Digital Engineering using IBM ELM

The Digital Thread - Traceability across the Lifecycle

Peter Schedl
Program Manager
IBM Engineering Lifecycle Management
peter.schedl@de.ibm.com

Jan Jancar
Solution Director
jan.jancar@softacus.com

Agenda

Why is Traceability Needed ?

Digital Thread offers Traceability

IBM Engineering Lifecycle Management (ELM) implements the Digital Thread

Traceability for Model-based Systems Engineering (MBSE)

Industry Examples

Summary

Market dynamic

Challenge

Software

Increasing importance

Connectedness/ intelligence

Transforming products

Competition

Global marketplace, demanding clients

Effect

- Importance of industry standards
- Quantity & complexity of requirements
- Agility, collaboration, traceability of environments
- Number, depth, and continuous testing
- Exponentially growing lines of code

Solution

Digital Engineering

Industry vision: Digital Engineering

Shifts from document centric to digital representations (aka “models”)

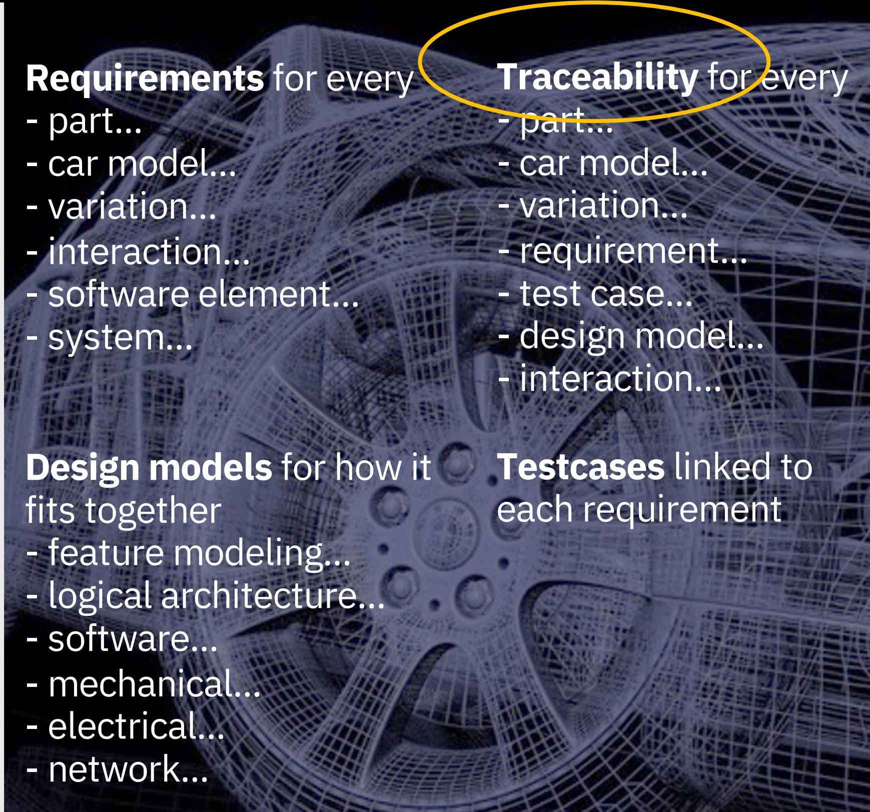
Ensures **data consistency** validity by managing “trusted” data sources

Facilitates **digital continuity** across providers to form lifecycle information models via digital threads

Enables **cross lifecycle digital viewpoints** to support the necessary insights from the data

Enables **data exchange** across domains and providers to foster collaboration, data consistency and automation

Improves productivity by adopting a **digital process** with full transparency of planned and performed activity integrated with the data



Requirements for every
- part…
- car model…
- variation…
- interaction…
- software element…
- system…

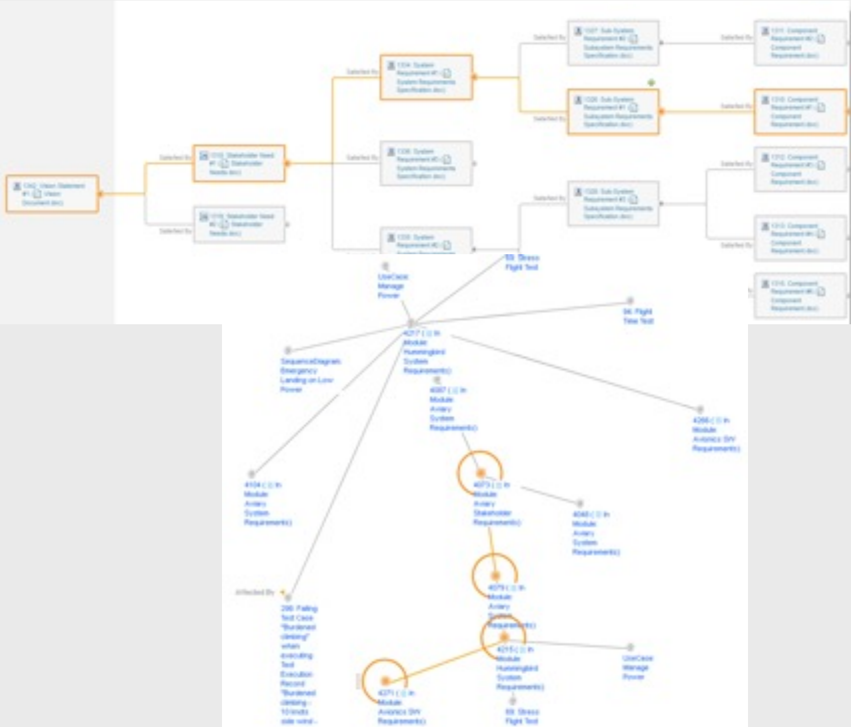
Traceability for every
- part…
- car model…
- variation…
- requirement…
- test case…
- design model…
- interaction…

Design models for how it fits together
- feature modeling…
- logical architecture…
- software…
- mechanical…
- electrical…
- network…

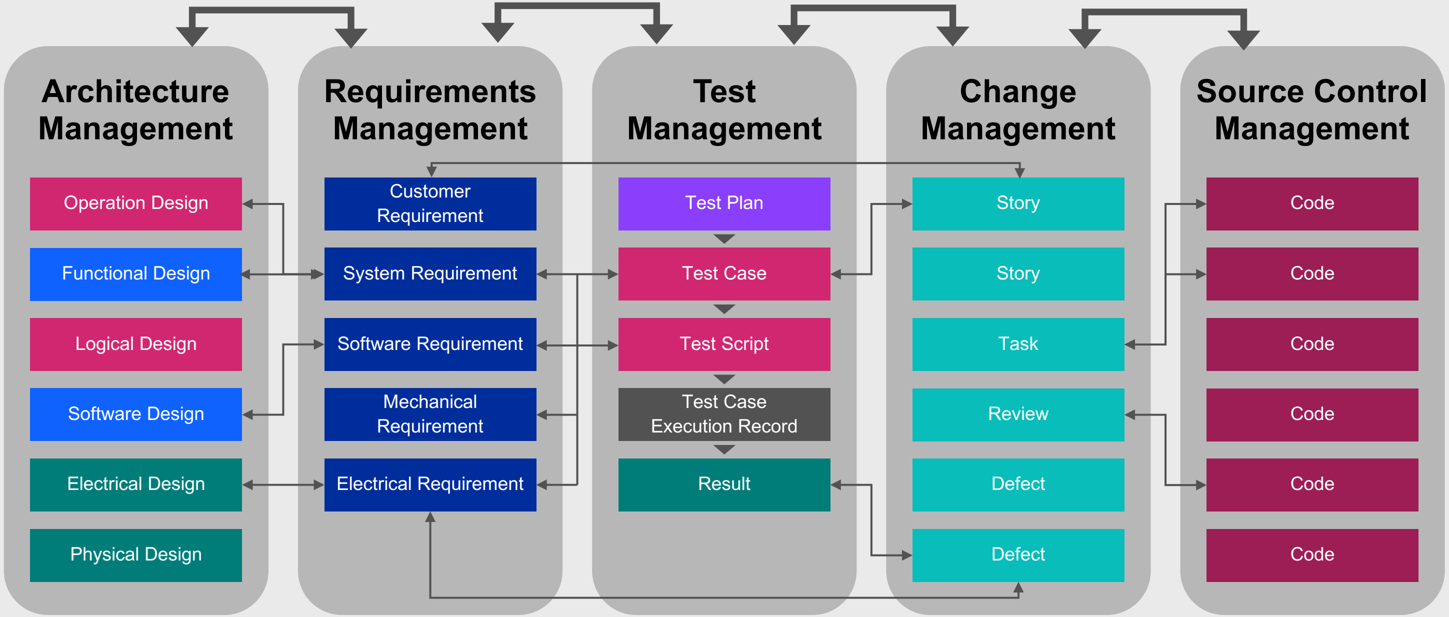
Testcases linked to each requirement

A Note about Traceability: Why?

- Impact Analysis
  - What is the impact of this requirements change?
  - Where in the design is this requirement met?
- Design Justification
  - What requirements does this design element satisfy?
  - Does the design support all the requirements?
- Verification Completeness
  - What test cases verify this requirement?
  - What requirements do this test case verify?
- Project Status
  - How many requirements are realized?
  - How many requirements have been verified?



The multi-dimensional Complexity Challenge Facing Customers Today



Architecture Management

- Operation Design
- Functional Design
- Logical Design
- Software Design
- Electrical Design
- Physical Design

Requirements Management

- Customer Requirement
- System Requirement
- Software Requirement
- Mechanical Requirement
- Electrical Requirement

Test Management

- Test Plan
- Test Case
- Test Script
- Test Case Execution Record
- Result

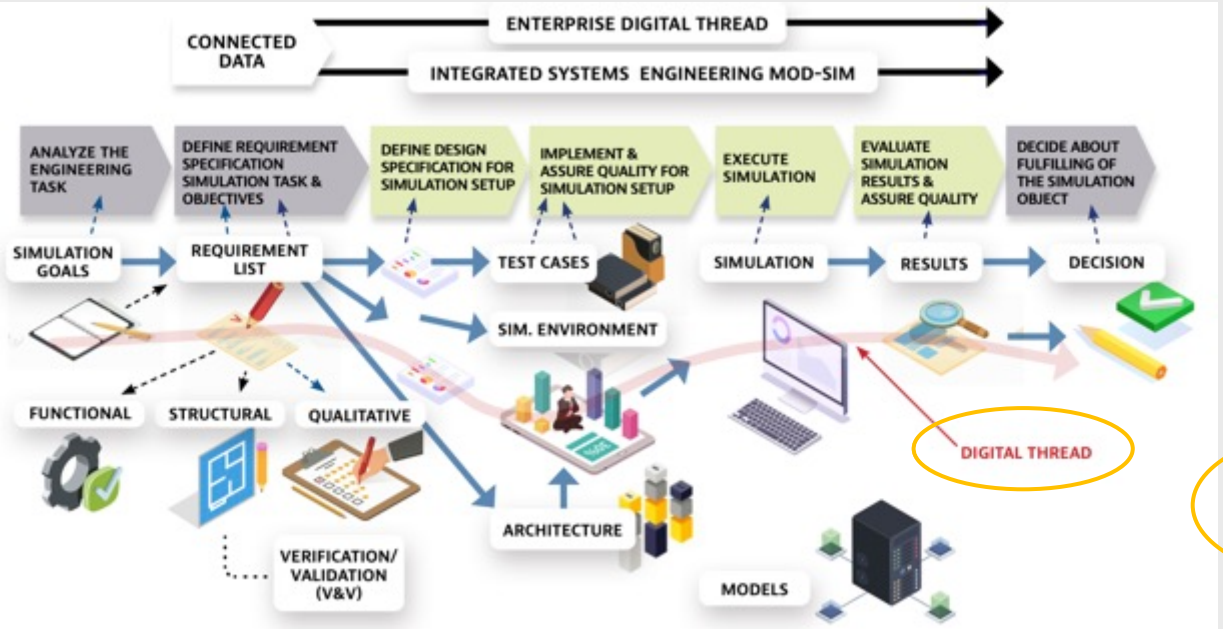
Change Management

- Story
- Story
- Task
- Review
- Defect
- Defect

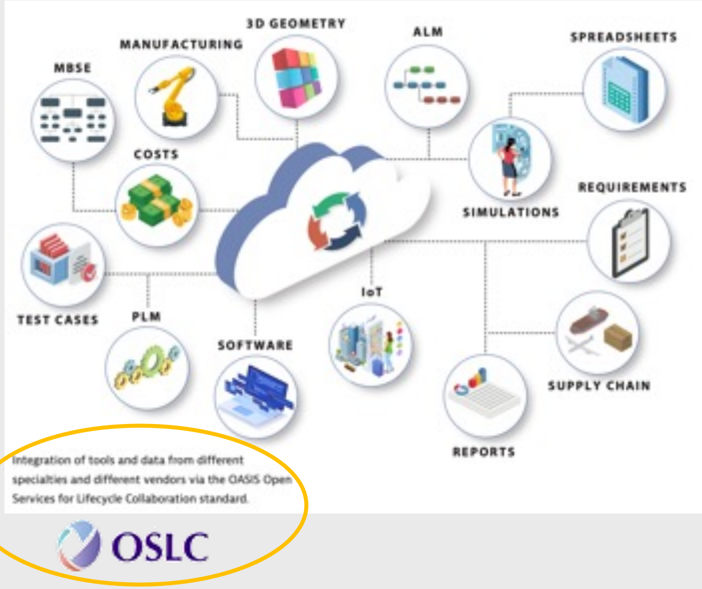
Source Control Management

- Code
- Code
- Code
- Code
- Code
- Code

How will we develop products in the future – INCOSE Vision 2035


CONNECTED DATA
ENTERPRISE DIGITAL THREAD
INTEGRATED SYSTEMS ENGINEERING MOD-SIM
ANALYZE THE ENGINEERING TASK
DEFINE REQUIREMENT SPECIFICATION SIMULATION TASK & OBJECTIVES
DEFINE DESIGN SPECIFICATION FOR SIMULATION SETUP
IMPLEMENT & ASSURE QUALITY FOR SIMULATION SETUP
EXECUTE SIMULATION
EVALUATE SIMULATION RESULTS & ASSURE QUALITY
DECIDE ABOUT FULFILLING OF THE SIMULATION OBJECT
SIMULATION GOALS
REQUIREMENT LIST
TEST CASES
SIMULATION
RESULTS
DECISION
SIM. ENVIRONMENT
FUNCTIONAL
STRUCTURAL
QUALITATIVE
DIGITAL THREAD
ARCHITECTURE
VERIFICATION/ VALIDATION (V&V)
MODELS

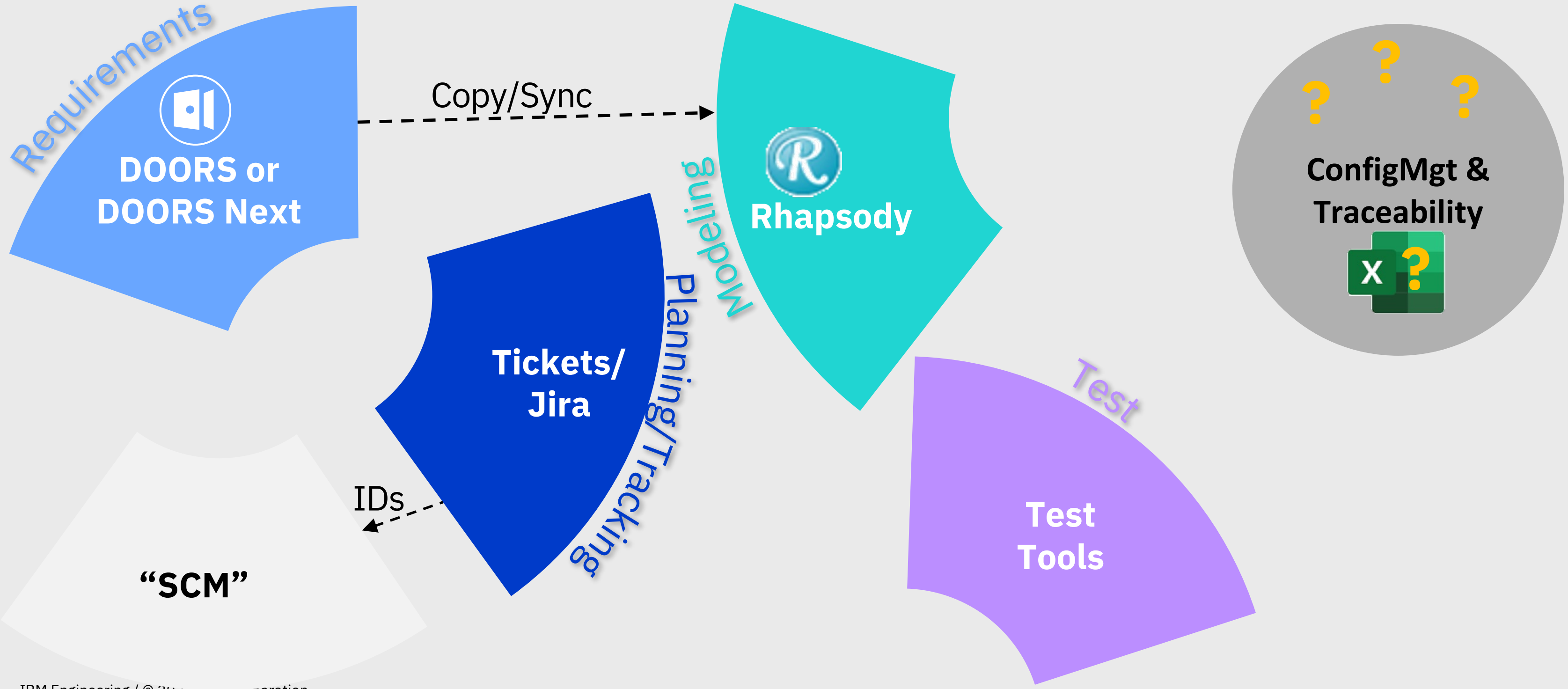
Tools & Data Integration



Integration of tools and data from different specialties and different vendors via the OASIS Open Services for Lifecycle Collaboration standard.

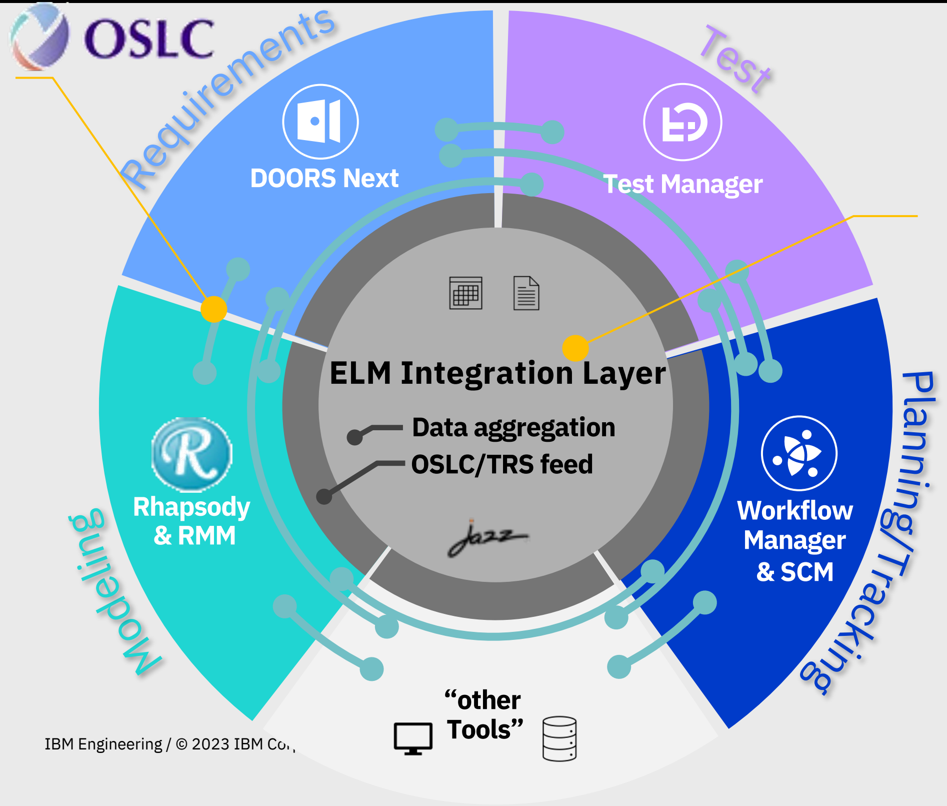
https://www.incose.org/about-systems-engineering/se-vision-2035

Product Development Example - Where Are You Today?



Multiple Tools (UIs, Configs, Permissions,…) often no or 'manual' Integration

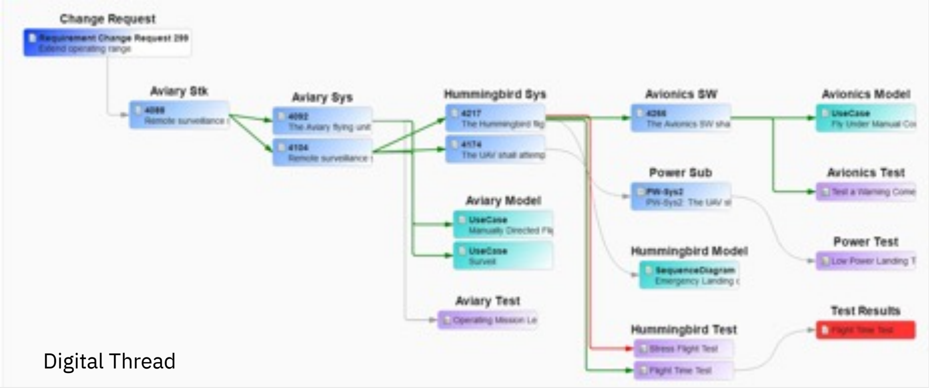
IBM Engineering Lifecycle Management (ELM) Integration Layer for Single Source of Truth enables the Digital Thread



Digital Thread is a communication framework that connects traditionally siloed elements in product development processes and provides an integrated view of a product throughout development

Cross domain integration layer

- Single source of truth data layer visualizes the **Digital Thread** (Versioned data w relations)



Digital Thread

Visibility across the Engineering Lifecycle via a Digital Thread provides multiple Benefits

Traceability

The bi-directional data in the digital thread will enable functional, software, mechanical, and electrical engineering domains with a single source of truth.

Impact Analysis

Iterations are costly. Ensure new features will be delivered on time and all sub-systems and vendors have accepted responsibility to reduce integration failures and delivery delays.

Compliance

Compliance and regulatory reporting are changing. Ensure you have full visibility into the lifecycle so you can easily produce reports to track progress towards compliance. Safety built in from the start.

Project Tracking & Collaboration

Real time insights to the status of the project through programmatic reports. Collaborate using a single system with process flows through the engineering workstream.

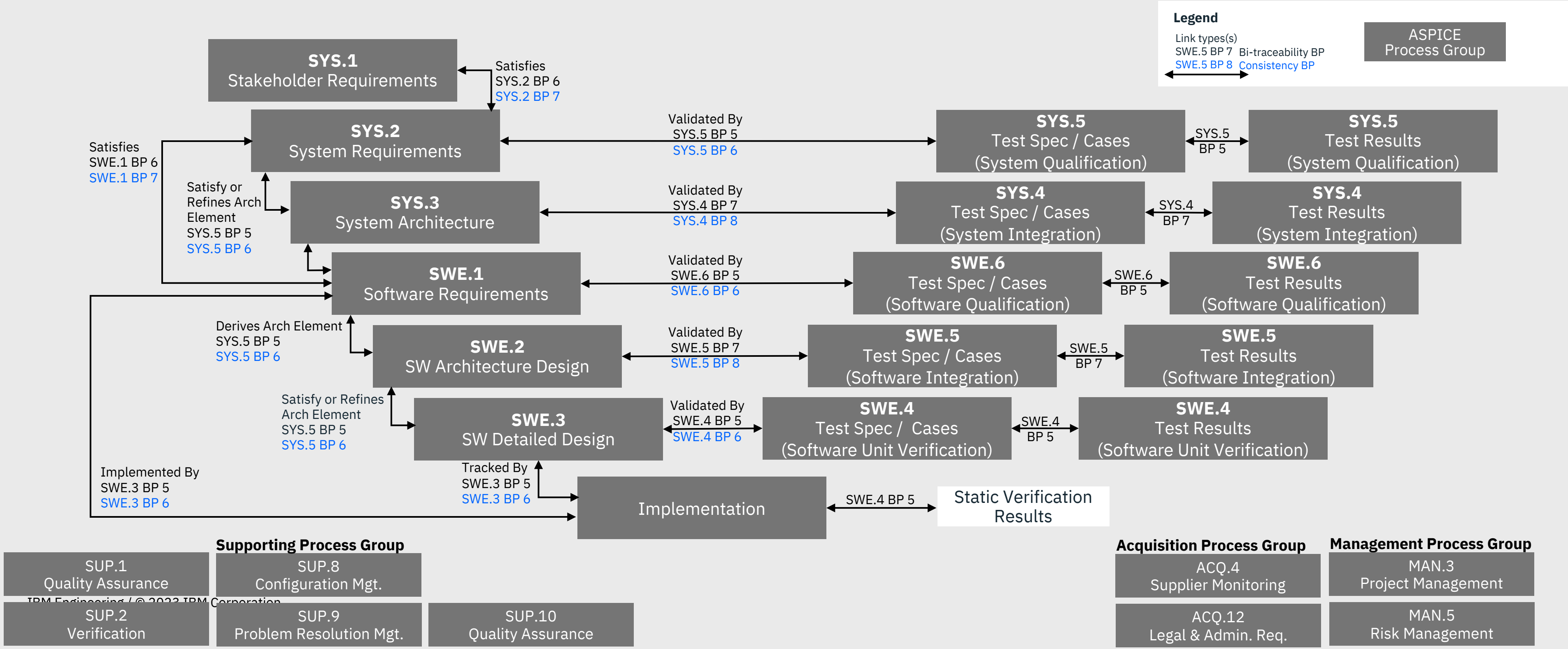
Strategic Reuse

Support product line variability and reuse through an integrated view of product targets and requirements that enable product level verification.
Take advantage of reuse for rapid innovation.

Agile SW Practices

Agile SW Development cycles needs to be integrated in the product lifecycle in both directions: Connection to Systems Engineering as well as towards CI/CD deployments.

Automotive SPICE (ASPICE) Traceability Requirements



Legend

Link types(s)

SWE.5 BP 7 Bi-traceability BP

SWE.5 BP 8 Consistency BP

ASPICE Process Group

SYS.1
Stakeholder Requirements

Satisfies
SYS.2 BP 6
SYS.2 BP 7

SYS.2
System Requirements

Satisfies
SWE.1 BP 6
SWE.1 BP 7

Validated By
SYS.5 BP 5
SYS.5 BP 6

SYS.5
Test Spec / Cases
(System Qualification)

SYS.5
BP 5

SYS.5
Test Results
(System Qualification)

Satisfy or
Refines Arch
Element
SYS.5 BP 5
SYS.5 BP 6

SYS.3
System Architecture

Validated By
SYS.4 BP 7
SYS.4 BP 8

SYS.4
Test Spec / Cases
(System Integration)

SYS.4
BP 7

SYS.4
Test Results
(System Integration)

SWE.1
Software Requirements

Validated By
SWE.6 BP 5
SWE.6 BP 6

SWE.6
Test Spec / Cases
(Software Qualification)

SWE.6
BP 5

SWE.6
Test Results
(Software Qualification)

Derives Arch Element
SYS.5 BP 5
SYS.5 BP 6

SWE.2
SW Architecture Design

Validated By
SWE.5 BP 7
SWE.5 BP 8

SWE.5
Test Spec / Cases
(Software Integration)

SWE.5
BP 7

SWE.5
Test Results
(Software Integration)

Satisfy or Refines
Arch Element
SYS.5 BP 5
SYS.5 BP 6

SWE.3
SW Detailed Design

Validated By
SWE.4 BP 5
SWE.4 BP 6

SWE.4
Test Spec / Cases
(Software Unit Verification)

SWE.4
BP 5

SWE.4
Test Results
(Software Unit Verification)

Implemented By
SWE.3 BP 5
SWE.3 BP 6

Tracked By
SWE.3 BP 5
SWE.3 BP 6

Implementation

SWE.4 BP 5

Static Verification
Results

Supporting Process Group

SUP.1
Quality Assurance

SUP.8
Configuration Mgt.

SUP.2
Verification

SUP.9
Problem Resolution Mgt.

SUP.10
Quality Assurance

Acquisition Process Group

ACQ.4
Supplier Monitoring

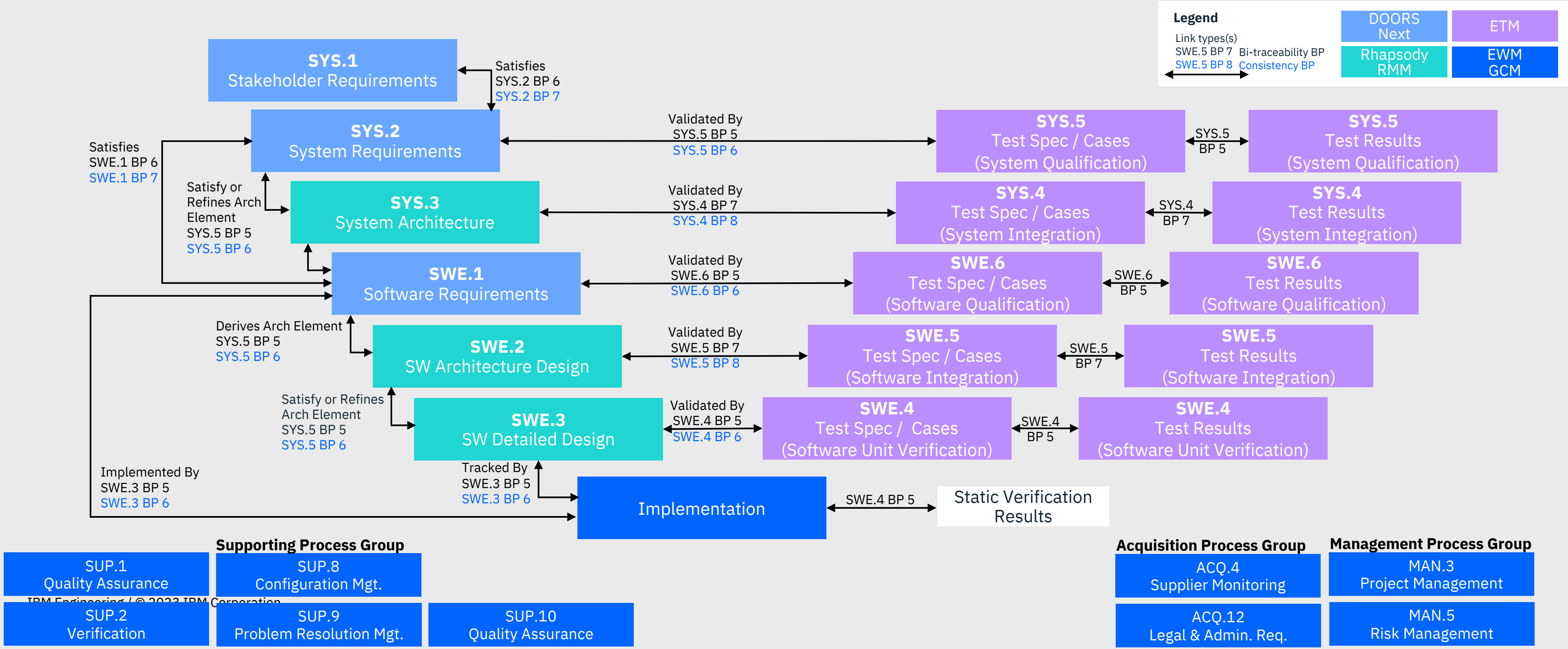
ACQ.12
Legal & Admin. Req.

Management Process Group

MAN.3
Project Management

MAN.5
Risk Management

Automotive SPICE (ASPICE) Traceability Requirements – IBM ELM Coverage



Legend

Link types(s)

SWE.5 BP 7 Bi-traceability BP

SWE.5 BP 8 Consistency BP

DOORS Next

ETM

Rhapsody RMM

EWM GCM

SYS.1 Stakeholder Requirements

Satisfies SYS.2 BP 6 SYS.2 BP 7

SYS.2 System Requirements

Validated By SYS.5 BP 5 SYS.5 BP 6

SYS.5 Test Spec / Cases (System Qualification)

SYS.5 BP 5

SYS.5 Test Results (System Qualification)

Satisfies SWE.1 BP 6 SWE.1 BP 7

Satisfy or Refines Arch Element SYS.5 BP 5 SYS.5 BP 6

SYS.3 System Architecture

Validated By SYS.4 BP 7 SYS.4 BP 8

SYS.4 Test Spec / Cases (System Integration)

SYS.4 BP 7

SYS.4 Test Results (System Integration)

SWE.1 Software Requirements

Validated By SWE.6 BP 5 SWE.6 BP 6

SWE.6 Test Spec / Cases (Software Qualification)

SWE.6 BP 5

SWE.6 Test Results (Software Qualification)

Derives Arch Element SYS.5 BP 5 SYS.5 BP 6

SWE.2 SW Architecture Design

Validated By SWE.5 BP 7 SWE.5 BP 8

SWE.5 Test Spec / Cases (Software Integration)

SWE.5 BP 7

SWE.5 Test Results (Software Integration)

Satisfy or Refines Arch Element SYS.5 BP 5 SYS.5 BP 6

SWE.3 SW Detailed Design

Validated By SWE.4 BP 5 SWE.4 BP 6

SWE.4 Test Spec / Cases (Software Unit Verification)

SWE.4 BP 5

SWE.4 Test Results (Software Unit Verification)

Implemented By SWE.3 BP 5 SWE.3 BP 6

Tracked By SWE.3 BP 5 SWE.3 BP 6

Implementation

SWE.4 BP 5

Static Verification Results

Supporting Process Group

SUP.1 Quality Assurance

SUP.8 Configuration Mgt.

SUP.2 Verification

SUP.9 Problem Resolution Mgt.

SUP.10 Quality Assurance

Acquisition Process Group

ACQ.4 Supplier Monitoring

ACQ.12 Legal & Admin. Req.

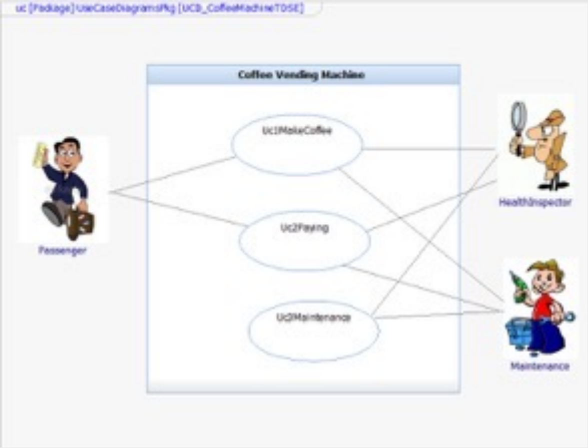
Management Process Group

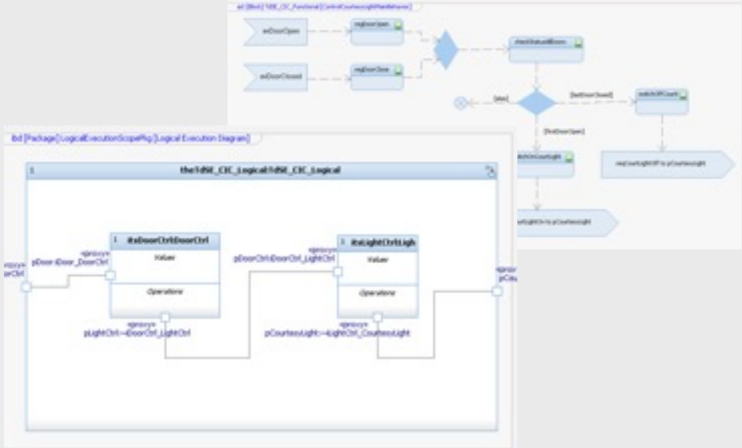
MAN.3 Project Management

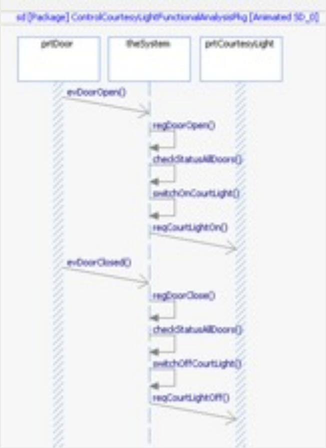
MAN.5 Risk Management

Model Based Systems Engineering

Model Based Systems Engineering (MBSE) complements traditional requirements definition and management techniques



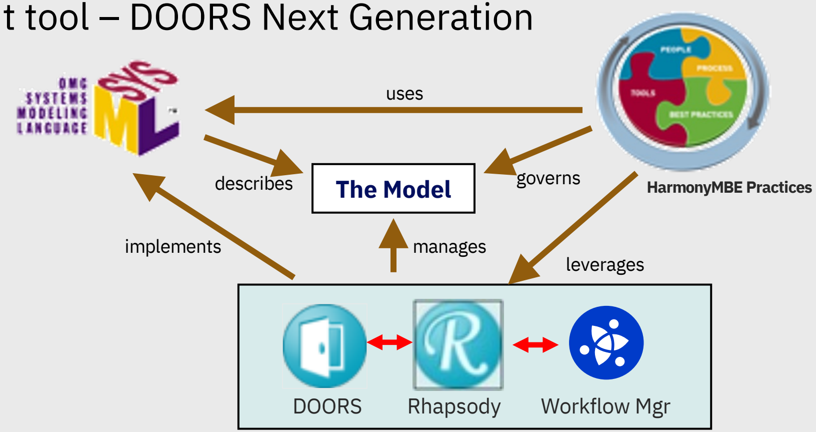

evDoorOpen
evDoorClosed
[else]
ibd [Package] LogicalExecutionScopePkg [Logical Execution Diagram]
theTdSE_CIC_Logical:TdSE_CIC_Logical
itsDoorCtrl:DoorCtrl
itsLightCtrl:LightCtrl
Values
Operations
pDoor:iDoor_DoorCtrl
pDoorCtrl:iDoorCtrl_LightCtrl
pLightCtrl:~iDoorCtrl_LightCtrl
pCourtesyLight:~iLightCtrl_CourtesyLight



IBM’s Model Based Systems Engineering (MBSE) Solution

MBSE is a standards based Systems Engineering practice that incorporates:

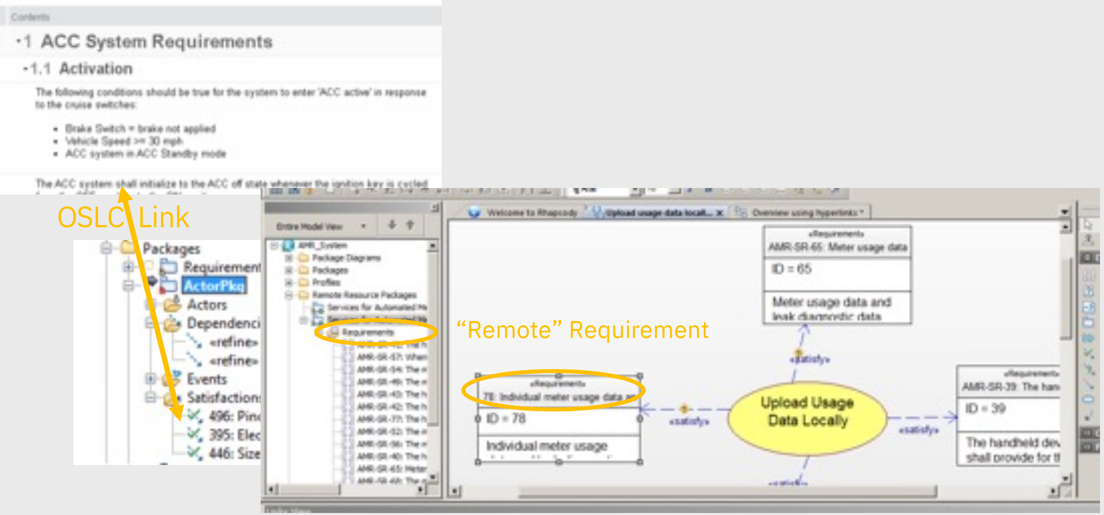
- Modeling language – SysML
- Modeling method – Harmony Systems Engineering Practices incl. Ticket System for Guidance & Com.
- Modeling tool – Rhapsody for Systems Engineers & Rhapsody Model Manager
- Requirements management tool – DOORS Next Generation



Models and external data like Requirements need to be integrated

Traceability to Requirements, Test Cases, Work Items,… using OSLC

Requirements Tool – DOORS Next



Contents

- 1 ACC System Requirements
- 1.1 Activation

The following conditions should be true for the system to enter 'ACC active' in response to the cruise switches:

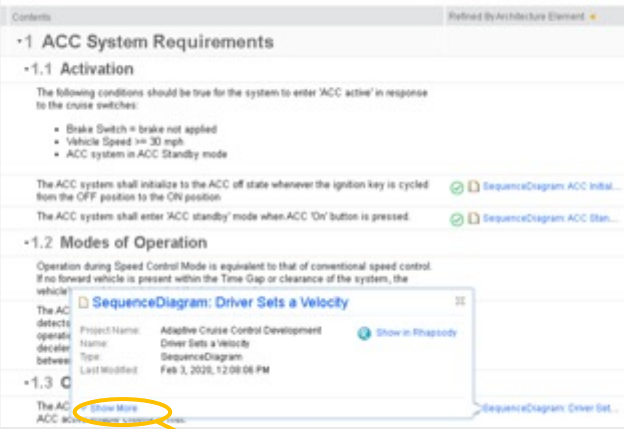
- Brake Switch = brake not applied
- Vehicle Speed >= 30 mph
- ACC system in ACC Standby mode

The ACC system shall initialize to the ACC off state whenever the ignition key is cycled

OSLC Link

“Remote” Requirement

Modeling Tool – Rhapsody w RMM



Contents | Refined By Architecture Element

- 1 ACC System Requirements
- 1.1 Activation

The following conditions should be true for the system to enter 'ACC active' in response to the cruise switches:

- Brake Switch = brake not applied
- Vehicle Speed >= 30 mph
- ACC system in ACC Standby mode

The ACC system shall initialize to the ACC off state whenever the ignition key is cycled from the OFF position to the ON position — SequenceDiagram: ACC Initial...

The ACC system shall enter 'ACC standby' mode when ACC 'On' button is pressed. — SequenceDiagram: ACC Stan...

- 1.2 Modes of Operation

Operation during Speed Control Mode is equivalent to that of conventional speed control. If no forward vehicle is present within the Time Gap or clearance of the system, the vehicle

The AC detects operati deceler betwee

- 1.3 C

The AC ACC ac — SequenceDiagram: Driver Set...

SequenceDiagram: Driver Sets a Velocity

Project Name: Adaptive Cruise Control Development

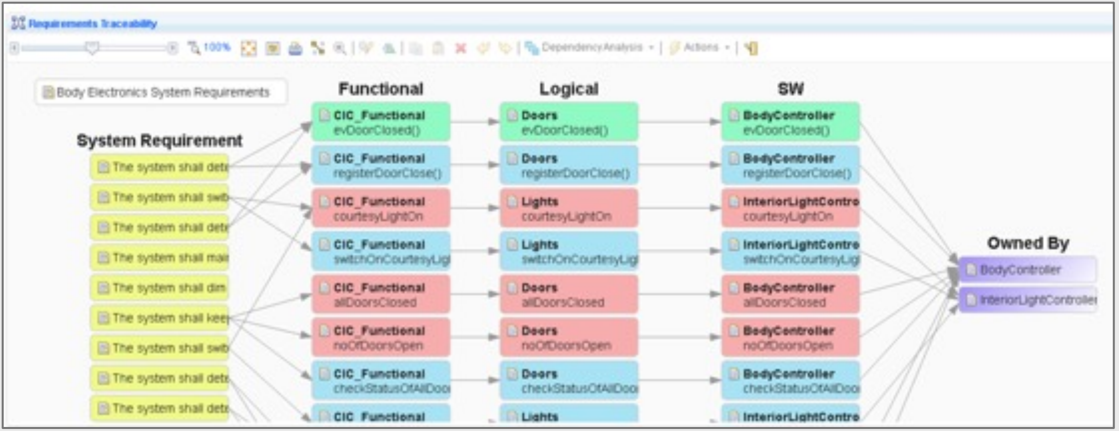
Name: Driver Sets a Velocity

Type: SequenceDiagram

Last Modified: Feb 3, 2020, 12:08:06 PM

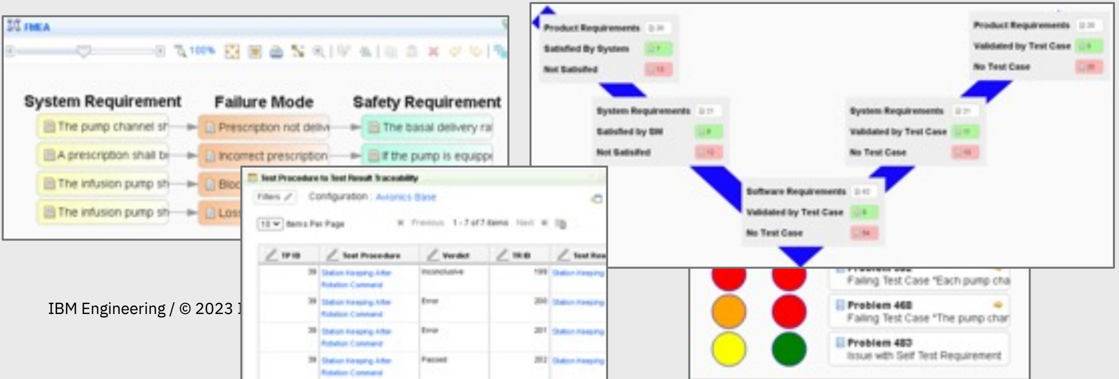
SequenceDiagram: Driver Sets a Velocity

Visualizing the Digital Thread


Requirements Traceability
100%
Dependency Analysis
Actions
Body Electronics System Requirements
Functional
Logical
SW
System Requirement
The system shall dete
The system shall swit
The system shall dete
The system shall main
The system shall dim
The system shall keep
The system shall swit
The system shall dete
The system shall dete
CIC_Functional evDoorClosed()
CIC_Functional registerDoorClose()
CIC_Functional courtesyLightOn
CIC_Functional switchOnCourtesyLig
CIC_Functional allDoorsClosed
CIC_Functional noOfDoorsOpen
CIC_Functional checkStatusOfAllDoo
CIC_Functional
Doors evDoorClosed()
Doors registerDoorClose()
Lights courtesyLightOn
Lights switchOnCourtesyLig
Doors allDoorsClosed
Doors noOfDoorsOpen
Doors checkStatusOfAllDoo
Lights
BodyController evDoorClosed()
BodyController registerDoorClose()
InteriorLightContro courtesyLightOn
InteriorLightContro switchOnCourtesyLig
BodyController allDoorsClosed
BodyController noOfDoorsOpen
BodyController checkStatusOfAllDoo
InteriorLightContro
Owned By
BodyController
InteriorLightController

System Requirements traced to Model Layers:

– Functional Architecture
– Logical Architecture
– SW Architecture
– Physical Architecture



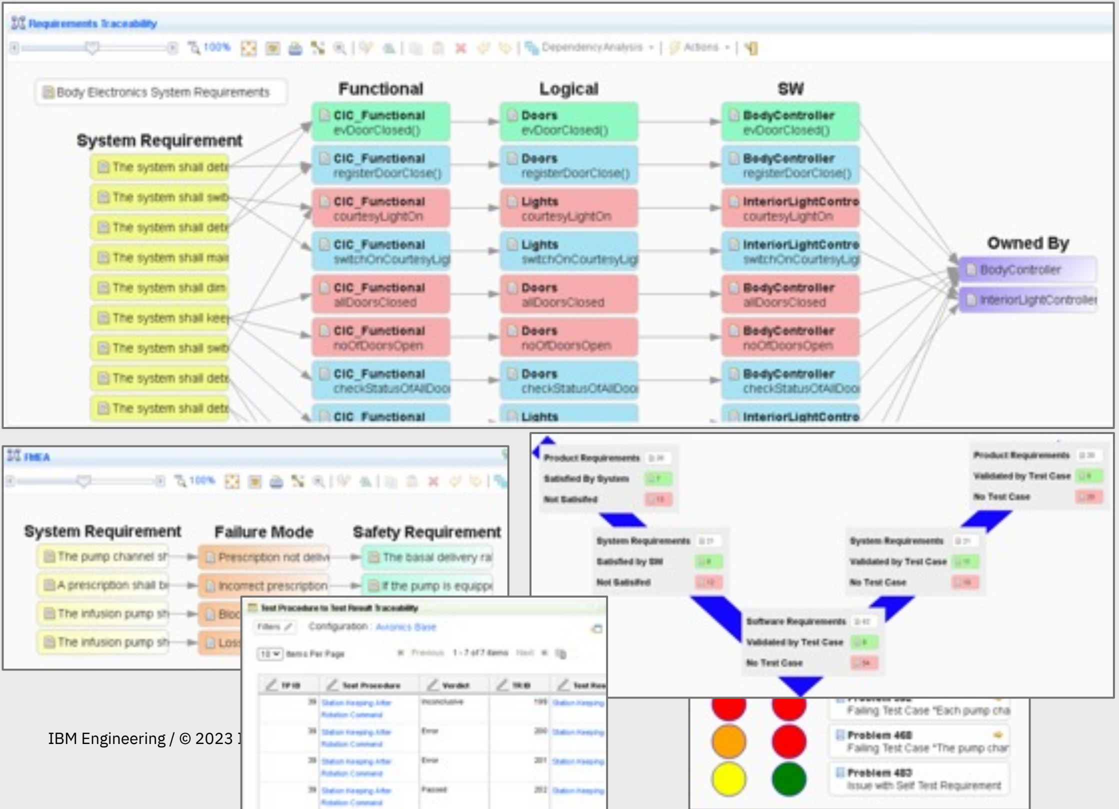
Status & Progress (KPI) Tracking

Customizable Dashboards

Real-time metrics & Reports

Impact & Gap Analysis

Visualizing the Digital Thread & Document Generation



Requirements Traceability

Body Electronics System Requirements

System Requirement	Functional	Logical	SW
The system shall dete	CIC_Functional evDoorClosed()	Doors evDoorClosed()	BodyController evDoorClosed()
The system shall swit	CIC_Functional registerDoorClose()	Doors registerDoorClose()	BodyController registerDoorClose()
The system shall dete	CIC_Functional courtesyLightOn	Lights courtesyLightOn	InteriorLightContro courtesyLightOn
The system shall mai	CIC_Functional switchOnCourtesyLig	Lights switchOnCourtesyLig	InteriorLightContro switchOnCourtesyLig
The system shall dim	CIC_Functional allDoorsClosed	Doors allDoorsClosed	BodyController allDoorsClosed
The system shall kee	CIC_Functional noOfDoorsOpen	Doors noOfDoorsOpen	BodyController noOfDoorsOpen
The system shall swit	CIC_Functional checkStatusOfAllDoo	Doors checkStatusOfAllDoo	BodyController checkStatusOfAllDoo
The system shall dete	CIC_Functional	Lights	InteriorLightContro
The system shall dete			

Owned By: BodyController, InteriorLightController

FMEA

System Requirement	Failure Mode	Safety Requirement
The pump channel sh	Prescription not deliv	The basal delivery ra
A prescription shall b	Incorrect prescription	If the pump is equippe
The infusion pump sh	Bloc	
The infusion pump sh	Loss	

Product Requirements	
Satisfied By System	7
Not Satisfied	13

System Requirements	
Satisfied by SW	9
Not Satisfied	12

Software Requirements	
Validated by Test Case	8
No Test Case	34

System Requirements	
Validated by Test Case	[illegible]
No Test Case	[illegible]

Product Requirements	
Validated by Test Case	[illegible]
No Test Case	[illegible]

Test Procedure to Test Result Traceability

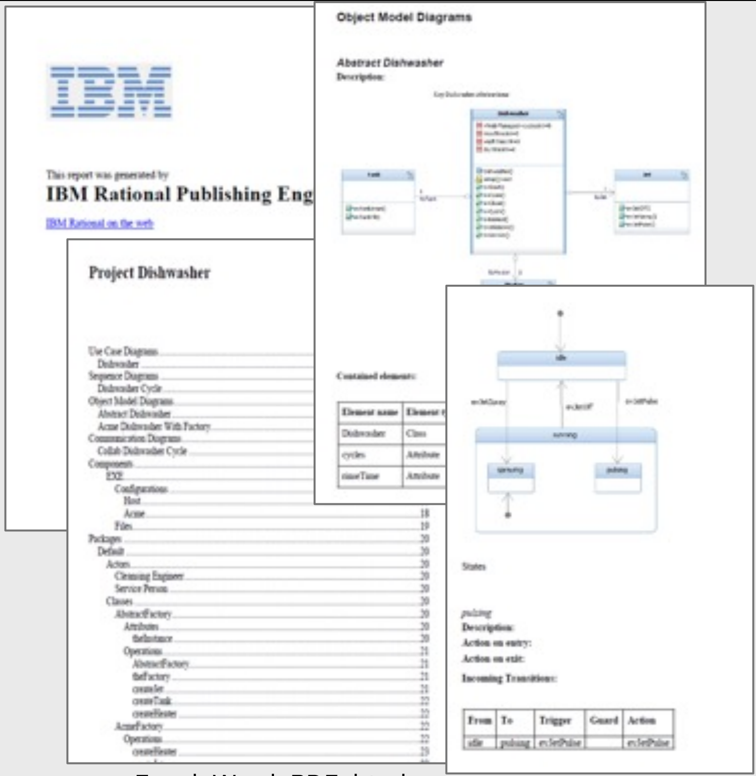
Configuration: Avionics Base

Problem 468
Failing Test Case "The pump char

Problem 483
Issue with Self Test Requirement

Failing Test Case "Each pump cha

IBM Engineering / © 2023 I



This report was generated by

IBM Rational Publishing Eng

IBM Rational on the web

Project Dishwasher

Use Case Diagrams
Dishwasher
Sequence Diagrams
Dishwasher Cycle
Object Model Diagrams
Abstract Dishwasher
Acme Dishwasher With Factory
Communication Diagrams
Collab Dishwasher Cycle
Components
EXE
Configurations
Host
Acme 18
Files 19
Packages 20
Default 20
Actors 20
Cleaning Engineer 20
Service Person 20
Classes 20
AbstractFactory 20
Attributes 20
theInstance 20
Operations 21
AbstractFactory 21
theFactory 21
createJet 21
createTank 22
createHeater 22
AcmeFactory 22
Operations 22
createHeater 23

Object Model Diagrams

Abstract Dishwasher

Description:

Contained elements:

Element name	Element t
Dishwasher	Class
cycles	Attribute
rinseTime	Attribute

States

pulsing

Description:

Action on entry:

Action on exit:

Incoming Transitions:

From	To	Trigger	Guard	Action
idle	pulsing	evJetPulse		evJetPulse

Excel, Word, PDF, html, …

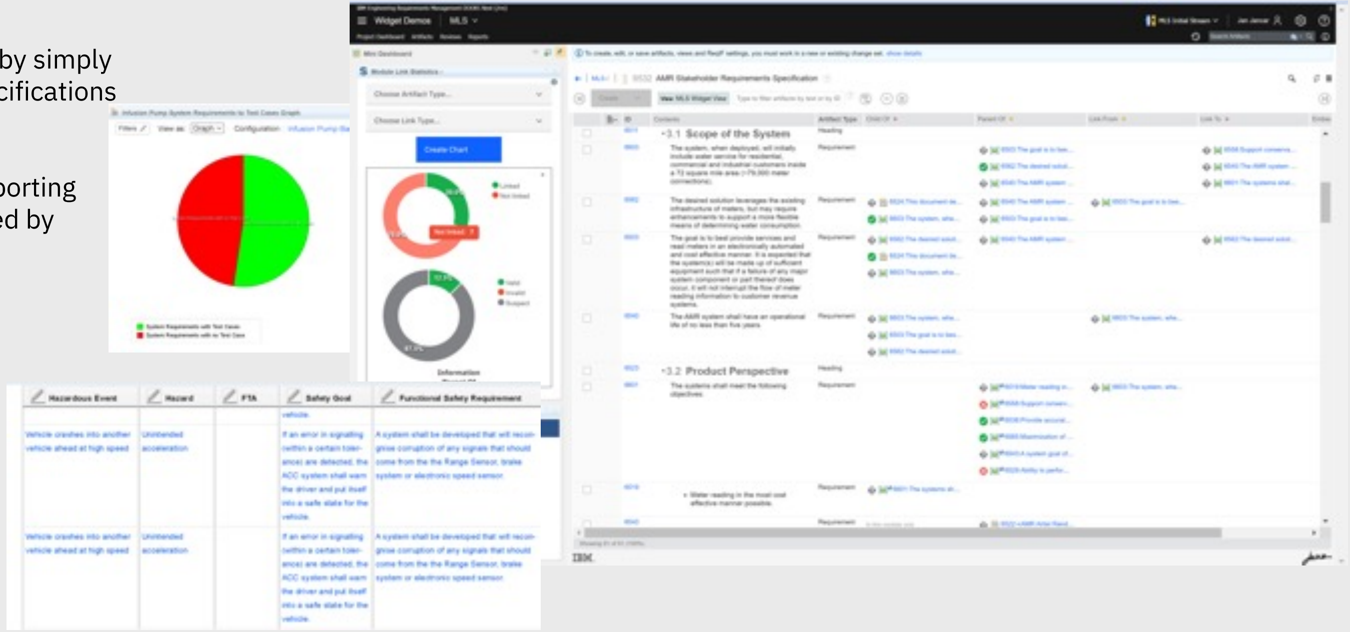
Project Example – Calculation of Traceability and Link Validity

Links KPI and Reports

Check **data treacebility metrics** by simply dynamically “browsing” over specifications

Report on **traceability KPI** in Reporting application. Requirements covered by Test Cases

Tabular, Pie Chart Reports & Live Excel Reports



3.1 Scope of the System — Heading

The system, when deployed, will initially include water service for residential, commercial and industrial customers inside a 72 square mile area (~79,000 meter connections). — Requirement

The desired solution leverages the existing infrastructure of meters, but may require enhancements to support a more flexible means of determining water consumption. — Requirement

The goal is to best provide services and read meters in an electronically automated and cost effective manner. It is expected that the system(s) will be made up of sufficient equipment such that if a failure of any major system component or part thereof does occur, it will not interrupt the flow of meter reading information to customer revenue systems. — Requirement

The AMR system shall have an operational life of no less than five years. — Requirement

3.2 Product Perspective — Heading

The systems shall meet the following objectives. — Requirement

Hazardous Event	Hazard	FTA	Safety Goal	Functional Safety Requirement
			vehicle.	
Vehicle crashes into another vehicle ahead at high speed	Unintended acceleration		If an error in signaling (within a certain tolerance) are detected, the ACC system shall warn the driver and put itself into a safe state for the vehicle.	A system shall be developed that will recongnise corruption of any signals that should come from the the Range Sensor, brake system or electronic speed sensor.
Vehicle crashes into another vehicle ahead at high speed	Unintended acceleration		If an error in signaling (within a certain tolerance) are detected, the ACC system shall warn the driver and put itself into a safe state for the vehicle.	A system shall be developed that will recongnise corruption of any signals that should come from the the Range Sensor, brake system or electronic speed sensor.

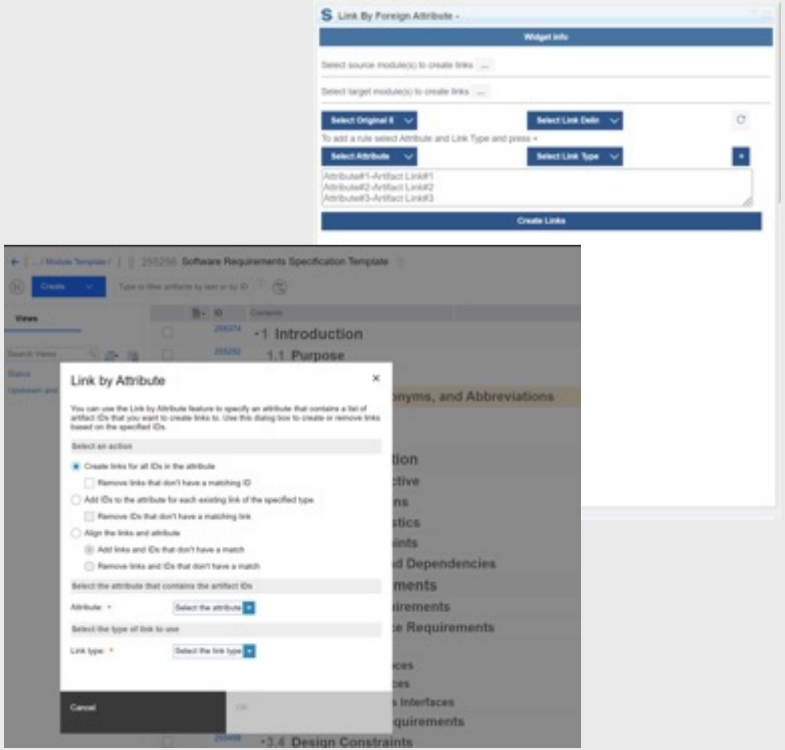
Traceability Recreation during Migration

Reestablishing of traceability

By Migration from DOORS to DOORS Next – ReqIF, Migiz,
By Migration from custom tools or Office documents

Various scripts, DOORS migrator and OOTB functionality

- Link by Foreign Attribute
- Link by Attribute
- Link Matrix only



Link By Foreign Attribute -

Widget info

Select source module(s) to create links ...

Select target module(s) to create links ...

Select Original ll | Select Link Defin

To add a rule select Attribute and Link Type and press +

Select Attribute | Select Link Type | +

Attribute#1-Artifact Link#1
Attribute#2-Artifact Link#2
Attribute#3-Artifact Link#3

Create Links

Link by Attribute

You can use the Link by Attribute feature to specify an attribute that contains a list of artifact IDs that you want to create links to. Use this dialog box to create or remove links based on the specified IDs.

Select an action

● Create links for all IDs in the attribute
  □ Remove links that don't have a matching ID
○ Add IDs to the attribute for each existing link of the specified type
  □ Remove IDs that don't have a matching link
○ Align the links and attribute
  ○ Add links and IDs that don't have a match
  ○ Remove links and IDs that don't have a match

Select the attribute that contains the artifact IDs

Attribute: * Select the attribute

Select the type of link to use

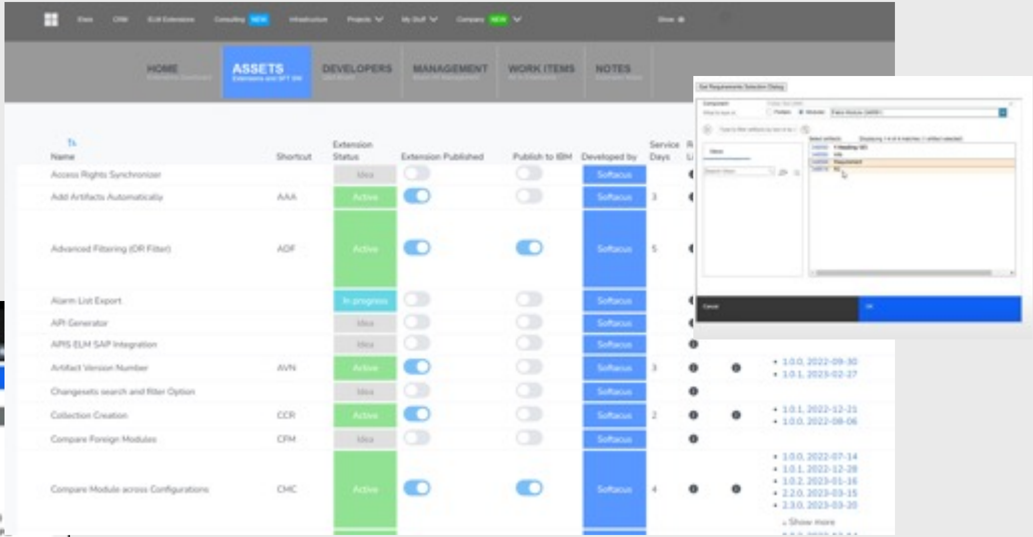
Link type: * Select the link type

Cancel | OK

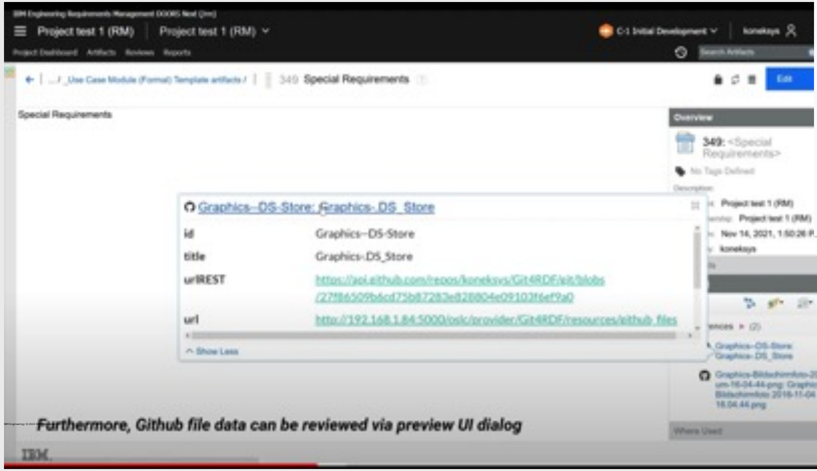
OSLC Traceability Generation

Traceability outside IBM

OSLC Links to custom own applications
OSLC Links PTC RVS – DOORS Next (in progress)
OSLC Links GIT Documents – DOORS Next Requirements



Name	Shortcut	Extension Status	Extension Published	Publish to IBM	Developed by	Service Days	R Li		
Access Rights Synchronizer		Idea	○	○	Softacus		●		
Add Artifacts Automatically	AAA	Active	●	○	Softacus	3	●		
Advanced Filtering (OR Filter)	AOF	Active	●	●	Softacus	5	●		
Alarm List Export		In progress	○	○	Softacus		●		
API Generator		Idea	○	○	Softacus		●		
APIS ELM SAP Integration		Idea	○	○	Softacus		●		
Artifact Version Number	AVN	Active	●	○	Softacus	3	●	●	1.0.0, 2022-09-30 1.0.1, 2023-02-27
Changesets search and filter Option		Idea	○	○	Softacus		●		
Collection Creation	CCR	Active	●	○	Softacus	2	●	●	1.0.1, 2022-12-21 1.0.0, 2022-08-06
Compare Foreign Modules	CFM	Idea	○	○	Softacus		●		
Compare Module across Configurations	CMC	Active	●	●	Softacus	4	●	●	1.0.0, 2022-07-14 1.0.1, 2022-12-28 1.0.2, 2023-01-16 2.2.0, 2023-03-15 2.3.0, 2023-03-20 Show more



.../ _Use Case Module (Formal) Template artifacts / | 349 Special Requirements

Special Requirements

349: <Special Requirements>
No Tags Defined
Description
Project test 1 (RM)
Project test 1 (RM)
Nov 14, 2021, 1:50:26 P...
koneksys

Graphics--DS-Store: Graphics-.DS_Store

id	Graphics--DS-Store
title	Graphics-.DS_Store
urlREST	https://api.github.com/repos/koneksys/Git4RDF/git/blobs/27f86509b6cd75b87283e828804e09103f6ef9a0
url	http://192.168.1.84:5000/oslc/provider/Git4RDF/resources/github_files

...ences (2)
Graphics--DS-Store: Graphics-.DS_Store
Graphics-Bildschirmfoto-2016-11-04-um-16-04-44.png: Graphics-Bildschirmfoto 2016-11-04 um 16.04.44.png

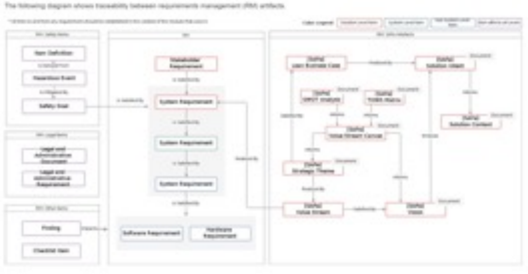
Furthermore, Github file data can be reviewed via preview UI dialog

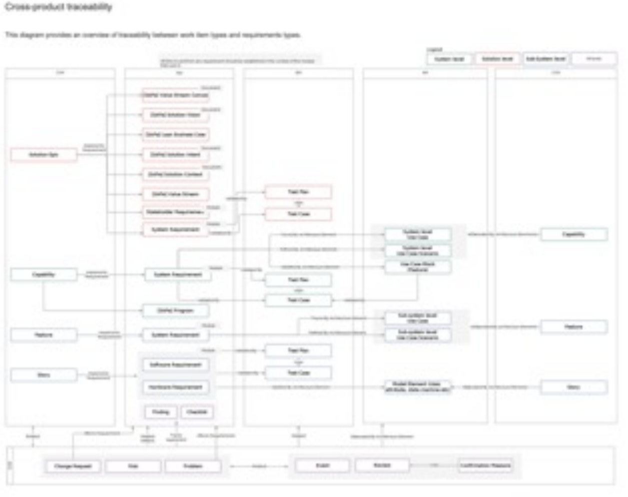
ASPICE Example

Traceability according to ASPICE

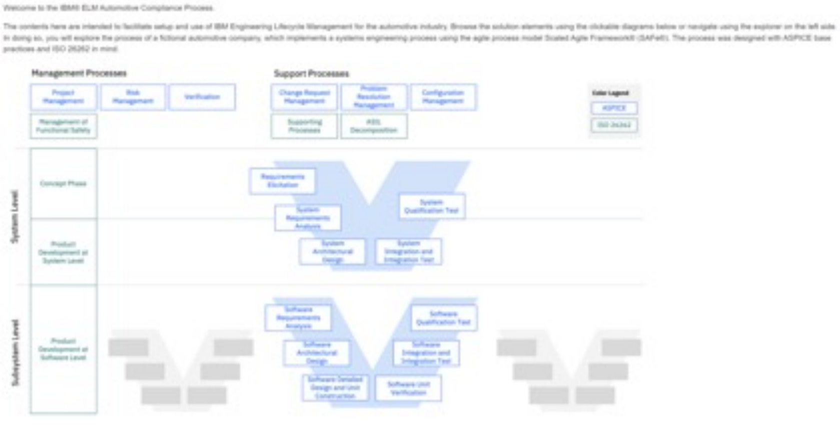
Customization of IBM ASPICE template
Cross Domain traceability
Real Live Example

Requirement – Design - Test Cases


The following diagram shows traceability between requirements management (RM) artifacts.

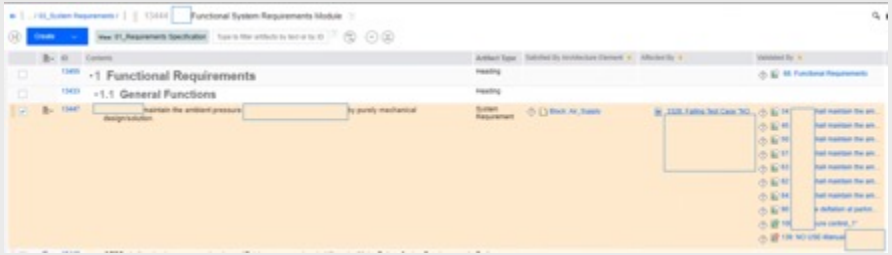

Cross-product traceability

This diagram provides an overview of traceability between work item types and requirements types.


Welcome to the IBM® ELM Automotive Compliance Process.

The contents here are intended to facilitate setup and use of IBM Engineering Lifecycle Management for the automotive industry. Browse the solution elements using the clickable diagrams below or navigate using the explorer on the left side. In doing so, you will explore the process of a fictional automotive company, which implements a systems engineering process using the agile process model Scaled Agile Framework® (SAFe®). The process was designed with ASPICE base practices and ISO 26262 in mind.

Management Processes
Project Management
Risk Management
Verification
Management of Functional Safety
Support Processes
Change Request Management
Problem Resolution Management
Configuration Management
Supporting Processes
ASIL Decomposition
Color Legend
ASPICE
ISO 26262
System Level
Concept Phase
Product Development at System Level
Requirements Elicitation
System Requirements Analysis
System Architectural Design
System Integration and Integration Test
System Qualification Test
Subsystem Level
Product Development at Software Level
Software Requirements Analysis
Software Architectural Design
Software Detailed Design and Unit Construction
Software Unit Verification
Software Integration and Integration Test
Software Qualification Test



Functional System Requirements Module
Create
View: 01_Requirements Specification
Type to filter artifacts by text or by ID
ID
Contents
Artifact Type
Satisfied By Architecture Element
Affected By
Validated By

ID	Contents	Artifact Type	Satisfied By Architecture Element	Affected By	Validated By
13455	1 Functional Requirements	Heading			66: Functional Requirements
13433	1.1 General Functions	Heading			
13447	maintain the ambient pressure by purely mechanical design/solution	System Requirement	Block: Air_Supply	1328: Failing Test Case TKO	[illegible]

IBM Engineering Insights

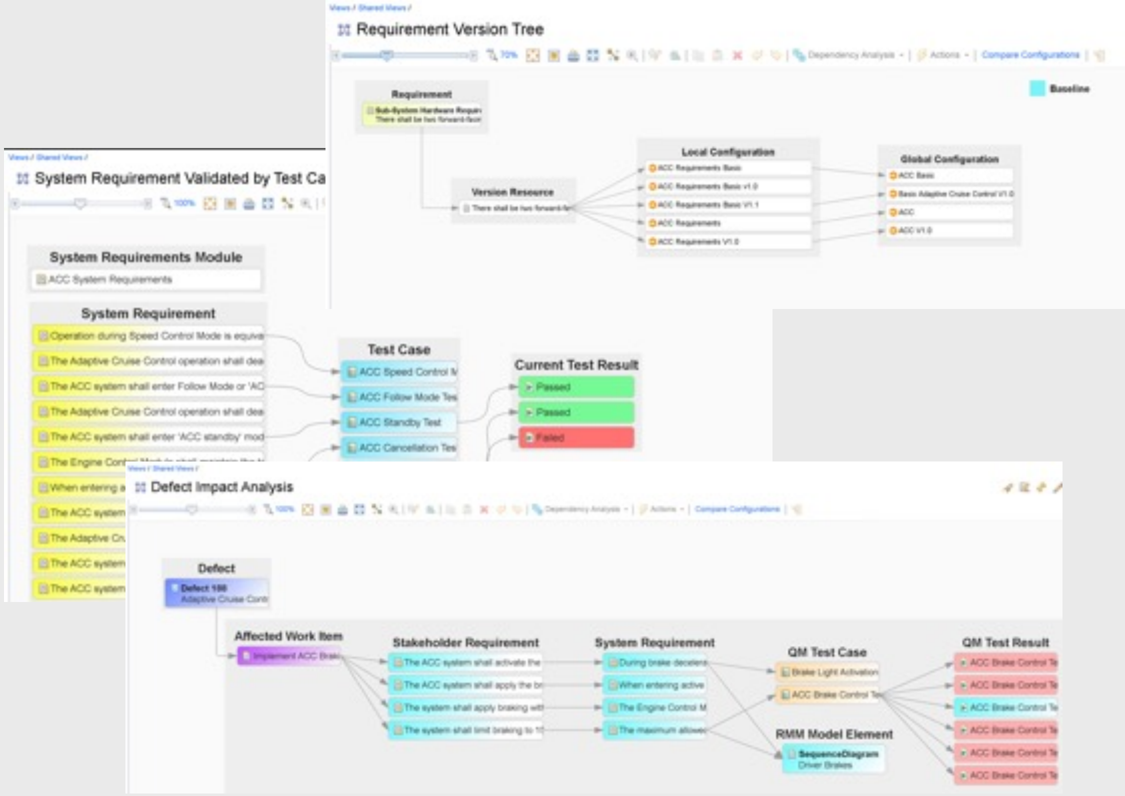
Advantages Achieved in Projects

Project Advantages

- Easy to demonstrate compliance
- Easy discovery of systems, capabilities, variants, versions, subsystems etc.
- Quick impact understanding
- Enhanced visibility, collaboration and productivity

Examples:

Requirements - Test – Result
Requirements – Version – Global Configuration
Defect Analysis – Requirement- Test - Results



RVTM Example

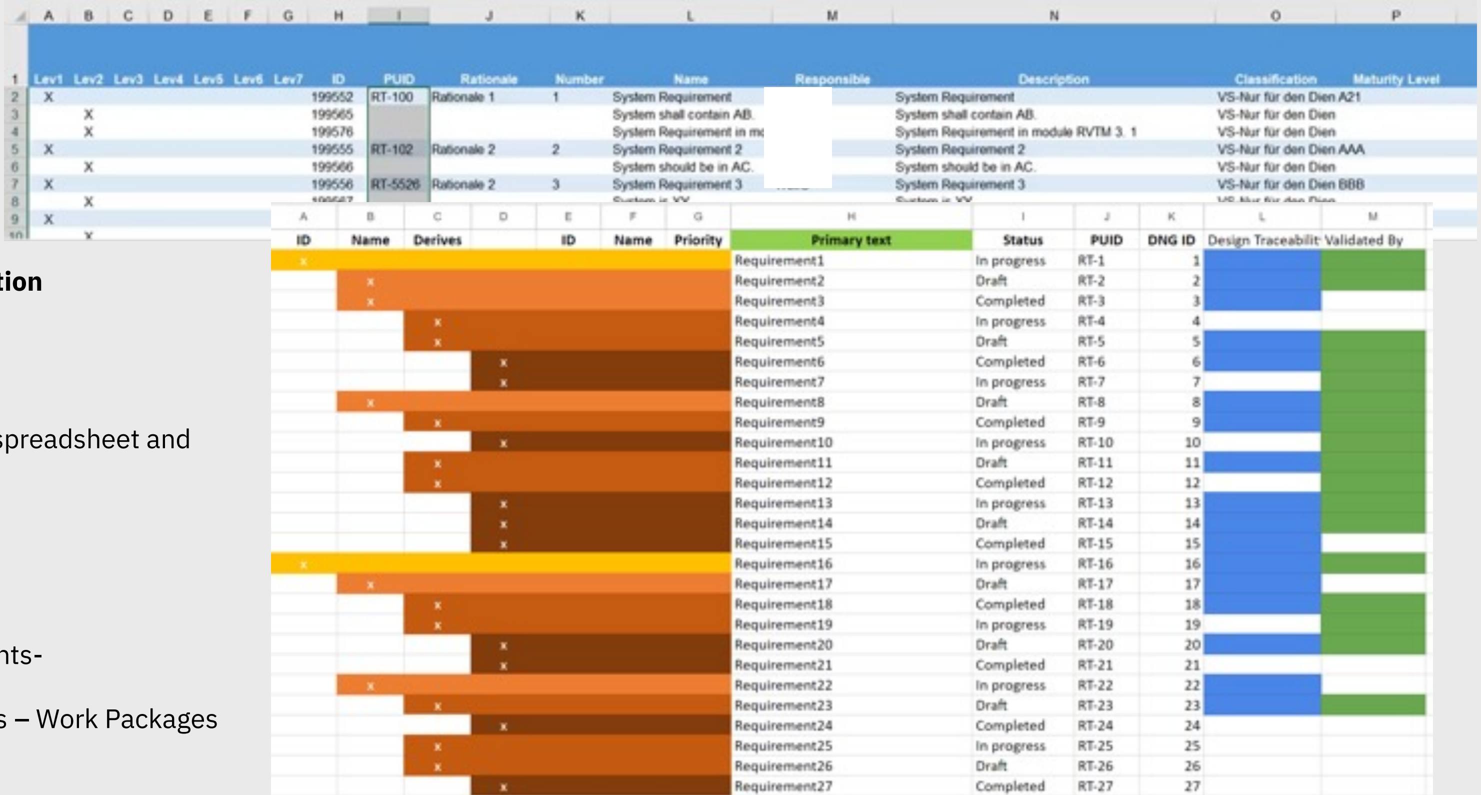
RVTM

Requirements Verification Traceability Matrix

Horizonal and Vertical traceability exported from IBM ELM to Excel spreadsheet and shared with customer

Traceability

Stakeholder Requirements-
System Requirements-
Design Elements – Tests – Work Packages



	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
1	Lev1	Lev2	Lev3	Lev4	Lev5	Lev6	Lev7	ID	PUID	Rationale	Number	Name	Responsible	Description	Classification	Maturity Level
2	X							199552	RT-100	Rationale 1	1	System Requirement	[redacted]	System Requirement	VS-Nur für den Dien A21	
3		X						199565				System shall contain AB.	[redacted]	System shall contain AB.	VS-Nur für den Dien	
4		X						199576				System Requirement in mo	[redacted]	System Requirement in module RVTM 3. 1	VS-Nur für den Dien	
5	X							199555	RT-102	Rationale 2	2	System Requirement 2	[redacted]	System Requirement 2	VS-Nur für den Dien AAA	
6		X						199566				System should be in AC.	[redacted]	System should be in AC.	VS-Nur für den Dien	
7	X							199556	RT-5526	Rationale 2	3	System Requirement 3	[redacted]	System Requirement 3	VS-Nur für den Dien BBB	
8		X						199567				System is XX		System is XX	VS-Nur für den Dien	
9	X															
10		X														

A	B	C	D	E	F	G	H	I	J	K	L	M
ID	Name	Derives		ID	Name	Priority	Primary text	Status	PUID	DNG ID	Design Traceabilit	Validated By
x							Requirement1	In progress	RT-1	1	🟦	🟩
	x						Requirement2	Draft	RT-2	2	🟦	🟩
	x						Requirement3	Completed	RT-3	3	🟦	
		x					Requirement4	In progress	RT-4	4		
		x					Requirement5	Draft	RT-5	5	🟦	🟩
			x				Requirement6	Completed	RT-6	6	🟦	🟩
			x				Requirement7	In progress	RT-7	7		🟩
	x						Requirement8	Draft	RT-8	8	🟦	🟩
		x					Requirement9	Completed	RT-9	9	🟦	🟩
			x				Requirement10	In progress	RT-10	10		🟩
		x					Requirement11	Draft	RT-11	11	🟦	🟩
		x					Requirement12	Completed	RT-12	12		🟩
			x				Requirement13	In progress	RT-13	13	🟦	🟩
			x				Requirement14	Draft	RT-14	14	🟦	🟩
			x				Requirement15	Completed	RT-15	15	🟦	
x							Requirement16	In progress	RT-16	16	🟦	🟩
	x						Requirement17	Draft	RT-17	17	🟦	
		x					Requirement18	Completed	RT-18	18	🟦	🟩
		x					Requirement19	In progress	RT-19	19		🟩
			x				Requirement20	Draft	RT-20	20	🟦	🟩
			x				Requirement21	Completed	RT-21	21		
	x						Requirement22	In progress	RT-22	22	🟦	
		x					Requirement23	Draft	RT-23	23	🟦	🟩
			x				Requirement24	Completed	RT-24	24		
		x					Requirement25	In progress	RT-25	25		
		x					Requirement26	Draft	RT-26	26		
			x				Requirement27	Completed	RT-27	27		

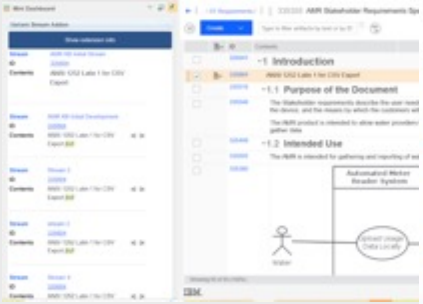
Reuse Based on Traceability Example

Traceability based Reuse

Reuse based on links
Requirements can be partially or fully updated from source

Compare of different variants and parameters

“Cherry picking” delivery to target



Mini Dashboard

Show extension info

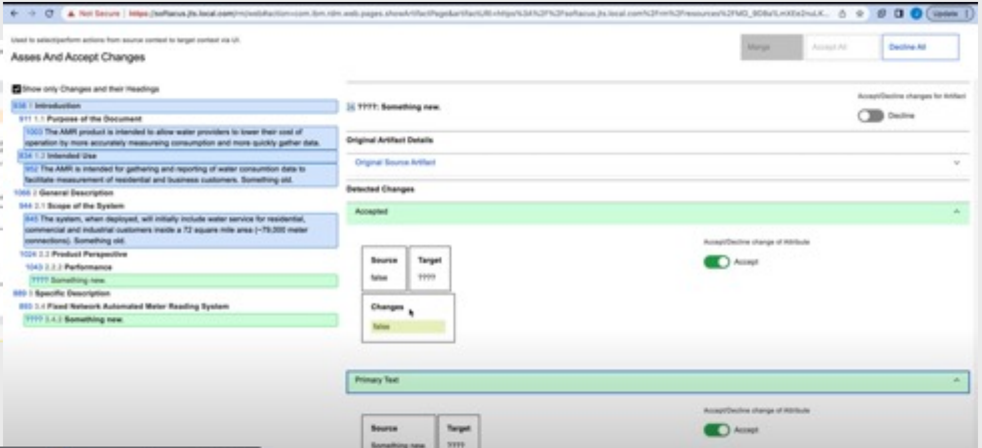
AMR Stakeholder Requirements Sp

Create

-1 Introduction

-1.1 Purpose of the Document

-1.2 Intended Use



Asses And Accept Changes

Merge | Accept All | Decline All

☑ Show only Changes and their Headings

838 1 Introduction
911 1.1 Purpose of the Document
1003 The AMR product is intended to allow water providers to lower their cost of operation by more accurately measureing consumption and more quickly gather data.
834 1.2 Intended Use
952 The AMR is intended for gathering and reporting of water consumption data to facilitate measurement of residential and business customers. Something old.
1066 2 General Description
944 2.1 Scope of the System
845 The system, when deployed, will initially include water service for residential, commercial and industrial customers inside a 72 square mile area (~79,000 meter connections). Something old.
1024 2.2 Product Perspective
1043 2.2.2 Performance
???? Something new
880 3 Specific Description
883 3.4 Fixed Network Automated Meter Reading System
???? 3.4.1 Something new

????: Something new.

Accept/Decline changes for Artifact: Decline

Original Artifact Details

Original Source Artifact

Detected Changes

Accepted

Source	Target
false	????

Changes: false

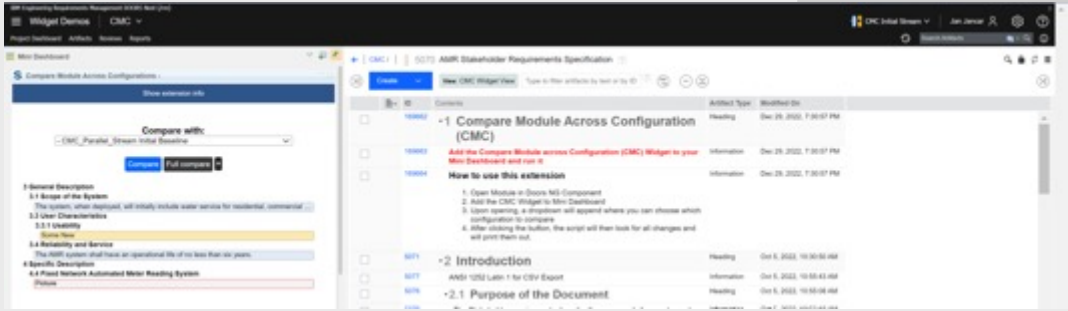
Accept/Decline change of Attribute: Accept

Primary Text

Source	Target
Something new	????

Changes

Accept/Decline change of Attribute: Accept



Widget Demos | CMC

Project Dashboard | Artifacts | Reviews | Reports

Mini Dashboard

Compare Module Across Configurations

Show extension info

Compare with:
CMC_Parallel_Stream Initial Baseline

Compare | Full compare

1 General Description
3.1 Scope of the System
The system, when deployed, will initially include water service for residential, commercial ...
3.2 User Characteristics
3.2.1 Usability
Some New
3.4 Reliability and Service
The AMR system shall have an operational life of no less than six years.
4 Specific Description
4.4 Fixed Network Automated Meter Reading System
Picture

CMC Initial Stream | Jan Jancar

CMC / | 5070 AMR Stakeholder Requirements Specification

Create | View: CMC Widget View

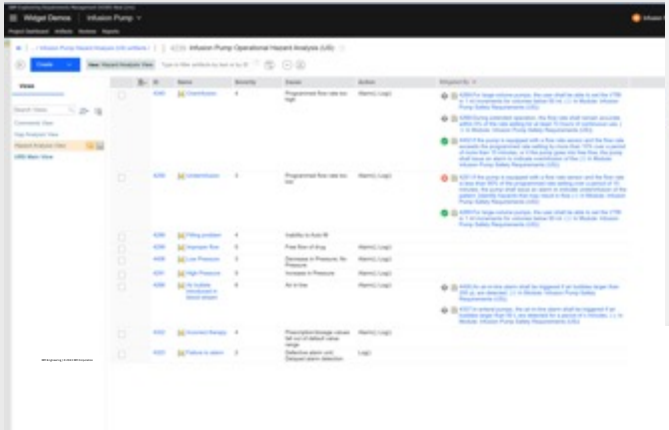
ID	Contents	Artifact Type	Modified On
	1 Compare Module Across Configuration (CMC)	Heading	Dec 29, 2022, 7:30:07 PM
	Add the Compare Module across Configuration (CMC) Widget to your Mini Dashboard and run it	Information	Dec 29, 2022, 7:30:07 PM
	How to use this extension 1. Open Module in Doors NG Component 2. Add the CMC Widget to Mini Dashboard 3. Upon opening, a dropdown will append where you can choose which configuration to compare 4. After clicking the button, the script will then look for all changes and will print them out.	Information	Dec 29, 2022, 7:30:07 PM
5071	2 Introduction	Heading	Oct 5, 2022, 10:30:50 AM
5077	ANSI 1252 Latin 1 for CSV Export	Information	Oct 5, 2022, 10:55:43 AM
5076	2.1 Purpose of the Document	Heading	Oct 5, 2022, 10:55:08 AM

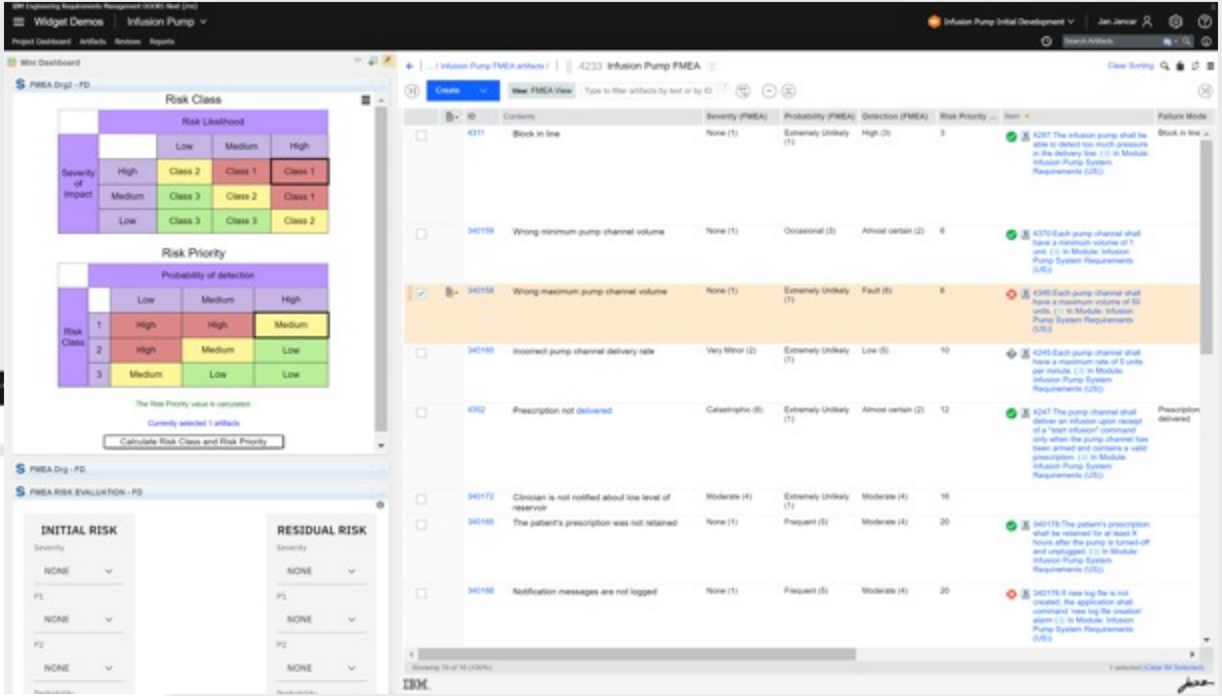
FMEA and Traceability

Traceability based FMEA

Various FMEA implementations for Medical, Military and Pharma

Hazard analysis, Risk Matrix...





Risk Class

		Risk Likelihood		
		Low	Medium	High
Severity of Impact	High	Class 2	Class 1	Class 1
	Medium	Class 3	Class 2	Class 1
	Low	Class 3	Class 3	Class 2

Risk Priority

		Probability of detection		
		Low	Medium	High
Risk Class	1	High	High	Medium
	2	High	Medium	Low
	3	Medium	Low	Low

The Risk Priority value is calculated

Currently selected 1 artifacts

INITIAL RISK

Severity: NONE

P1: NONE

P2: NONE

RESIDUAL RISK

Severity: NONE

P1: NONE

P2: NONE

4233 Infusion Pump FMEA

ID	Contents	Severity (FMEA)	Probability (FMEA)	Detection (FMEA)	Risk Priority ...	Item	Failure Mode
4311	Block in line	None (1)	Extremely Unlikely (1)	High (3)	3	4297 The infusion pump shall be able to detect too much pressure in the delivery line. In Module: Infusion Pump System Requirements (US)	Block in line
340159	Wrong minimum pump channel volume	None (1)	Occasional (3)	Almost certain (2)	6	4370 Each pump channel shall have a minimum volume of 1 unit. In Module: Infusion Pump System Requirements (US)	
340158	Wrong maximum pump channel volume	None (1)	Extremely Unlikely (1)	Fault (6)	6	4346 Each pump channel shall have a maximum volume of 50 units. In Module: Infusion Pump System Requirements (US)	
340160	Incorrect pump channel delivery rate	Very Minor (2)	Extremely Unlikely (1)	Low (5)	10	4245 Each pump channel shall have a maximum rate of 5 units per minute. In Module: Infusion Pump System Requirements (US)	
4352	Prescription not delivered	Catastrophic (6)	Extremely Unlikely (1)	Almost certain (2)	12	4247 The pump channel shall deliver an infusion upon receipt of a "start infusion" command only when the pump channel has been armed and contains a valid prescription. In Module: Infusion Pump System Requirements (US)	Prescription delivered
340172	Clinician is not notified about low level of reservoir	Moderate (4)	Extremely Unlikely (1)	Moderate (4)	16		
340165	The patient's prescription was not retained	None (1)	Frequent (5)	Moderate (4)	20	340176 The patient's prescription shall be retained for at least X hours after the pump is turned-off and unplugged. In Module: Infusion Pump System Requirements (US)	
340168	Notification messages are not logged	None (1)	Frequent (5)	Moderate (4)	20	340178 If new log file is not created, the application shall command 'new log file creation' alarm. In Module: Infusion Pump System Requirements (US)	

Showing 15 of 15 (100%)

1 selected (Clear All Selected)

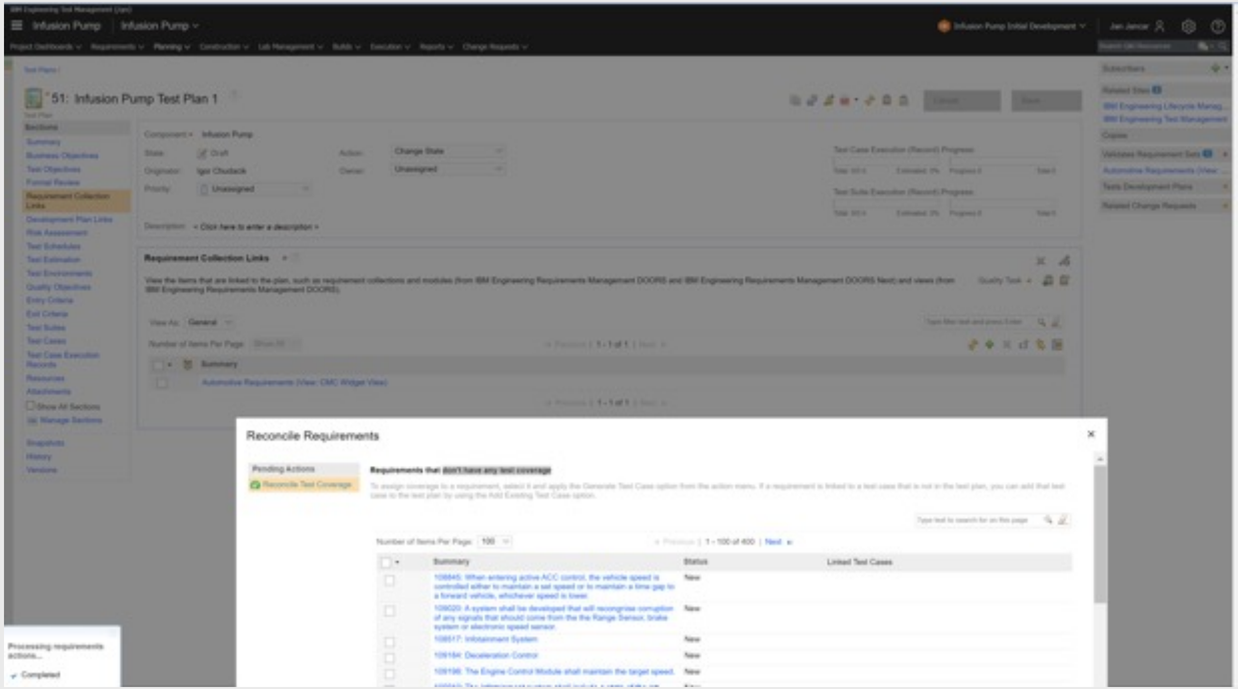
Generation of Artifacts based on Traceability

Traceability Autocreation

Generation of automated traceability between requirements and **test cases**

Generation of automated traceability between requirements and **tasks**

Artifact autocreation with few clicks OOTB and scripts



51: Infusion Pump Test Plan 1

Component: Infusion Pump
State: Draft
Action: Change State
Originator: Igor Chudack
Owner: Unassigned
Priority: Unassigned
Description: < Click here to enter a description >

Requirement Collection Links

View the items that are linked to the plan, such as requirement collections and modules (from IBM Engineering Requirements Management DOORS and IBM Engineering Requirements Management DOORS Next) and views (from IBM Engineering Requirements Management DOORS).

View As: General

Summary
Automotive Requirements (View: CMC Widget View)

1 - 1 of 1

Reconcile Requirements

Requirements that don't have any test coverage

To assign coverage to a requirement, select it and apply the Generate Test Case option from the action menu. If a requirement is linked to a test case that is not in the test plan, you can add that test case to the test plan by using the Add Existing Test Case option.

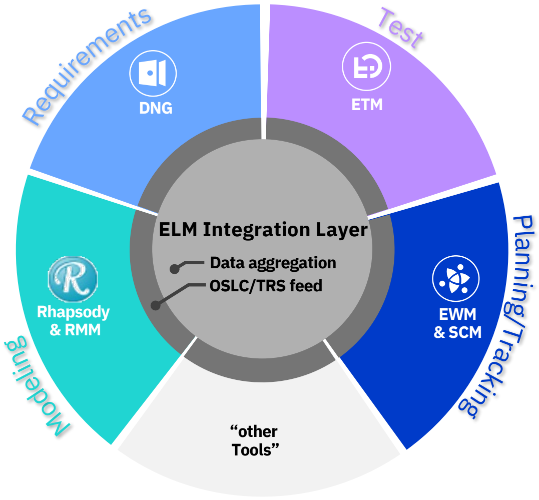
Number of Items Per Page: 100 — 1 - 100 of 400 | Next

Summary	Status	Linked Test Cases
108845: When entering active ACC control, the vehicle speed is controlled either to maintain a set speed or to maintain a time gap to a forward vehicle, whichever speed is lower.	New	
109020: A system shall be developed that will recongnise corruption of any signals that should come from the the Range Sensor, brake system or electronic speed sensor.	New	
108517: Infotainment System	New	
109184: Deceleration Control	New	
109198: The Engine Control Module shall maintain the target speed.	New	

Processing requirements actions...
✓ Completed

IBM Engineering Lifecycle Management Benefits

- Combines Systems & SW Engineering with Integration Layer
- Built on open integration and standards
- Automates transparency and traceability
- Includes Model Based Systems Engineering & SW Design
- Accelerates industry solutions w/compliance, safety critical and security standards
- Can be customized to meet industry needs



IBM Engineering Lifecycle Management (ELM) is an integrated systems and software development solution –empowering engineers and their teams to more easily manage each stage of the engineering process end-to-end.

Thank You

Peter Schedl
peter.schedl@de.ibm.com

Further information:

[IBM Engineering Lifecycle Management Automotive Compliance]

[IBM Engineering Management Overview]