</tr>
<tr>
<td>1 Nm High Torque : Load Motor</td>
<td>1</td>
</tr>
<tr>
<td>61 Digital : Port Limit</td>
<td>1</td>
</tr>
<tr>
<td>75 C : Motor Temp Limit</td>
<td>1</td>
</tr>
<tr>
<td>1800 rpm Rated Speed : Load Motor</td>
<td>1</td>
</tr>
<tr>
<td>Gearbox 15 Nm : Desired Torque</td>
<td>1</td>
</tr>
<tr>
<td>Measurement 1 s : Time Limit</td>
<td>1</td>
</tr>
<tr>
<td>Range +/- 250 mA : Zero Current Deviation</td>
<td>1</td>
</tr>
<tr>
<td>Shutdown 2 s : Time Limit</td>
<td>2</td>
</tr>
<tr>
<td>Speed 2000 rpm : Gearbox</td>
<td>1</td>
</tr>
<tr>
<td>Weight 1300 lbs : Weight Limit</td>
<td>1</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Structure</th>
<th>Testbed</th>
</tr>
</thead>
<tbody>
<tr>
<td>Gearbox</td>
<td>1</td>
</tr>
<tr>
<td>Load Motor</td>
<td>2</td>
</tr>
<tr>
<td>Open Circuit Current Response</td>
<td>2</td>
</tr>
<tr>
<td>Temperature : celsius</td>
<td>1</td>
</tr>
<tr>
<td>Microcontroller</td>
<td>1</td>
</tr>
<tr>
<td>Digital Port [0..*]</td>
<td>1</td>
</tr>
<tr>
<td>Power Converter</td>
<td>1</td>
</tr>
<tr>
<td>Weight : mass[pound]</td>
<td>2</td>
</tr>
<tr>
<td>Testbed</td>
<td>2</td>
</tr>
</tbody>
</table>

| Transition to Safe Mode | 2 |
→ Discussion Points (1)

1. **Complete yet Focused Requirement Statements**
   - It is still **too common to try to include as much as possible in the text statement**, with the justification being that this information is required to understand the requirement
   - MBSRs help focus the requirement definition on the attributes while supporting the full contextual description

2. **Reusability**
   - Often reuse or derive requirements, test plans, conditions, etc., from our previous efforts
   - Reuse of and traceability to elements (e.g., blocks, test activities, etc.) can bring many benefits

3. **Consistent Visualizations**
   - As a graphical language, SysML provides an opportunity to standardize and reduce work in creating visualizations
   - Changes to an element in one diagram are automatically reflected in all other diagrams
4. **Design for Testability (DfT)**

   - In DfT approaches, testability issues, such as **gaps between design decisions and practical testing of those choices**, are identified much early\(^1\)
   - Testing and V&V concerns and decisions brought earlier into the development through a system model capturing “all” aspects of system development

5. **Automated Test Plans and Other Documents**

   - Many MBSE tools support the generation of artifacts from the model (e.g., diagrams, tables, matrices, and maps but also full-featured documents and presentations)
   - Can meet organizational guidelines for documents like test plans

6. **Digital Threads and Completeness**

   - Creating documentation and keeping track of changes and decisions throughout system development can be challenging
   - Enables change propagation analysis and automated quality checks

\(^1\) Grout 2006
Conclusions
Conclusions

- This paper discussed model-based structured requirements (MBSRs) in SysML as a modern, model-based approach for developing high-quality requirements.
- Illustrated on the ASET Lab testbed, MBSRs can help release complete yet focused requirement statements with reusability, design for testability, digital threads, and more.
- MBSRs have the structure to incrementally automate and reduce tensions today while making our systems engineering machinery ready to better take advantage of digital transformation and AI as they become available.
- Base MBSR profile/model is available at https://github.com/danielrherber/model-based-structured-requirements
References

References (Continued)

- No Magic (2020). *Cameo Systems Modeler 19.0 SP4*. 19.0 SP4. URL: https://docs.nomagic.com/display/NMDOC/19.0+LTR+SP4+Version+News
Questions?

Building a Requirements Digital Thread from Concept to Testing using Model-Based Structured Requirements Applied to Thrust Reverser Actuation System Development

HERBER Daniel R.
Colorado State University — Department of Systems Engineering
daniel.herber@colostate.edu

EFTEKHARI-SHAHROUDI Kamran
Woodward, Inc.
Colorado State University — Department of Systems Engineering
kamran.eftekhari.shahroudi@colostate.edu