On a Recursive Methodology for the Analysis of System Requirement Tolerances

Bogdan S. Racolta, Integrated Product Team Lead

July 3rd, 2024

Part I

Introduction & Overview

Author Role

- Systems Engineer at Raytheon
- Supported requirement traceability, statistical analyses, and failure investigations
  - Supported AI/ML methods through development of verification techniques

- Statistical Tools
- Systems Theory
- Program Implementation

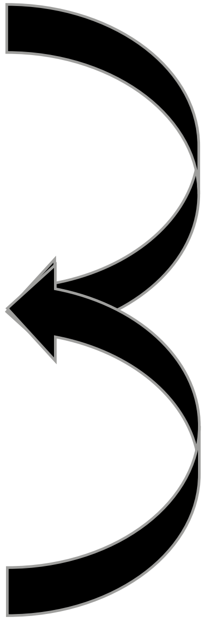
Author Responsibility

- AI/ML methods

Software Team

- Subject-Matter-Expert (SME) interactions

Program Personnel



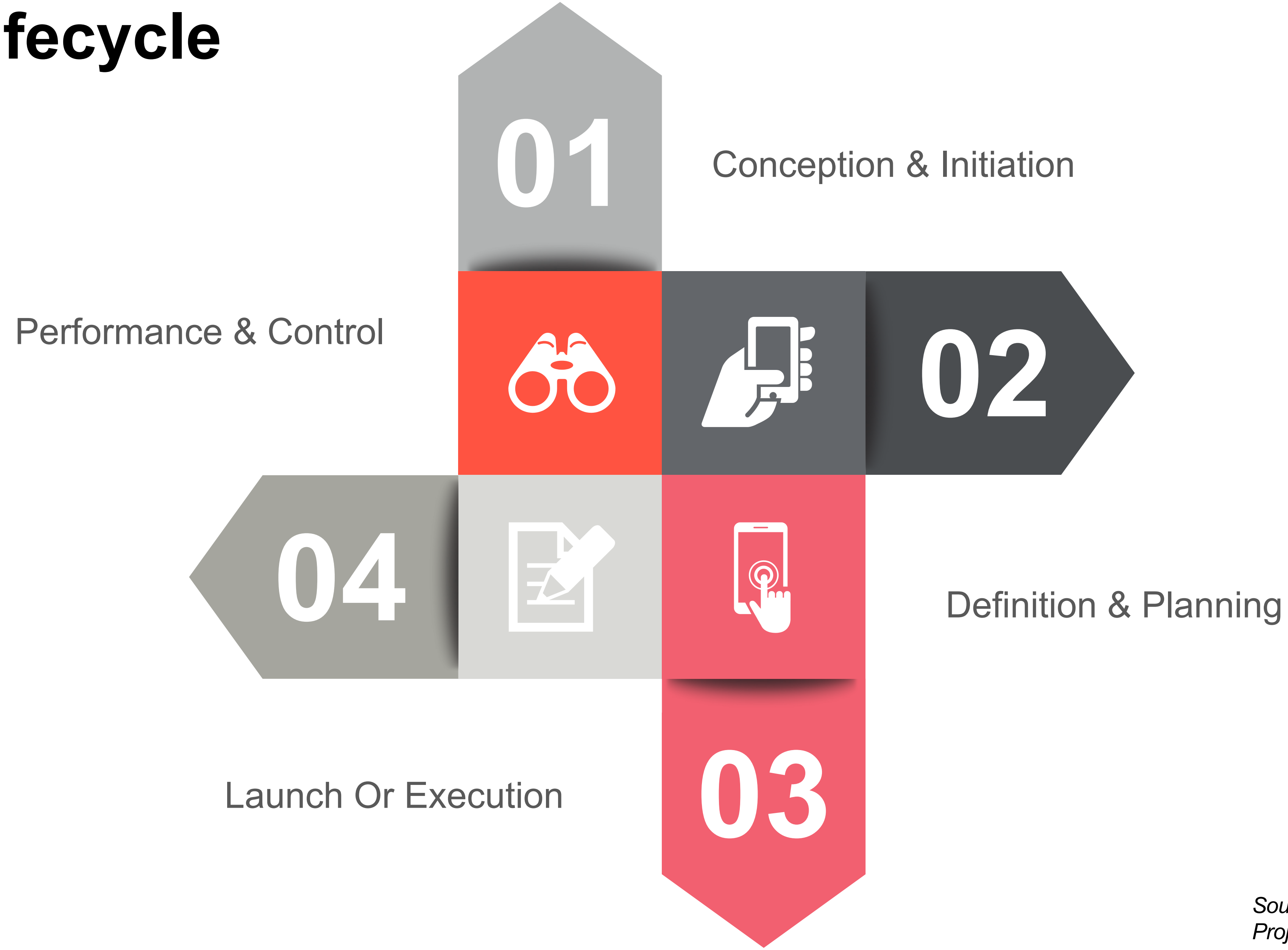
Integration

Methodology Objectives

Methodology Usage

No.	Objective	Description
01	Reduce Rework	Predicting test failure occurrence at a lower level of assembly than the SOI
02	Test Traceability	Accurately trace test requirements across test suite
03	Test Power	Identify inconsistencies, redundancies, and gaps in requirements specifications
04	Test Coverage	Prevent obscuring of valid system performance (minimize Type I error propagation)
05	Test Documentation Updates	Generate new test requirement revisions with updated bounds
06	Pattern Recognition	Leverage AI for pattern recognition and statistical methods for results verification

Project Lifecycle



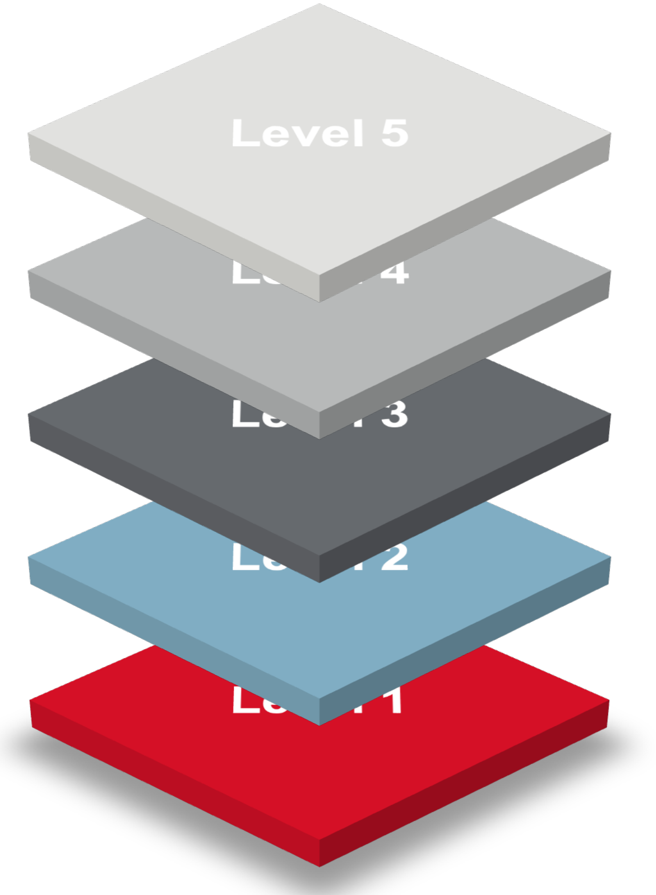
Source: PMI Institute
Project close is omitted

Rework Implications

System Development

System Complexity

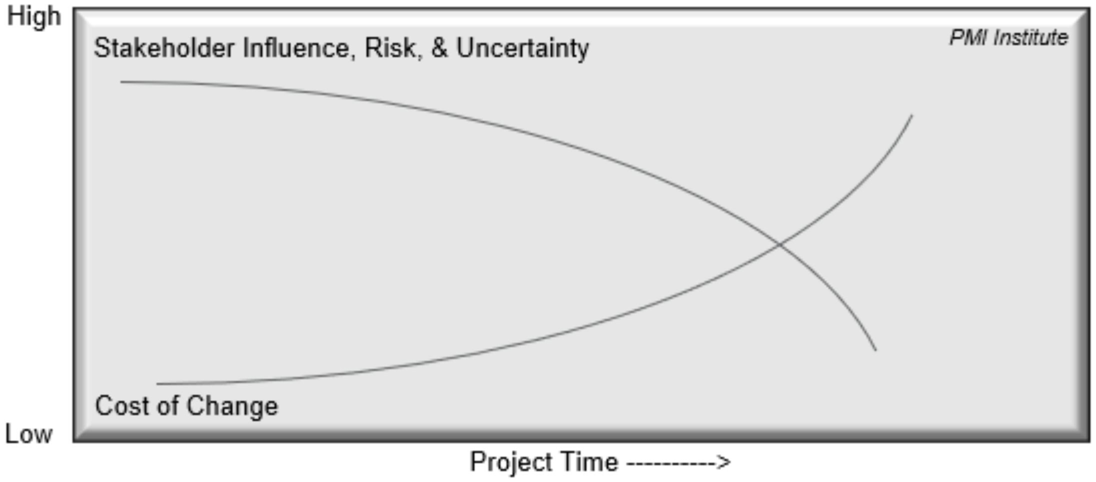
Project Timeline



CCA Level Build Structures

Subsystems

Final System Build



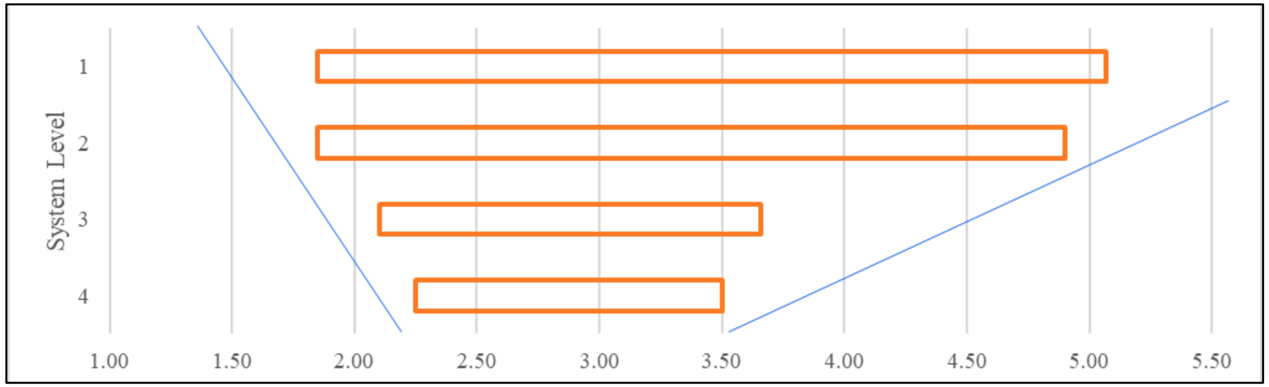
Part II

Methodology

Test ID	Function	Port Start	Port End	Lower Limit	Upper Limit	Units
Test #X	What does it do?	Port #Y	Port #Z	A	B	[units]

- Lower-level requirement bounds can be truncated to mitigate potential future failures
- Increase test precision
- Lower-level UUT changes are more cost-effective
    - Absence of confounding & system-system coupling effects

Requirement Bounds Across System Levels



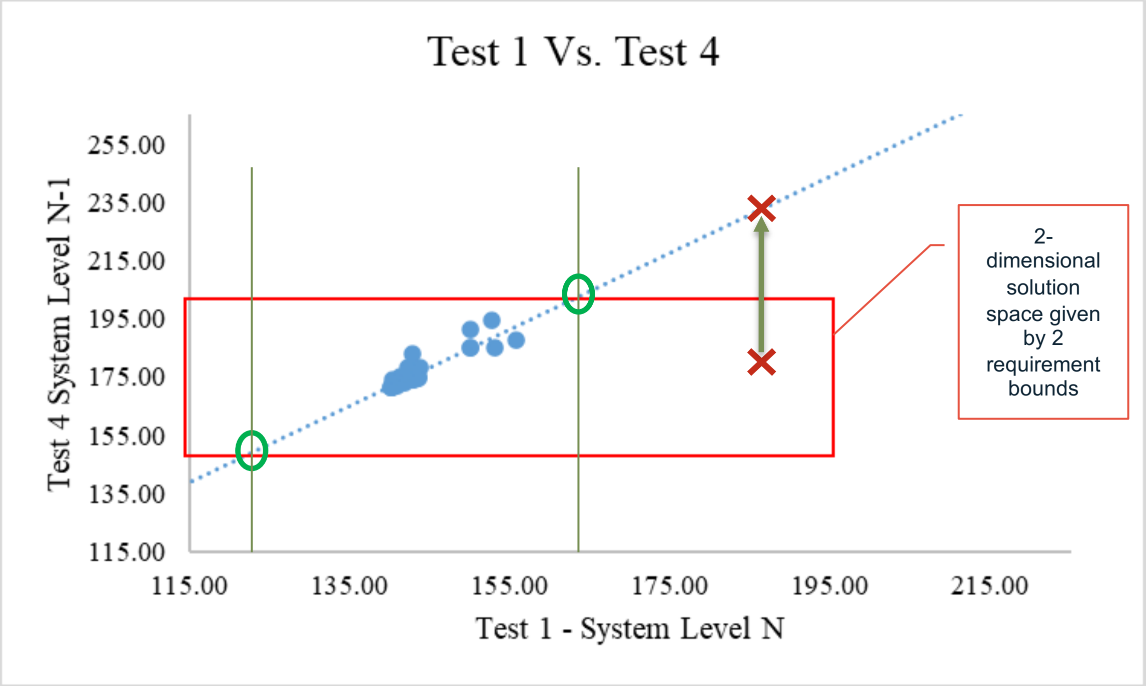
Units [Notional]

Funnel relationships amongst requirements with high correlation or derived traceability is desirable to capture ***true*** *system performance*

UUT – Unit Under Test

Requirement Bound Truncation I

- Given a function describing the two data sets and the requirement bounds with some degree of correlation:
  - Truncate bounds along the regression line to limit allowable variation
  - Maintain funnel relationship
- Implication:
  - Allowing 2-dimensional solution space can lead to future test failures (see ‘X’ data point)
  - Requirement bound construction can yield inadvertent failures
- Post-Truncation:
  - Requirement bound truncation can prevent future test error propagation



One-To-One Relationships

- Test relationships of the form: F(Test x1) = F(Test x2)

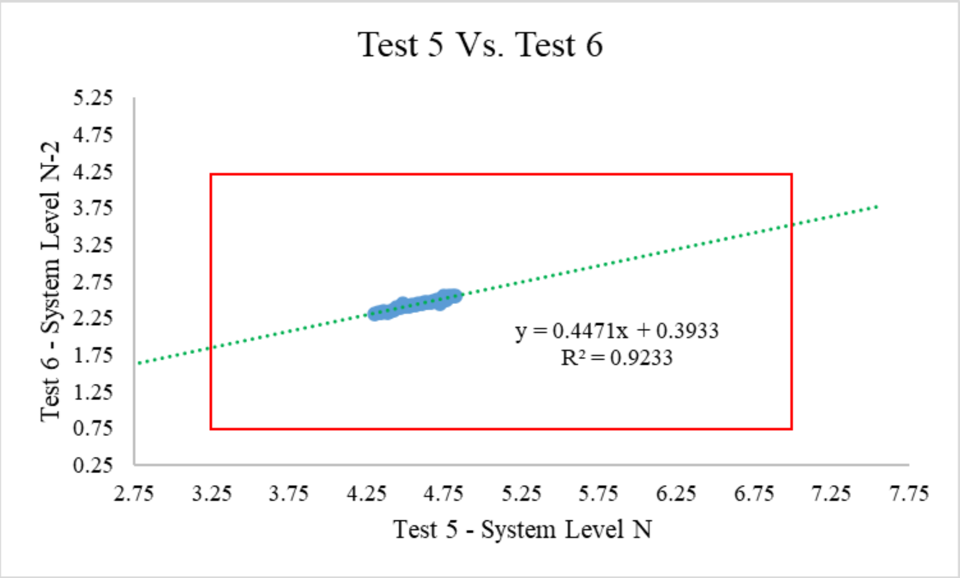
Where Test Suite = $[x_1, x_2, x_3, \ldots, x_n]$

- Test Criteria:
  - If tests are similar → Can the data (samples) from x_1 and x_2 be shown to be from the same population?
  - If so, there is strong evidence to allocate the same lower & upper limits
    - Maintains funnel structure
  - ANOVA, Welch Z-Test as diagnostic tools
- Indicates incorrect bounds on requirements across test requirements

One-to-One Relationships can maintain funnel relationship if they can be shown the come from the same population

Linear Relationships

- Test relationships of the form: $F(x_1) = m*x_2 + b$
  - Exists where SOS interactions/coupling effects influence test behavior
- Test Criteria:
  - Truncate bounds based on the intersection of the trendline, and requirement bounds (see slide 9)
  - Truncate bounds at some point and calculate the Type I Error Propagation for the proposed new bounds.



Linear Relationships may not always violate the funnel relationship, but allows for improving test coverage

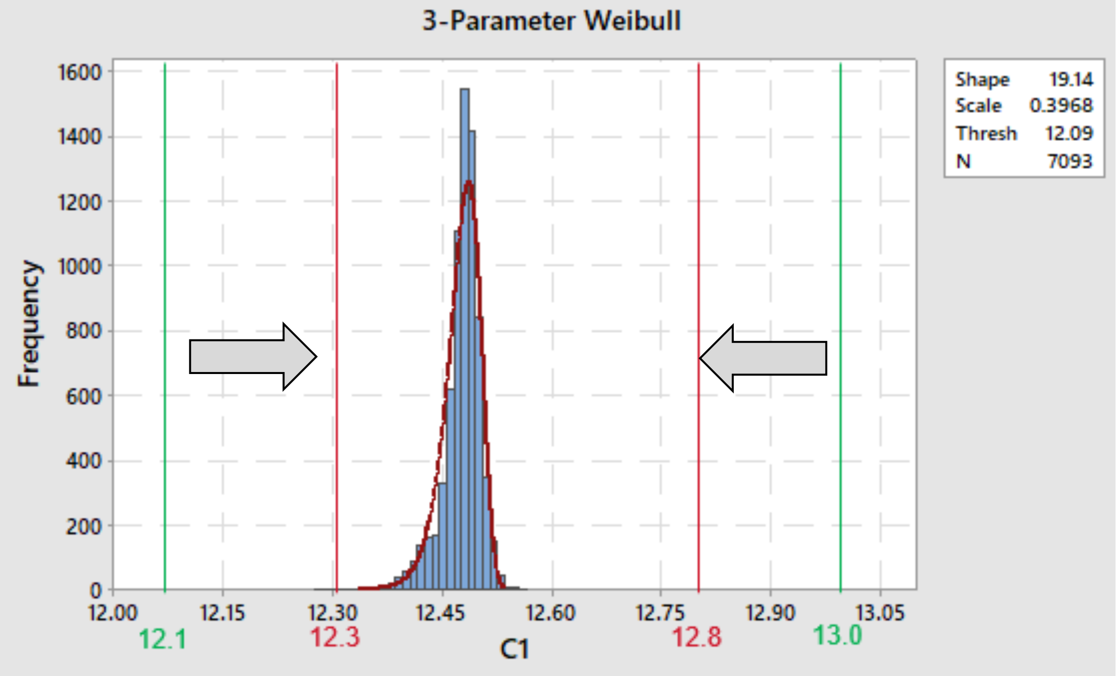
Type I Error Propagation

- Determine best distribution to fit to test data
  – KS, AD statistics used as criteria using statistical software
- [Test 7] has the following distribution based on AD statistic
- Green bounds → Requirement bounds
- Red bounds → Truncated bounds

$$Type\ I\ Error = \int_{12.1}^{12.3} P(x)dx + \int_{12.8}^{13.0} P(x)dx \leq \eta$$

Where P(x)~Weibull (19.14, 0.3968, 12.09)

Test 7 Distribution



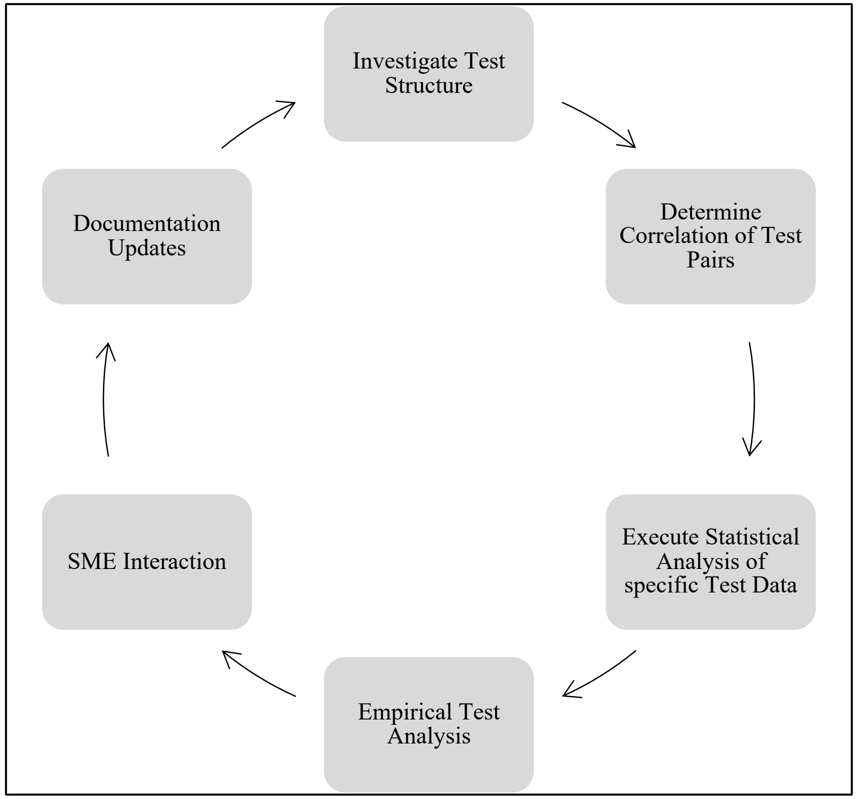
Given a change to requirement bounds, the resulting Type I Error probability must remain below a threshold value, η

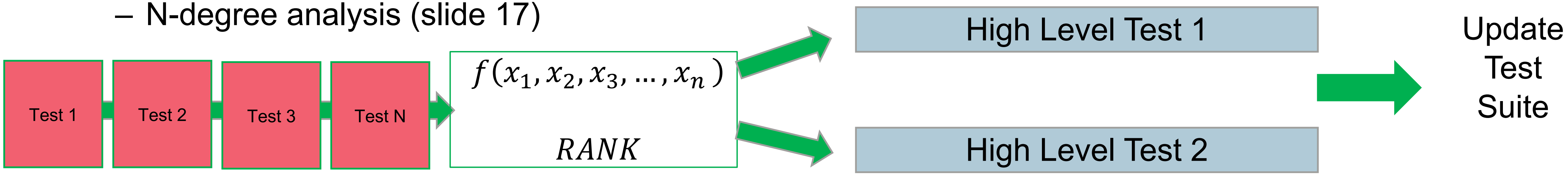
Part III

Results

Overall Findings

- 30+ unique tests across 7+ requirement sets identified as valid changes
  - Highly correlated pairs selected for analysis
  - Test execution investigated to understand test conduct
  - Type I error propagation minimized ($\eta \leq 0.1\%$)
  - SME interaction to validate & verify changes reinforced findings
- More complex relationships discovered
  - Nonlinear relationships
- Greater inter-test dependency discovered
  - N-degree analysis (slide 17)





Requirement changes were approved by program document change authorities and new bounds were implemented

Raytheon An RTX Business

Analysis Set

Part/Level	Test	Function	Part/Level	Corr.	LL (ms)	UL (ms)
CCA/3	Switch Close Time	Measure close time from specific Voltage	Battery Assembly/2	0.85	100	155
Aft Wing Assembly/1	Position Control	Measure deflection of wing following command	Test 1 (level 2 – wing assy.) Test 3 (level 4 – squib circuit 1) Test 7 (level 4 – tank release valve)	See slide 17	Multiple	Multiple

Let's investigate 2 examples

Analysis Set Example I

Bound truncation at higher system level from to lower-level tests

CCA Test

Part/ Level	Test	Function
CCA/ 3	Switch_Close_Time	Measure close time from specific voltage

Lower Limit: 80 ms
Upper Limit: 140 ms

R^2 = 0.921 →

Battery Assembly Test

Part/Level	Test	Function
Batt Assembly/2	Voltage Steady State	Record time battery reaches steady state (41 V for time duration)

Lower Limit: 70 ms
Upper Limit: 225 ms

- Highest correlated pairs from the full factorial design investigated first
- Lower limit on level 2 violates funnel relationship; investigate truncation

Analysis Set Example II

Test suite investigation for opaque predictive relationships – generalized linear model

Test Set with High correlation to Final Assembly Build

- Test 1 (level 2 – wing assy.)
- Test 3 (level 4 – squib circuit 1)
- Test 7 (level 4 – tank release valve)

Check-Out Test at Final Build Phase

- Test 44A – Aft Assembly wing switch position control

Units: mrad, tolerance of wing position

Multiple Liner Regression between Tests 1, 3, & 7 to Test 44A
Generate generalized linear model to determine failure modes on the independent var.

Given by general form:

$$Y_i = \beta_0 + \beta_1 X_{i1} + \beta_2 X_{i2} + \cdots + \beta_p X_{ip} + \epsilon_i$$

for each observation $i = 1, \ldots, n.$

The parameters $\beta_{i \ldots p}$ *can be estimated to determine the effect of each sub – test to a test of interest at the higher system level*

Part IV

Concluding Remarks

Conclusion

Type I Error Propagation

New bounds showed no increase in false alarms

Increase in Test Robustness

Eliminate Test Uncertainty
Decrease Allowable Performance Variation

Efficacy on High Failure Rate Components

Greatest cost savings on high failure rate components

- Touch Time
- Rework Operations

Statistical Analysis

Using statistical methods as verification techniques for AI-driven results

Functional Team Benefits

1	System Performance	Test Evaluation	Increase test precision
2	Project Management	Project Costs	Resource savings
3	Factory Production	Reproducibility	Simplification of tests

Methodology shows efficacy in RTX's domain:

- Complex systems with SOS interactions
- Expensive test operations
- High-failure rate components/technologies
- High-volume production
- Low-level production

Questions

Bogdan S. Racolta
Raytheon Technologies
bogdan.racolta@rtx.com

www.incose.org/symp2024
#INCOSEIS

34th Annual INCOSE international symposium

hybrid event

Dublin, Ireland
July 2 - 6, 2024