



Exposing Hidden Parts of the SE Process: MBSE Patterns and Tools for Tracking and Traceability

Maddalena Jackson

Marcus Wilkerson

Jean-Francois Castet

Agenda



- Introduction
- Why do we need tracking & traceability
- Case studies
- Conclusions



INTRODUCTION

Introduction (who is talking to you?)



- Me: NASA's proposed Europa Mission for ~2 years
- MBSE for 7
- Roles:
 - Practitioner
 - Systems Engineer on FS requirements team
 - Do requirements engineering, *happen* to use MBSE as tool of choice
 - SW developer for query, automation, tool, visualization, and any other as-need infrastructure
 - Model System Engineer for PSE
 - *One* interface between SEs with more traditional skill sets and system model
 - My particular role is software management



MBSE ON THE EUROPA PROJECT

Europa



*"Europa, with its probable vast subsurface ocean sandwiched between a potentially active silicate interior and a highly dynamic surface ice shell, offers **one of the most promising extraterrestrial habitable environments**, and a plausible model for habitable environments beyond our solar system"*

"Visions and Voyages", 2011 Planetary Decadal Survey

- How do we solve Europa's mysteries? By potentially sending a spacecraft and instruments to collect data for our investigation!
- Europa Project:
 - Early phase
 - Dual focus on system/design architecture and closing big trades and requirements derivation, analysis, and flow-down.

MBSE on the Europa Project

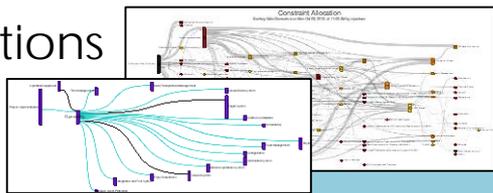


- Europa is fully MBSE mission concept (Project and SC level)
 - We use MBSE to do our SE
 - MBSE is not the product
- Specifically, for our phase:
 - MELs, PELs, resource allocation and analysis, system decomposition, etc
 - All systems engineering activities
 - Requirements (derivation, justification, traceability, analysis, maturity, history, verification, document generation, metrics, etc.)
 - This talk will focus on the SE aspects

What does the Europa Project do with that?



Custom visualizations



Spreadsheets

Parent Concept	Child Concept	Action	Passing Through the Following Concepts	Out
Parent Concept 1	Child Concept 1	Active	Concept A, Concept B	Value 1
Parent Concept 2	Child Concept 2	Active	Concept C, Concept D	Value 2
Parent Concept 3	Child Concept 3	Active	Concept E, Concept F	Value 3
Parent Concept 4	Child Concept 4	Active	Concept G, Concept H	Value 4
Parent Concept 5	Child Concept 5	Active	Concept I, Concept J	Value 5

PEL

Resources and

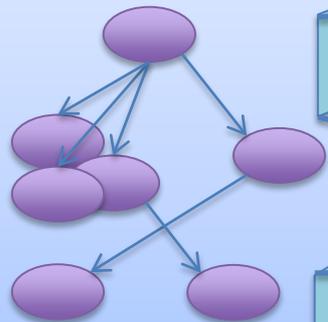
Margins

(read-write)

MEL

Gate Products for Release (read-only)

System Model



Spacecraft and Project Requirements Documents

Results for Rule: hasConstrainedElement

Alert: CONSTRAINED ELEMENT must have at least one constrained element

Description: Constraints that constrain anything other than ONE constrained element will FAIL this rule.

Applies To: ConstrainedBlock

Total Elements Evaluated: 31

VIOLATORS: 14 | PASSED: 1 | SKIPPED: 0 | SKIPPED (N/A): 16 |

AltId	Name of Validated Element	Validation Result	Model ID
CT100.807	Duration Between Off-Sun Turns in Inner Cruise	FAILED	_PBKXQwEvEOA16G4SXLN8A
CT100.804	Temperature Limits during Faults in Adverse Environments	FAILED	_PBKXQwEvEOA16G4SXLN8A
CT100.900	Supplementary Heater Power on Off Hardware	FAILED	_Jb1FDQwvEOA16G4SXLN8A
CT100.895	Temperature Limits during Faults with All Power Dispositions	FAILED	_Jb1FDQwvEOA16G4SXLN8A

Model v...
complete
reports;

Burndown,
work to go,
quality, and
other metrics

Automation and
query plugins and
libraries

Model
(read-write)



Other
engineering
analysis tools



- Assertion: MBSE can enable better (clearer, more automated) SE process
- Particularly in areas:
 - Maturity of requirements (and other elements)
 - Evolution of maturity and design
 - Metrics generation, tracking, reporting
 - Model V&V, confidence, and integrity

The problem

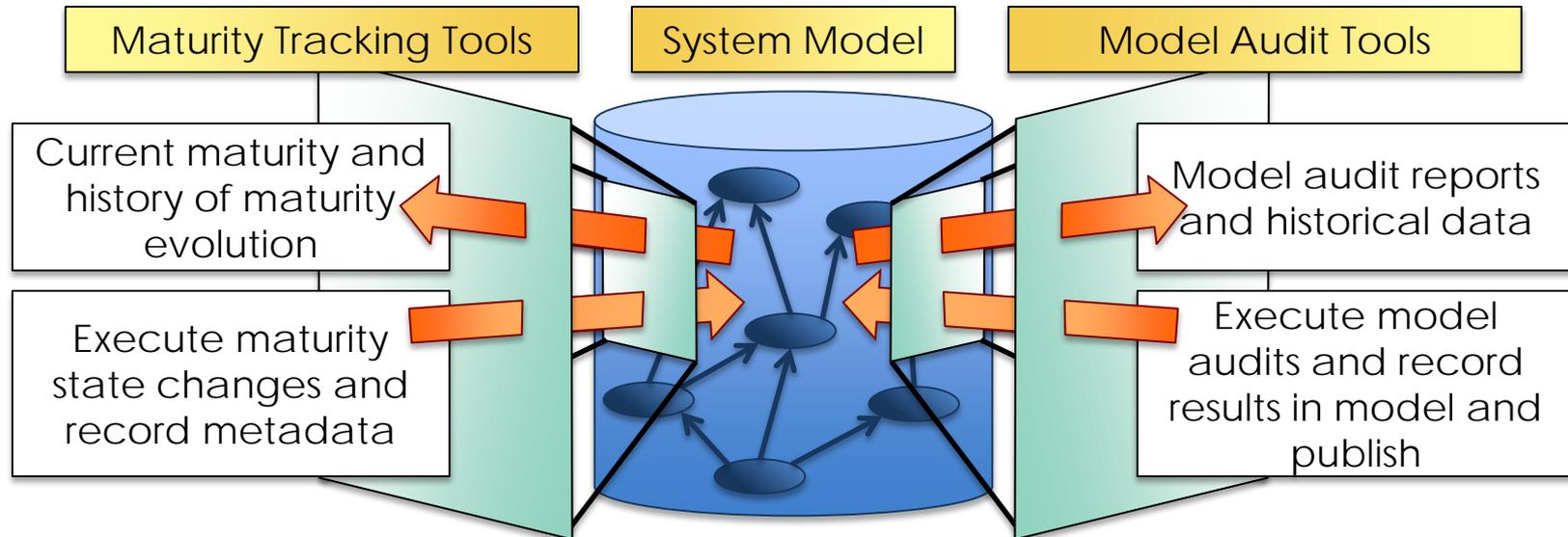


- Situation 1:
 - A: We have over a thousand requirements. How do I keep track of my work? Can the model-based approach help?
 - B: Someone allocated a requirement to me. I want to be able to reject or accept it. I also want to be able to propose a requirement to my parent. Can the model help me keep track of the agreement?
- Situation 2:
 - A: We know elements should be related in certain ways for the model to be “complete” and patterns to be “correct.” Can I check the model against assertions?
 - B: How many of x , y , z are there? How does that compare to last week?

MBSE + SE process – what we did



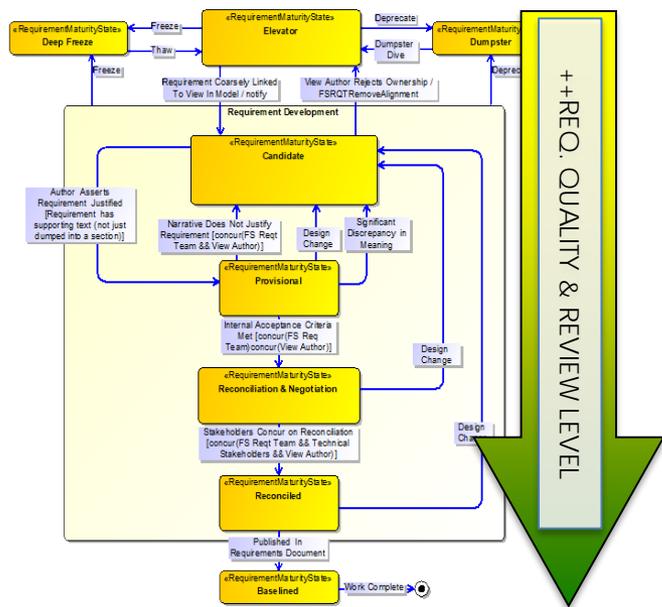
- These are common questions in any SE effort.
- Often have to answer these questions during reviews, in reporting to managers, for own confidence, or as part of modeling engineering role.
- What we did:





CASE 1: MATURITY EVOLUTION AND TRACKING

Maturity Evolution



Update 1 Element: Identified ----> ?

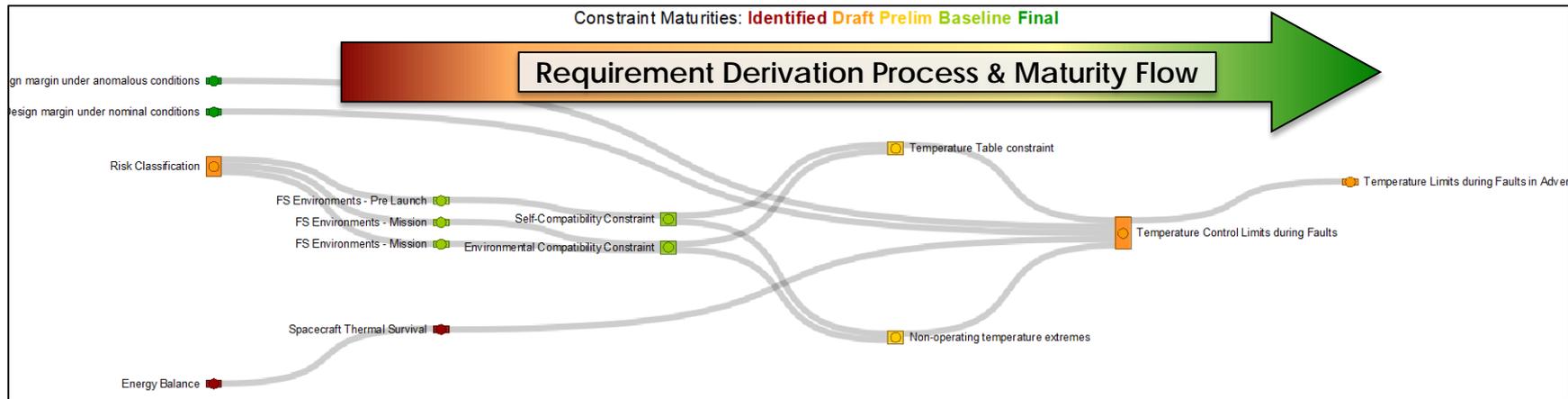
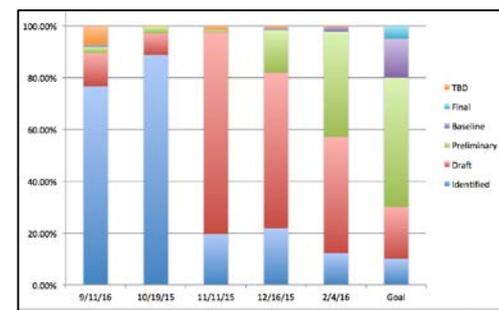
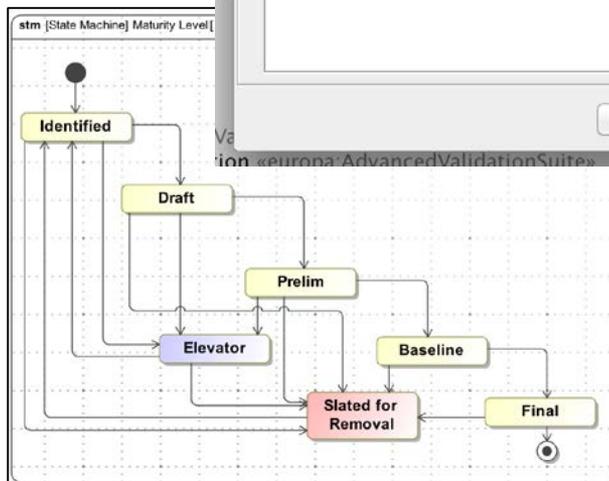
Select Next State (Identified ----> ?)

Draft Display all possible states (not recommended)

Comments about this change

Constraint approved at review

OK



Allocation and Negotiation



Technical Discipline Area
Responsible SE

constraint

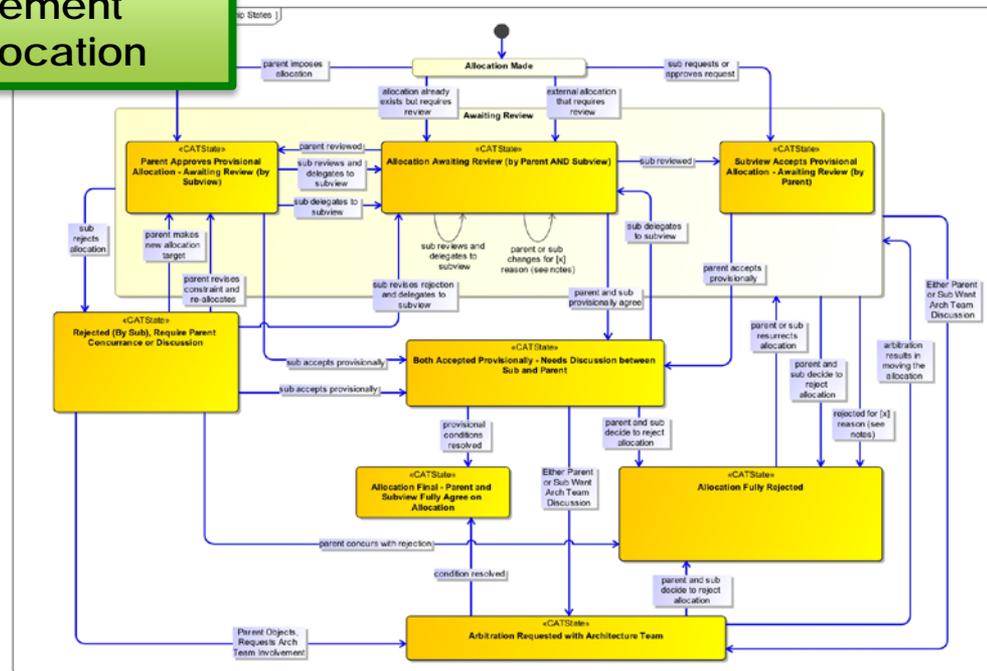
Constraint Allocation

- Parent allocates constraint
- Constraint is addressed by child
- Parent and child both approve the allocation

We want to track the
state of agreement
about the allocation

Technical
Discipline Area
Responsible SE

This is the machinery: abstracted
into a simple UI by code



Negotiation tracking



Update 1 Constraint Allocation: Both Accepted Provisionally - Needs ...

Select Transition out of Current State (Both Accepted Provisionally - Needs ...)

sub delegates to subview

Next State

Allocation Awaiting Review (by Parent AND Subview)

Who Initiated This Change?

Maddalena Jackson

Comments about this change

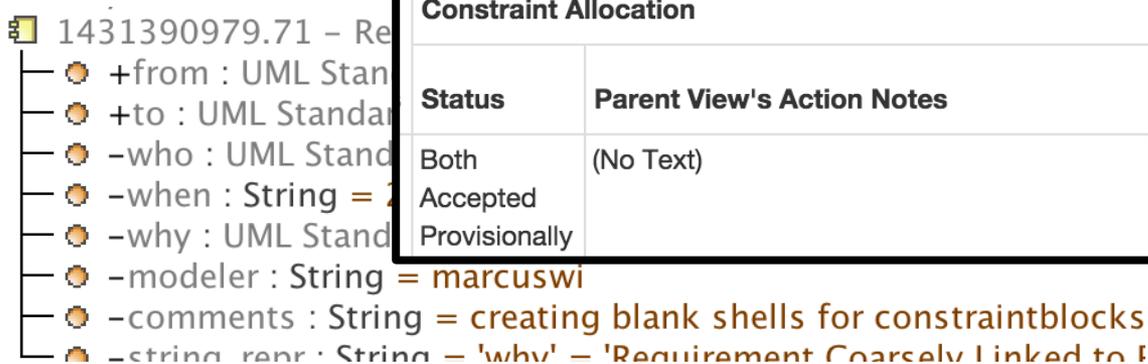
Constraint needs to be addressed at a lower level

Do you want to notify (email) anyone about this change?

Yes, I want to notify people!

OK

- Queryable: work to go, what needs attention, what is stale, what a responsible engineer has on plate
- Notifications
- History
- Metrics
- Negotiation status is live-linked in documents



Constraint Allocation		
Status	Parent View's Action Notes	Sub-View's Action Notes
Both	(No Text)	mjackson, 12.19.14: seems redundant with ETC 23 and ETC 24, although there's a caveat. Caveat is that 23 only talks about "launch induced thermal
Accepted		
Provisionally		



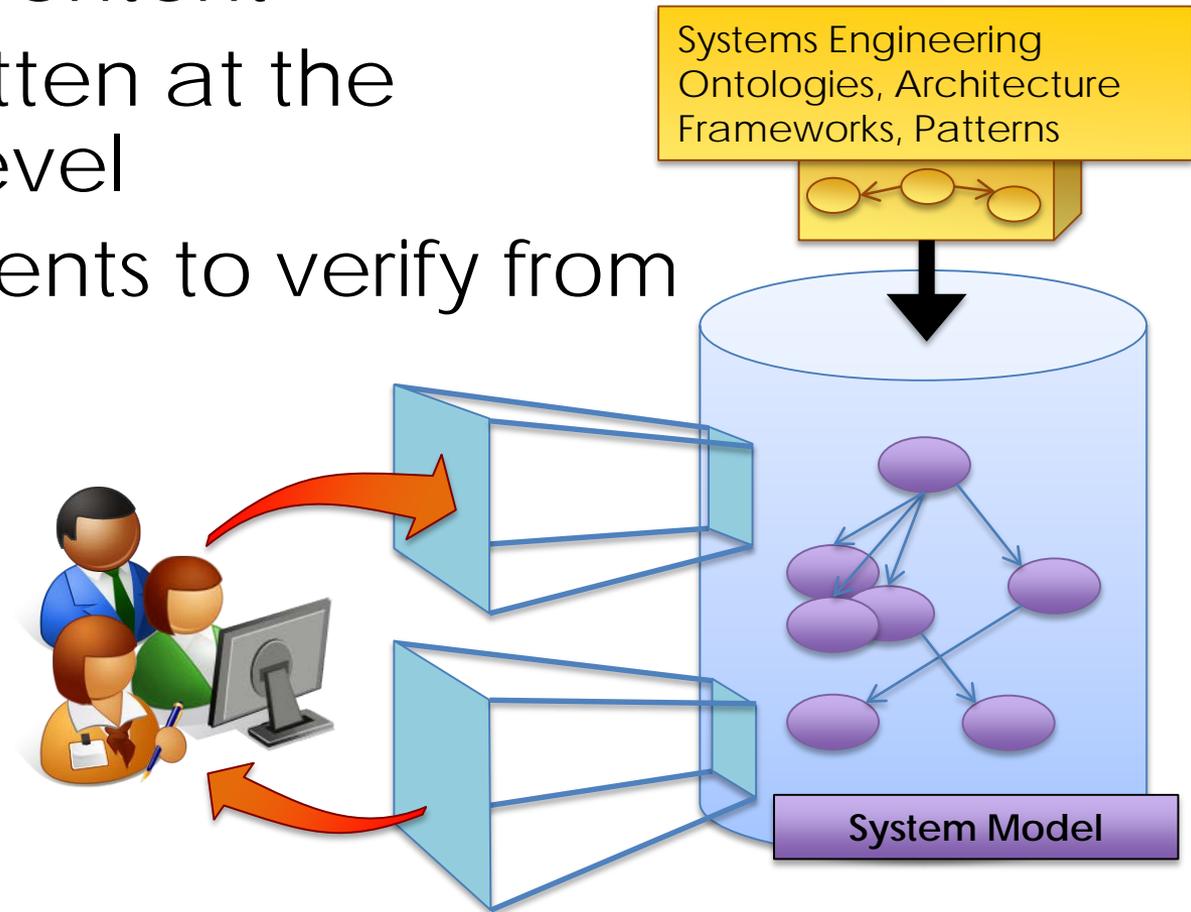
CASE 2: METRICS, MODEL V&V, AND HISTORY

Model V&V



- Need ability to make, check, and track assertions about the model and content
- Rules are written at the framework level
- Extract elements to verify from model

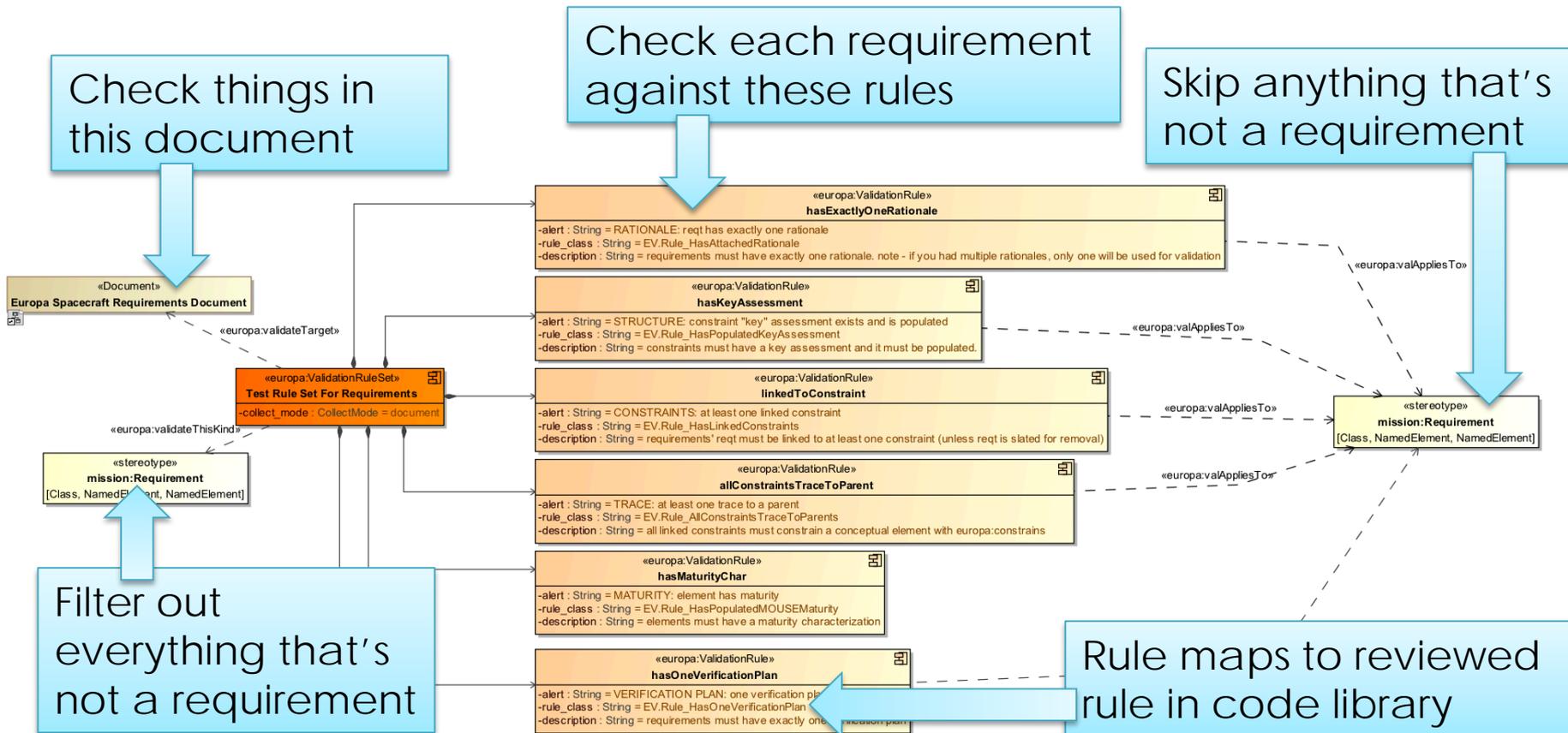
Is everything modeled completely?
Correctly?
Does my modeled information conform to my framework and patterns?



EVF Generic Execution Example



- Compose rules into "RuleSets"
- Model the configuration (simple pattern)
- Execute Generically



EVF Interactive In-Model Report



```
12.29:16:57:48 [Info] Elements (1):
12.29:16:57:48 [Info] collection mode is document
12.29:16:57:48 [Info] LOOKING IN 193 VIEWS
12.29:16:57:50 [Info] Server response: HTTP/1.1 200 OK
12.29:16:57:55 [Info] OK, we're validating 895 things
12.29:16:57:55 [Info] code class found for ValidationRule: Rule_HasAttachedRationale
12.29:16:57:55 [Info] code class found for ValidationRule: Rule_HasPopulatedKeyAssessment
12.29:16:57:55 [Info] code class found for ValidationRule: Rule_HasLinkedConstraints
12.29:16:57:55 [Info] code class found for ValidationRule: Rule_AllConstraintsTraceToParents
12.29:16:57:55 [Info] code class found for ValidationRule: Rule_HasPopulatedMOUSEMaturity
12.29:16:57:55 [Info] code class found for ValidationRule: Rule_HasOneVerificationPlan
12.29:16:57:55 [Info] number of rules: 6
12.29:16:57:55 [Info]
12.29:16:57:55 [Info] found all validation classes!
12.29:16:57:56 [Info]
12.29:16:57:56 [Info] Rules with violations:
12.29:16:57:56 [Info] STRUCTURE: constraint "key" assessment exists and is populated: (288 violations) selectAll
12.29:16:57:56 [Info] .....constraints must have a key assessment and it must be populated.
12.29:16:57:56 [Info] CONSTRAINTS: at least one linked constraint: (1 violation) selectAll
12.29:16:57:56 [Info] .....requirements' reqt must be linked to at least one constraint (unless reqt is slated for
12.29:16:57:56 [Info] TRACE: at least one trace to a parent: (615 violations) selectAll
12.29:16:57:56 [Info] .....all linked constraints must constrain a conceptual element with europa:constrains
12.29:16:57:56 [Info] VERIFICATION PLAN: one verification plan: (336 violations) selectAll
12.29:16:57:56 [Info] .....requirements must have exactly one verification plan
12.29:16:57:56 [Info]
12.29:16:57:56 [Info] All elements passed:
12.29:16:57:56 [Info] RATIONALE: reqt has exactly one rationale: (0 violations)
12.29:16:57:56 [Info] .....requirements must have exactly one rationale. note - if you had multiple rationales,
12.29:16:57:56 [Info] MATURITY: element has maturity: (0 violations)
12.29:16:57:56 [Info] .....elements must have a maturity characterization
12.29:16:57:56 [Info]
12.29:16:57:56 [Info] -----REPORT-----
12.29:16:57:56 [Info] RATIONALE: reqt has exactly one rationale: (failed: 0 | passed: 895 | skipped: 0)
12.29:16:57:56 [Info] requirements must have exactly one rationale. note - if you had multiple rationales, only one will be used for validation (APPLIES TO STEREOTYPES: ['mission:Requirement'] (all others skipped))
12.29:16:57:56 [Info]
12.29:16:57:56 [Info] STRUCTURE: constraint "key" assessment exists and is populated: (failed: 288 | passed: 606 | skipped: 1) select all \(fix all sequentially\)
12.29:16:57:56 [Info] constraints must have a key assessment and it must be populated. (APPLIES TO STEREOTYPES: ['mission:Requirement', 'Const', 'Block'] (all others skipped))
12.29:16:57:56 [Info] violator: Cross-Strapping Symmetry \(fix it!\)
12.29:16:57:56 [Info] violator: Gravity Science Signal \(fix it!\)
12.29:16:57:56 [Info] violator: Rotation Thresholds \(fix it!\)
12.29:16:57:56 [Info] violator: Use of Spacecraft for Gravity Science Measurements \(fix it!\)
12.29:16:57:56 [Info] violator: Record specific values for faults \(fix it!\)
12.29:16:57:56 [Info] violator: Reporting Instrument Data Telemetry Errors \(fix it!\)
12.29:16:57:56 [Info] violator: Autonomous JOL Execution \(fix it!\)
12.29:16:57:56 [Info] violator: Sunpoint without absolute time capability \(fix it!\)
12.29:16:57:56 [Info] violator: Vent only once \(fix it!\)
12.29:16:57:56 [Info] violator: Configurability of FS States by Onboard FSW \(fix it!\)
12.29:16:57:56 [Info] violator: Reconstructed Knowledge - ICEMAG \(fix it!\)
12.29:16:57:56 [Info] violator: RWA swap \(fix it!\)
12.29:16:57:56 [Info] violator: Updating On-board Files and Parameters \(fix it!\)
12.29:16:57:56 [Info] violator: Reconstructed Knowledge - REASON \(fix it!\)
12.29:16:57:56 [Info] violator: Visibility into Downlink Buffer Space Utilization \(fix it!\)
12.29:16:57:56 [Info] violator: Vital Telemetry from Direct Measurement Sources \(fix it!\)
12.29:16:57:56 [Info] violator: Best-effort Downlink Mode \(fix it!\)
12.29:16:57:56 [Info] violator: H/W Undervoltage Response \(fix it!\)
12.29:16:57:56 [Info] violator: Maintain Quiescent Attitude State \(fix it!\)
12.29:16:57:56 [Info] violator: Summarizing Spacecraft State to Fit 30 minutes of Downlink Bandwidth \(fix it!\)
12.29:16:57:56 [Info] violator: Commanding backup devices \(fix it!\)
12.29:16:57:56 [Info] violator: Downlink or Record Selection Criteria for Telemetry Channels \(fix it!\)
12.29:16:57:56 [Info] violator: Thrust axes \(fix it!\)
12.29:16:57:56 [Info] violator: Autonomous launch activities \(fix it!\)
12.29:16:57:56 [Info] violator: Downlink Priority Configurability \(fix it!\)
12.29:16:57:56 [Info] violator: Keep Out Zone Limitations \(fix it!\)
12.29:16:57:56 [Info] violator: Communication timing in safing \(fix it!\)
12.29:16:57:56 [Info] violator: initialize attitude on RWAs \(fix it!\)
```

Successfully fetched all 895 requirements from document on server

Rule code found and initialized

Summary section (violations)

Summary section (all pass)

Launch rule-fixer (batch mode)

Detailed report of elements that violate the rule. Can navigate to element to fix manually or can fix with fixer tool.

EVF Web Reporting



Validation Rule Set Report

Executed Rule Set: Requirement Rule Set (_18_0_2_8af0285_1456424903741_891483_385359)
Collection Source: Flight System, Requirements
Validated Element Types: mission:Requirement

← Summary of validation rule set

Results for Rule: hasExactlyOneRationale

Alert: RATIONALE: reqt has exactly one rationale
Description: requirements must have exactly one rationale. note - if you had multiple rationales, only one will be used for validation
Applies To: mission:Requirement
Total Elements Evaluated: 3811

↓ Detailed results for each rule (ID for finding element quickly)

VIOLATORS: **1** | PASSED: **3810** | SKIPPED: **0** | SKIPPED (N/A): **0** |

Afid	Name of Validated Element
RQ101.011	Nadir pointing accuracy
RQ103.177	Alliveness test or touch test
RQ105.231	FS TLM via LV
RQ103.294	Spectral Resolution below 2500nm

Results for Rule: allConstraintsTraceToParent

Alert: TRACE: at least one trace to a parent
Description: all linked constraints must constrain a conceptual element with <<europa:constrains>>
Applies To: mission:Requirement
Total Elements Evaluated: 7

VIOLATORS: **5** | PASSED: **2** | SKIPPED: **0** | SKIPPED (N/A): **0** |

Afid	Name of Validated Element	Validation Result	Model ID
RQ101.916	Uplink Re-transmit Attempts	FAILED	_17_0_2_3_f36036c_1380586409090_1713_30703
RQ101.913	Processing at Maximum Uplink Rates	FAILED	_17_0_2_3_f36036c_1380586629373_871672_31028
RQ100.696	Instrument Sequences	FAILED	_17_0_2_3_f36036c_1380654798802_58135_30666
RQ100.673	Information Structure	FAILED	_17_0_2_3_f36036c_1380570837251_801942_28653
RQ101.910	Processing Uplink Data Stream	FAILED	_17_0_2_3_f36036c_1380586561865_227644_30963
RQ101.911	Uplink Frame Error Rate	PASS	_17_0_2_3_f36036c_1380586734624_886838_31074
RQ101.907	Uplink Accountability Report	PASS	_17_0_2_3_f36036c_1380586511996_571134_30900

Automatic EVF execution and web publication

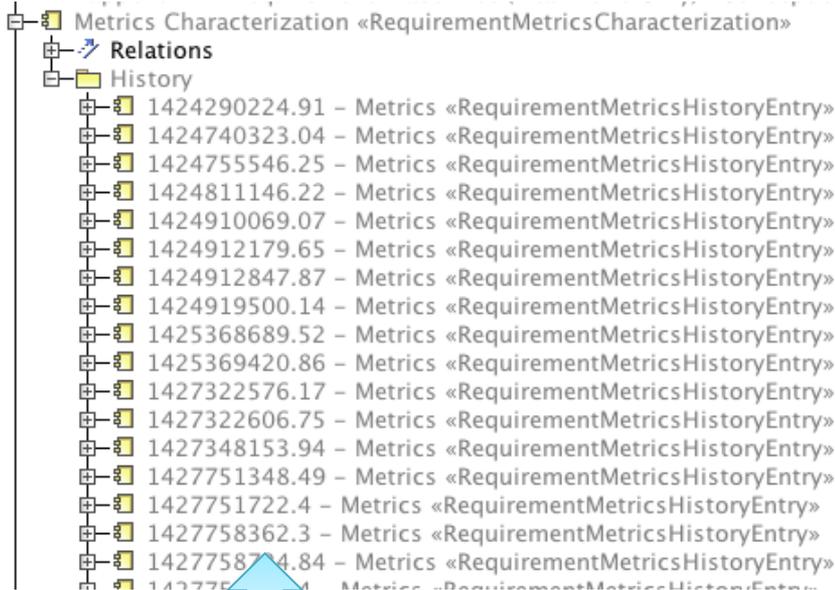
Easily adjustable web report template

Results for Rule: hasMaturityChar

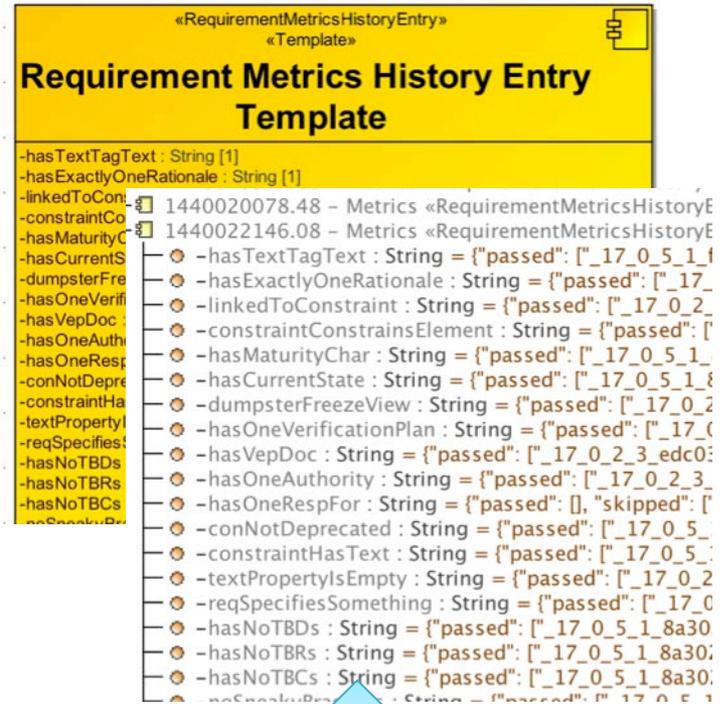
History



Inside of our Spacecraft Requirements Document...



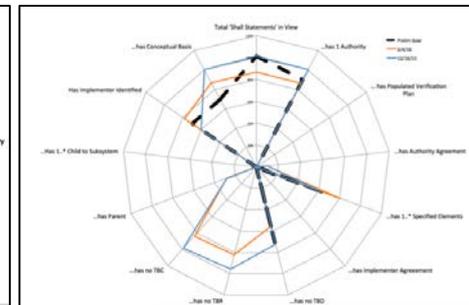
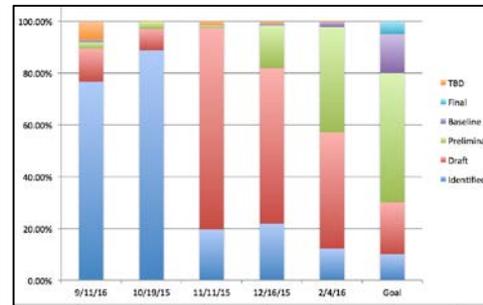
Each of these History Entries...



Contains Pass/Fail/Skip records for each requirement for each rule

History can be queried and used for

- Metrics
- Trending
- Generating historical reports
- Individual element history





- Lightweight
- Minimal software development
- Model reusable rule configurations
- Code is easy to review and unit test

- In use:
 - Modeling team internal quality checks
 - Metrics for burndown, management, trending



CONCLUSIONS

Conclusions



- SE process can easily be supported (and enriched) by MBSE
 - Tracking maturity, negotiation state → work to go, situational awareness, historical traceability
 - Rules-checking in model → confidence, metrics, simple reporting, easy fix interface, history
- SE process needs not always well understood until runtime
- Most time spent hashing out patterns and rules (code is easy once algorithms agreed upon)

Recommendations



- Maturity definitions and rules in SEMP
- Explicit study of SE process → requirements for infrastructure