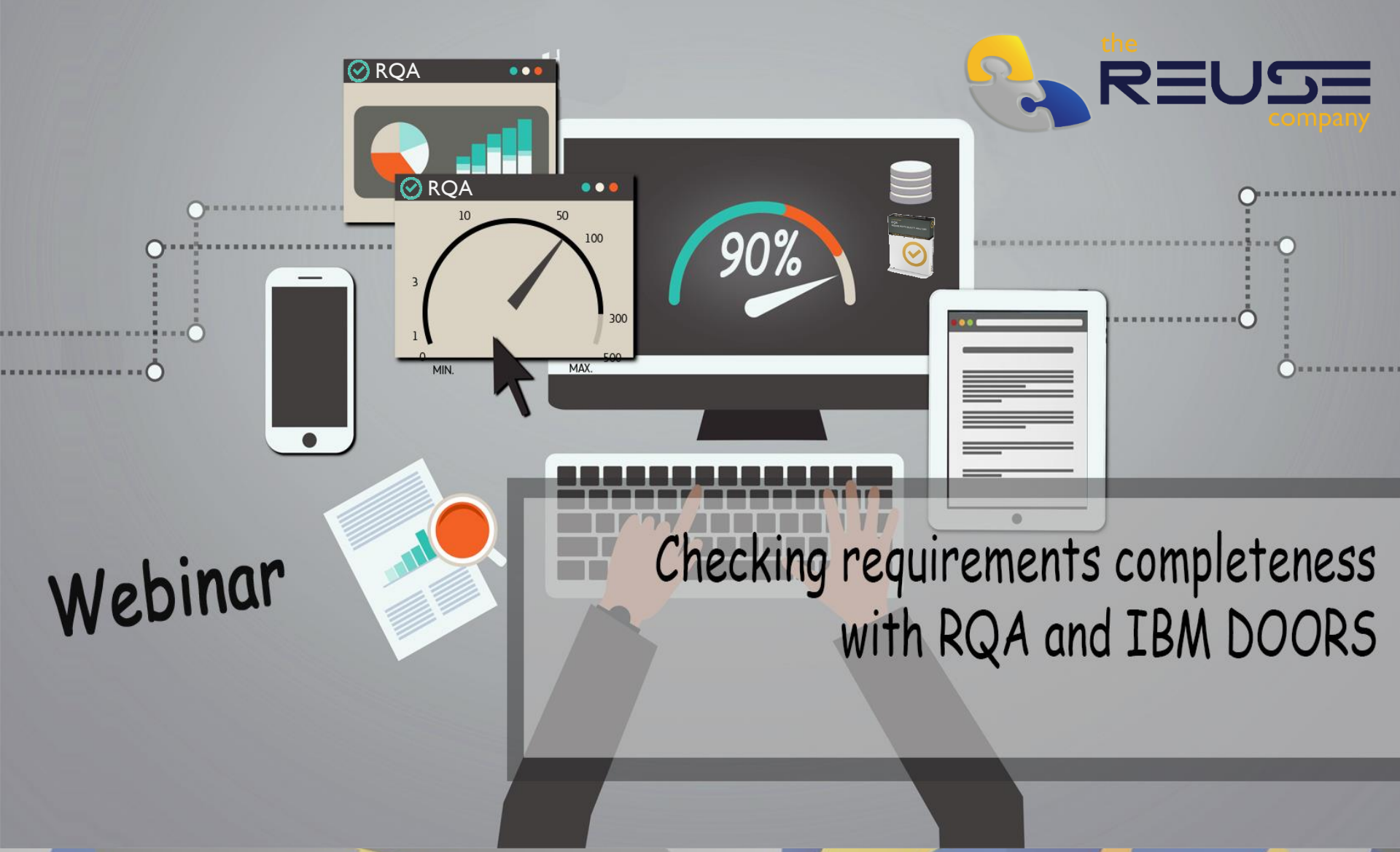


the REUSE company

Webinar

Checking requirements completeness with RQA and IBM DOORS

José Fuentes

Monday, November 20, 2017

Content

- Introduction
- The problem of requirements completeness
- Completeness metrics in RQA
- How to check completeness in RQA
- Live demo
- Q&A

Introduction

Introduction: Webinar rules

- Webinar rules:
  - The Webinar will start in few minutes
  - You'll be muted all along the Webinar
  - There's a chatting box to ask your questions or send your comments when you want
  - Please address these comments and questions to the user "The REUSE Company" and not to me directly
  - If you have any technical issue please use this chatting box, or mail us at: support@reusecompany.com
  - The Webinar will be recorded. A link to the recording will be sent to you in few days

Presentación

José M. Fuentes

jose.fuentes@reusecompany.com

+34 912 17 25 96

@ReuseCompany

https://www.linkedin.com/in/josemiguelfuentes/

TRC - Our competences

T(he) R(euse) Q(ompany)(y)

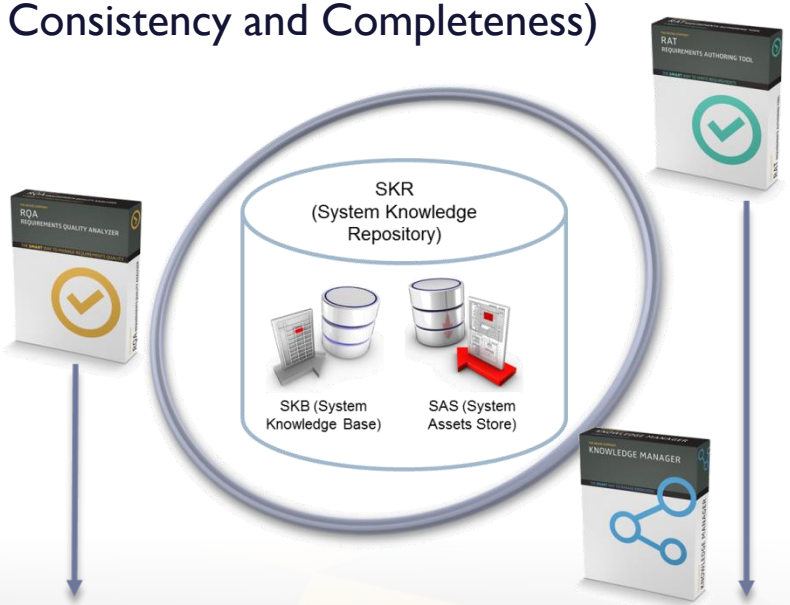
Trace + Retrieval + Quality

Towards systematic Reuse

By means of : **Repositories** containing **Ontologies and Assets**

RQS – Requirements Quality Suite

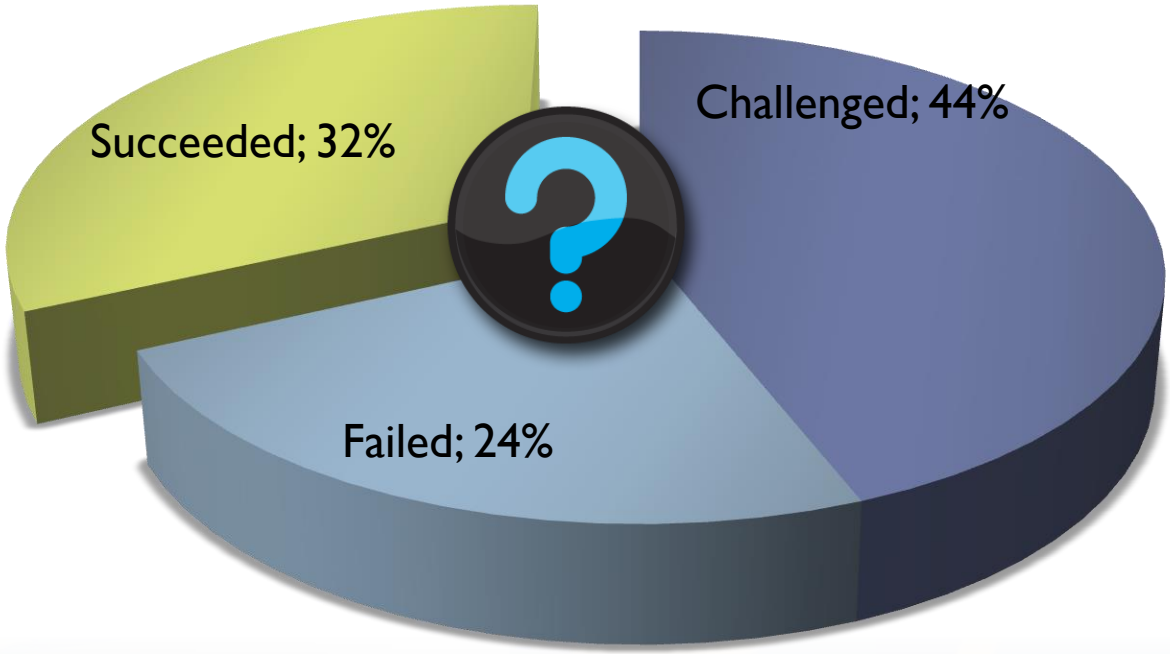
- The Requirements Quality Suite (RQS) intends to tackle requirements quality management by offering a set of tools and processes
- Automatic measurement of requirements quality metric
- Support to Requirements Authoring
- RQS models requirements quality metrics using the CCC approach (Correctness, Consistency and Completeness)



- **Requirements Quality Analyzer (RQA):** to setup, check and manage the quality of a requirements specification
- **Requirement Authoring Tool (RAT):** to assist authors while they are creating or editing requirements.
- **Knowledge Manager (KM):** to manage knowledge around a requirements specification: dictionaries, glossaries, concept maps, knowledge models, ontologies, patterns…

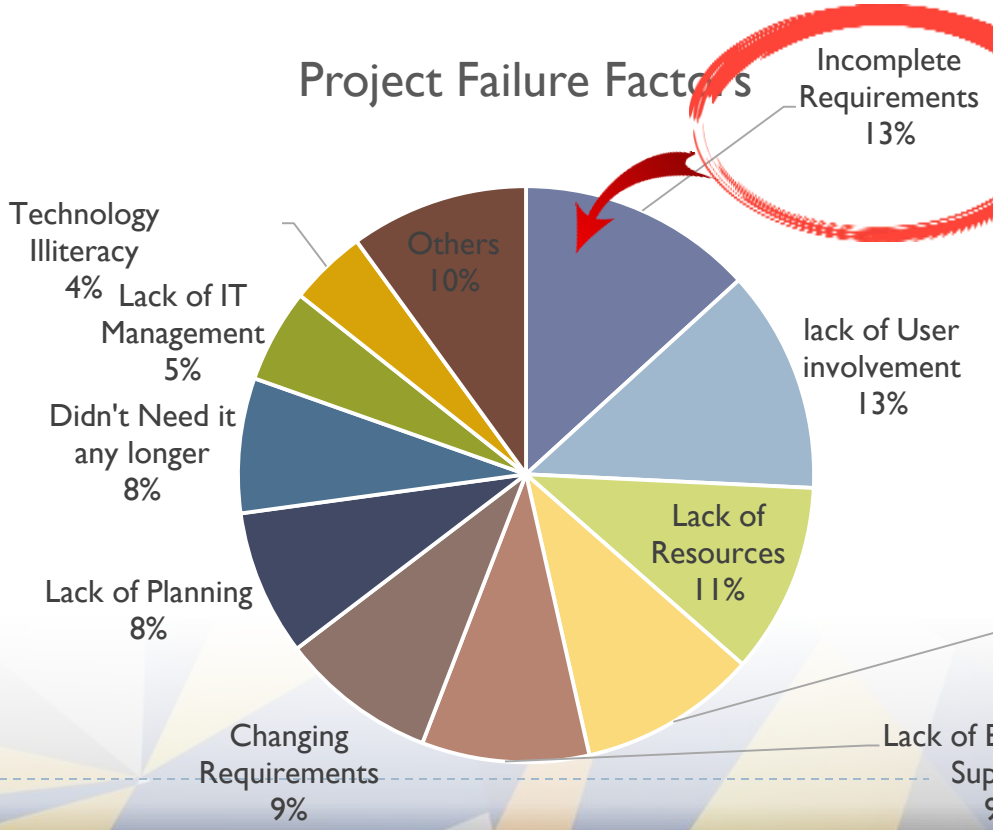
Current state: Chaos report

Chaos Report, 2009



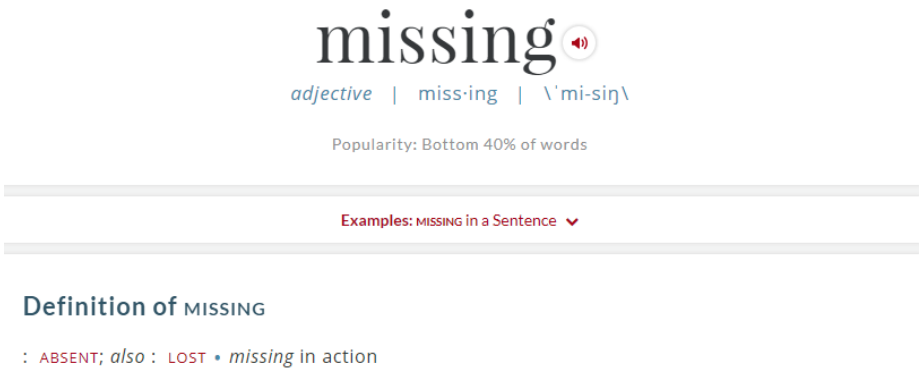
Current status : Chaos report

Project Success Factors	% of Responses
1. User Involvement	15.9%
2. Executive Management Support	13.9%
3. Clear Statement of Requirements	13.0%
4. Proper Planning	9.6%
5. Realistic Expectations	8.2%
6. Smaller Project Milestones	7.7%
7. Competent Staff	7.2%
8. Ownership	5.3%
9. Clear Vision & Objectives	2.9%
10. Hard-Working, Focused Staff	2.4%
Other	13.9%



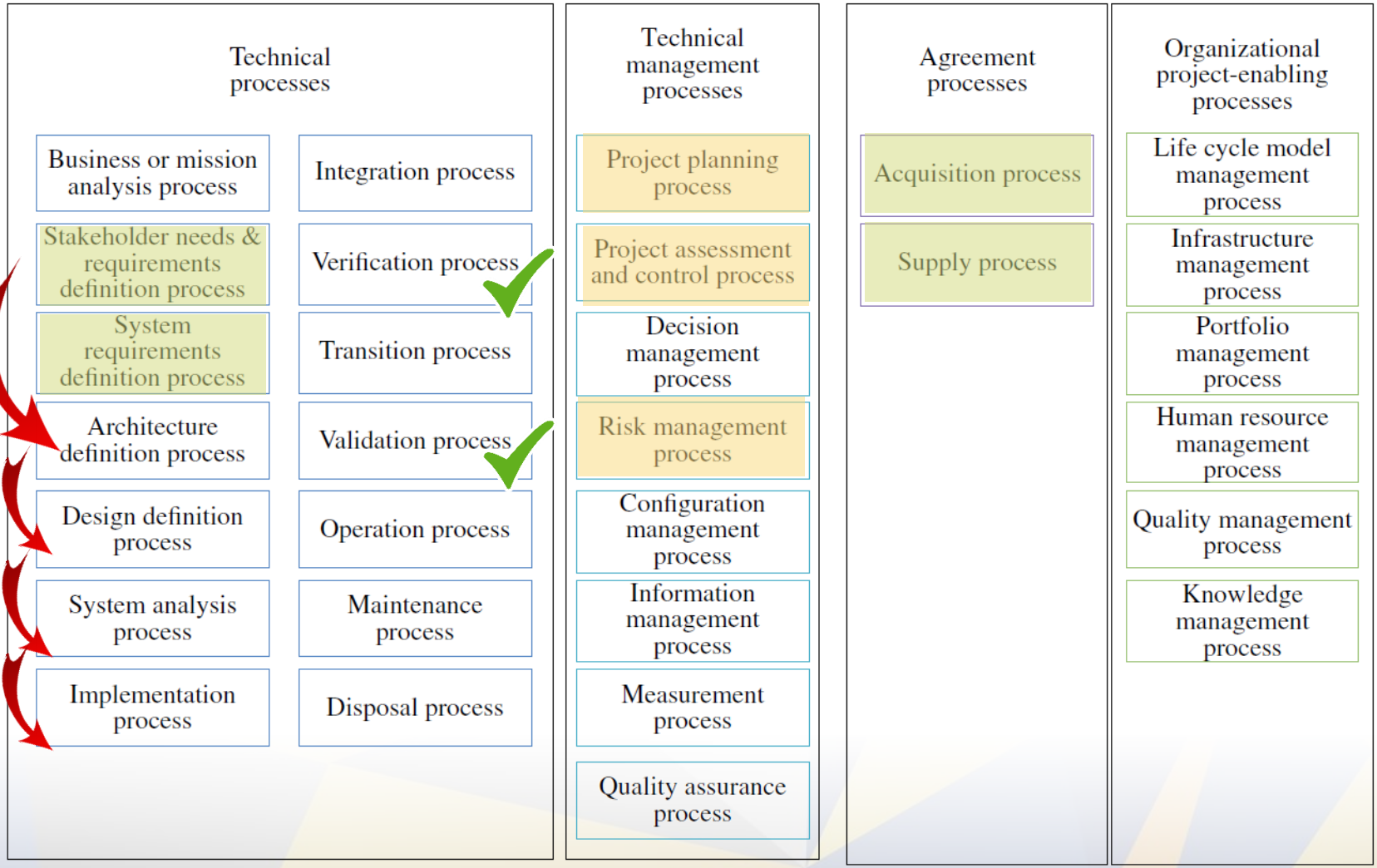
Completeness and missing requirements

- Merriam-Webster dictionary:



- INCOSE Guide for Writing Requirements:
  - The requirements set stands alone such that it sufficiently describes the necessary capabilities, characteristics, constraints, interfaces, standards, regulations, and/or quality factors to meet the entity needs without needing other information
  - Minimum and sufficient

Impact of missing requirements

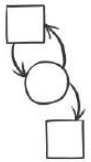
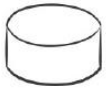


How to address completeness

for our requirements

Missing requirements: it's all about asking the proper questions

- All different dimensions have to be taken into account

						
User	Interface	Action	Data	Control	Environment	Quality Attribute
Users interact with the product	The product connects to users, systems, and devices	The product provides capabilities for users	The product includes a repository of data and useful information	The product enforces constraints	The product conforms to physical properties and technology platforms	The product has certain properties that qualify its operation and development

- The 7 product dimensions: "Discover to deliver: Agile product planning and analysis". Ellen Gottesdiener, Mary Gorman

Missing requirements: it's all about asking the proper questions

- What triggers the system to do X?
- In what system state(s) shall the system do (or not do) X?
- What are the performance requirements related to the system when doing X? Specifically, are there minimum/maximum limits on throughput, response time, jitter, etc. when the system does X?
- In what state shall the system be left when it is done with doing X?
- What are the data and interface requirements associated with doing X? Specifically, what data shall flow in and out of the system when the system is doing X? What data must the system store and what stored data must the system use when doing X? What interfaces shall the system use when doing X?
- What capacity requirements are associated with doing X? Specifically, how shall the system's performance when doing X change as system load nears the limits of its specified capacity?

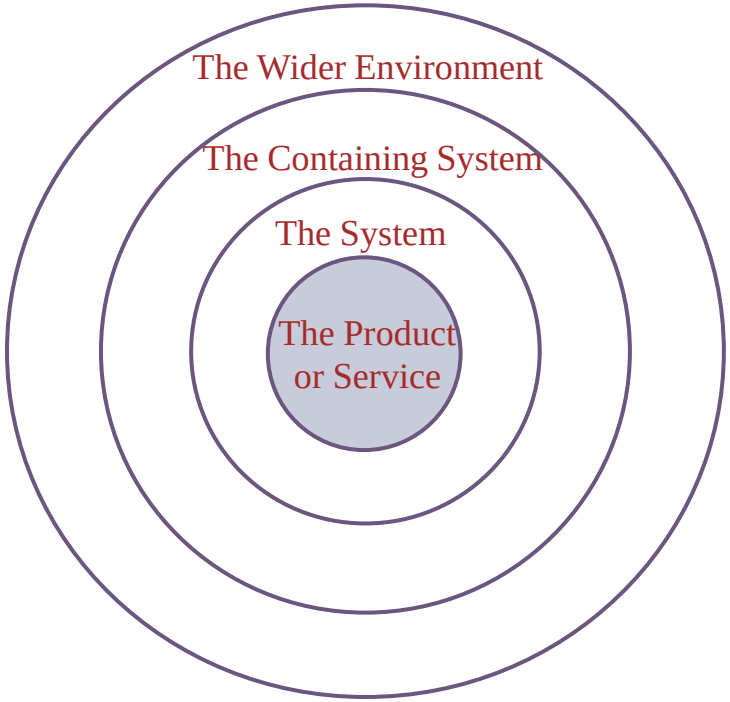
Missing requirements: it's all about asking the proper questions

- Is doing X a critical function that must be preserved when the system goes into degraded mode? In other words, is performing X related to the system's survivability requirements?
- What is the safety trust level (STL) of the system doing X? What are the safety integrity levels (SILs) of the system components involved in doing X[7]?
- Does the system doing X have related security requirements? Specifically, are only certain external entities (e.g., individual people, roles people play, groups of people, external systems) authorized to request the system to do X? Must the system ensure the security of private data or messages when doing X?
- What shall the system do if it cannot do X? For example, too often requirements engineers only specify the normal case ("sunny day") paths of use cases. What about the "rainy day" paths? Does the alternative exceptional behavior to performing X vary depending on system state?

Source: Donald Firesmith (SEI)

Missing requirements: involve every stakeholder

- The Product or Service
  - Contains no stakeholders
- The System (of interest)
  - The Product or Service plus the people who operate the product or deliver the Service
  - Also often includes training, support and maintenance
- The Containing System
  - Those who immediately benefit from the functions carried out by the System or Interface with it
  - Are usually, but not necessarily, different from the operators
- The Wider Environment
  - People who are affected indirectly, such as derived benefit of induced harm.



role s

Missing requirements: dimensions

- Requirements taxonomy:
  - E.g. The NASA requirements taxonomy

Technical Requirements – Allocation Hierarchically to PBS
Functional Requirements
Performance Requirements
Interface Requirements

Operational Requirements – Drive Functional Requirements
Mission Timeline Sequence
Mission Configurations
Command and Telemetry Strategy

Reliability Requirements – Project Standards – Levied Across Systems
Mission Environments
Robustness, Fault Tolerance, Diverse Redundancy
Verification
Process and Workmanship

Safety Requirements – Project Standards – Levied Across Systems
Orbital Debris and Reentry
Planetary Protection
Toxic Substances
Pressurized Vessels
Radio Frequency Energy
System Safety
…

Specialty Requirements – Project Standards – Drive Product Designs
Producibility
Maintainability
Asset Protection
…

Missing requirements: the entire lifecycle

- Requirements is not just the *operational phase*
- Other requirements are also necessary

- Consider also requirements for the transition between the current system and the system to be built

Completeness for individual requirements

- Events **trigger** the system to perform the function including any input data, requests received, or exceptions being handled
- **Preconditions** must hold for the system to be able to successfully perform the function, including system mode and state, the state of system externals, and the values of any system data
- **Actions** the system must perform when receiving the triggers when the preconditions hold
- **Postconditions** must hold once the system successfully performs its function

Completeness for individual requirements

- Links
  - Missing links is also a source of incompleteness
- Attributes:
  - Missing the necessary information in other attributes makes your specification not complete
  - And also it makes difficult to meet other quality characteristics:
    - Traceable
    - Ranked
    - Verifiable
    - …

Completeness: not only requirements

- A requirements specification is not the only document to be created
  - Dictionaries
  - Models
  - Stakeholders matrixes
  - …
- Completeness of all these other artifacts is also a must

Missing requirements: other ways to tackle

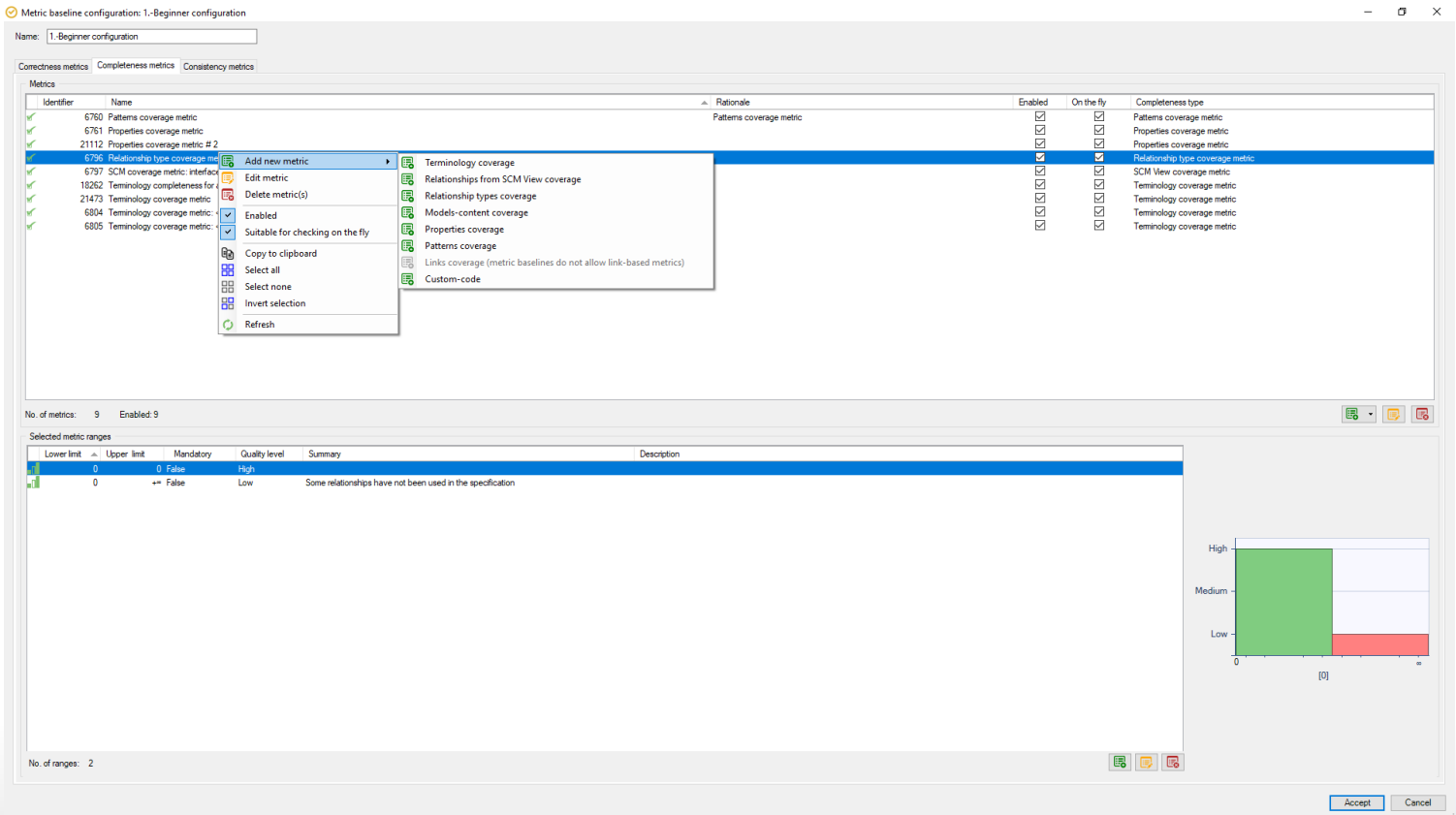
- Use of requirements checklists
- Requirements reuse
- Improve communication with stakeholders
- Improve the communication skills of your business analysts
- Establish a formal inspection process
- Consider the elicitation process as an iterative and recursive process

Completeness metrics

in RQA

Completeness metrics in RQA

- Based on *parameterized* metrics:



Completeness metrics in RQA: terminology

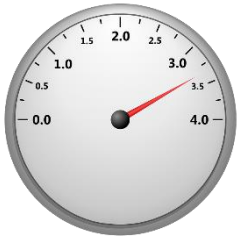
- Answers the following questions:
  - Are we involving all different system/sub-systems/components/parts?
  - Are we reacting under all different *states* and *modes*?
  - Are we representing all the stakeholders?
  - Does the spec still contains TBDs, TBCs…?
  - Do our specs include the counterparts of the involved actions. E.g.:
    - If a requirement is about inserting *entityX*, shouldn't it be another requirement to delete *entityX*
    - If a system allows to *raise the temperature* shouldn't we expect another requirement to *lower the temperature*
    - …
  - If a specific action is performed at level X of our spec, shouldn't we expe same action to be performed by the derived requirements at level X+1?

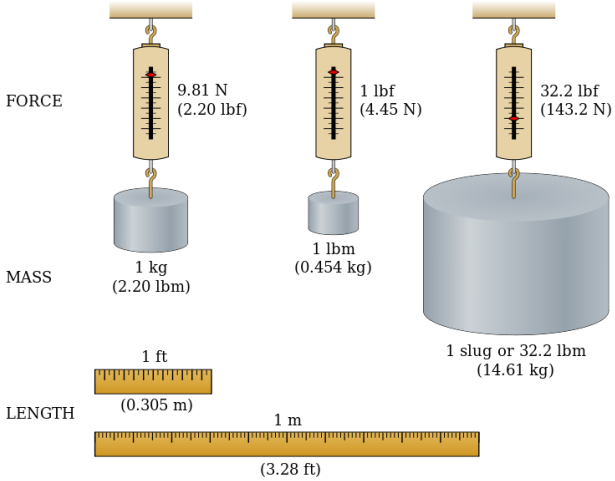
Completeness metrics in RQA: relationships

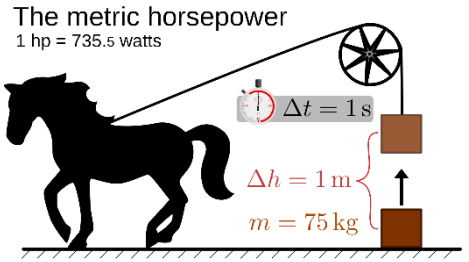
- Answers the following questions:
  - Do the requirements represent the entire PBS?
  - Are all the identified interfaces really "used" in the requirements?
  - Do the transitions among states in your *statecharts* (e.g. UML, SysML) in accordance with the transitions stated in natural language in our requirements?

Completeness metrics in RQA: properties

- Answers the following questions:
  - Do the requirements include the actual value for the different properties identified in the physical elements of my system?



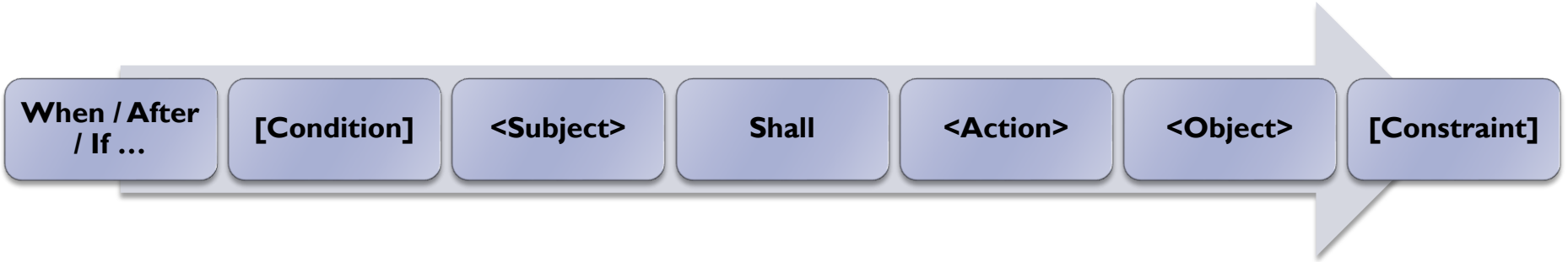




- Remember the counterpart of this kind of metrics in the consistency dimension. Check our previous Webinar at: https://goo.gl/933kZ4

Completeness metrics in RQA: patterns

- Answers the following questions:
  - Does the specification address all the expected types of requirements?
  - Use KM to create different patterns for different types of requirements
  - Check which of those patterns are matched in your specifications, and the number of requirements matching with the different patterns



Completeness metrics in RQA: links

- Answers the following questions:
  - Are all the requirements properly linked to each other?
  - And to other elements outside the requirements management tool?

Sample traceability matrix

Requirement Identifiers	Reqs Tested	REQ1 UC 1.1	REQ1 UC 1.2	REQ1 UC 1.3	REQ1 UC 2.1	REQ1 UC 2.2	REQ1 UC 2.3.1	REQ1 UC 2.3.2	REQ1 UC 2.3.3	REQ1 UC 2.4	REQ1 UC 3.1	REQ1 UC 3.2	REQ1 TECH 1.1	REQ1 TECH 1.2	REQ1 TECH 1.3
Test Cases	321	3	2	3	1	1	1	1	1	1	2	3	1	1	1
Tested Implicitly	77														
1.1.1	1	x													
1.1.2	2		x	x											
1.1.3	2	x											x		
1.1.4	1			x											
1.1.5	2	x												x	
1.1.6	1		x												
1.1.7	1			x											
1.2.1	2				x		x								
1.2.2	2					x		x							
1.2.3	2								x	x					
1.3.1	1										x				
1.3.2	1										x				
1.3.3	1											x			
1.3.4	1											x			
1.3.5	1											x			
etc....															
5.6.2	1														x

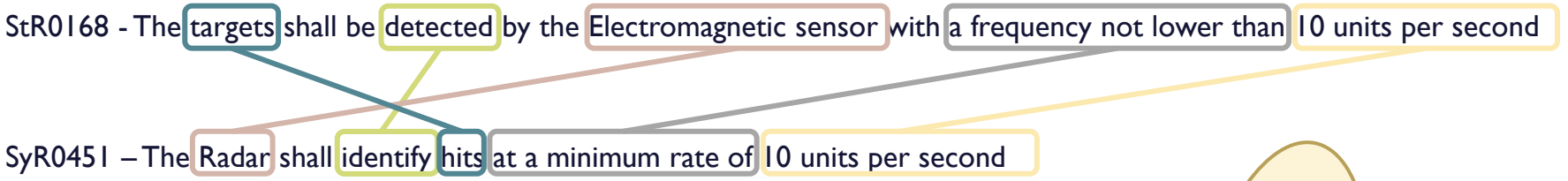
Completeness metrics in RQA: requirements reuse

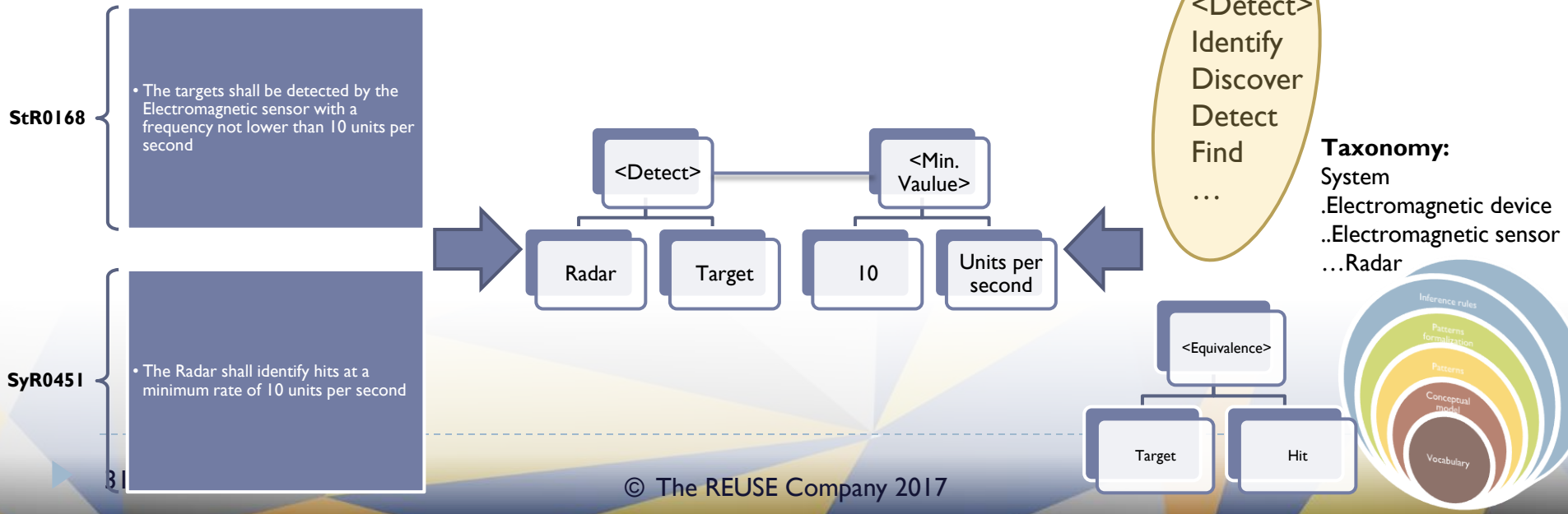
- Coarse grain reuse (strategic reuse):
  - Requirements libraries
  - Parameterized requirements

  - RM tools normally offer solutions to address this need
  - Product Line Engineering specific tools to tackle the variability issues, feature models….

Completeness metrics in RQA: requirements reuse

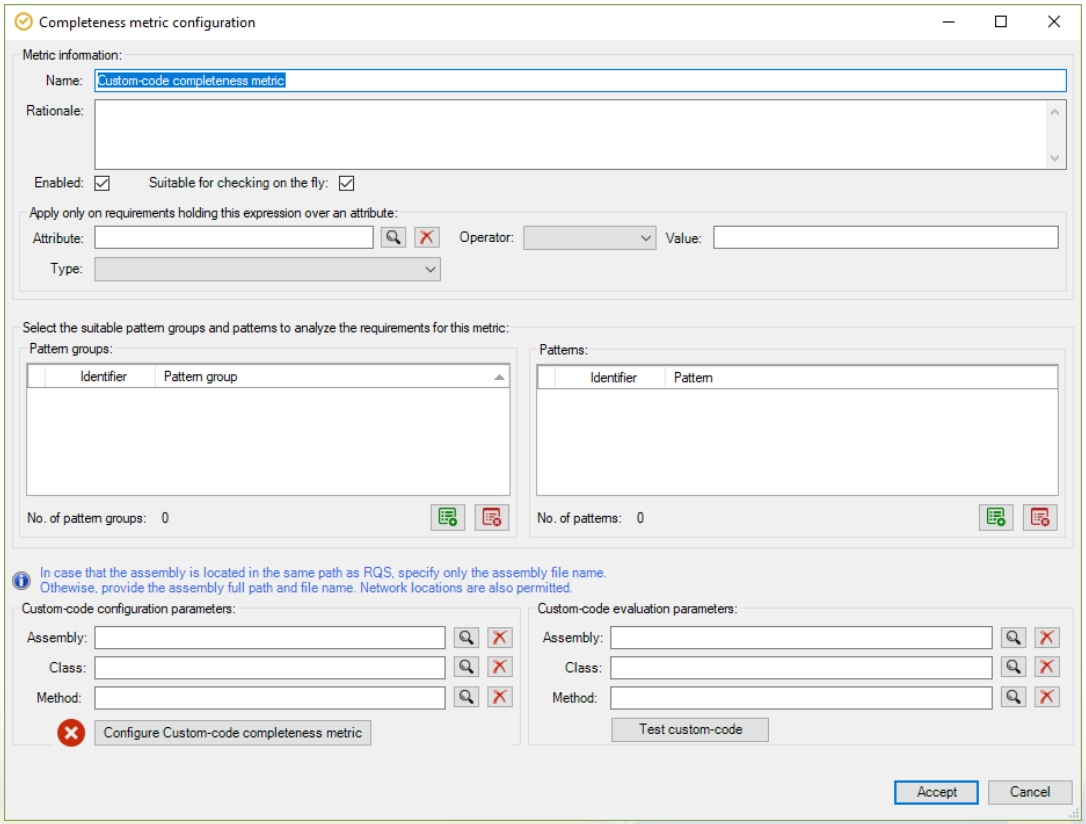
- Fine grain reuse (ah-hoc reuse): semantic search engine
- Based on patterns, formalization and knowledge bases:





Completeness metrics in RQA: custom

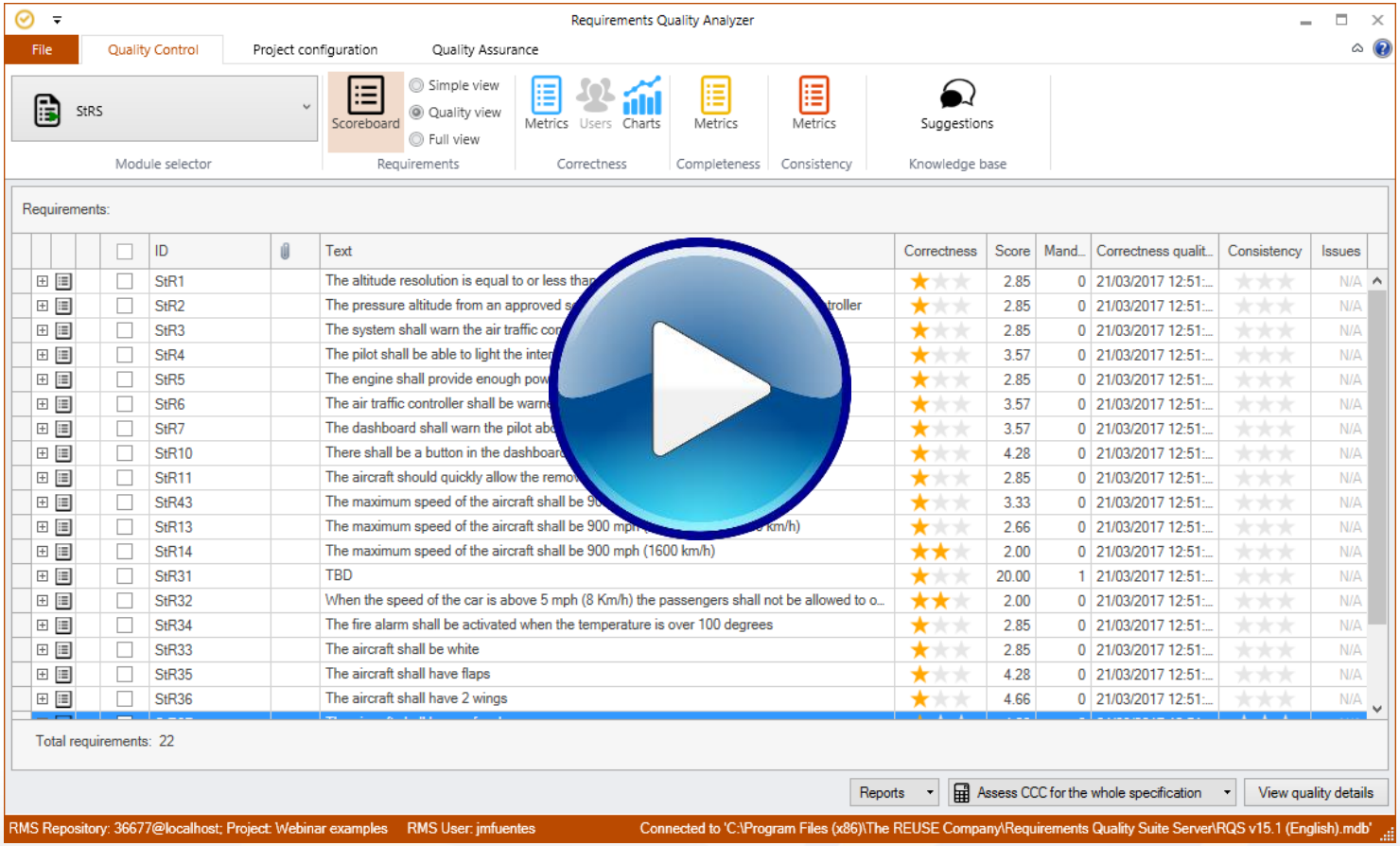
- Answers the following questions:
  - Now you! Ask your own question, and answer it based on your own custom code



Missing requirements : conclussions

- Workout the appropriate **process, techniques, tools** and **skills**:
  - **Inspection** process
  - **Reuse** process
  - **Elicitation** techniques, **inspection** techniques
  - **Requirements management** and **traceability** tools
  - **Checklists, patterns, prototypes**…
  - **Communication is the key:** soft skills, assertiveness
- But this is **not enough**:
  - A complete specification is normally too large
  - Difficult to address all different dimensions by humans
  - The help of tools is **a must**

Live demo



Special gift

- I'd like to offer a special gift for you, the audience of this Webinar
- **Order RQS** tools **before end of 2017**
- And provide us this **promotional code:** WebinarCompleteness17
- And you'll get:
  - A **30% discount** on your licenses
  - One full day **free-training**


Q&A

the REUSE company

Margarita Salas, 16 2nd Floor
Innovation Center
LEGATEC Technology Park
28919 Leganés – Madrid
SPAIN – EU

Tel: (+34) 912 17 25 96
Fax: (+34) 916 80 98 26

http://www.reusecompany.com

@ReuseCompany

contact@reusecompany.com