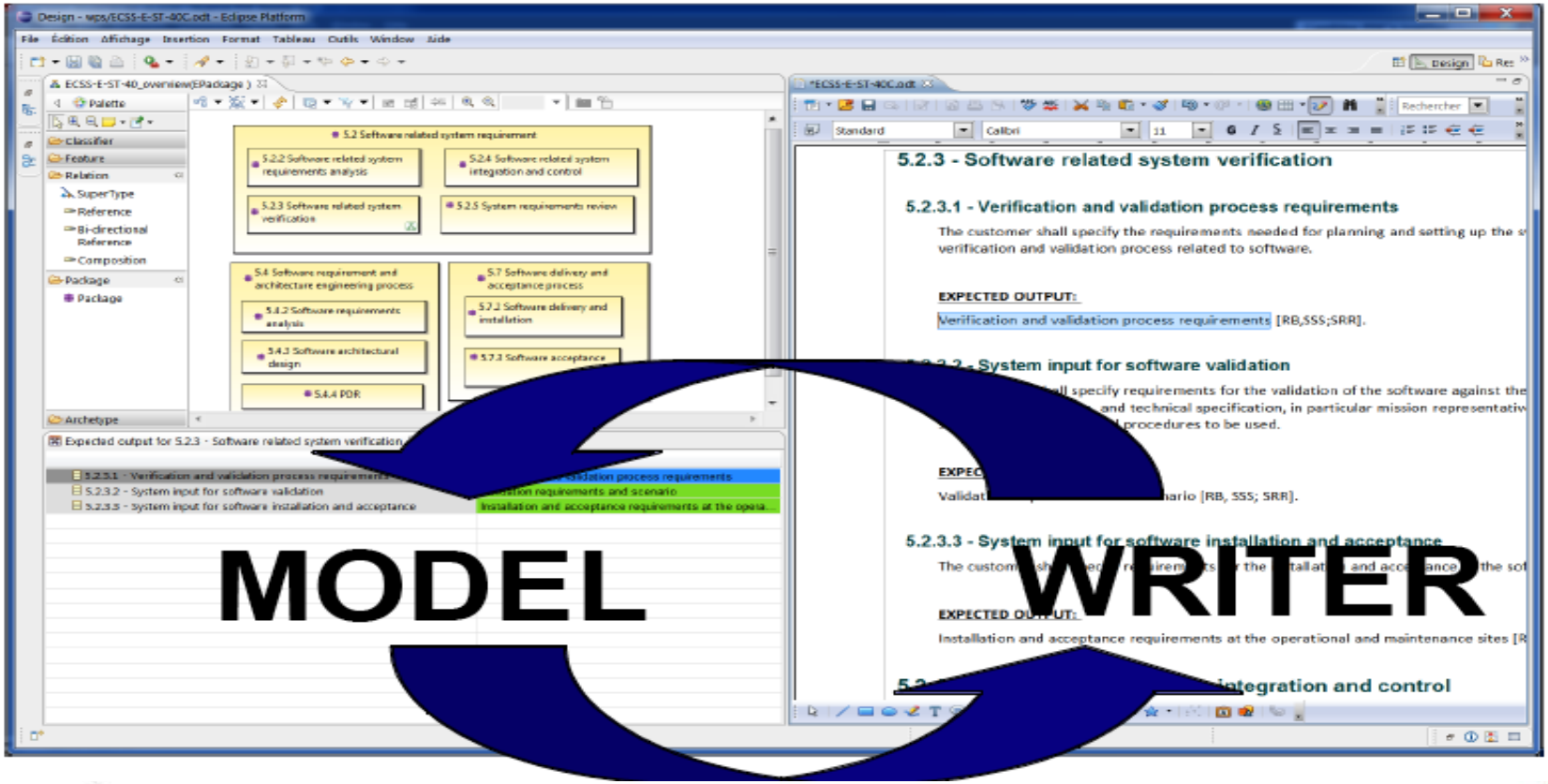
ModelWriter

Text & Model-Synchronized Document Engineering Platform

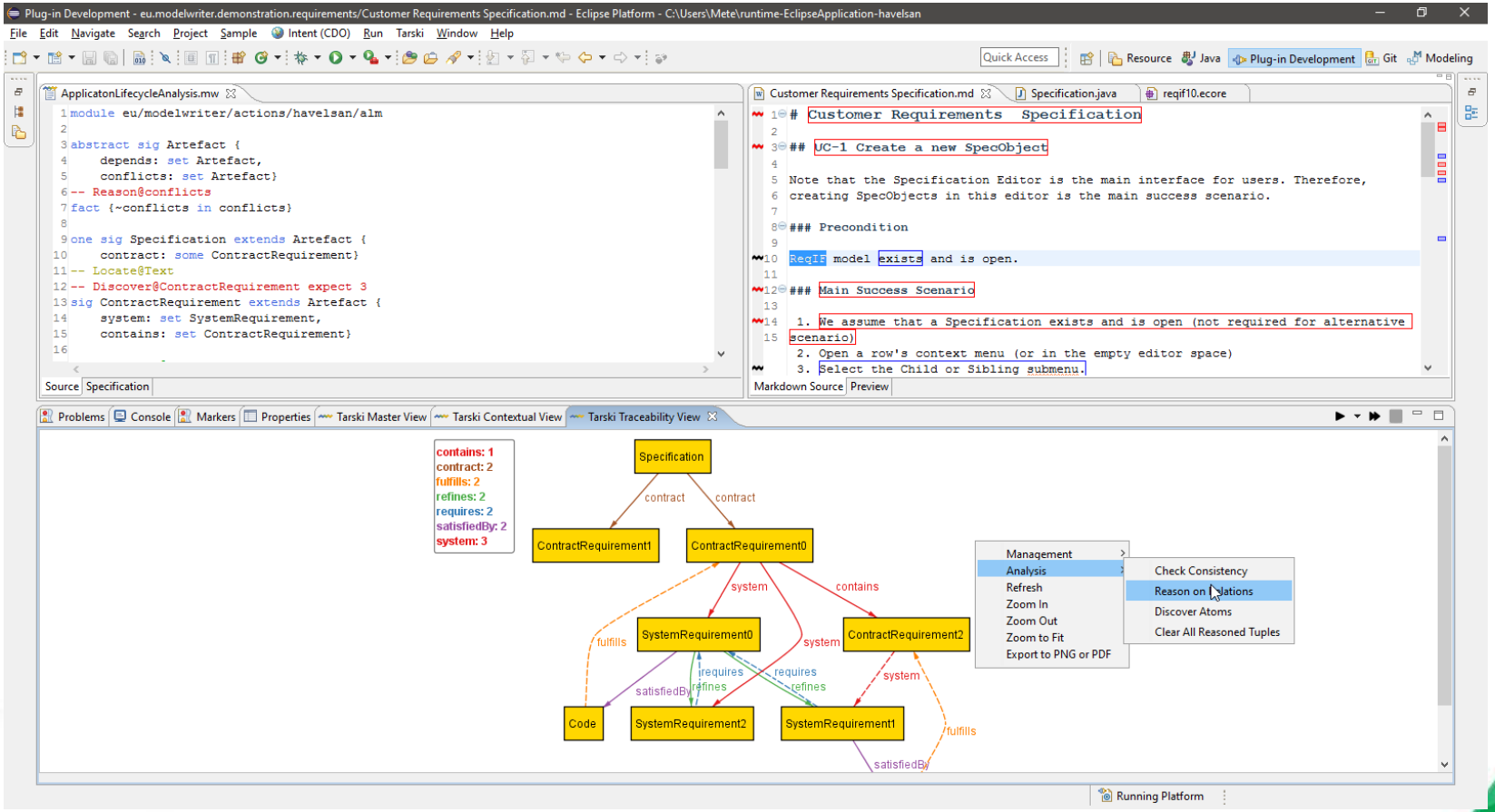


Project Leader: Ferhat Erata (ferhat@computer.org)
Project Email: project@modelwriter.eu

ITEA 3

ModelWriter

Text & Model-Synchronized Document Engineering Platform



Project Leader: Ferhat Erata (ferhat@computer.org)
Project Email: project@modelwriter.eu

ITEA 3

1 Introduction

Ferhat Erata
UNIT, ModelWriter Project Leader

Participants
ModelWriter #1 Review

UNIT
- Ferhat Erata
- Dr. Moharram Challenger
- Hasan Emre Kirmizi

beo Model Driven Company
- Yvan Lussaud
- Etienne Juliot

HAVELSAN
- Dr. Eray Tuzun
- Yagup Macit

cnrs
- Dr. Claire Gardent
- Dr. Samuel Cruz-Lara
- Dr. Bikash Gyawali
- Anastasia Shimorina

FORD OTOSAN
- Tuğçe Yiğiter
- Alan Endersoy
- Dr. Emrah Kinav

Mantis Software Company
- Dr. Guven Köse
- Omurhan Soysal
- Aydin Can Polatkan

KoçSistem
- Dr. Geylani Kardas
- Mehmet Onat
- Hale Gezgen

HİSBİM
- Ersan Gürdoğan
- Taskin Kızıl
- Oguz Yavuz

AIRBUS GROUP
- Dr. Anne Monceaux

Agenda [13:00 - 17:00]

1. Introduction (5 mins) [13:00 - 13:05]
2. Overview of the project (WP5) (15 min) [13:05 - 13:20]
3. Industrial Use Cases (WP1) (30 min) [13:20 - 13:50]
4. Exploitation Related Activities (WP7) (10 mins) [13:50 - 14:00]
5. Technical Innovations (WP2, 3 & 4) (30 mins) [14:00 - 14:30]
6. User Interfaces & Integration (WP6) (10 mins) [14:30 - 14:40]

- Break (20 mins) [14:40 - 15:00]

6. Demonstrations (60 mins) [15:00 - 16:00]
7. Conclusions (10 mins) [16:00 - 16:10]

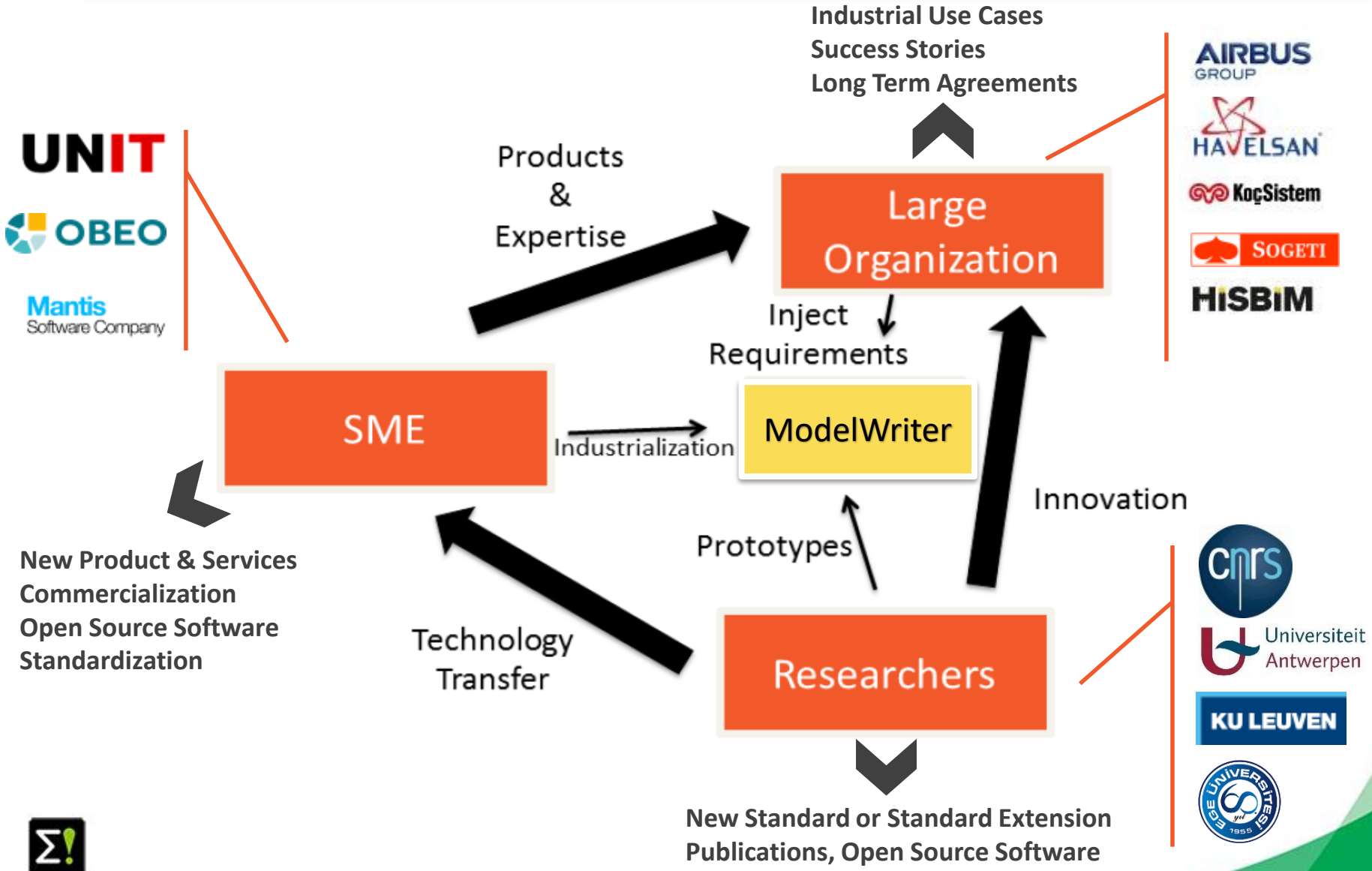
- Reviewers' Private Section (30 min) [16:10 - 16:40]
- Feedback Session (15min) [16:40 - 16:55]
- Final Words (5 min) [16:55 - 17:00]

2 Overview of the Project

Ferhat Erata
UNIT, ModelWriter Project Leader

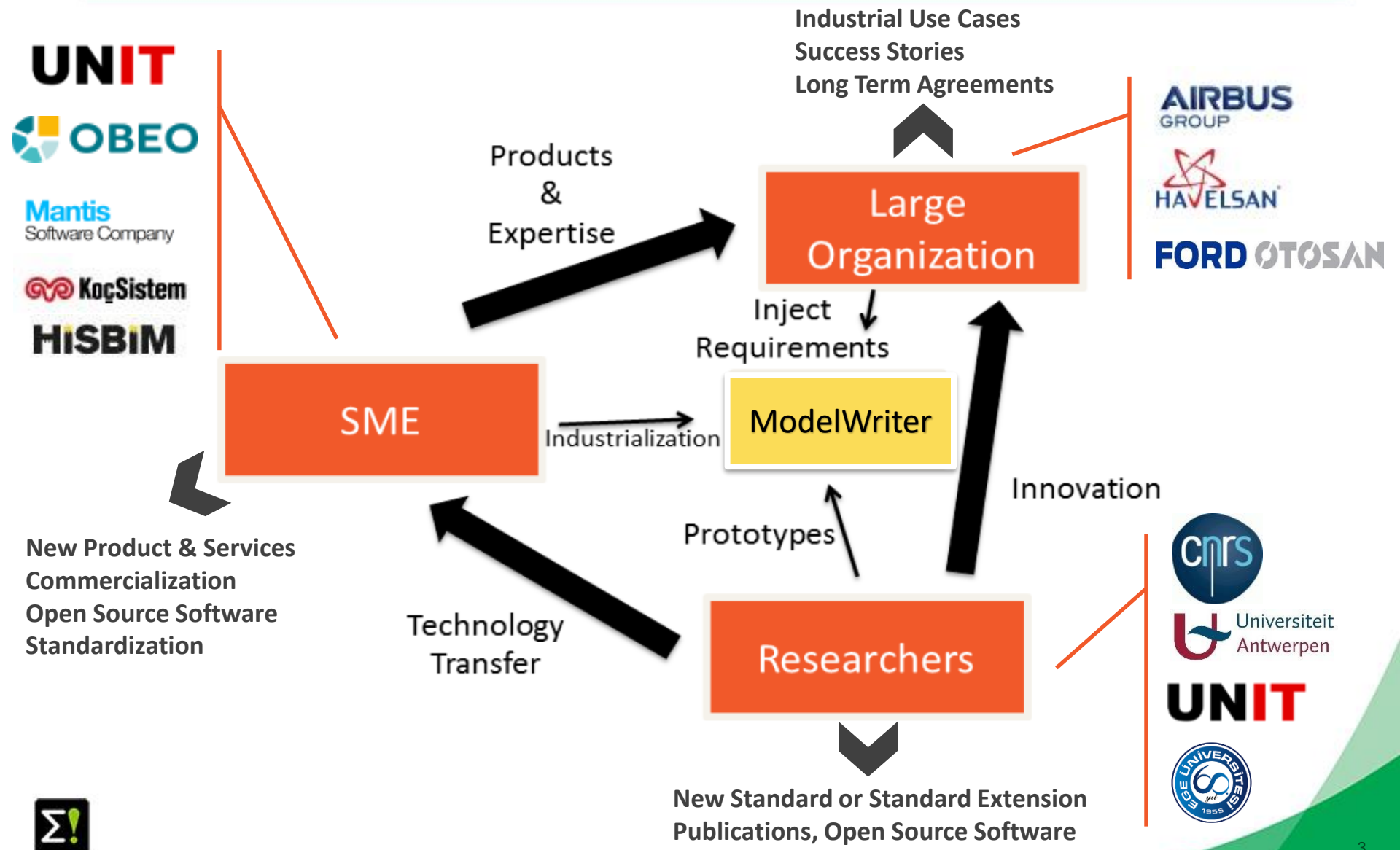
Industrialization Triangle in ModelWriter

Open Source Software (year #1)

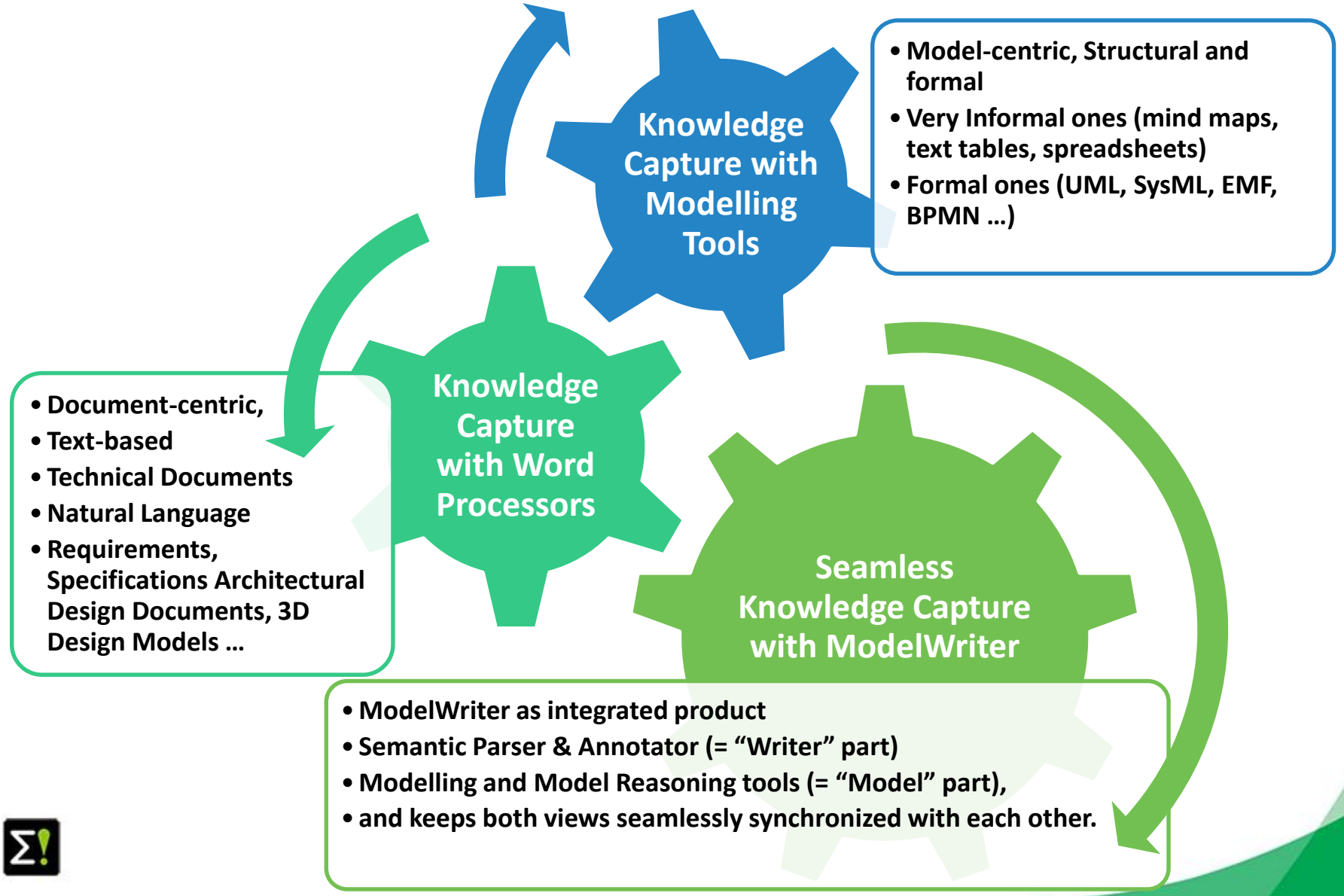


Industrialization Triangle in ModelWriter

Open Source Software (year #2)

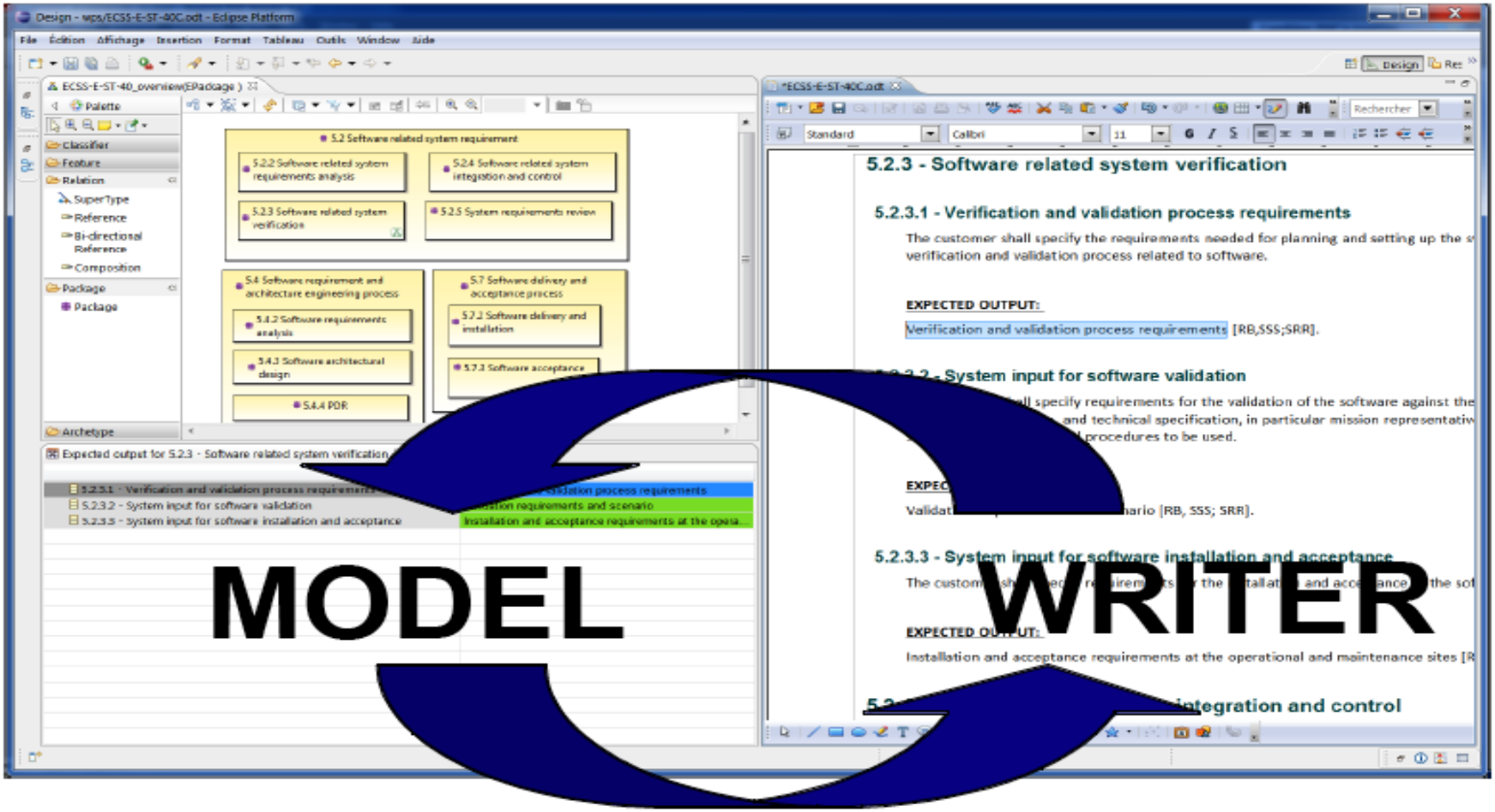


ModelWriter
Today's Knowledge Capture



ModelWriter

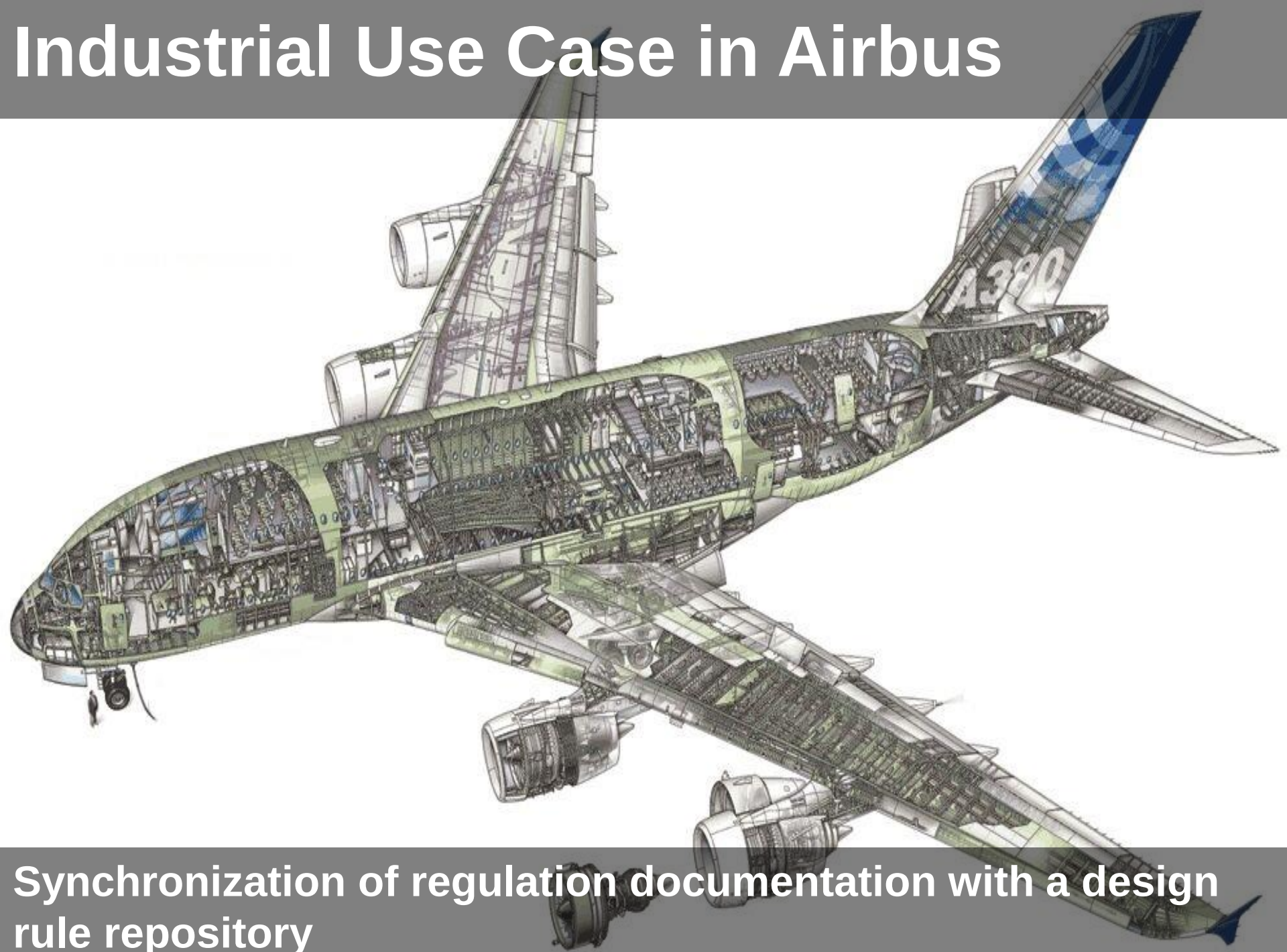
bi-directional Knowledge Capture tool



What is the problem?

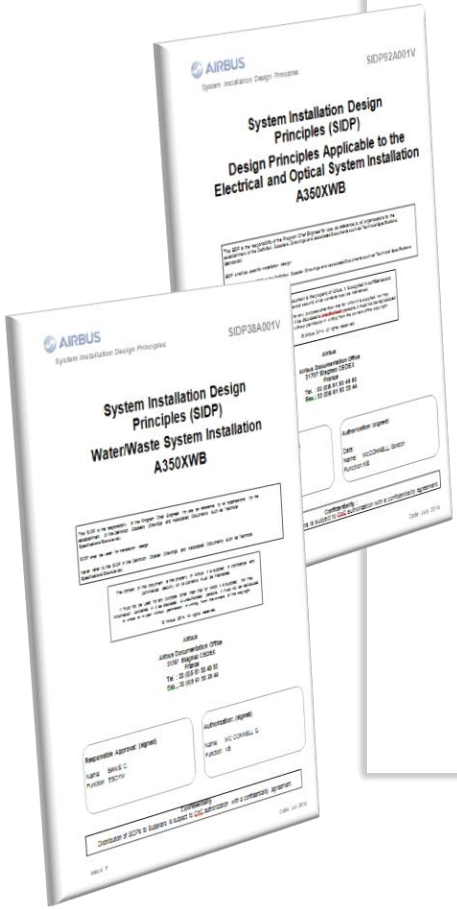
ITEA 3

Industrial Use Case in Airbus



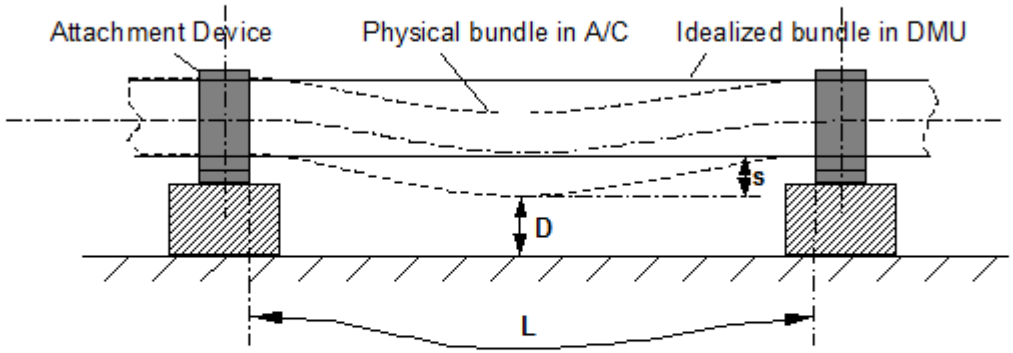
Synchronization of regulation documentation with a design rule repository

SIDP: System Installation Design Principles



SIDP92A001V-A-784

For installation of optical and electrical harnesses additional clearance for sagging (s) shall be provided as detailed below:



s...Sagging of bundle (real behavior of physical bundle in A/C due to gravity, ageing, etc.)
D...Required Distance
L...Actual length of a bundle segment between two Attachment Points (as designed in DMU)

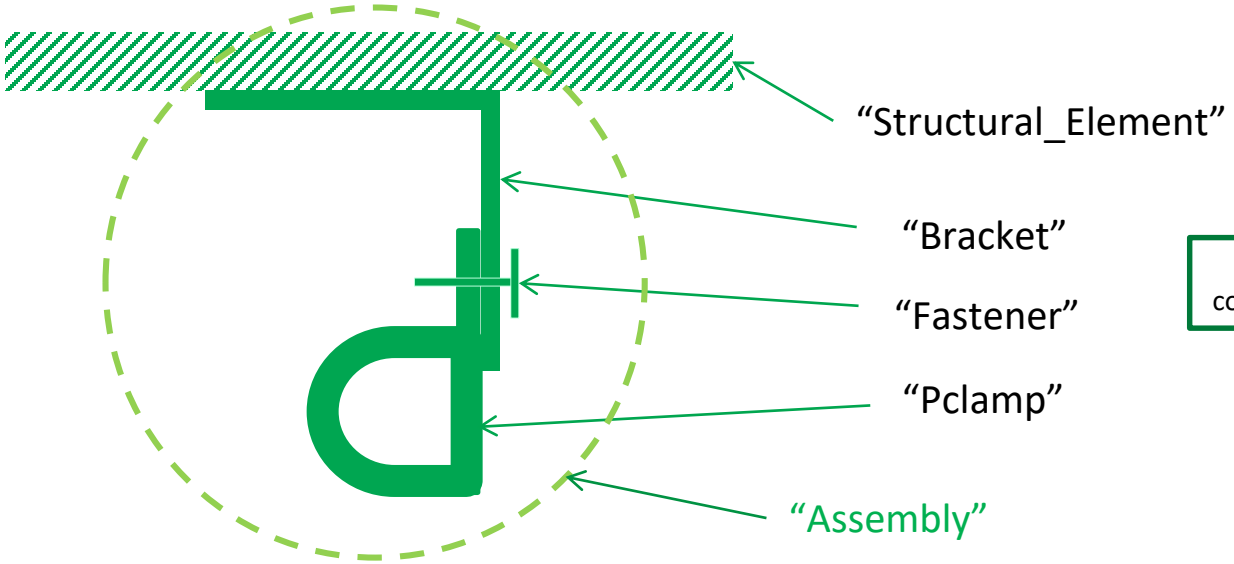
Figure 6: Sagging of bundles between attachment points

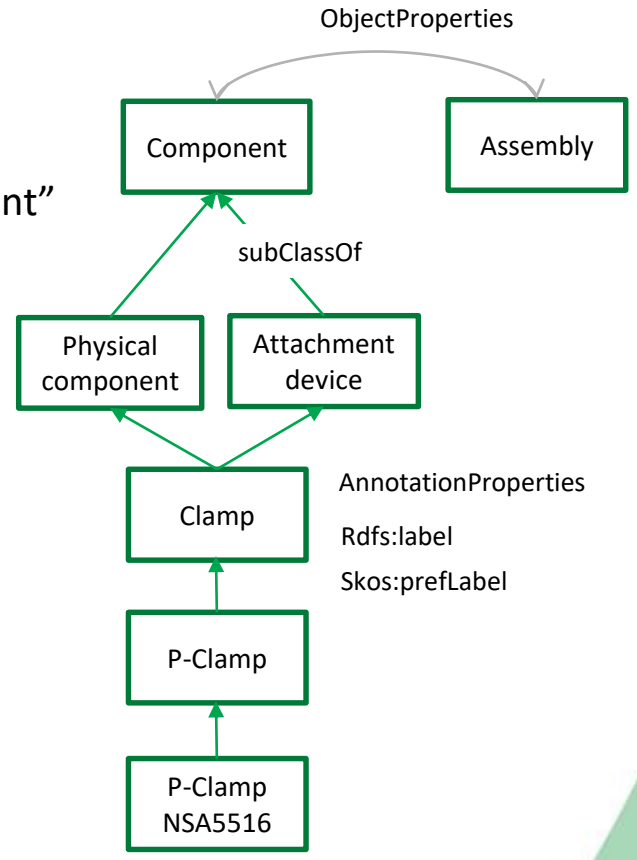
Note: Unless the bundle has a straight routing, L is bigger than the pitch between the Attachment Points.

Component Ontology and Rules

Objectives:

- Manage rules/design principles and improve traceability
- Automate identification of design conflicts against rules

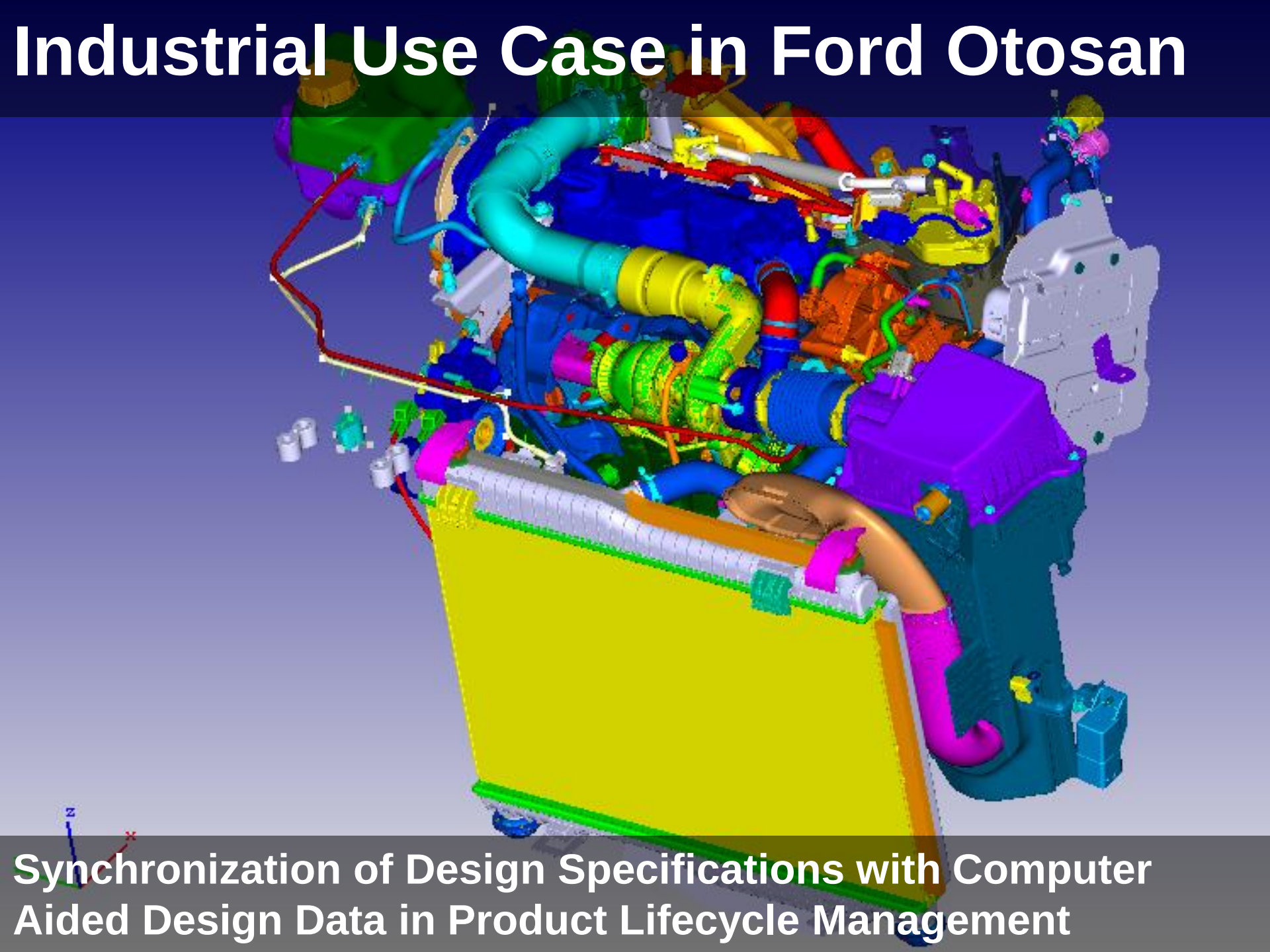




"P-clamp NSA5516 can be fixed on X with Y"

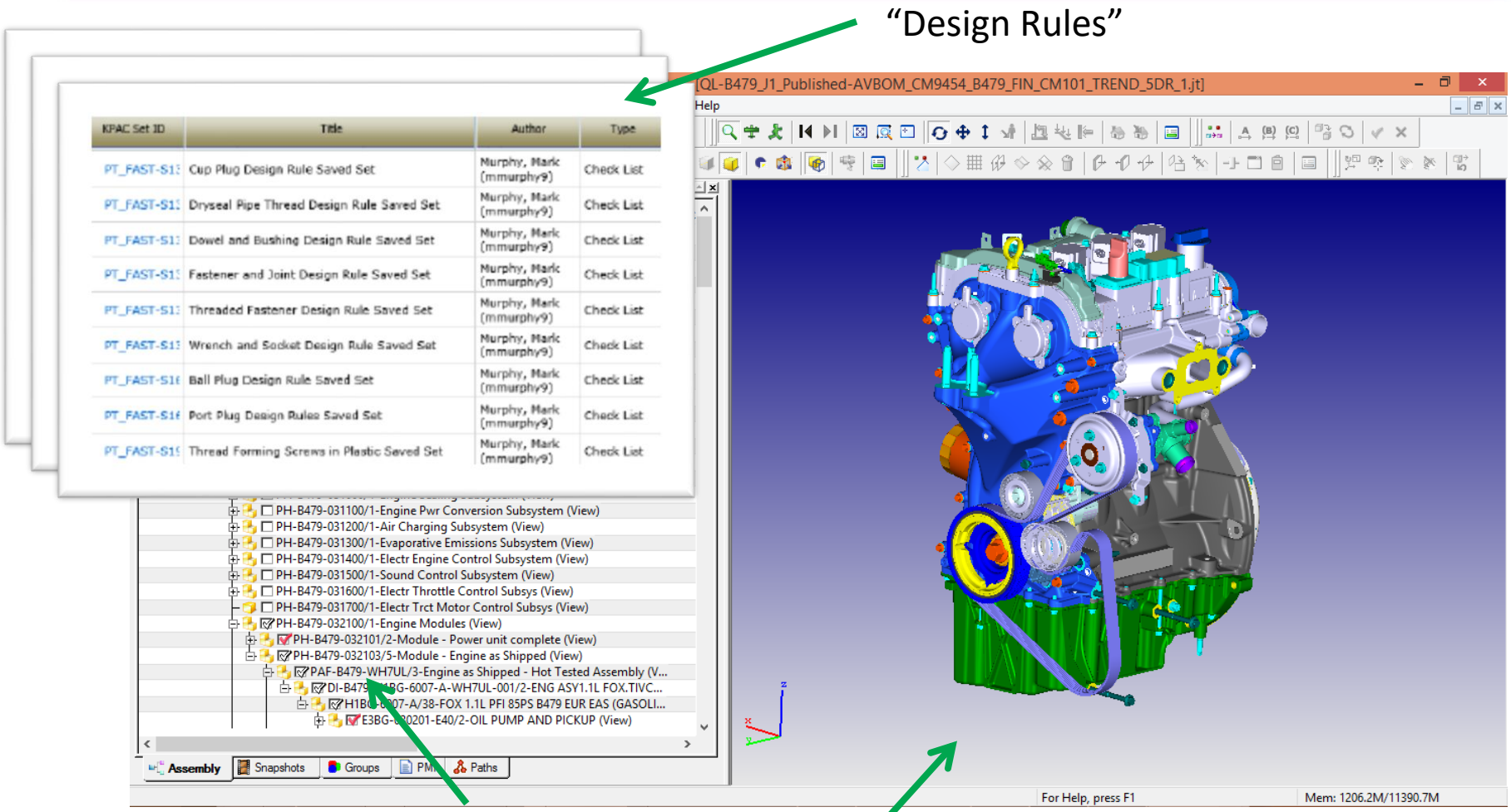
"Physical component" "Standard reference"

Industrial Use Case in Ford Otosan

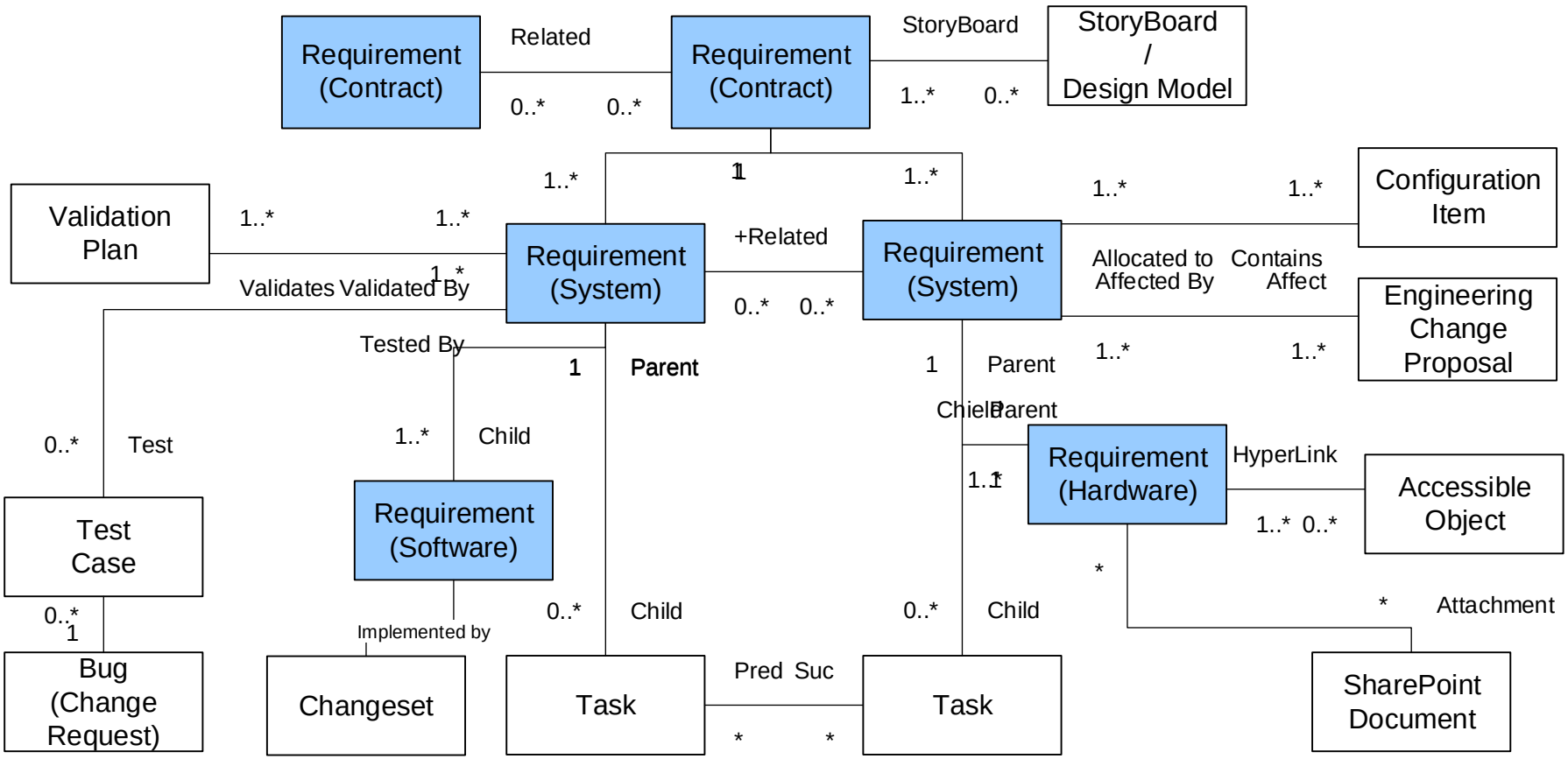


Synchronization of Design Specifications with Computer Aided Design Data in Product Lifecycle Management

BOM and Design Specifications



Industrial Use Case in Havelsan

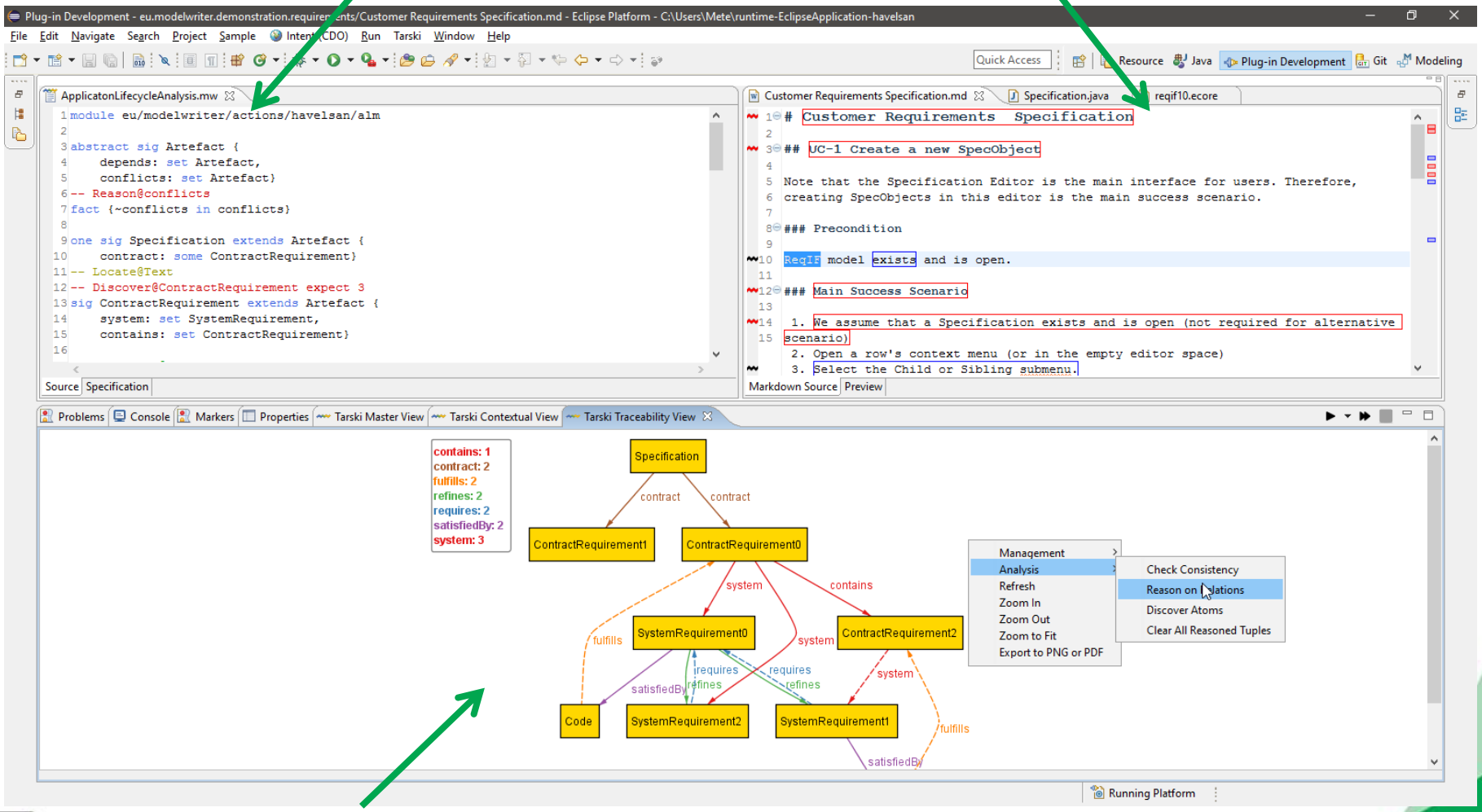


Integration with Application Lifecycle Management to ensure reliability and consistency in the system under development.

Automated Analysis of Dynamically Configured Traceability Semantics

"Traceability Rules to define traceability semantics"

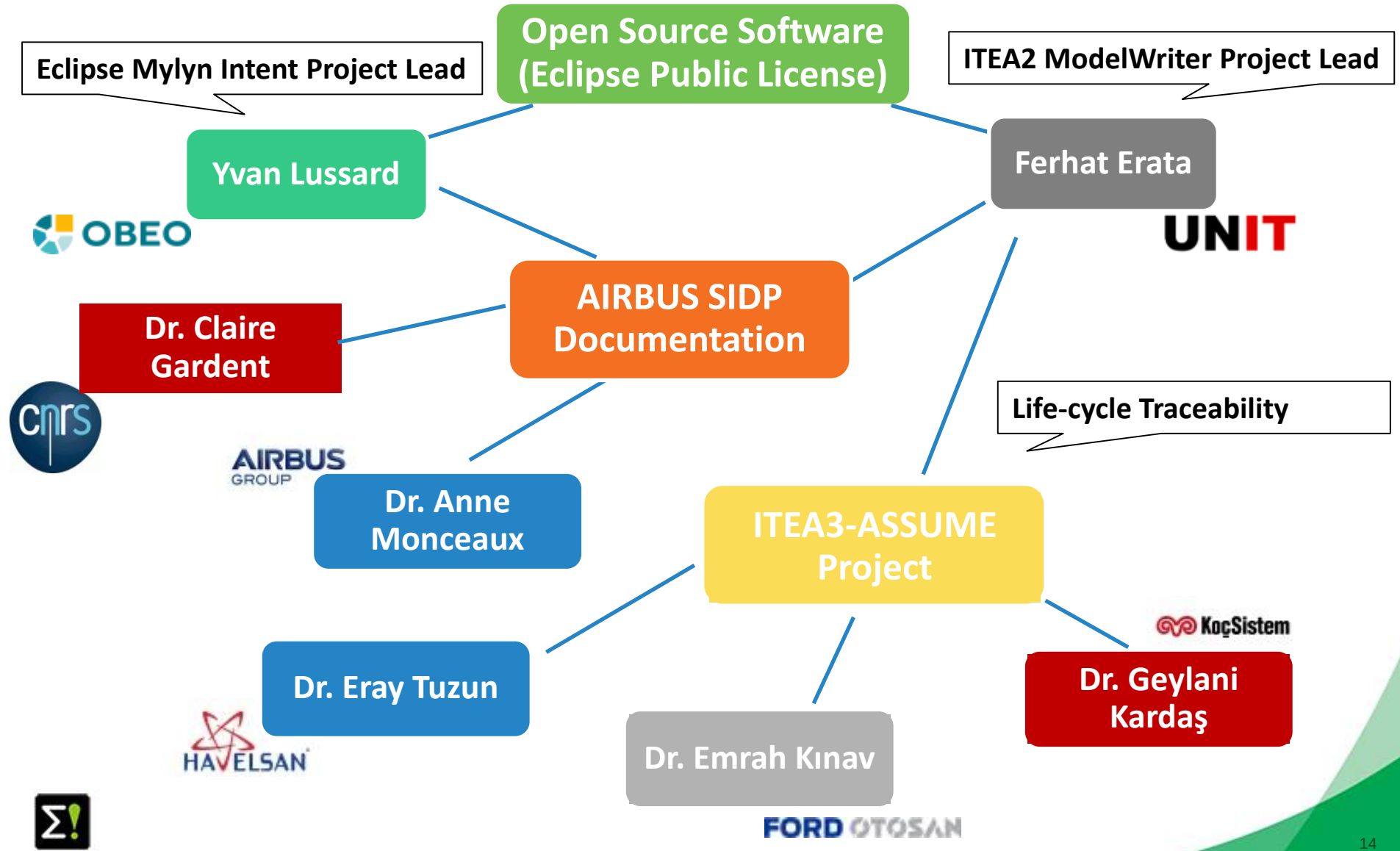
Artefacts or part of artefacts



"Various Traceability Analysis might be performed"

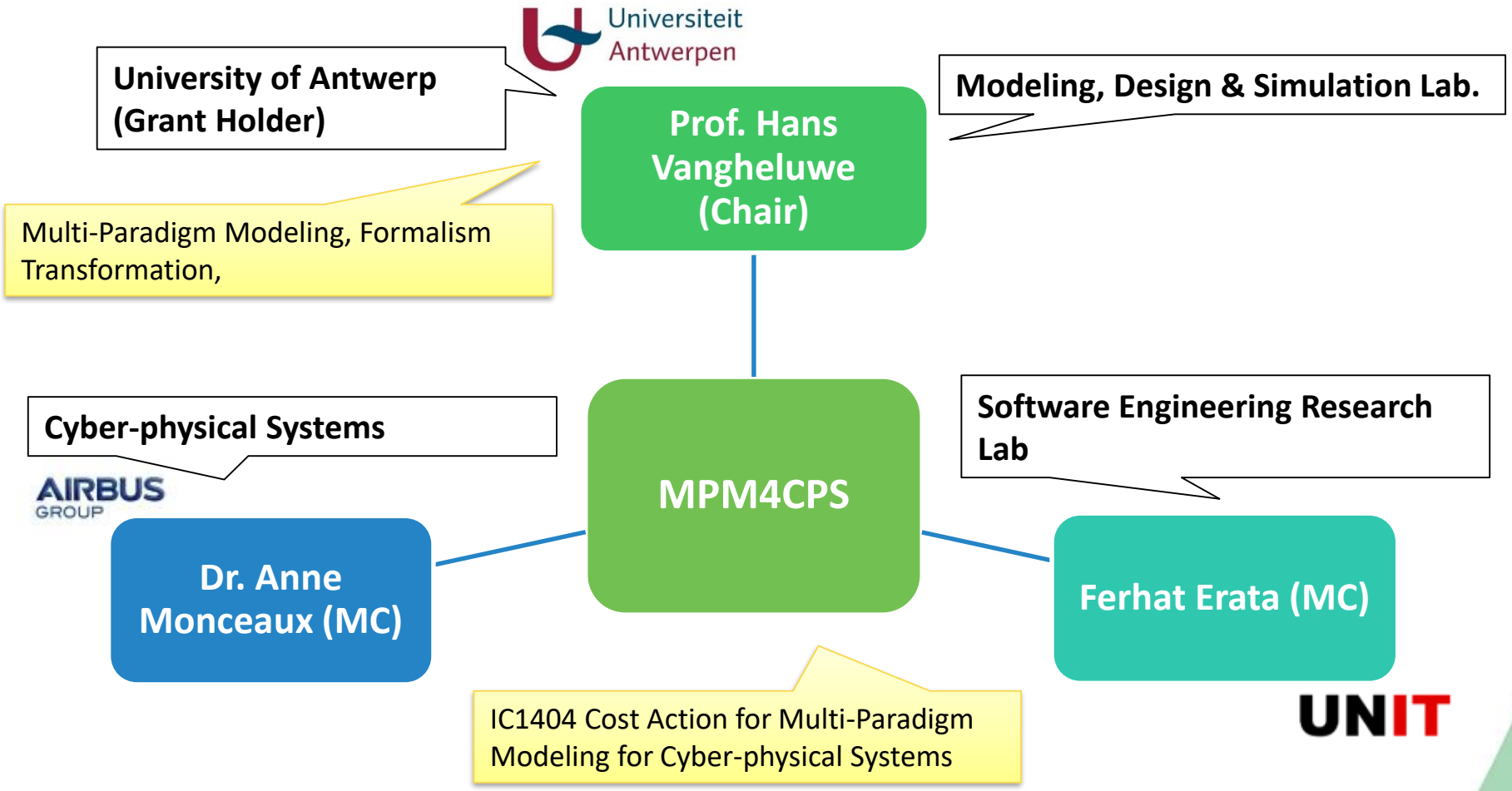
Level of Collaboration within ModelWriter

International Collaboration



Level of Collaboration within ModelWriter

International Collaboration



ModelWriter Activities in the First Year

https://github.com/modelwriter/workshops

Project Kick-off in Istanbul, Turkey (Nov 08, 2014) [M1]

Initial Architectural Design, Indutrial Use Cases, Technical WP discussions

Collaboration Infrastructure

The 1st International ModelWriter Workshop in Izmir, Turkey (Jan 15-17, 2015) [M4]

Exploitation: Havelsan's participation

The 1st International Eureka Project Exhibition in Berlin, Germany (Mar 10-11, 2015) [M6]

Consolidated User Requirements & Review

The 2nd International ModelWriter Workshop in Brussels, Belgium (Apr 30, 2015) [M7]

Software Requirements Review & Architecture

The 3rd International ModelWriter Workshop in Toulouse, France (Jun 22-23, 2015) [M10]

Rehearsal & Review

The 4th International ModelWriter Workshop in Brussels, Belgium (Sep 23-24, 2015) [M12]

Integration of software components

The 5th International ModelWriter Workshop in Ludwigsburg, Germany (Nov 2-5, 2015) [M16]

ModelWriter Activities in the Second Year

https://github.com/modelwriter/workshops

Product Owner Review Meeting

The 6th International ModelWriter Workshop in Paris, France (Feb 15-16, 2016)

ModelWriter is positioned on the Working Groups of this Cost Action

ICT Cost Action - MPM4CPS WG meeting at Vienna, Austria, on the 15-16 April, 2016

The 7th International ModelWriter Technical Workshop in Toulouse, France (6 June 2016)

Future of ModelWriter is discussed

The 7th Int'l ModelWriter Brainstorming Session in Toulouse, France at Airbus (9 June 2016)

The 7th Int'l ModelWriter Coordination Meeting in Toulouse, France at Airbus (10 June 2016)

Poster Presentation

ModelWriter Poster Presentation SAT/SMT/AR Summer School 2016

Participation in International Joint Conference on Automated Reasoning (IJCAR) 2016

ModelWriter Activities in the Second Year

https://github.com/modelwriter/workshops

Verification Technology, Systems & Applications (VTSA) Summer School (Aug 29- Sept 02, 2016)

ICT Cost Action - MPM4CPS WG meeting at Gdansk, Poland (Sep 13-16, 2016)

A paper is submitted to ACM Applied Computing Symposium and under review.

Participation in Workshop on Software Correctness and Reliability (Oct 7-8 2016)

The 8th International ModelWriter Technical Workshop in Toulouse, France (13 June 2016)

ICT Cost Action - MPM4CPS WG meeting at Malaga, Spain (Nov 24-25, 2016)

ModelWriter
Work Packages & Technical Innovations

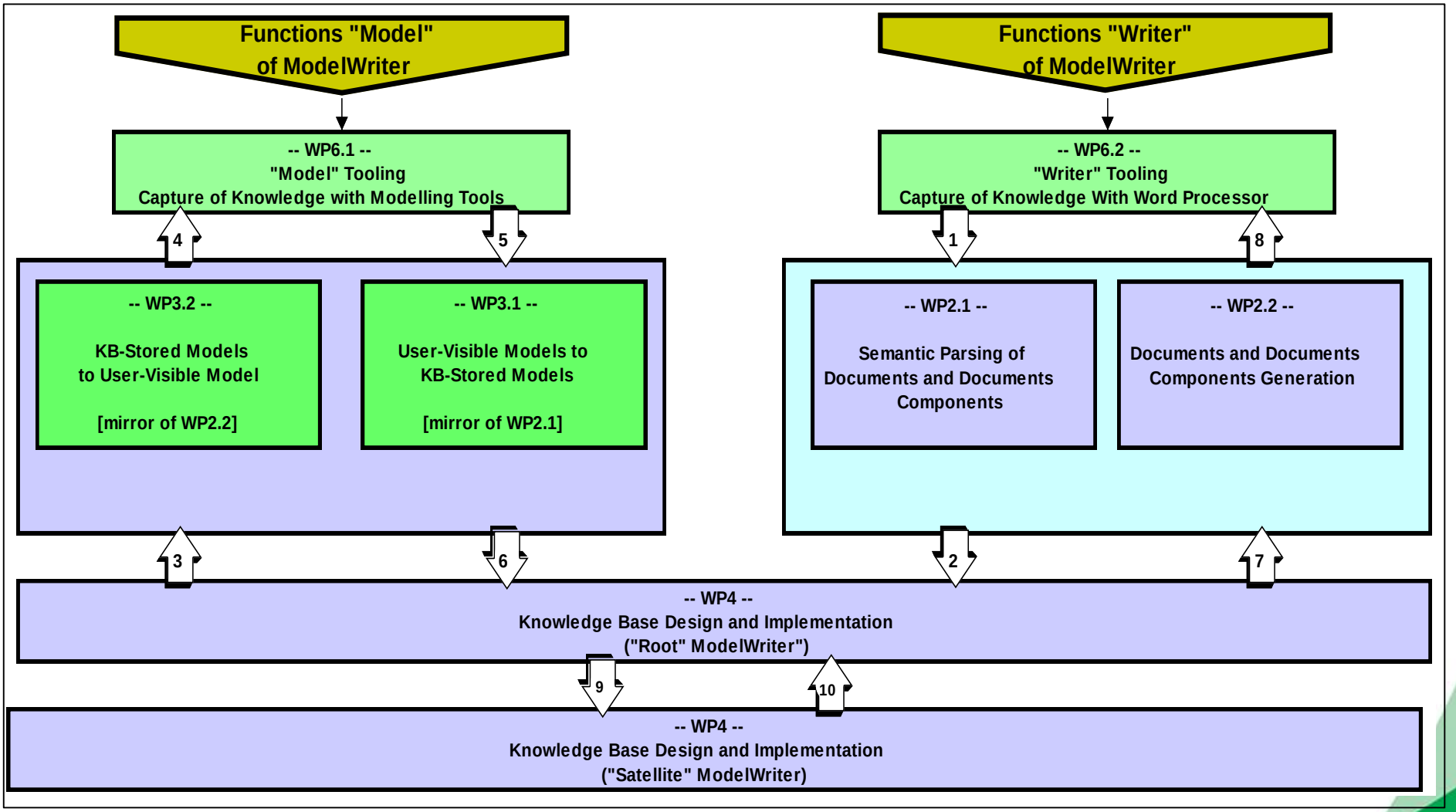
WP1 Industrial Use Cases and Requirements (AIRBUS)

WP2 (LORIA)	WP3 (UNIT)	WP4 (MANTIS)	WP6 (OBEO)
• Semantic Parser • Document Generation • bi-directional transformation between text and formal knowledge representation	• Bi-directional synchronization mechanism between texts and models • Configuration & Traceability Components • Consistency checker plug-in for consistency	• A federated Knowledge Base and its API • Synchronization mechanism between texts/models & knowledge base	• A complete “ModelWriter” tool integrating of all these in a consistent way • User Interfaces

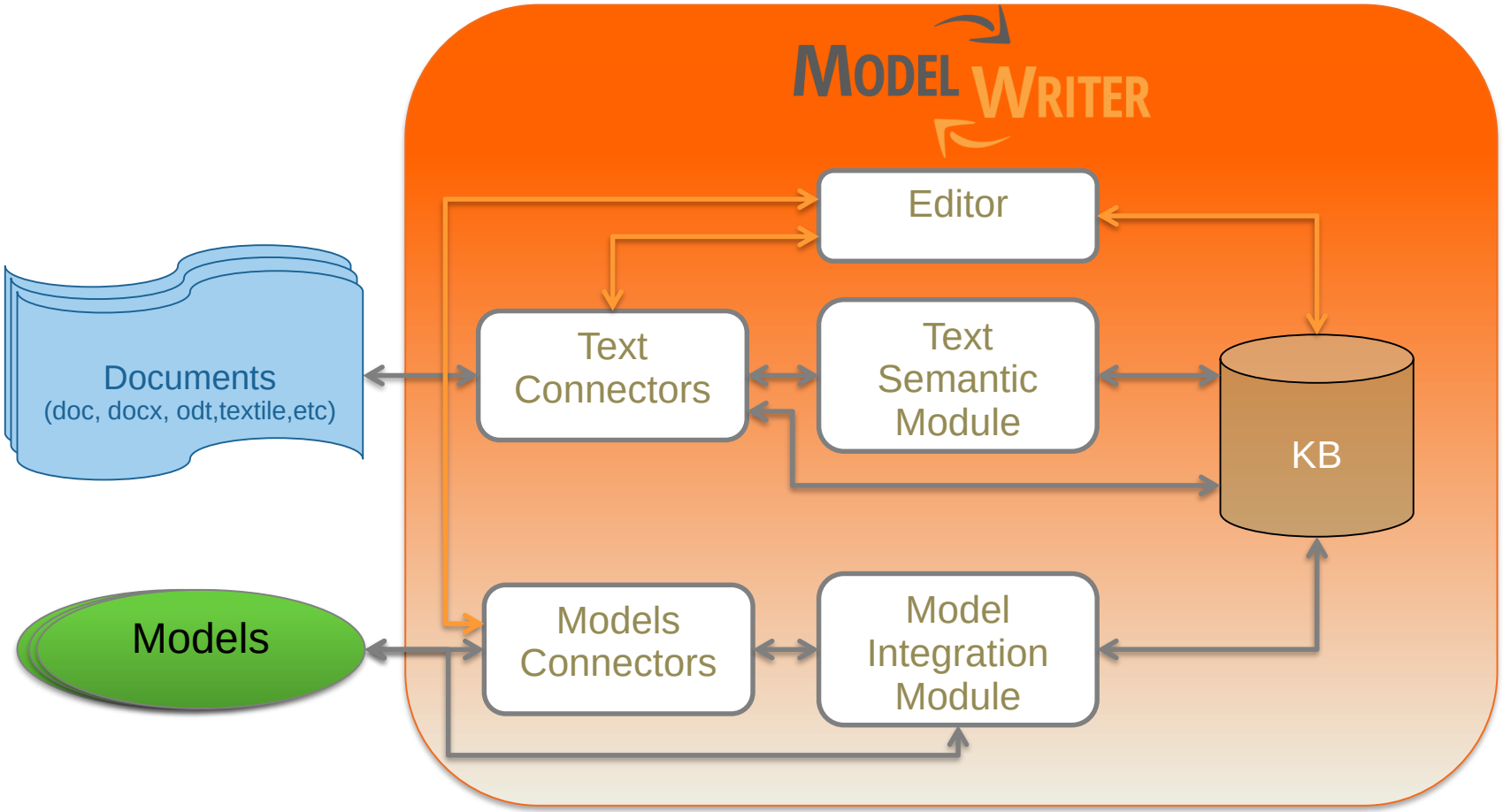
WP5 Project Management (UNIT)

WP7 Standardization, Dissemination and Exploitation (OBEO)

ModelWriter
Technological components & interactions



ModelWriter
Conceptual Architecture



Thank you for your attention
We value your opinion and questions.

DRAFT

Model Writer

Progress on the Industrial Use Cases

DRAFT

AIRBUS industrial use cases

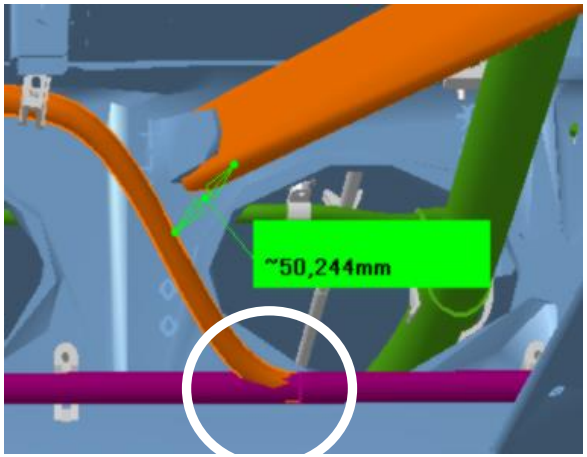
Anne MONCEAUX

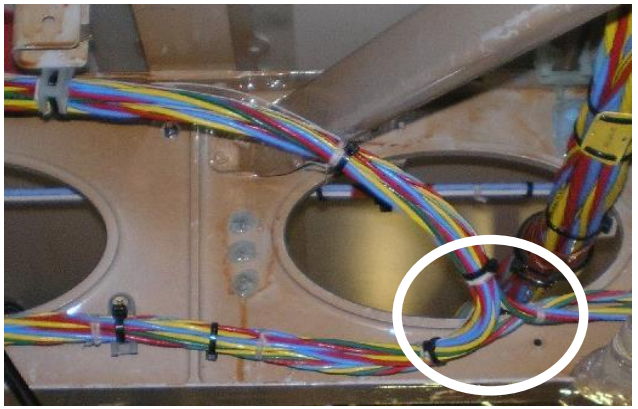
AIRBUS GROUP INNOVATIONS

Airbus use cases in ModelWriter

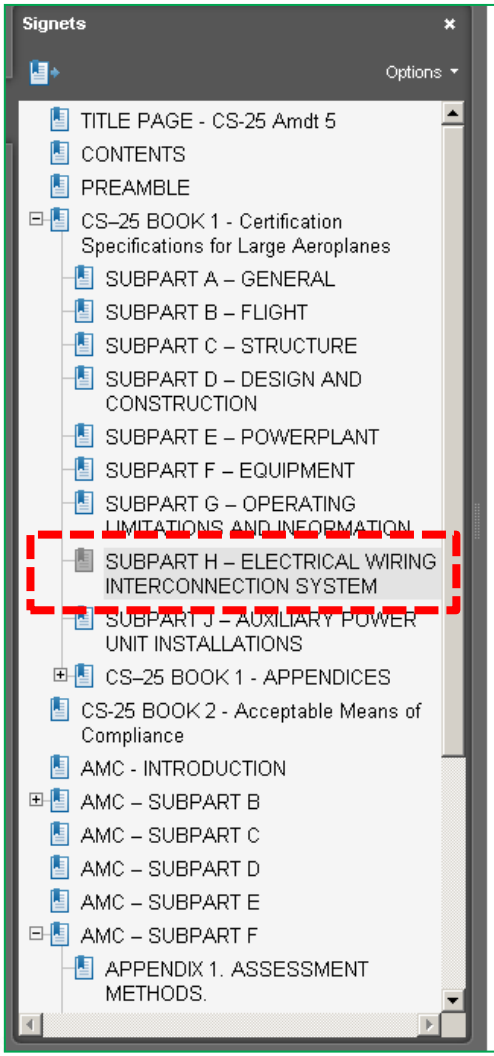
- We decided to concentrate on some problematics related to System Installation
- We primarily focus on Electrical, Water Waste, and Fuel Systems

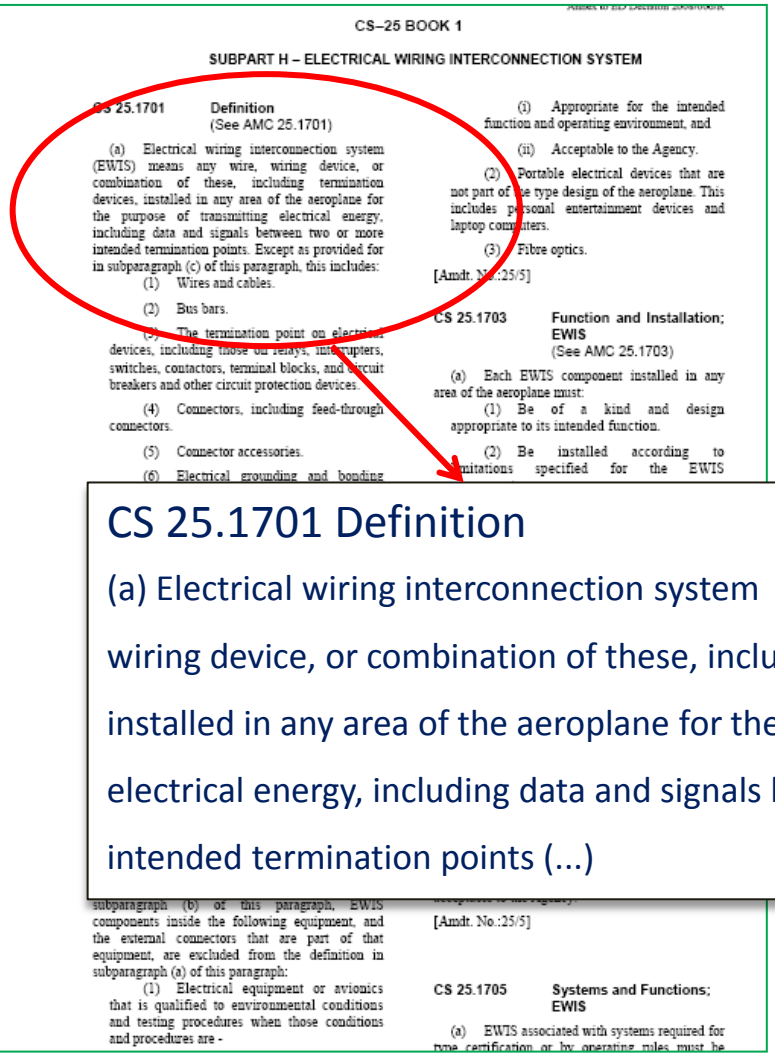
- System installation:
  - obeys to some Regulations
  - is specified by some rules/requirements/principles
  - the design and the manufactured product are checked against these rules





Certification Specification – CS 25 (EASA)





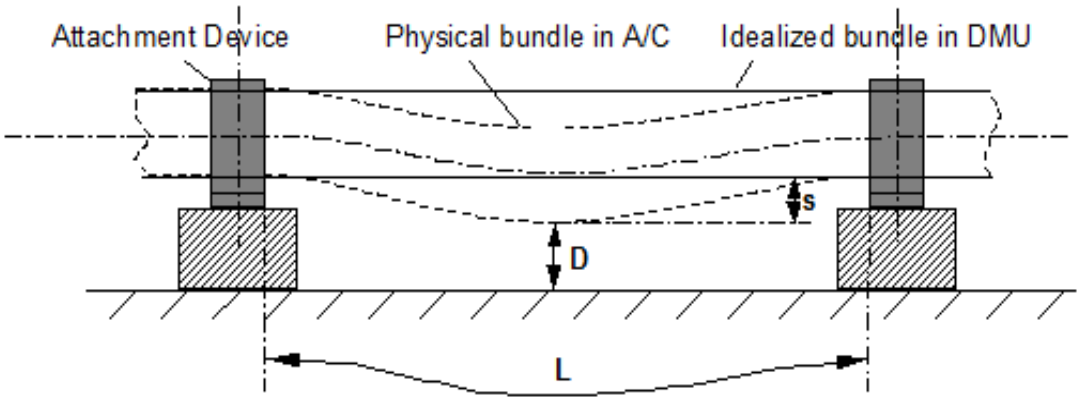
CS 25.1701 Definition

(a) Electrical wiring interconnection system (EWIS) means any wire, wiring device, or combination of these, including termination devices, installed in any area of the aeroplane for the purpose of transmitting electrical energy, including data and signals between two or more intended termination points (...)

System Installation Design Principles

SIDP92A001V-A-784

For installation of optical and electrical harnesses additional clearance for sagging (s) shall be provided as detailed below:



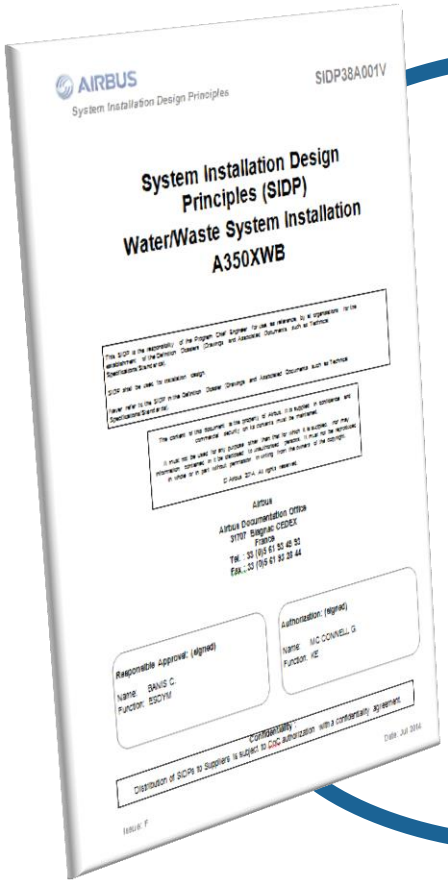
s...Sagging of bundle (real behavior of physical bundle in A/C due to gravity, ageing, etc.)
D...Required Distance
L...Actual length of a bundle segment between two Attachment Points (as designed in DMU)

Figure 6: Sagging of bundles between attachment points

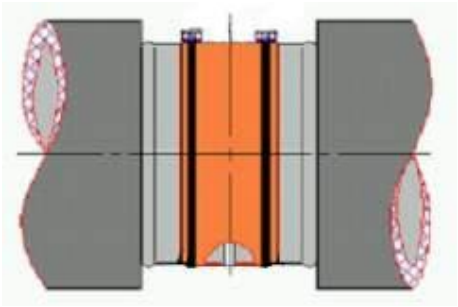
Note: Unless the bundle has a straight routing, L is bigger than the pitch between the Attachment Points.

5

Rule traceability & design check



Traceability



Check

Problem statement & objectives

1. Model and manage the design rules: since SIDP (design principles) are todays kinds of natural language requirements and explanations:
   - not all the rules can be directly formalized in a way to be used to verify the design
   - rules are spread over various products/programs
   - rules evolve
   - rules are complex artefacts made of text and picture and tables….

   ⇒ This problem is the one we have been considering primarily in the scope of ModelWriter, trying to define a rule model and knowledge base

2. Enable mapping between rule model and design model, in order to automate identification of conflicts between rules and design, and then automate analysis of the impact of changes in rules or in design.

   ⇒ This 2d objective we think cannot be reached in the frame of Modelwriter because of the challenge of mapping 3D design models to the BOM … and because lack of skills and data in the context of the project

   BUT to be consider to make the system amenable for formal analysis

Problem statement & objectives

(1) Manage rules/design principles and improve traceability	
Easier search/retrieval of rules	- save costs/time by supporting DP consultation / retrieval in accordance to the needs
Easier change impact analysis at rules level	- save DP updating costs/time - avoid non-compliance of design caused by DP updating lead time
Easier traceability between artefacts (BOM DB, design models, CAD models…)	- keep traceability from upstream regulations to requirements and to downstream design models
(2) Automate identification of design conflicts against rules	
Easier consistency checking of the design (CAD) data	- Save time to retrieve applicable rules - avoid non-compliance of design wrt rules

Solutions explred in MW

		1	2	3	4
Use a Component taxonomy (BOM) as a reference to annotate rules (rules as Text part / Taxonomy as Model part)	MW synchronization mechanism to help annotation management	+	+	-	-
Extend first approach to other artefacts/models	MW synch between several artefacts	-	-	+	+
Add semantics to BOM to enable reasoning	MW synch	+	+	+	
Model rules written in natural language by formalizing the rules in description logic / ability to dynamically update the BOM/component ontology and apply some checks on design rules	MW writer part features that LORIA developed so far in the formalization process + MW synch			+	+
Model rules written in natural language by formalizing the rules in relational logic	MW writer part + Tarsky part + Synch mech			+	+

1=Rule retrieval – 2=Rule change – 3= traceability – 4=design check

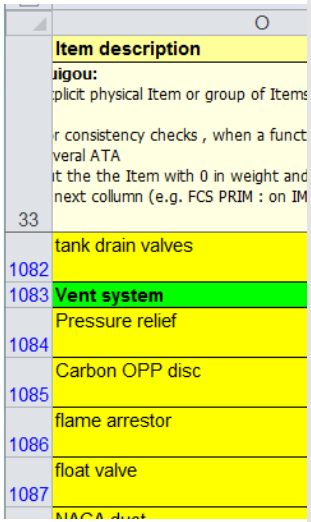
DRAFT

UC-FR-03 - Synchronization of Regulation Documentation with a Design Rule Repository

Anne MONCEAUX
Airbus Group Innovation

ITEA 3

Ex. BOM, Component taxonomy, Component ontology

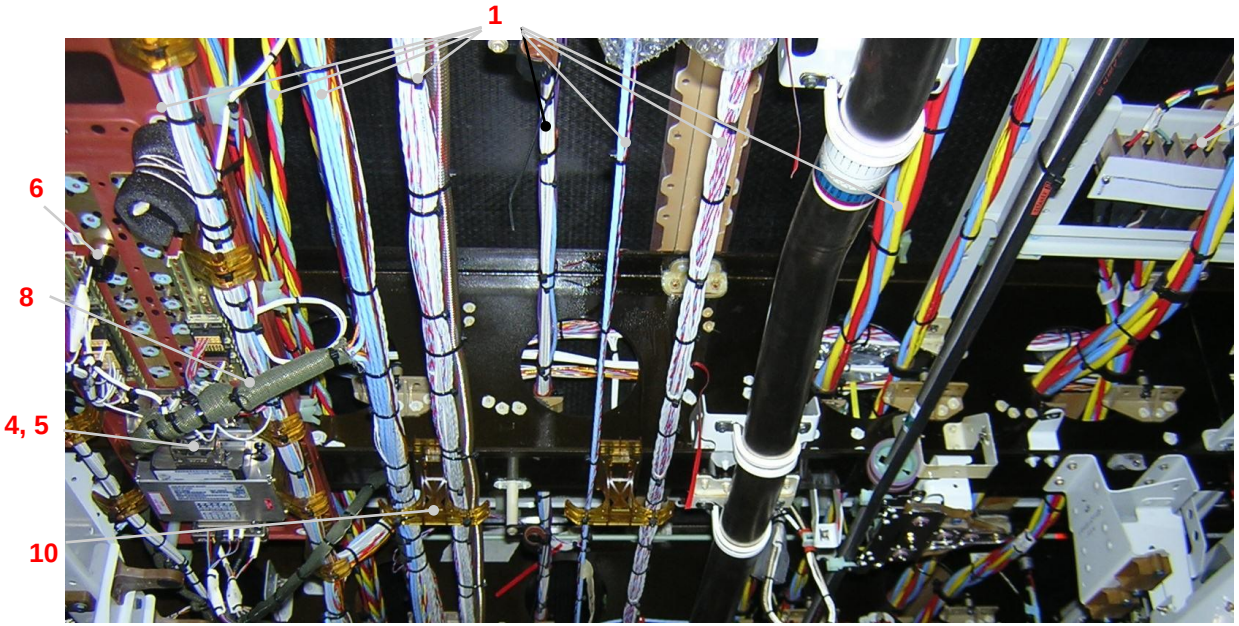


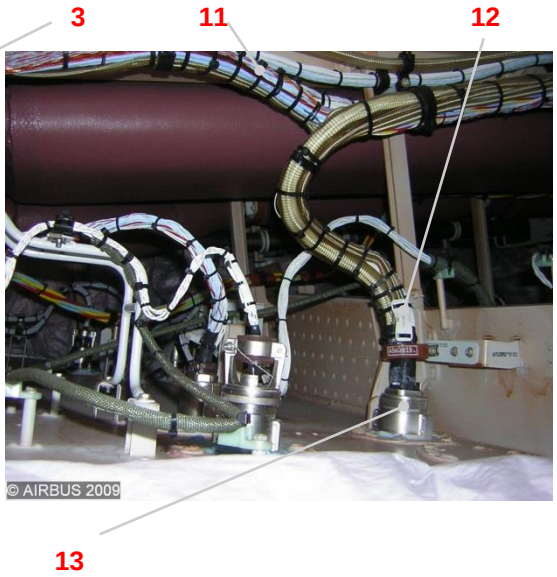
	O
	Item description
	ıigou:
	:plicit physical Item or group of Items
	r consistency checks , when a funct
	veral ATA
	ıt the the Item with 0 in weight and
	next collumn (e.g. FCS PRIM : on IM
33	
1082	tank drain valves
1083	**Vent system**
1084	Pressure relief
1085	Carbon OPP disc
1086	flame arrestor
1087	float valve

```
.
<http://airbus-group.installsys/component>
  rdf:type owl:Ontology ;
  rdfs:comment "Component definitions are given and validated by Airbus ESIR dpt"^^xsd:string ;
  owl:imports <http://qudt.org/schema/qudt> ;
  owl:imports <http://qudt.org/vocab/unit> ;
  owl:versionInfo "Ontology for System Installation Components"^^xsd:string ;
.
comp:ABS1759
  rdf:type owl:Class ;
  rdfs:label "ABS 1759 cable tie mount"^^xsd:string ;
  rdfs:label "ABS1759"^^xsd:string ;
  rdfs:label "cable tie mount ABS1759"^^xsd:string ;
  rdfs:subClassOf comp:CableTieMount ;
  skos:prefLabel "ABS 1759 cable tie mount"@en ;
.
comp:AFDXcable
  rdf:type owl:Class ;
  rdfs:label "AFDXcable"^^xsd:string ;
  rdfs:subClassOf comp:BusCable ;
  skos:prefLabel "AFDXcable"^^xsd:string ;
.
comp:Active-Fastener
  rdf:type owl:Class ;
  rdfs:label "active fastener"^^xsd:string ;
  rdfs:subClassOf comp:Fastener ;
  rdfs:subClassOf [
      rdf:type owl:Restriction ;
      owl:hasValue "true"^^xsd:boolean ;
      owl:onProperty comp:isActive ;
    ] ;
  skos:prefLabel "active fastener"@en ;
.
comp:Active_component
  rdf:type owl:Class ;
  rdfs:label "active component"@en ;
  rdfs:label "active element"@en ;
  rdfs:label "active item"@en ;
```

Electrical wiring interconnection system (EWIS)

⇒ Mainly: **Aircraft electrical common installation (ATA92)**

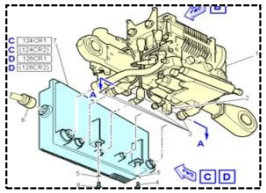


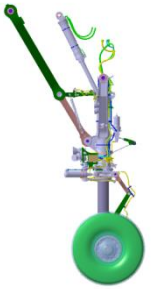


⇒ But also: **part of Systems equipment wiring:**

- Power distribution centers wiring
- External wiring of equipment
- Wiring of equipment not qualified to appropriate standards e.g. EUROCAE ED-14 / RTCA DO-160

cf CS 25.1701(b)&(c)





Illustrate traceability between all aspects of A/C wiring

DRAFT

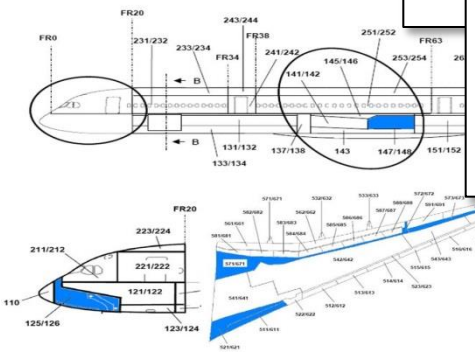
EWIS components design,

EWIS architecture - routes defi

EWIS installation design pr
requirements

Systems functional des
Systems equipment de

Environmental c



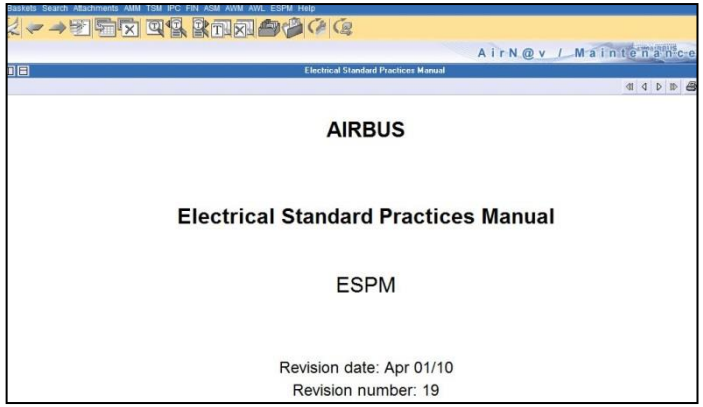
Safety analyses

EWIS definition process, V&V

Interfaces

Manufacturing procedures

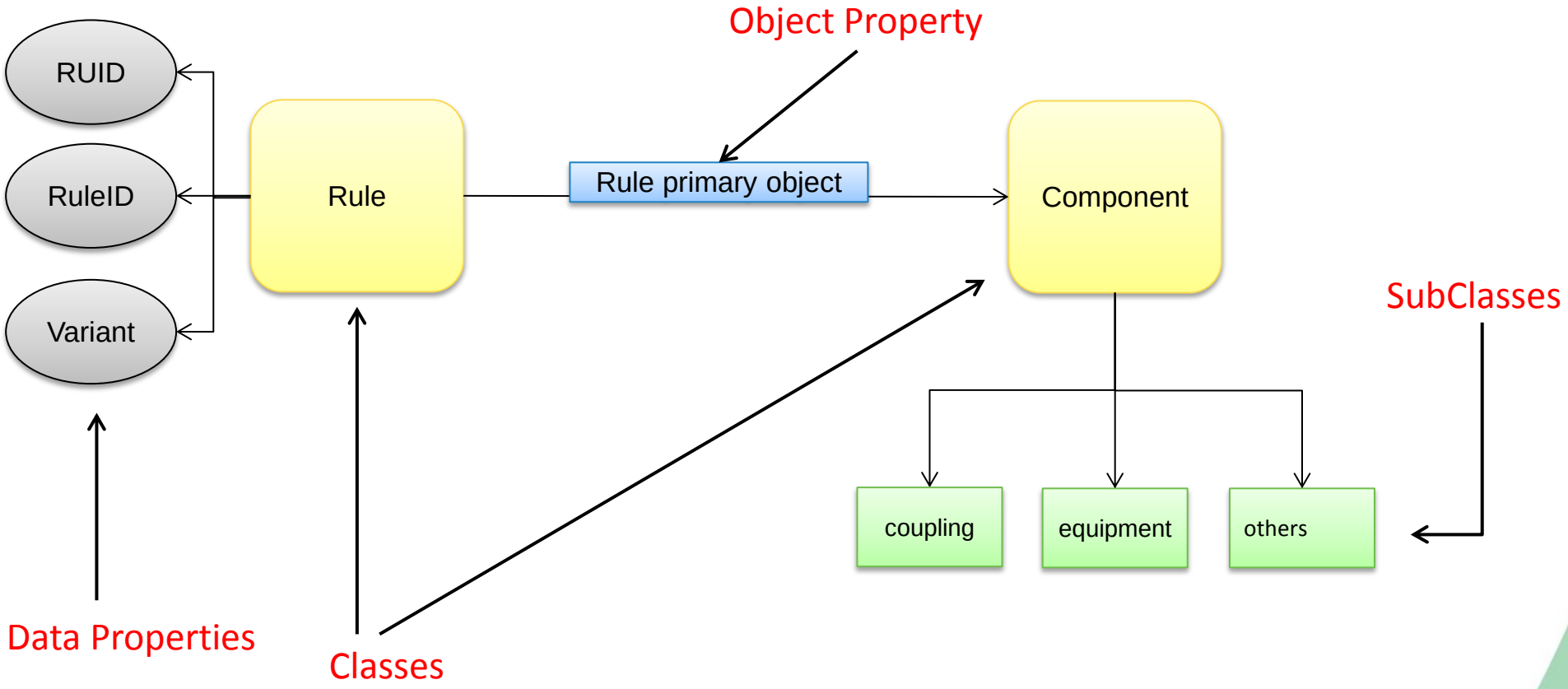
EWIS ICA, maintenance instructions



Etc…

Back-up slides

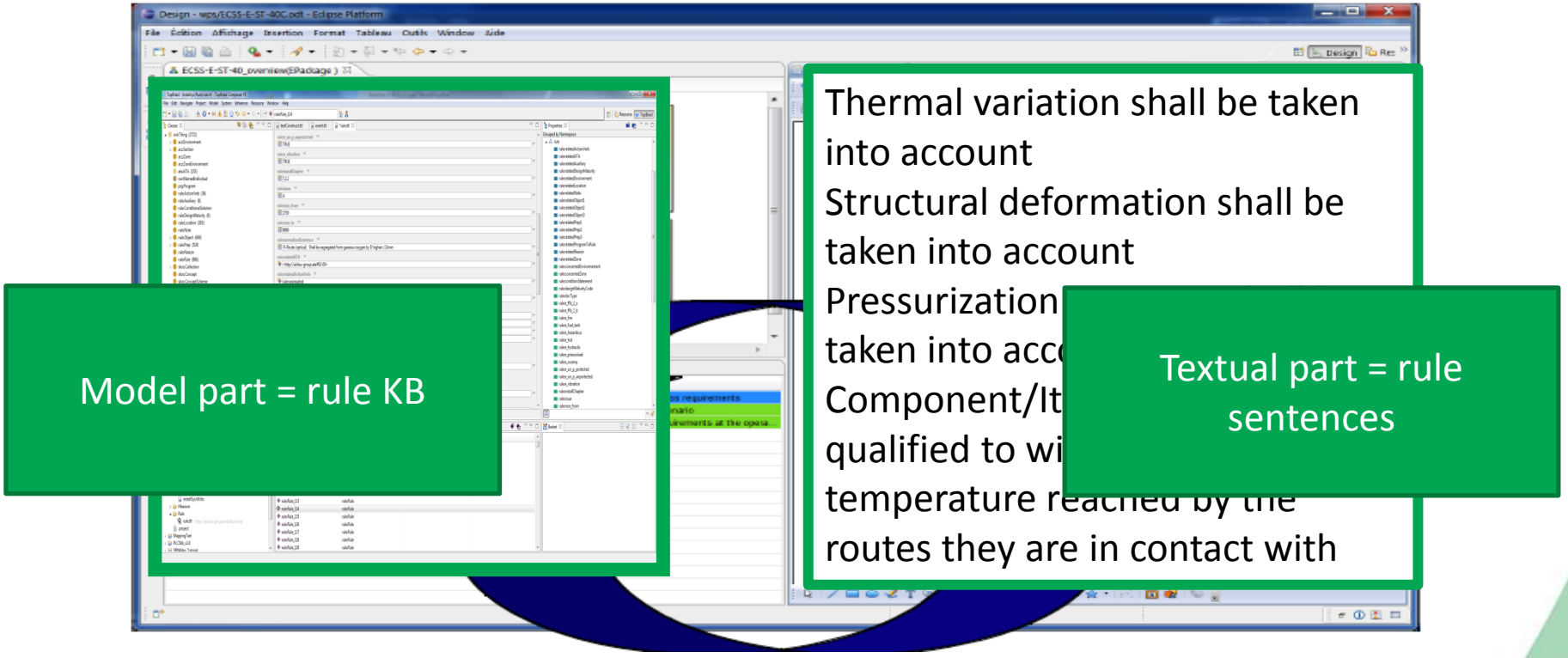
Représentation des données



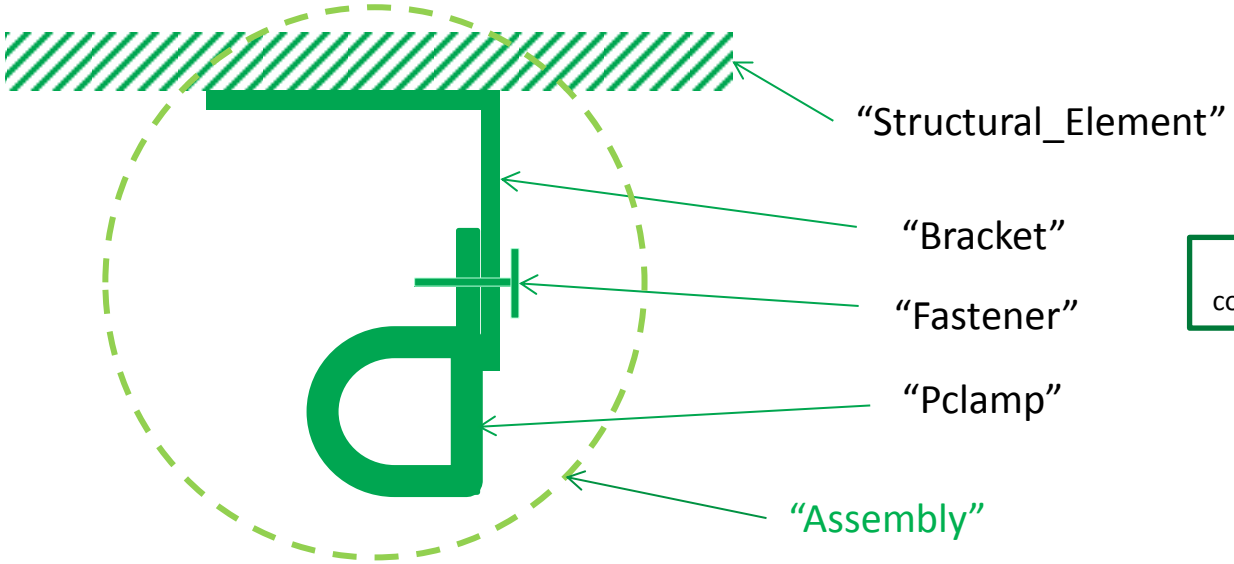
15

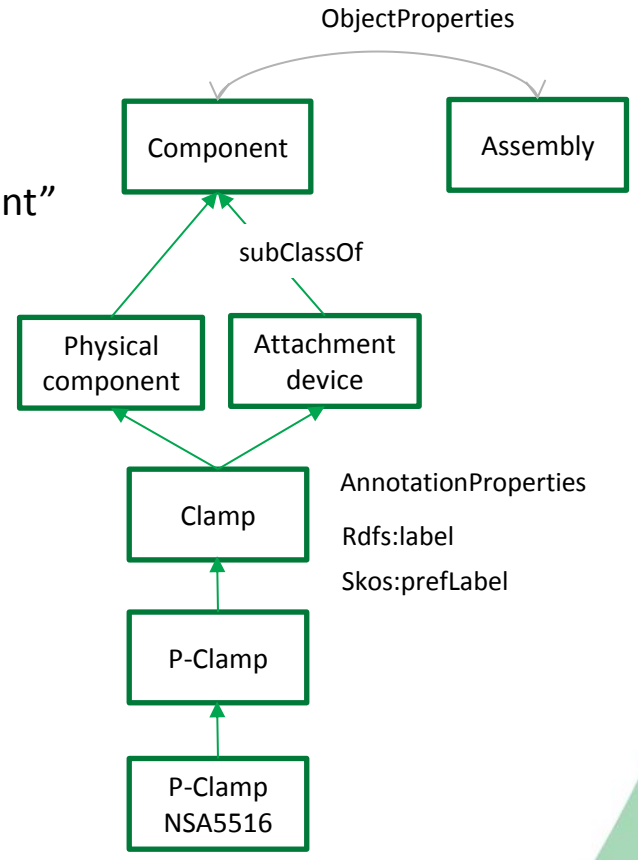
UC-FR-03 Synchronization of regulation documentation with a design rule repository

Status 1st iteration: synchronizing the SIDP KB content and text part

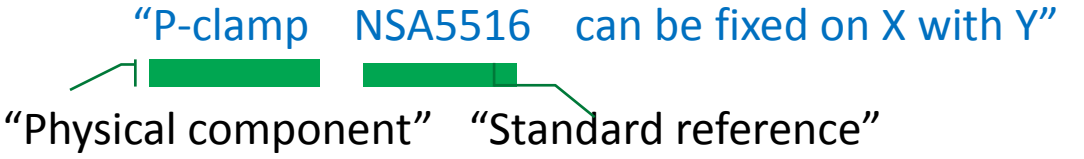


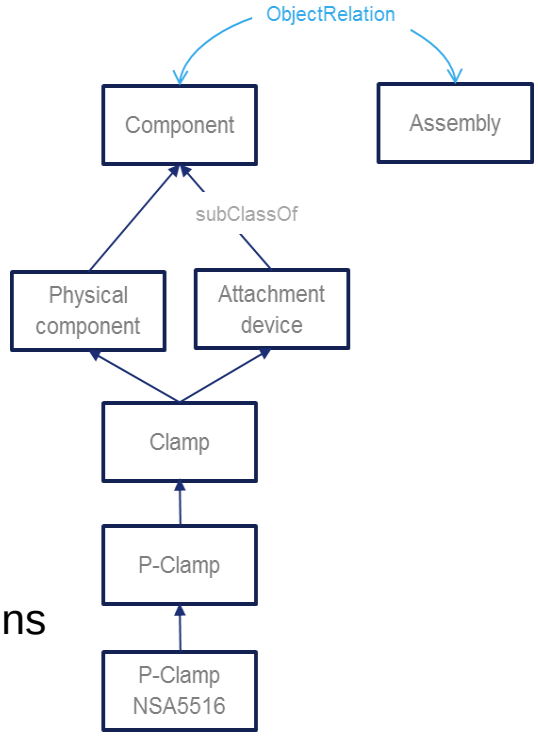
Component classes taxonomy





Component classes taxonomy



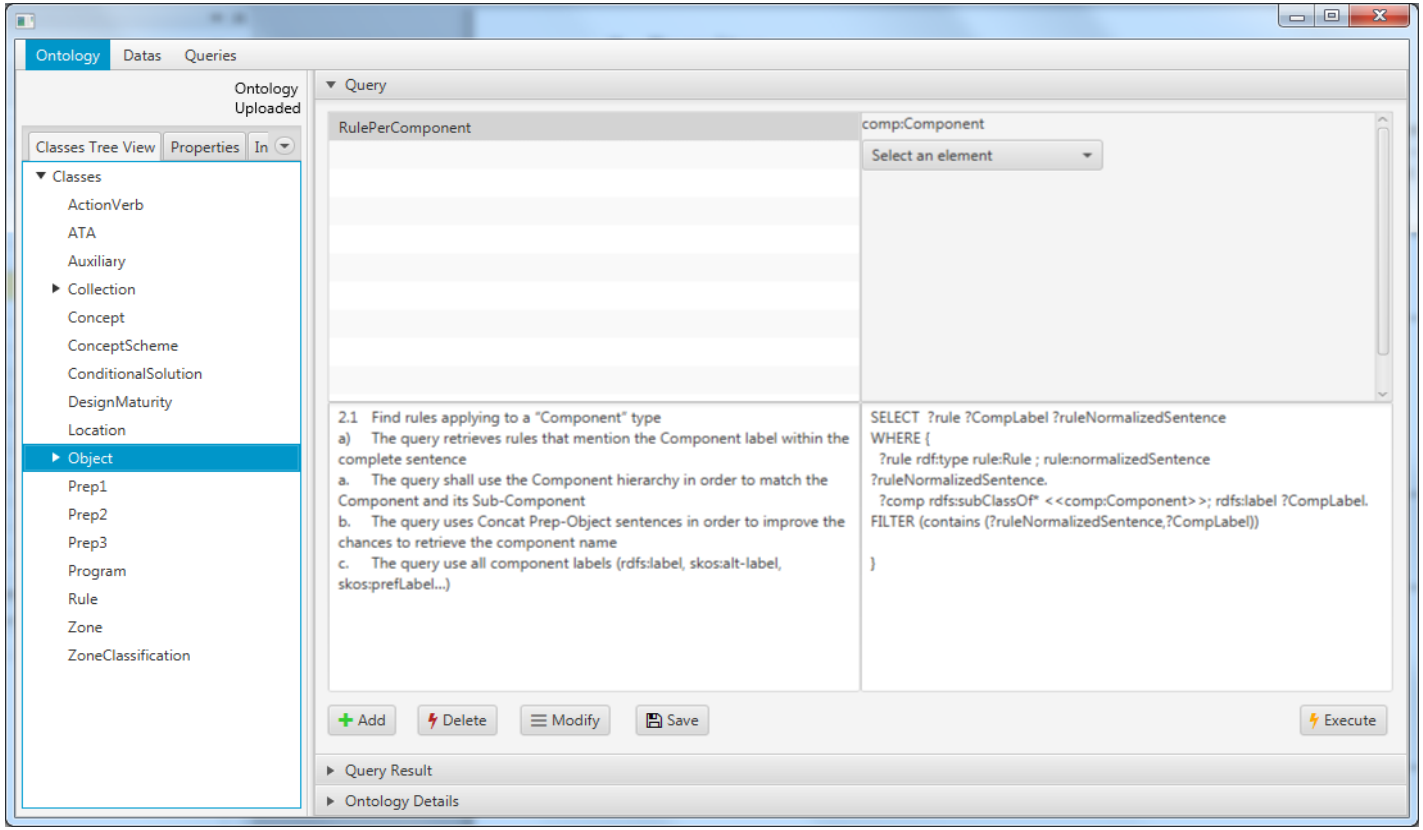


- NLP Parsing uses this taxonomy. Labels + assumptions such as a physical component may be referred to using its name or its reference or both concatenated
- Inference rule: a rule applying to a component type (Attachment device) applies to its subtypes (P-clamp)
- Document display : when searching rules applying to a component type (P-Camp) → retrieve and display rules applying to super-types

SPARQL queries

Status : model based queries specification

- Preliminary study with Loria (Text to RDF)
- 1 internship on RDFizer and Query management



DRAFT

Thank you for your attention!
Any question ?

ITEA 3

FORD OTOSAN

Model Writer

FEAD (Front End Accessory and Drive) & EGR (Exhaust Gas Recycling)

06.11.2016

SYNCHRONIZATION OF DESIGN SPECIFICATIONS WITH CAD DATA

Design Rules, BoM and CAD Data

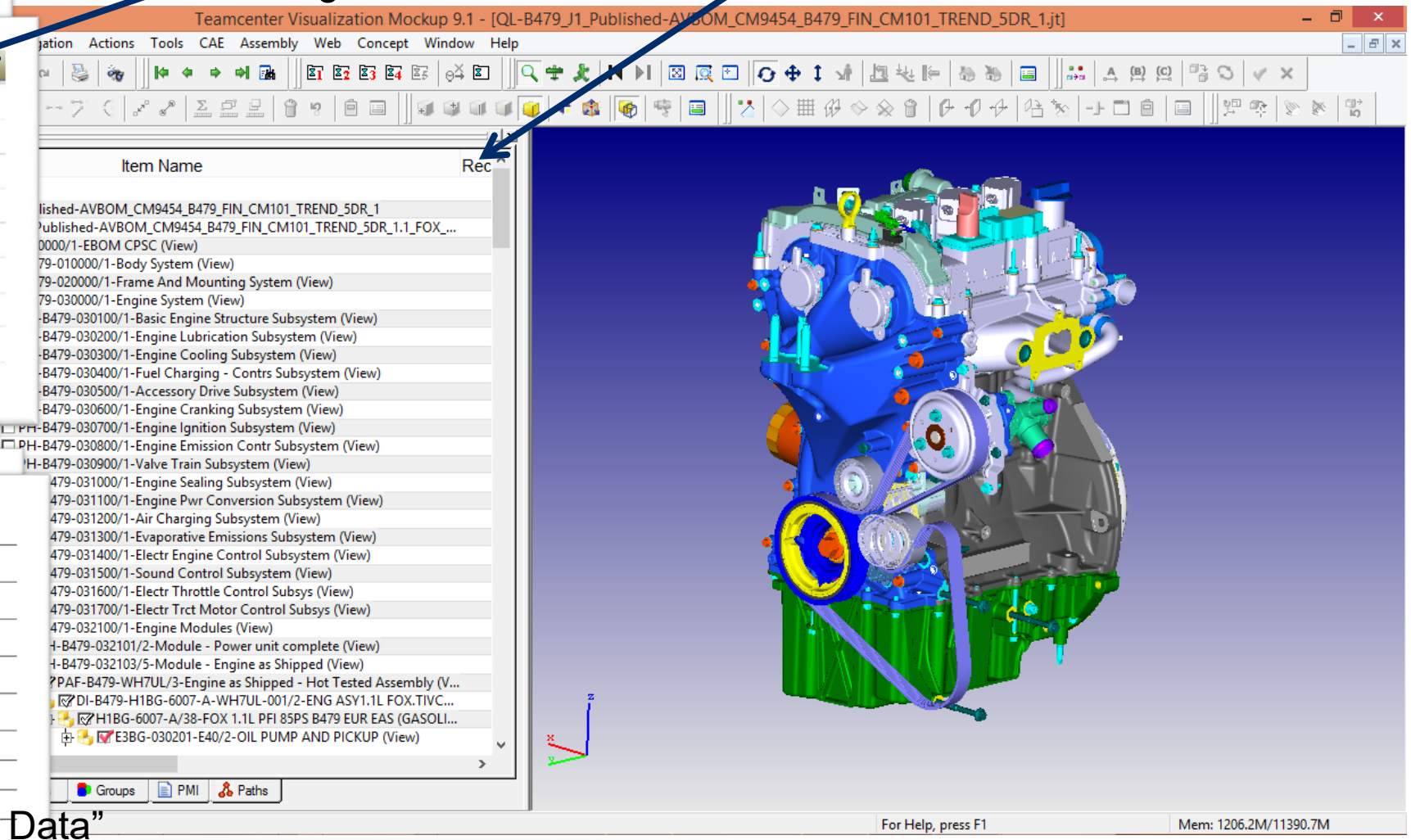
“Design Rules”

“Computer-aided Design Data”

KPAC Set ID	Title	Author	Type
PT_FAST-S1	Cup Plug Design Rule Saved Set	Murphy, Mark (mmurphy9)	Check List
PT_FAST-S1	Dryseal Pipe Thread Design Rule Saved Set	Murphy, Mark (mmurphy9)	Check List
PT_FAST-S1	Dowel and Bushing Design Rule Saved Set	Murphy, Mark (mmurphy9)	Check List
PT_FAST-S1	Fastener and Joint Design Rule Saved Set	Murphy, Mark (mmurphy9)	Check List
PT_FAST-S1	Threaded Fastener Design Rule Saved Set	Murphy, Mark (mmurphy9)	Check List
PT_FAST-S1	Wrench and Socket Design Rule Saved Set	Murphy, Mark (mmurphy9)	Check List
PT_FAST-S1	Ball Plug Design Rule Saved Set	Murphy, Mark (mmurphy9)	Check List
PT_FAST-S1	Port Plug Design Rules Saved Set	Murphy, Mark (mmurphy9)	Check List
PT_FAST-S1	Thread Forming Screws in Plastic Saved Set	Murphy, Mark (mmurphy9)	Check List

Prefix*	Base*	Suffix*	Name*	CPSC*	Engineering Commodity	Remarks (Fastener Attaching Statement)
	W500025	S437	BOLT M8X30 HF 8	030501	FASTENERS (PMT400)	FROM: 6A228; TO: 6059;
	W500214	S437	BOLT M6X20 HF PIL 8	030501	FASTENERS (PMT400)	FROM: 19A216; TO: 6059 ;
	W704693	S442	BOLT M8X95 HF PIL 8	030501	FASTENERS (PMT400)	FROM: 10300; TO: 6059 & 6015 ;
	W704693	S442	BOLT M8X95 HF PIL 8	030501	FASTENERS (PMT400)	FROM: 19D259; TO: 6059 & 6675 ;
	W714914	S303	BOLT M8X13 HF SERR STL	030501	FASTENERS (PMT400)	FROM: 8509; TO: 6059;
C1BQ	8509	AC	CVR-ACC DRV	030501	FEAD	
CM5Q	19A216	AB	IDL ASY ACC DRV BEL	030501	FEAD	
CM5Q	6A228	BA	TENS ASY ACC DRV	030501	FEAD	

“Bill of Material Data”

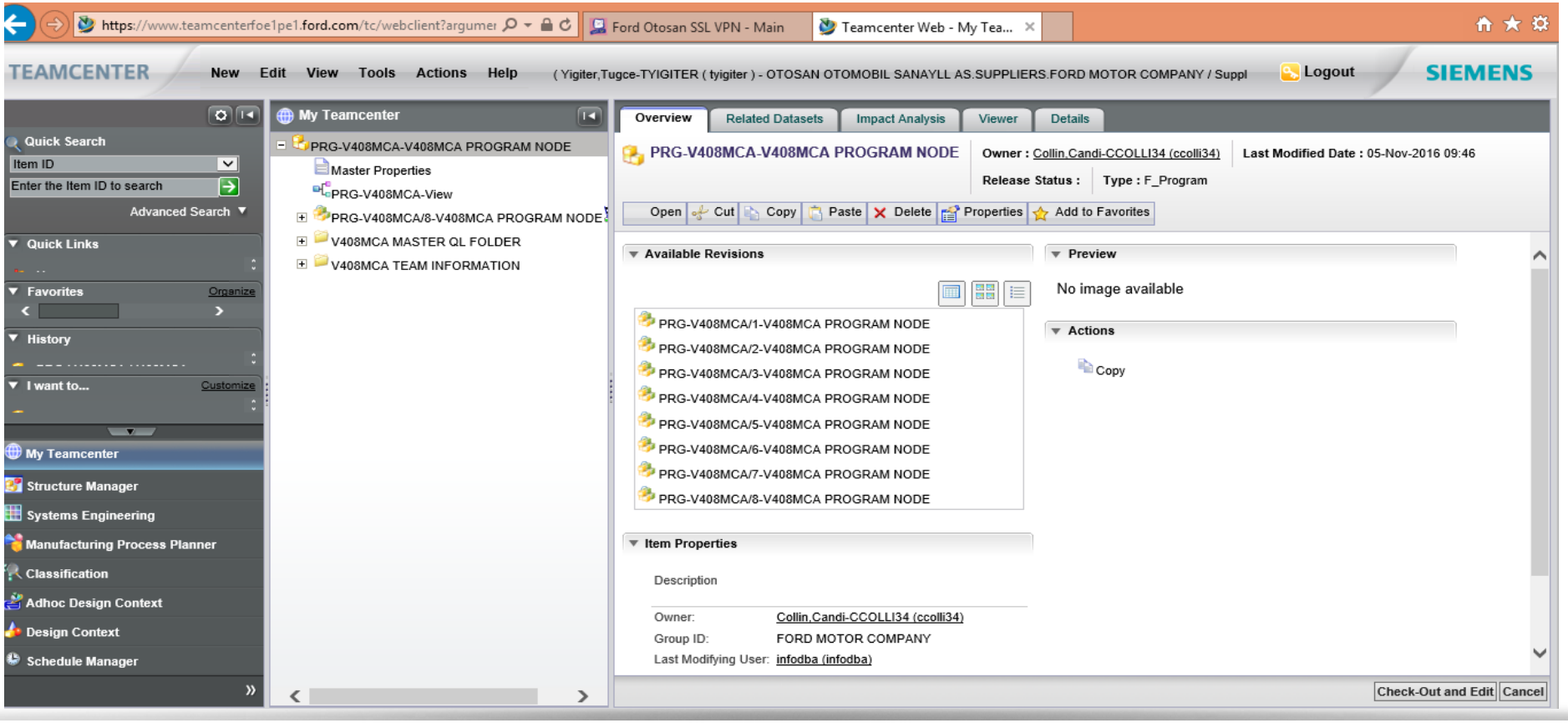


SIEMENS COLLABORATION WITH FORD

Siemens – TeamCenter

Ford Otosan is using Siemens Team Center for sharing and uploading the internal documents. Siemens has 6 server for Ford. Ford Otosan is currently using Ford of Europe (FOE) servers.

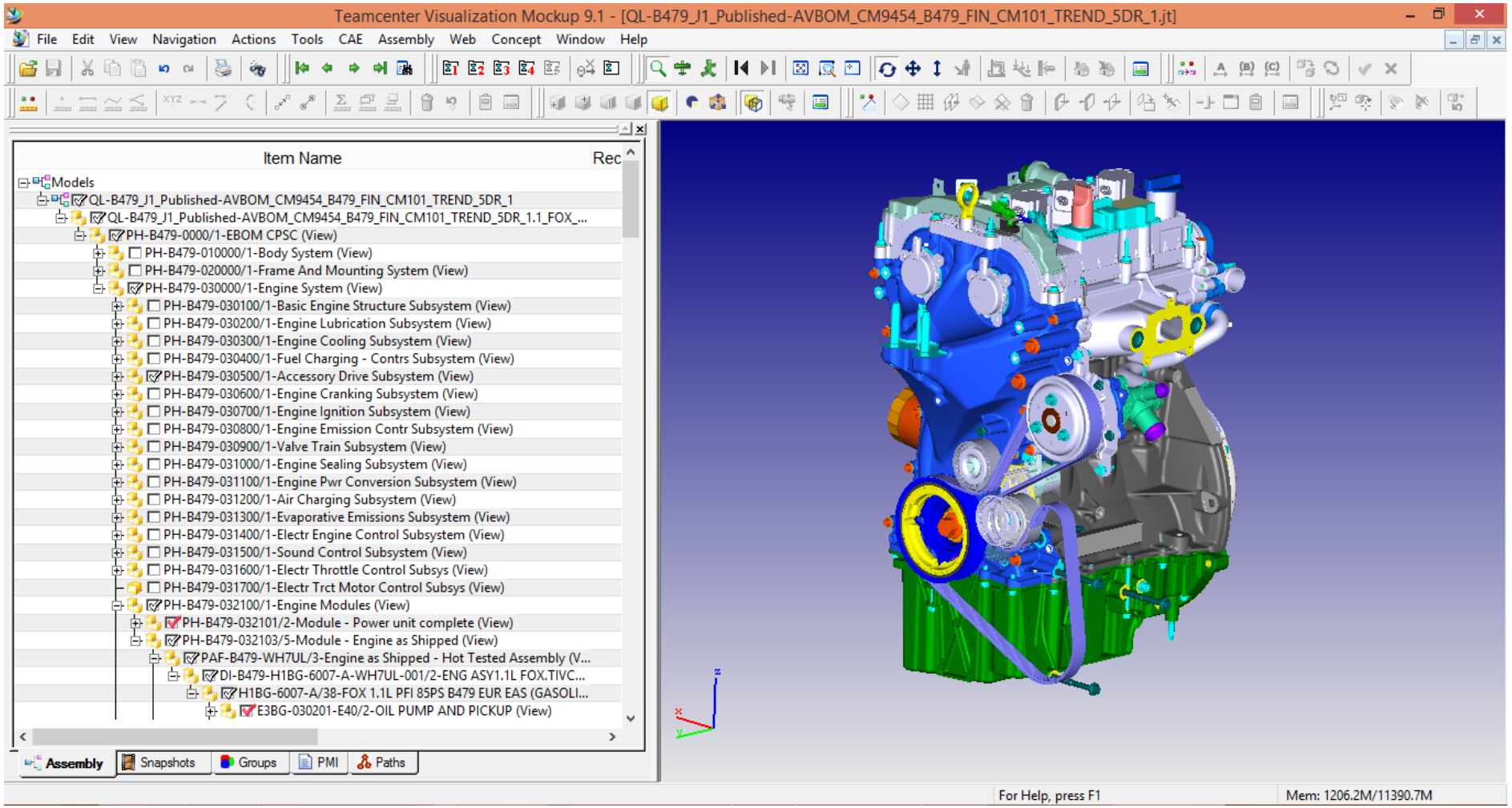
https://www.teamcenterfoe1pe1.ford.com/tc/webclient



USER INTERFACE FOR 3D MODELS

Siemens – TeamCenter Visualization Mockup 9.1

Ford Otosan is using Siemens TeamCenter Visualization Mockup 9.1 for displaying the documents.



SEVERAL FACTS ABOUT THE USE CASE

Facts, Impact, and KPIs

- Total number of rules in the whole system? About 2000 rules
- Expected Rule Coverage in the use cases? At least 50%
- Impact in FO: Total number of systems in which the tool can be applied? 70/80 teams (disiciplines)
- Principles for each disciplines (
  - EGR: Thermodynamics, Fluids Dynamic
  - FEAD: Mechanical Construction
  - Suspension: Mechanical Construction
  - Electronic: Electrical Engineering and Embedded Systems
- Total number of items in a vehicle with respect to CPSC system? About 10000 items per vehicle
- Performance Metrics
  - Expected Response time in seconds per each validation operation? At most 2 seconds
  - Server-side High Performance Computing is need for checking consistency and reasoning processes.
- Ergonomics
  - Siemens PLM Visualization Integration

FORD OTOSAN

Model Writer

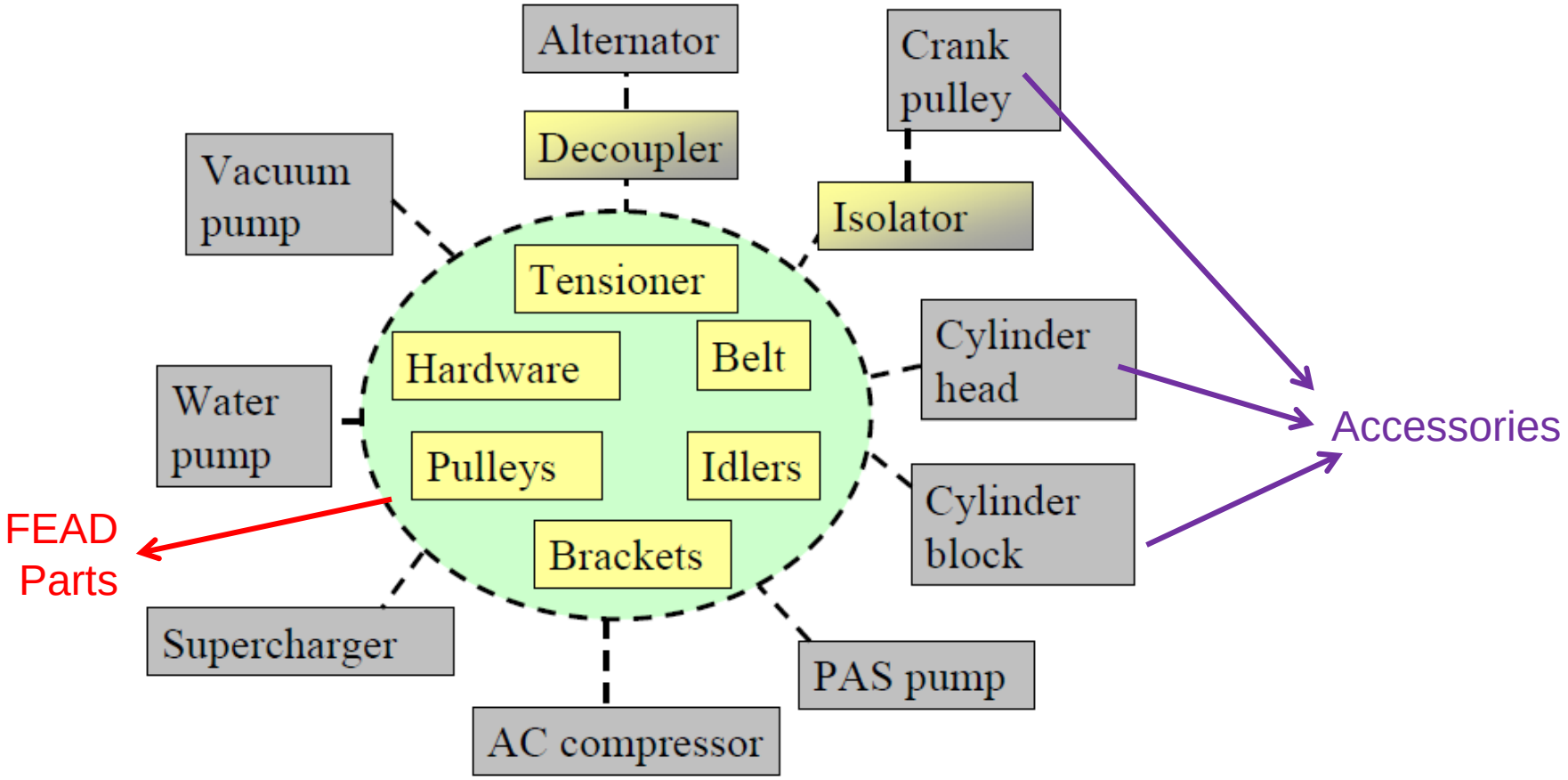
FEAD (Front End Accessory and Drive)

06.11.2016

WHAT'S FEAD?

Front End Accessory Drive Design

- The objective of good accessory drive design is to transmit power to turn the various accessories with maximum efficiency under all driving conditions, without the customer knowing the job is being done.



FEAD PARTS

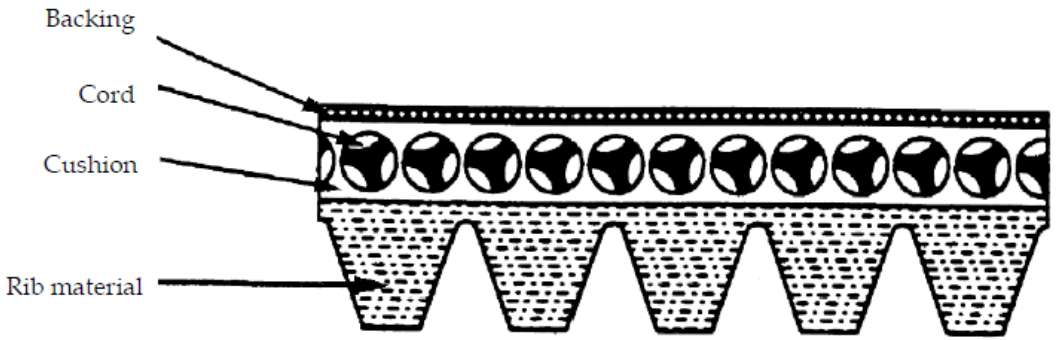
Tensioner

- The function of a tensioner is to control system tension at a sufficient level to prevent belt slip under any driving condition.

Manual Rotary - automatic Strut - automatic

Belt

- The belt transmits the driving power from the crankshaft to the accessories.



FEAD PARTS

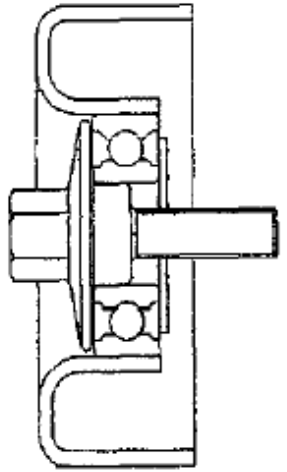
Idler Pulleys

- An idler pulley is introduced into the system to guide the belt to achieve the following requirements:

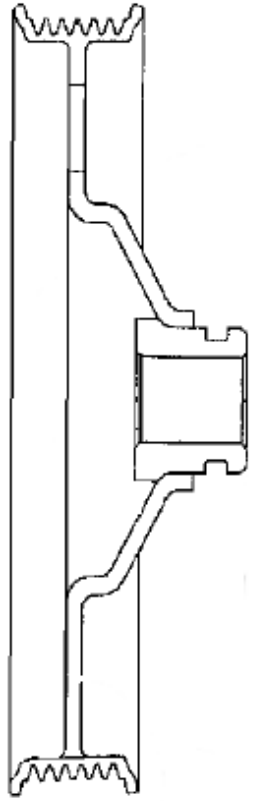
a) Reposition the belt path to avoid obstacles,

b) Create more belt wrap on adjacent pulleys,

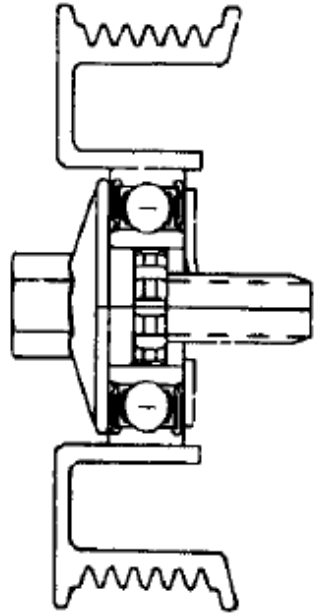
c) Control vibration on long spans.



Flat idler



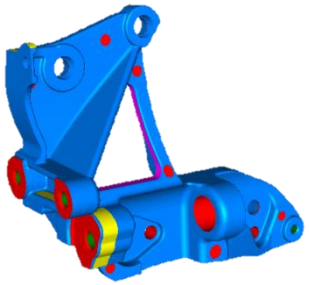
Grooved pulley

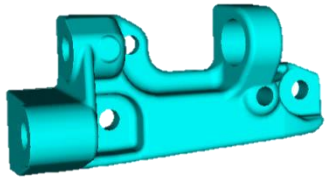


Grooved idler

Brackets

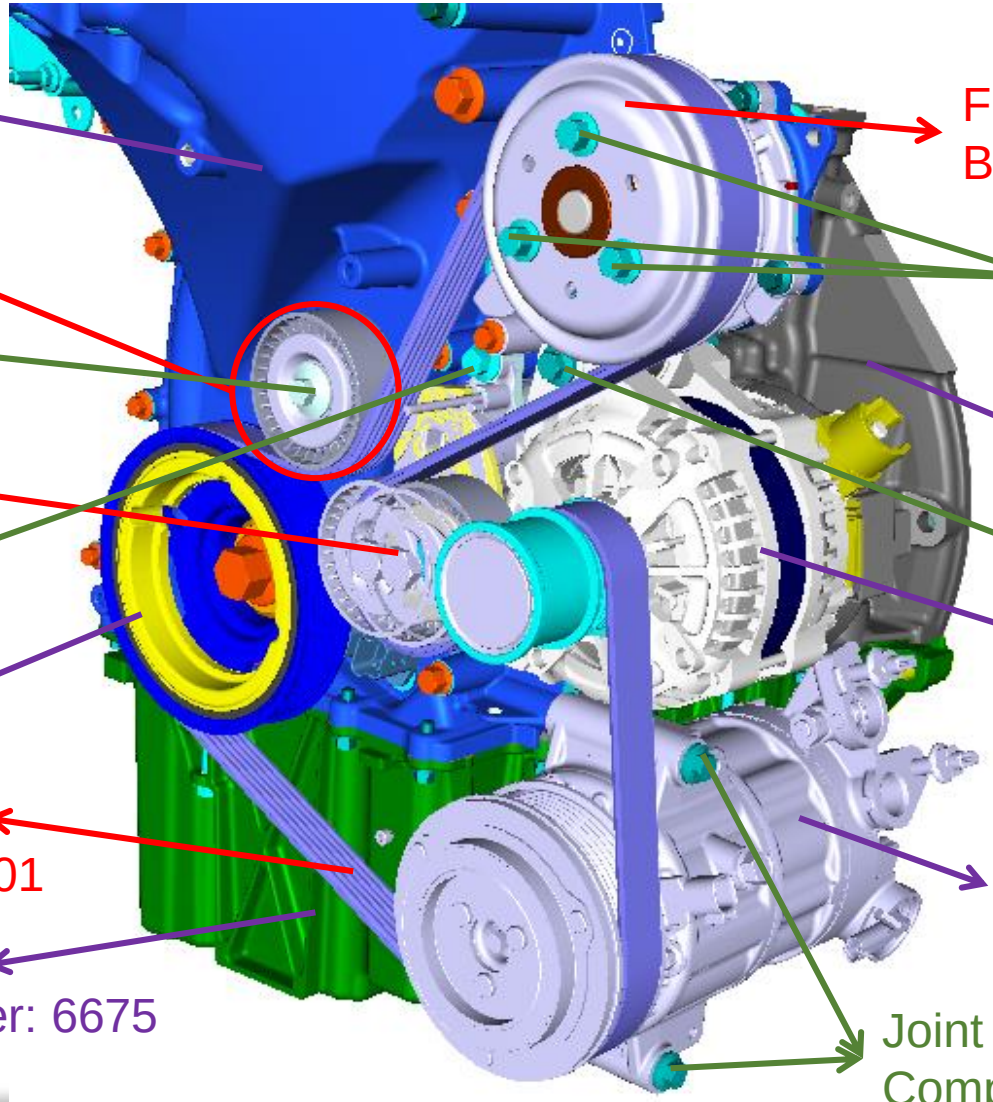
- Creates suitable positioning points for accessories.





RELATİON BETWEEN FEAD PARTS

Please see the FEAD part (red and green) and the other related parts (purple) layout below.



Front Cover
Base Number: 6059

FEAD Idler Base
Number: 19A216

Joint of Idler

FEAD Tensioner
Base Numbe: 6A228

Joint of Tensioner

Crankshaft Pulley
Base Number: 6316

FEAD Belt
Base Number: 6C301

Oil Pan
Base Number: 6675

FEAD Water Pump Pulley
Base Number: 8509

Joint of
WP Pulley

Cyclinder Block
Base Number: 6015

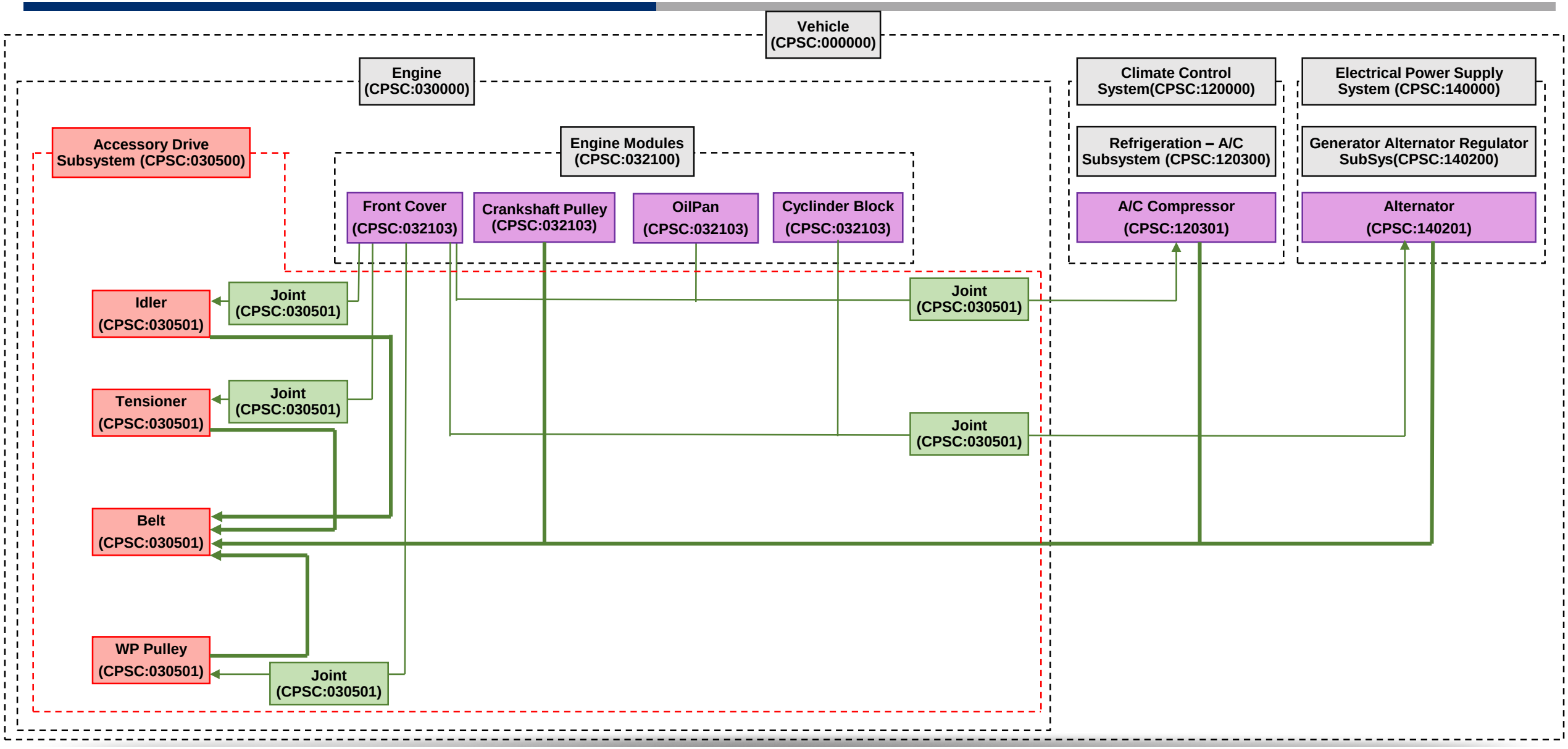
Joint of Alternator

Alternator
Base Number: 10300

AC Compressor
Base Number: 19D629

Joint of AC
Compressor

BOUNDARY DİAGRAMS OF THE PARTS



FEAD PART LIST AND REQUIRED INFORMATIONS

Prefix*	Base*	Suffix*	Name*	CPSC*	Engineering Commodity	Remarks (Fastener Attaching Statement)
	W500025	S437	BOLT M8X30 HF 8	030501	FASTENERS (PMT400)	FROM: 6A228; TO: 6059;
	W500214	S437	BOLT M6X20 HF PIL 8	030501	FASTENERS (PMT400)	FROM: 19A216; TO: 6059 ;
	W704693	S442	BOLT M8X95 HF PIL 8	030501	FASTENERS (PMT400)	FROM: 10300; TO: 6059 & 6015 ;
	W704693	S442	BOLT M8X95 HF PIL 8	030501	FASTENERS (PMT400)	FROM: 19D259; TO: 6059 & 6675 ;
	W714914	S303	BOLT M8X13 HF SERR STL	030501	FASTENERS (PMT400)	FROM: 8509; TO: 6059;
C1BQ	8509	AC	CVR-ACC DRV	030501	FEAD	
CM5Q	19A216	AB	IDL ASY ACC DRV BEL	030501	FEAD	
CM5Q	6A228	BA	TENS ASY ACC DRV	030501	FEAD	
CM5Q	6C301	BA	BEL ACC DRV	030501	FEAD	

FORD OTOSAN

Model Writer

EGR (Exhaust Gas Recycling) System

06.11.2016

EGR (EXHAUST GAS RECYCLING) SYSTEM

- ***EGR – Exhaust Gas Recycling*** is a method of controlling combustion through the addition of exhaust gases into the intake side of an internal combustion engine.

In its simplest form an EGR system comprises of the following steps

1. Exhaust gases pass through a take-off tube connected to the exhaust system.
2. The exhaust gas flow is metered through a control valve
3. A proportion of exhaust gases are passed into the combustion chamber through the engine intake

Internal EGR - Engines equipped with variable valve timing on the intake and exhaust camshafts are able to trap exhaust gasses within the cylinder by not fully expelling it during the exhaust stroke, dramatically increasing the residual gas fraction. This is called internal EGR. All the benefits of EGR are not realised by this method. However, in certain cases, internal (uncooled) EGR is preferred:
Gasoline – At low loads to reduce pumping losses while maintaining combustion stability
Diesel – At low loads and during cold start

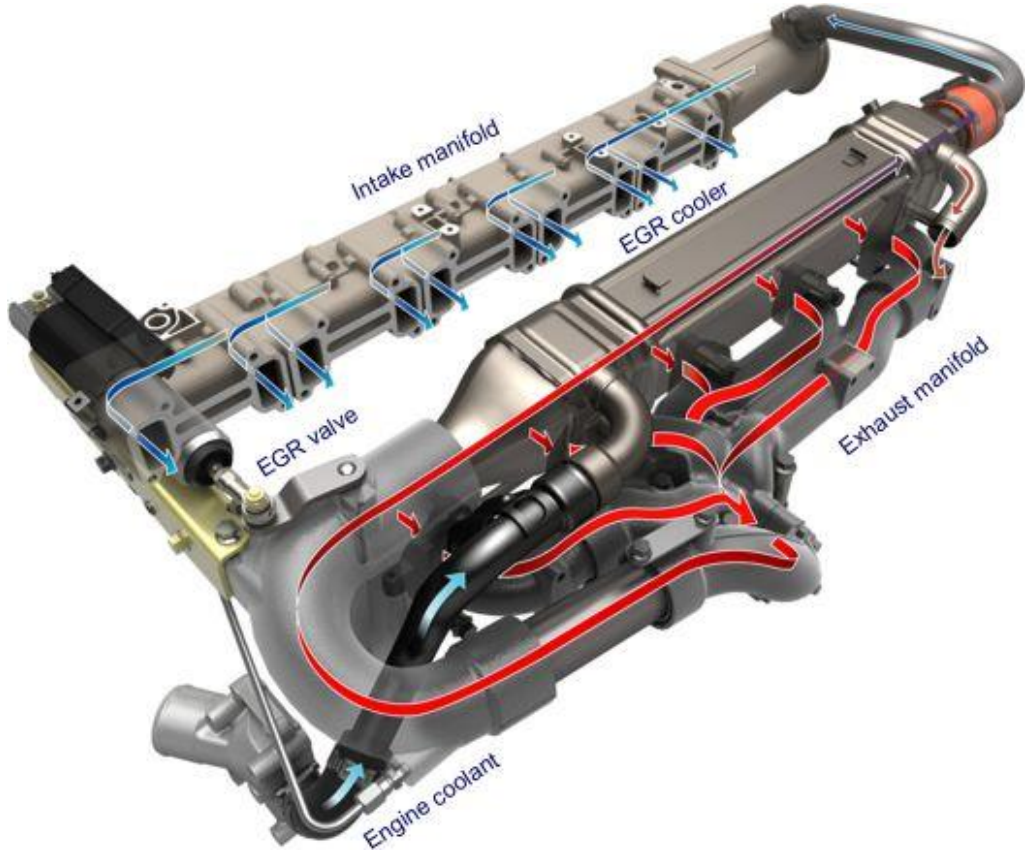
EGR Cooler Function

Diesel - EGR cooling reduces engine out NOx and increases intake charge density, allowing for increased EGR % without compromising desired air/fuel ratio.

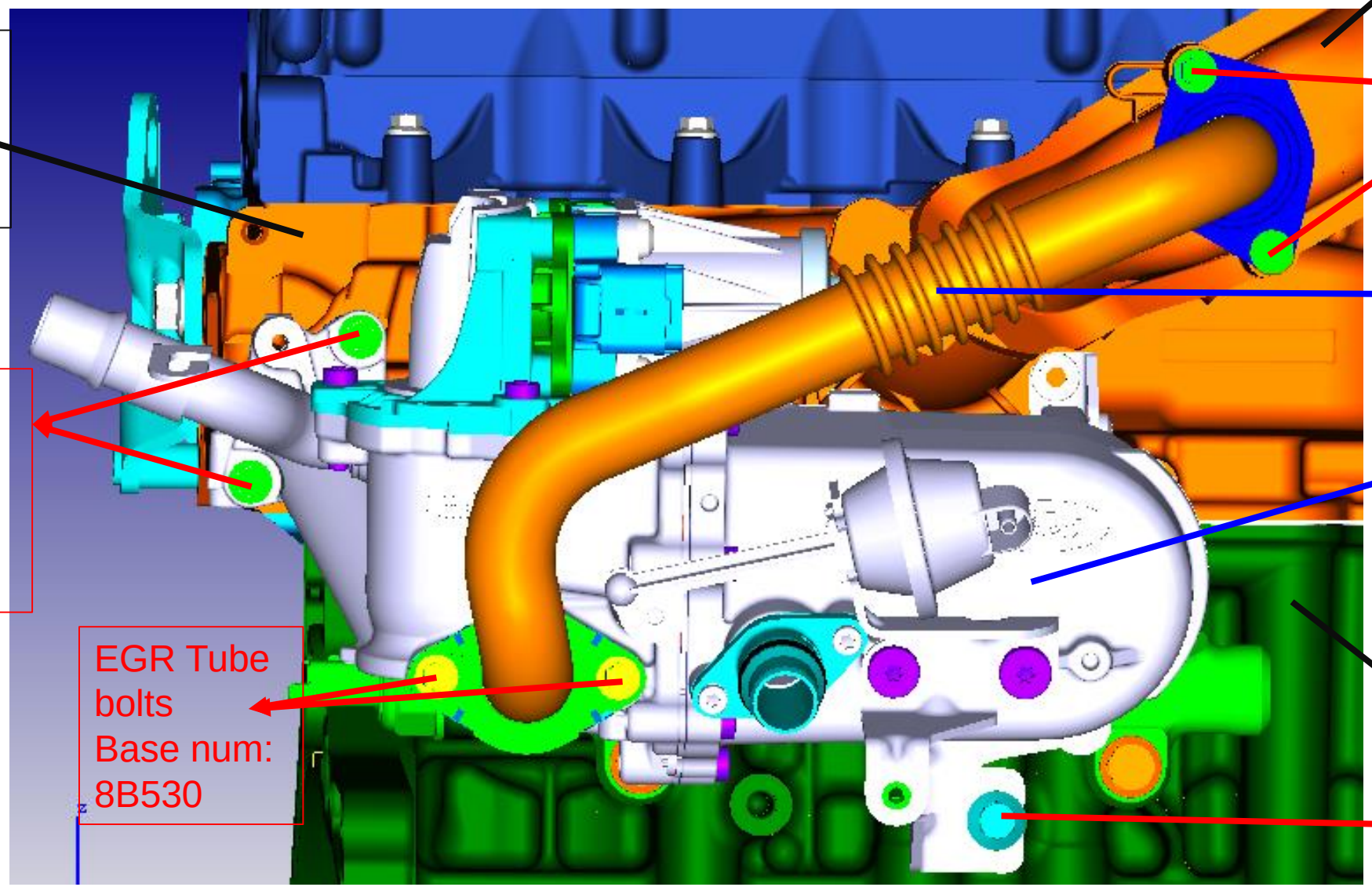
Gasoline - For downsized boosted applications, cooled EGR not only reduces NOx but also improves fuel economy by mitigating knock and reducing or eliminating enrichment to maintain proper turbine temperatures.

EGR Valve Function

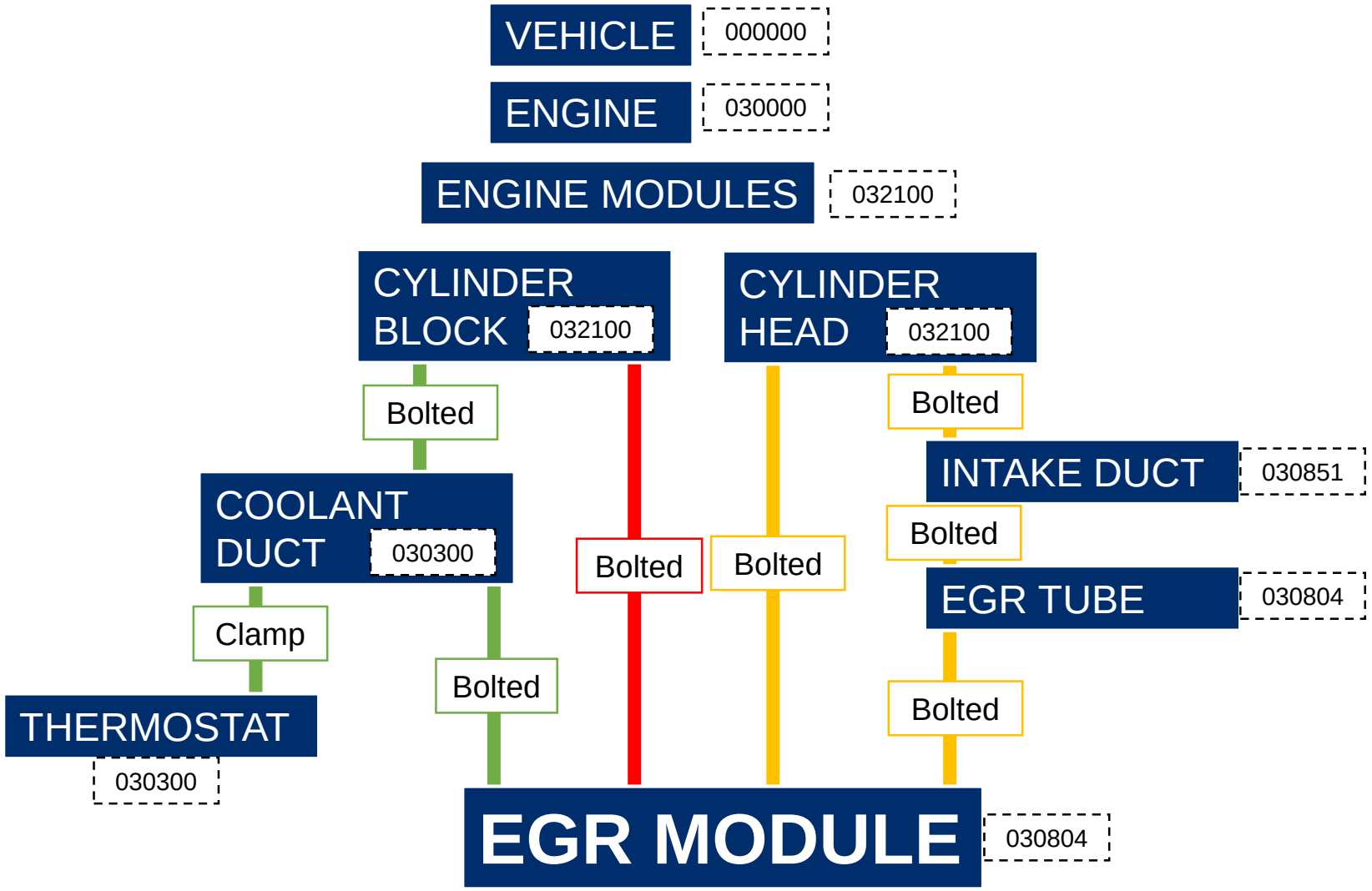
An EGR Valve is used to control the amount of EGR flow which enters the intake system. The valve assembly contains a DC electronically controlled motor, a gearbox and usually a poppet on seat valve. Depending on design and material specification, the assembly can be placed either side of the EGR cooler. Valves exposed to high temperatures will have additional channels in the casting for coolant flow.



OVERVIEW OF EGR SYSTEM



BOUNDARY DIAGRAMS



Joints are shown with lines.

- Liquid transfer
- Gas transfer
- Structural
- CPSC

CPSC Code is a 6-digit numerical code that is defined to identify commodities. In our example, our commodity is in ENGINE AS SHIPPED. Which means, the parts that are assembled in engine plant, there are also other engine parts that are assembled in vehicle plant afterwards, those parts usually have other unique CPSC codes.

BOM

Prefix*	Base*	Suffix*	Name*	CPSC*	Engineering Commodity	Remarks (Fastener Attaching Statement)
	W500223	S437	BOLT M8X20 HD PIL	030804	FASTENERS (PMT400)	FROM: 6A228; TO: 6059;
FM5Q	9D475	AA	VLV-ASY E/G/R	030804	DRESSED ENGINE	
7M5Q	8B530	AA	SC CVR WTR PM	030804	DRESSED ENGINE	FROM: 9D475; TO: 6090
FM5Q	9D477	AA	TUB ASY E/G/R OLET	030804	DRESSED ENGINE	
AV6Q	9S331	AA	INTK MANF TUB	030851	DRESSED ENGINE	

RELATED DESIGN RULESETS

EGR Systems Design Rules

KPAC Set ID	Title	Author	Type
0308_EG-S1	Exhaust Gas Recirculation Design Rule Saved Set for Diesel	Ewen, Ken (kewen)	Check List
0308_EG-S1	Exhaust Gas Recirculation Design Rule Saved Set for Gasoline	Ewen, Ken (kewen)	Check List
0308_EG-S1	Exhaust Gas Recirculation Design rule Saved Set for Low Pressure	Ewen, Ken (kewen)	Check List

Engine Fasteners (Bolted joints) Design Rules

KPAC Set ID	Title	Author	Type
PT_FAST-S1	Cup Plug Design Rule Saved Set	Murphy, Mark (mmurphy9)	Check List
PT_FAST-S1	Dryseal Pipe Thread Design Rule Saved Set	Murphy, Mark (mmurphy9)	Check List
PT_FAST-S1	Dowel and Bushing Design Rule Saved Set	Murphy, Mark (mmurphy9)	Check List
PT_FAST-S1	Fastener and Joint Design Rule Saved Set	Murphy, Mark (mmurphy9)	Check List
PT_FAST-S1	Threaded Fastener Design Rule Saved Set	Murphy, Mark (mmurphy9)	Check List
PT_FAST-S1	Wrench and Socket Design Rule Saved Set	Murphy, Mark (mmurphy9)	Check List
PT_FAST-S1	Ball Plug Design Rule Saved Set	Murphy, Mark (mmurphy9)	Check List
PT_FAST-S1	Port Plug Design Rules Saved Set	Murphy, Mark (mmurphy9)	Check List
PT_FAST-S1	Thread Forming Screws in Plastic Saved Set	Murphy, Mark (mmurphy9)	Check List

Engine Sealing Design Rules

KPAC Set ID	Title	Author	Type
0310_ES-S1	Cam Cover Design Rule Saved Set	Smith, Thomas (tsmith57)	Check List
0310_ES-S1	Dynamic Seal Rule Saved Set	Smith, Thomas (tsmith57)	Check List
0310_ES-S1	Fasteners Rules for Covers Saved Set	Smith, Thomas (tsmith57)	Check List
0310_ES-S1	Front Cover Design Rule Saved Set	Smith, Thomas (tsmith57)	Check List
0310_ES-S1	Oil Pan Design Rule Saved Set	Smith, Thomas (tsmith57)	Check List
0310_ES-S1	Rear Seal Retainer Design Rule Saved Set	Smith, Thomas (tsmith57)	Check List
0310_ES-S1	Static Seal Design Rule Saved Set	Smith, Thomas (tsmith57)	Check List

SYSTEM CHECK

- Three types of EGR systems are listed. We need to choose the correct system to see the correct rules.

KPAC Set ID	Title	Author	Type
0308_EG-S1	Exhaust Gas Recirculation Design Rule Saved Set for Diesel 	Ewen, Ken (kewen)	Check List
0308_EG-S1	Exhaust Gas Recirculation Design Rule Saved Set for Gasoline 	Ewen, Ken (kewen)	Check List
0308_EG-S1	Exhaust Gas Recirculation Design rule Saved Set for Low Pressure 	Ewen, Ken (kewen)	Check List

The particular example we are working on is using a **High Pressure** EGR System on a **DIESEL** engine. So we will need to choose the first ruleset. The other 2 rulesets are not related to our system, they are there for **GASOLINE** and **Low Pressure EGR** engines..

SYSTEM CHECK

- We have identified the joints in our systems. EGR means Exhaust Gas Recirculation, which means it includes a lot of gas transfers. Additionally, EGR systems usually includes a cooler as well. In our example system, the cooler and the valve of the EGR system is packaged inside one module. So our system also includes coolant (liquid) transfer, to cool down the gases.
- As we have all these gas and liquid transfers, some of our related joints should be leak proof, sealed properly. To support this, there are design rules for **ENGINE SEALING.**

KPAC Set ID	Title	Author	Type
0310_ES-S1:	Cam Cover Design Rule Saved Set	Smith, Thomas (tsmith57)	Check List
0310_ES-S1:	Dynamic Seal Rule Saved Set	Smith, Thomas (tsmith57)	Check List
0310_ES-S1:	Fasteners Rules for Covers Saved Set	Smith, Thomas (tsmith57)	Check List
0310_ES-S1:	Front Cover Design Rule Saved Set	Smith, Thomas (tsmith57)	Check List
0310_ES-S1:	Oil Pan Design Rule Saved Set	Smith, Thomas (tsmith57)	Check List
0310_ES-S1:	Rear Seal Retainer Design Rule Saved Set	Smith, Thomas (tsmith57)	Check List
0310_ES-S1:	Static Seal Design Rule Saved Set	Smith, Thomas (tsmith57)	Check List

- In this list, there are some design rulesets for specific systems. As there is no specific design ruleset for EGR systems inside here, we will choose Static Seal Design Ruleset. The others, for example Cam Cover design ruleset should be used when the related part is indeed a cam cover.

SYSTEM CHECK

- Lastly, as there are also fasteners that connects the parts to eachother, we have general rules for these fixations. As you can see on the below list there are a few different rulesets for that.

KPAC Set ID	Title	Author	Type
PT_FAST-S1	Cup Plug Design Rule Saved Set	Murphy, Mark (mmurphy9)	Check List
PT_FAST-S1	Dryseal Pipe Thread Design Rule Saved Set	Murphy, Mark (mmurphy9)	Check List
PT_FAST-S1	Dowel and Bushing Design Rule Saved Set	Murphy, Mark (mmurphy9)	Check List
PT_FAST-S1	Fastener and Joint Design Rule Saved Set	Murphy, Mark (mmurphy9)	Check List
PT_FAST-S1	Threaded Fastener Design Rule Saved Set	Murphy, Mark (mmurphy9)	Check List
PT_FAST-S1	Wrench and Socket Design Rule Saved Set	Murphy, Mark (mmurphy9)	Check List
PT_FAST-S1	Ball Plug Design Rule Saved Set	Murphy, Mark (mmurphy9)	Check List
PT_FAST-S1	Port Plug Design Rules Saved Set	Murphy, Mark (mmurphy9)	Check List
PT_FAST-S1	Thread Forming Screws in Plastic Saved Set	Murphy, Mark (mmurphy9)	Check List

You can see that some special types of fasteners (like ball plug, dowel/bushing, cup plug etc. has unique design rulesets. In our example, we do not have a special feature on any of our fixing points, so we will just use the generic Fastener and Joint design rule and Threaded Fastener design ruleset.

Model Writer

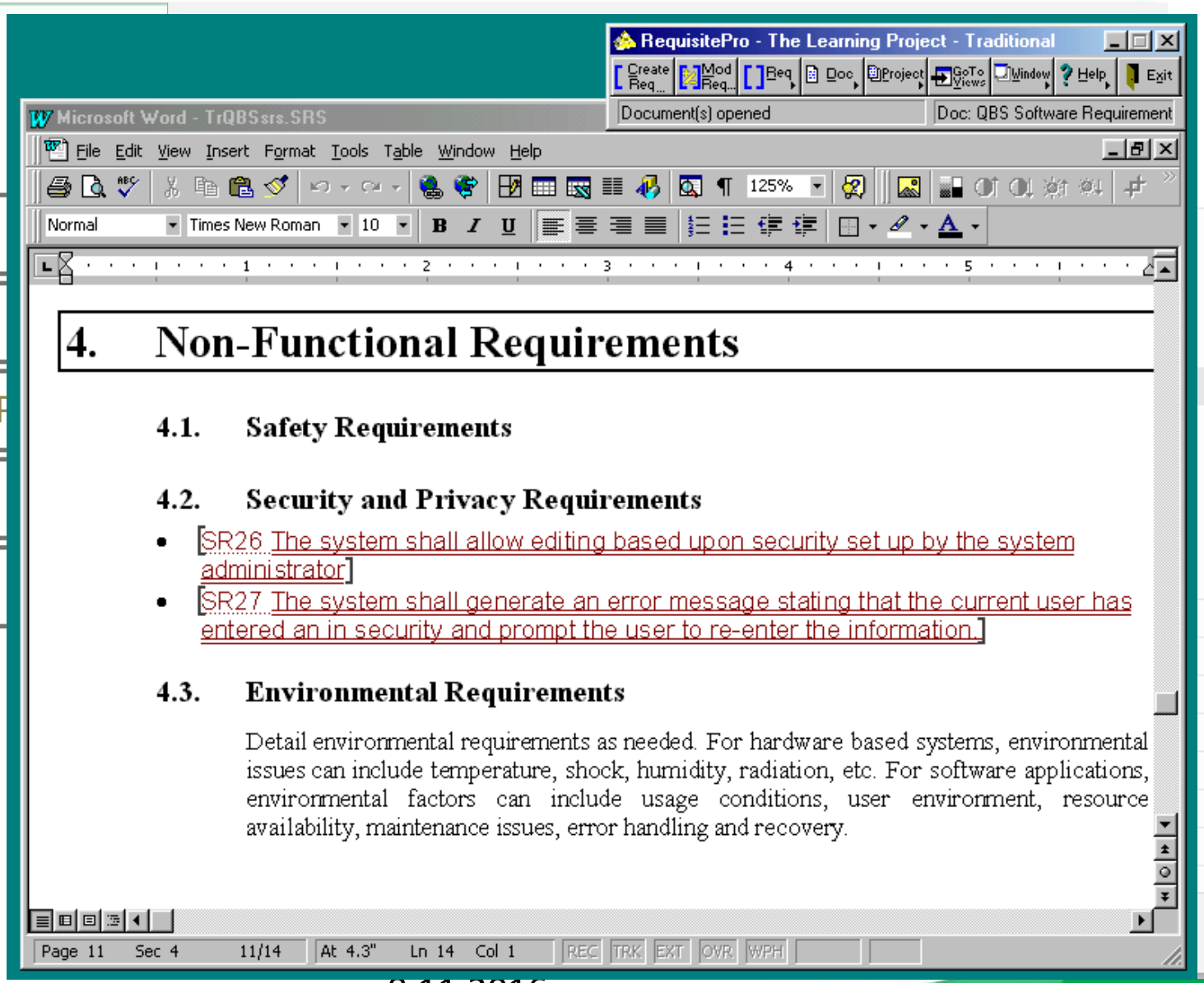
Work Package 1 - Industrial Use Cases and Requirements

Integration with Application Lifecycle Management tools

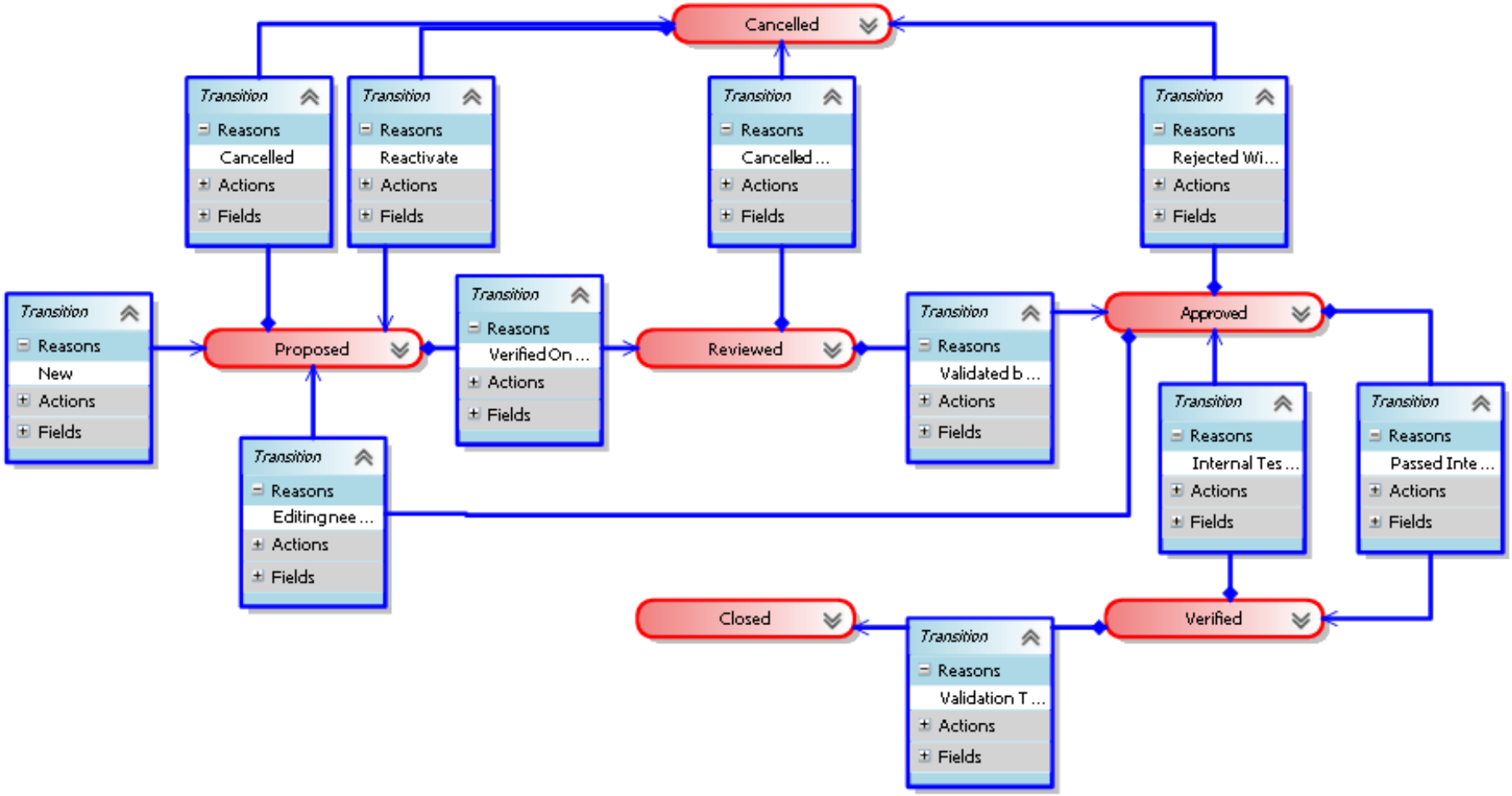
Eray TÜZÜN (HAVELSAN)

Requirement Work Item

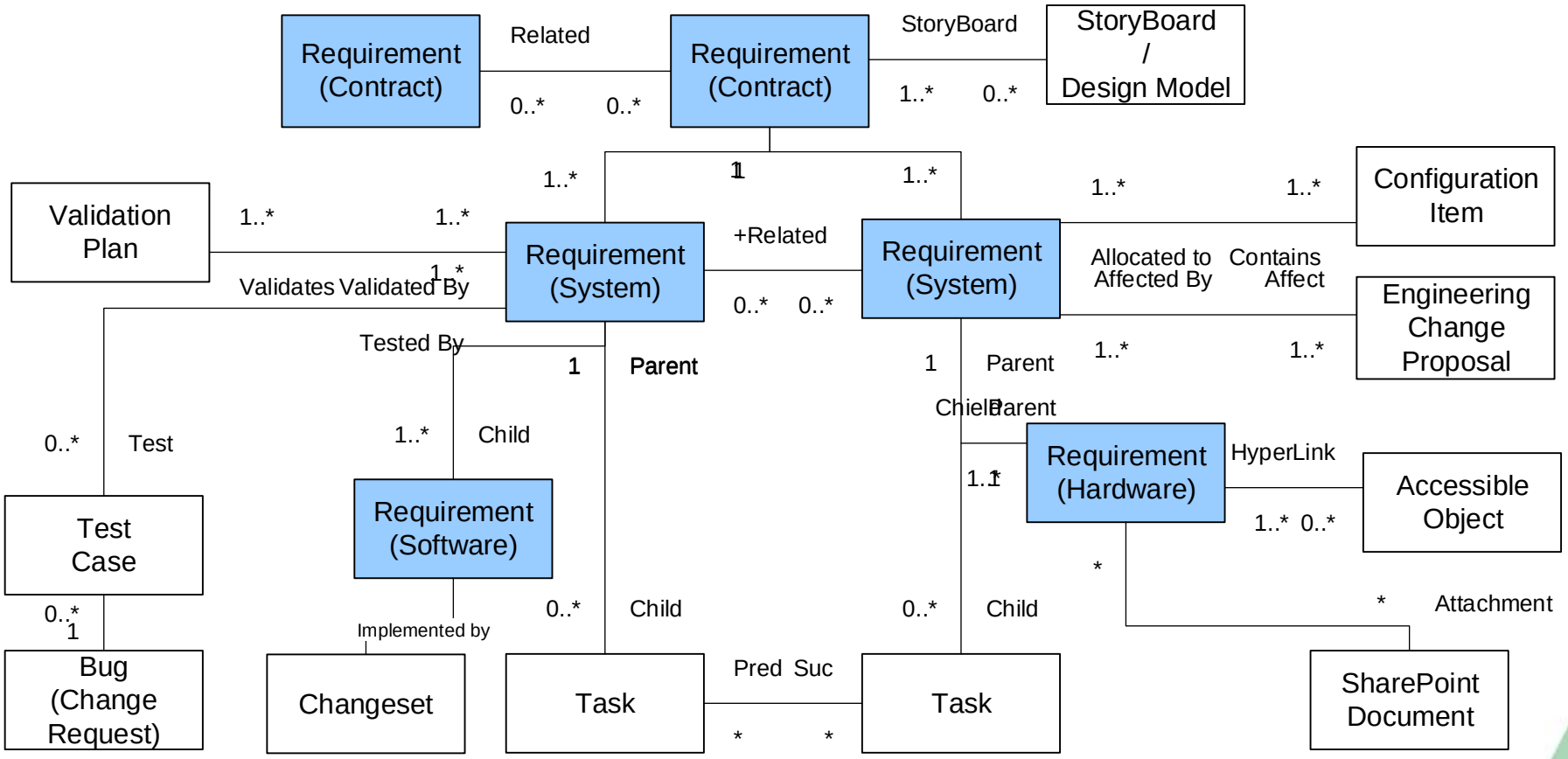
- Customized Form
- Attributes
- WorkItem Number F
- History
- Discussion



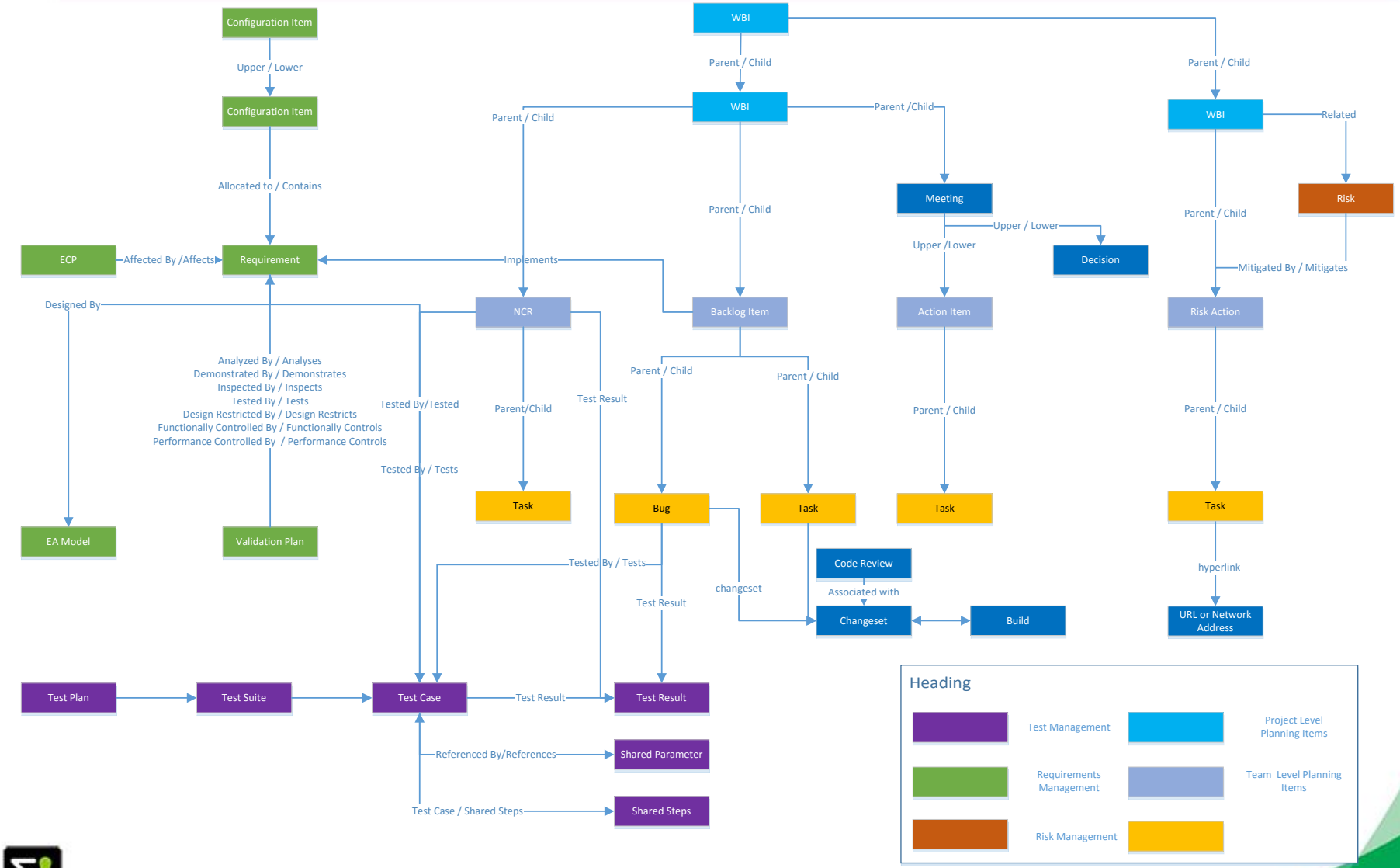
Requirement LifeCycle



Requirements Traceability



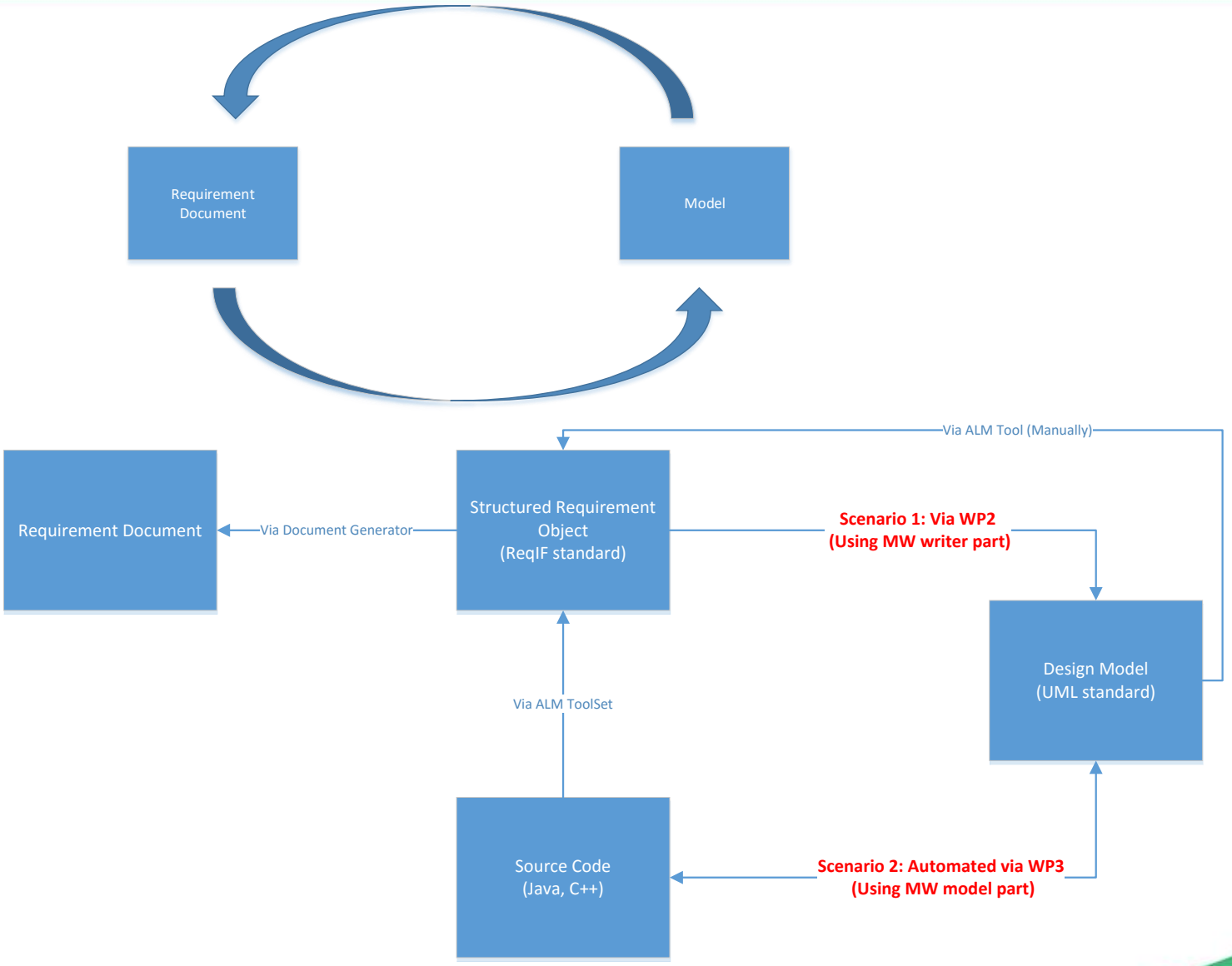
Requirements in ALM



Requirements in ALM

- Traceability with other artifacts is key
  - Requirements to other requirements
    - Customer/System/Software/Hardware..
    - Dependency relation between requirements
  - Requirements to tasks (Project management)
  - Requirements to Test Cases
  - Requirements to Design elements
  - Requirements to generated documents
  - Requirements to source code
  - Requirements to Build
  - Requirements to bugs
  - Requirements to risks
  - …

HAVELSAN Use case for ModelWriter



Exploitation, Dissemination, Standardisation (WP7)

Yvan Lussaud, OBEO

Business Model Canvas - Obeo

Key Partners

Sirius Existing Users

MW Partners

eclipse

Evangelists (MBSE, Req, Wiki)

Key Activities

Communication & Events

Software Edition

Professional Services

Key Resources

- MW Engine of WP6
- Expertise on MW internals
- Listing of existing users / customers of Obeo products

Value Propositions

Dev outsourcing / expertise for
- extraction rules
- core new feature
- high quality

Better **Collaboration** between doc & model authors

Support & Maintenance
Training

Integration with SCM, ALM, Editors

Customer Relationships

1. Open Source
2. Software Editor guaranties

Channels

1. Conferences
2. Eclipse.org
3. Direct sales

Customer Segment

1. M&T Architect of Sirius community
2. System Engineers
3. Requirement Engineers
4. Enterprise Architects
5. Pre-sales

Cost Structure

1. Salaries of consultants & MW committers
2. Marketing (conferences, communication, web)
3. Commercial

Revenue Streams

1. **Professional Services** per working day (on site or remote)
2. **New features** funding
3. New option in Obeo Designer annual **support contract**
4. Obeo Designer Team **licenses** for collaborative features

3

WP7
UNIT as a tool and technology provider

Integration with Siemens Teamcenter PLM visualization

Required for FORD-OTOSAN use case

Using JT Open Toolkit's C++ application programming interface for 3D product information

ISO 14306:2012 Industrial automation systems and integration -- JT file format specification for 3D visualization

Integration with IBM Rational DOORS or Esterel SCADE Suite

Required for HAVELSAN use case

using ReqIF (Requirement Interchange Format) Standard and Eclipse's Requirement Management Framework (RMF)

WP7
Market

Obeo Sirius	• 200 clients and about10000 prospects • 60% of Obeo's revenue
MBSE	• System analysis and live cycle tooling • About 50 million euros (IBM, Siemens, Dassault System, PTC, …)
Enterprise Architecture	• 100 million euros in France alone • 9000 people certified Togaf 9 in the world
Competitors	• Reqtify, Recycle, DoorsNG • Assets of MW: Open source, Interoperability, not only focused on requirements

WP7
KPI

	Actual	Goal
Number of users (not counting MW participants)	0	10
Number of projects using MW outcomes	0	6
Unique visitors of MW website per month	0	300

WP7
Exploitation

Release of Tarski platform (WP3) – UNIT and KOCSISTEM

- Automated analysis and dynamically configurable semantics of traceability links
- Standalone and open source https://github.com/ModelWriter/WP3

Achievement: Synchronization of EGR & FEAD designs with specifications

- Improve review process of engineering teams
- Impact analysis of design changes

Release of M2Doc – Obeo

- ModelWriter helped us to ramp-up MS Word document creation and Java API
- New product of MS word document generation M2Doc
- Future new feature for traceability using ModelWriter mapping API

WP7
Exploitation

Release of Synchronization Engine and UI – Obeo

- Synchronization engine frist version
- First implementation of GUI

Ontology consistency checking – CNRS and Airbus

- Identify and remove inconsistencies from the knownledge base
- Solution based on Airbus needs

MS Word plug-in – UNIT and HISBIM

- Enable seemless integration with ModelWriter on Eclipse platform
- GUI is almost ready
- Working on integration of Eclipse platform and MS Word plug-in with respect to ModelWriter workspace

WP7
Exploitation

Release of Component Ontology – Airbus

- partially describes the vocabulary used in the SIDP documents and database
- Improves the SIDP model developed in first semester
- Used for semantic annotation and link creation

Web based services for Turkish language support – Mantis

Highlight experience of industrial end users

- Airbus
- Havelsan
- Ford-Otosan

WP7
Dissemination

6th international ModelWriter Workshop (25 February 2015)

- Product owner review
- Integration of first release

7th international ModelWriter Workshop (6 June 2016)

- At Airbus headquarter
- Brain storming and progress presentation
- Mapping base design and sketches of GUI by Obeo

Airbus Friday TechTalk

- Demonstration of current state
- Presentation of objectives

WP7
Dissemination

MPM4CPS (Multi-paradigm Modeling for Cyber-physical Systems) Cost Action

- UNIT and AIRBUS are Management Committee Members representing Turkey and France in this COST action

The European Research Network on Software-intensive Systems-of-Systems (SiSoS)

- UNIT is a participant in the network

ITEA3-Assume project

- Ford-Otosan, Havelsan, KocSistem and UNIT participates in that project

WP7
Dissemination

ACM Applied Computing Symposium
- Paper submited and uneder review

SAT/SMT/AR Summer School
- Poster presentation

VTSA Summer School
- Poster presentation

WP7
Dissemination

Use Obeo deployment of MW on Sirius documentation

Target developers as early adopters

- EclipseCon, SiriusCon, Eclipse Demo Camp

Move MW project to Eclipse foundation

WP7
Standardization

Modular architecture

- Open source
- API
- Definition of domain model for mapping

Used standards

- EMF based of MOF
- XMI serialization
- Standard ontology formats (owl, rdf, turtle, …)

WP2 - Semantic Parsing and Generation of Documents and Documents Components

Claire GARDENT, Bikash GYAWALI,
Anastasia SHIMORINA
CNRS / LORIA
Samuel CRUZ-LARA
University of Lorraine / LORIA

ITEA 3

WP2

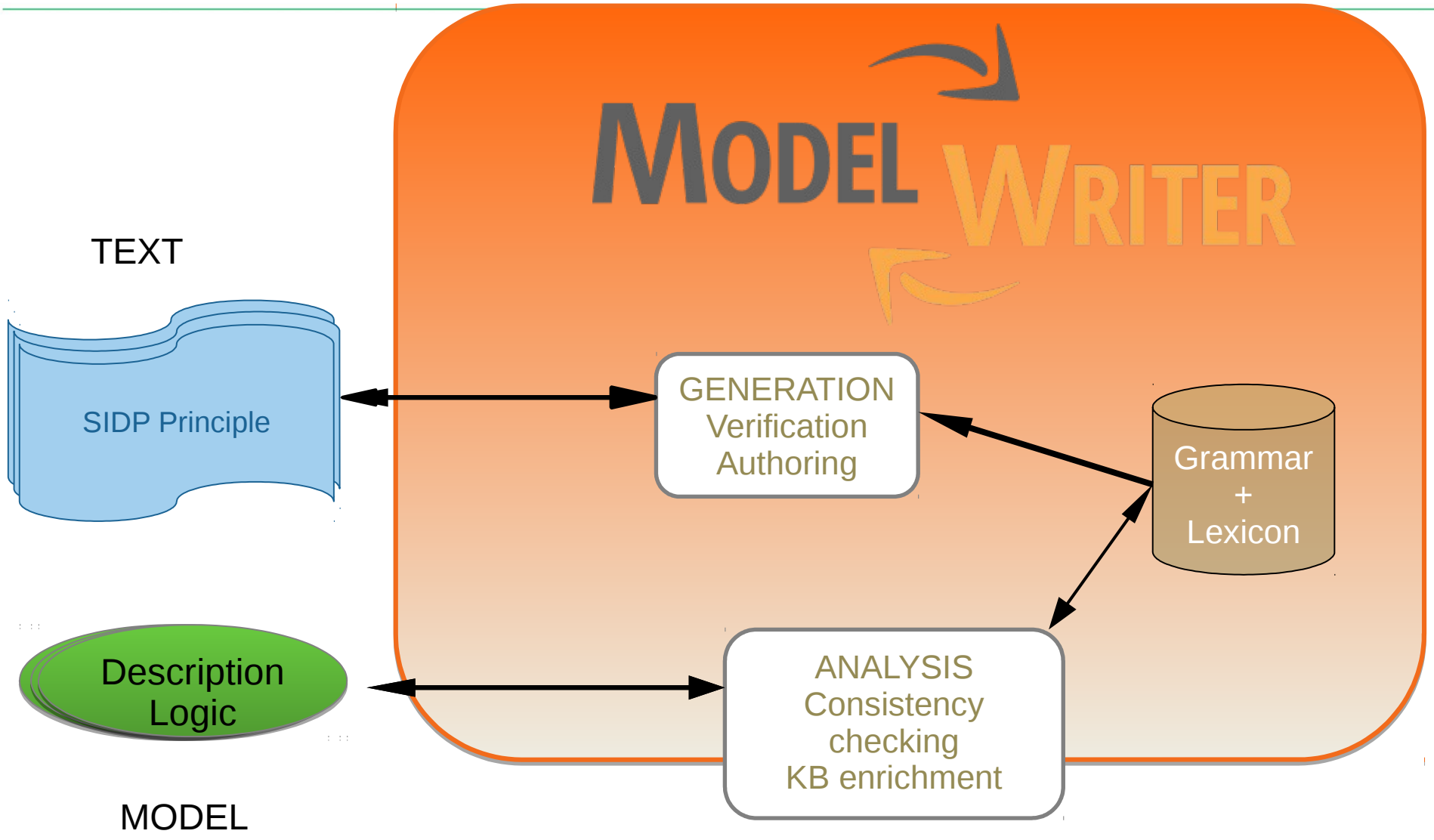
Goal: Provide tools and methods for:

- Converting texts to models and models to text
- Annotating text fragments with model elements

Tasks:

- T2.1 Data Collection
- T2.2 Semantic Parsing
- T2.3 Natural Language Generation
- T2.4 Definition of a common target semantic language
- T2.5 Development of a Semantic Parser and of a Natural Language Generator

Synchronising Text and Model



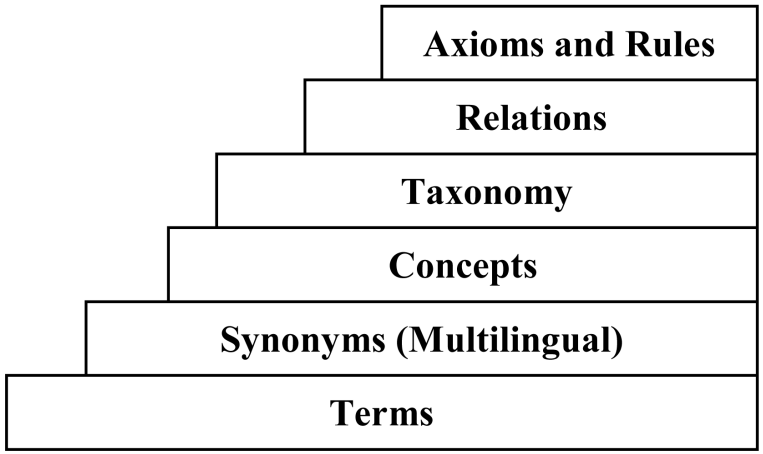
Related Work on Ontology Learning

Ontology Learning from Text

[Mädche and Staab 2000, Volker et al. 2007, Tablan et al. 2006, Zouaq and Nkambou 2008]

Restricted expressivity. Not applied to sentences (complex axioms).

Limited to : definition of new classes, creation of hierarchies between classes, definition of object and data-type properties, creation of instances, and setting of property values for instances



Related Work on Semantic Parsing

Deep parsing

[Currant et al. 2007, McCartney and Manning 2007]

First Order Logic Representations close to initial text. Trained on newspaper text (Penn Tree Bank).

==> *Not easily adaptable to Description Logic and SIDP text.*

Domain Specific Semantic Parsing

[Ge and Mooney 2009,Wong and Mooney 2007]

==> *Require parallel text-data training corpus.*

Open Domain Semantic parsing

[Kwiatkowski et al 2010, Bordes et al. 2012, Kwiatkowski et al 2013, Berant et al., 2013, Bordes et al. 2014, Wang et al. 2015]

==> *Restricted to questions. Require parallel question-answer training corpus.*

Related Work on Text Generation

Symbolic Approaches

[Dimitrios et al. 2007, Androtsopoulos et al. 2013, Power et al. 2010, Bontcheva et al. 2004]

Heavily dependent on hand-written modules.

Machine Learning Approaches

[Wong et al. 2007, Belz 2008, Angeli et al. 2010, Chen et al. 2008, Konstas and Lapata 2012a and 2012b]

Require parallel data-text training corpus.

Pattern-Based

[Duma et al. 2010, Blake et al. 2013, Schilder et al. 2013]

Require large quantity of parallel or comparable text-data training corpus. Limited Semantic Variability (set of RDF triples).

Contributions

Reversible Processing: Text <--> Model

Analysis: SIDP Rule → Description Logic
Generation: Description Logic → SIDP Rule

Verification by generation

Semantic Parsing of Complex Axioms

Pipe shall be identified by labels

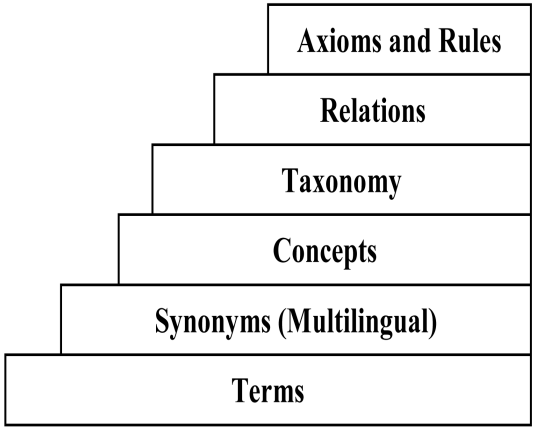
$$Pipe \sqsubseteq \exists identificationArg2^{-}.(Identification \sqcap \exists identificationArg3.Label)$$

Executable Semantic Parsing on DL KBs

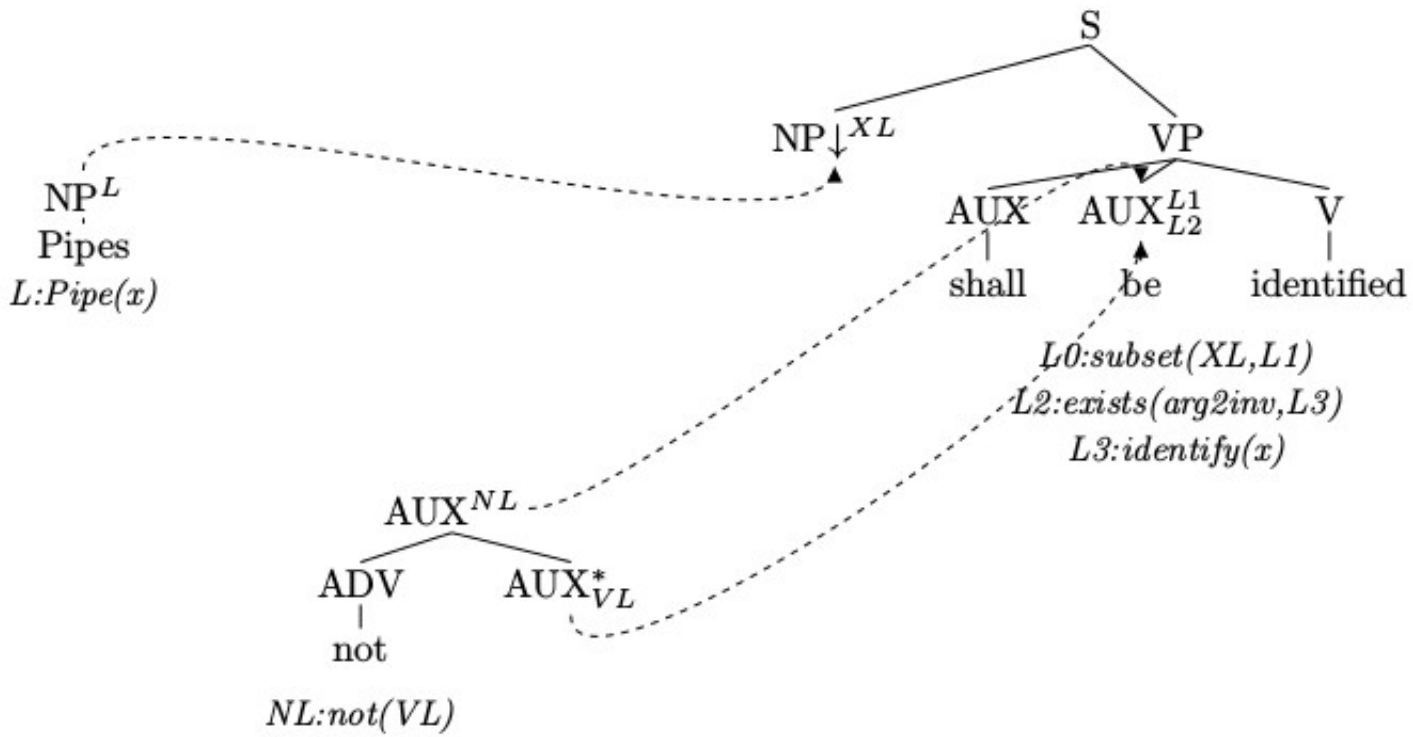
The output of semantic parsing is used to update a Description logic Knowledge Base and check the consistency of SIDPs (system installation design principle)

Genericity

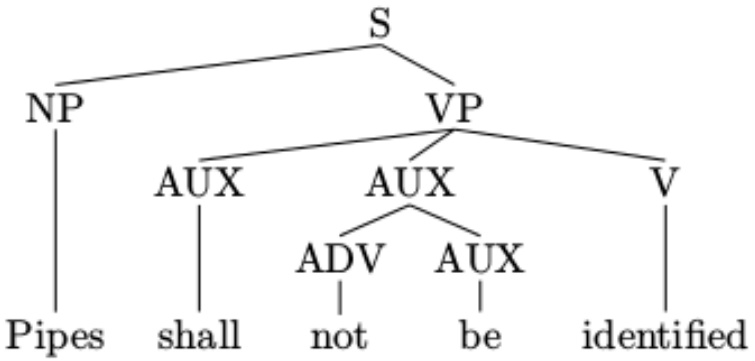
No training corpus required.
Adaptation to a new domain through grammar adaptation, extension or induction



Grammar-Based Parsing and Generation



Grammar-Based Parsing and Generation



PL:Pipe(x) L0:subset(PL,NL) NL:not(VPL)
VPL:exists(identifyA2inv,VL) VL:Identify(x)

$$Pipe \sqsubseteq \exists \neg identifyA2^{-}.(Identify)$$

Semantic Variations

Logical Operators	
Only S shall be used by O	$\neg S \sqsubseteq \neg \exists useA2^{-}.(use \sqcap \exists by.O)$
S should be used by all O	$O \sqsubseteq \exists by^{-}.(Use \sqcap \exists useA2.S)$
S shall not be used by O	$S \sqsubseteq \neg \exists useA2^{-}.(Use \sqcap \exists by.O)$
Word Order	
S shall be used by O only	$S \sqsubseteq \neg \exists useA2^{-}.(Use \sqcap \exists by.\neg O)$
All S shall be used by O	$S \sqsubseteq \exists useA2^{-}.(Use \sqcap \exists by.O)$
Arity	
S shall be used	$S \sqsubseteq \exists useA2^{-}.(use)$
S shall be used by O	$S \sqsubseteq \exists useA2^{-}.(use \sqcap \exists by.O)$
S shall be used by O on PO	$S \sqsubseteq \exists useA2^{-}.(use \sqcap \exists by.O \sqcap \exists on.PO)$
Sentence Structure	
S shall be used by O before entering connections	$(Use \sqcap \exists useA2.S \sqcap \exists by.O) \sqsubseteq \exists before.(Enter \sqcap \exists enterA2.Connections)$
Modifiers	
S shall be used directly by O	$S \sqsubseteq \exists useA2^{-}.(Use \sqcap \exists directly.(\exists by.O))$
S shall be used by O between C and D	$S \sqsubseteq \exists useA2^{-}.(Use \sqcap \exists useA3.(O \sqcap \exists betweenA1^{-}.(Between \sqcap \exists betweenA2.C \sqcap \exists betweenA3.D)))$

Updating the Model and Checking Consistency

The Semantic Representations output by the parser are converted to Description Logic

$l_0 : A(x)$ ⇒ :A

$l_0 : exists(R, l_1)\ \ l_1 : C$ ⇒ ObjectSomeValuesFrom(:R τ(C))

$l_0 : subset(l_1, l_2)\ \ l_1 : C_1\ \ l_2 : C_2$ ⇒ SubClassOf(τ(C_1) τ(C_2))

$l_0 : and(l_1, l_2)\ \ l_1 : C_1\ \ l_2 : C_2$ ⇒ ObjectIntersectionOf(τ(C_1) τ(C_2))

$l_0 : not(l_1)\ \ l_1 : C$ ⇒ not(τ(C))

E.g.,

Pipes should be identified by labels

$l_1 : Pipe(x) l_0 : subset(l_1, l_2)\ l_2 : exists(identifyA2inv, l_3)\ \ l_3 : and(l_4, l_5)$
$l_4 : Identify(z)\ \ l_5 : exists(by, l_6) l_6 : Label(y)$

```
SubClassOf(Pipe ObjectSomeValuesFrom(identifyA2inv
ObjectIntersectionOf(Identify ObjectSomeValuesFrom(by Label))))
```

The formula is added to the AIRBUS KB and Hermit is used to check for consistency

Experimental Setup and Results for Parsing

Grammar: 52 trees

Lexicon: 10781 lexical entries

Parsing algorithm: CKY + Robustness mechanism to skip unknown words

Input: 991 System Installation Design Principles

	Complete Parse	Partial Parse	Failure
Simple SIDP	132	329	24
Complex SIDP	0	496	10
All SIDP	132 (13%)	825 (83%)	34 (3%)

Updating the Model using Parsing Results (Complete Parses)

CONCEPTS	Nb. of new Concepts	184
	Nb. of Existing Concepts	30
PROPERTIES	Nb. of New Properties	62
SIDP AXIOMS	Nb. of SIDP Axioms (from Parsing)	132
	Nb of Invalid Axioms	0
	Nb. of Redundant Axioms	2
ALL	Total Nb. Of Added Elements	376
	Nb. of Axioms in Initial KB	12469
	Nb. of Axioms in Enriched KB	13029

Updating the Model using Parsing Results (All Parses)

CONCEPTS	Nb. of new Concepts	667
	Nb. of Existing Concepts	79
PROPERTIES	Nb. of New Properties	98
SIDP AXIOMS	Nb. of SIDP Axioms (from Parsing)	957
	Nb of Invalid Axioms	61
	Nb. of Redundant Axioms	125
ALL	Nb. of Inconsistent Axioms	20
	Nb. of added SIDP Axioms	749
	Total Nb. Of Added Elements	1514
	Nb. of Axioms in Initial KB	12469
	Nb. of Axioms in Enriched KB	14650

Generation Results

Grammar: 52 trees

Lexicon: 10781 lexical entries

Generation algorithm: Tabular + Polarity Filtering

Input: 957 Description Logic Axioms derived from the AIRBUS System Installation Design Principles

	Success	Failure
Simple SIDP	448	13
Complex SIDP	470	26
All SIDP	918 (96%)	39 (4%)

Verifying Parsing Results Using Generation

BLEU	< 0.33	> 0.32 and < 0.67	> 0.66
Complete Parses (S)	1	0	131 (14%)
Complete Parses (C)			
Partial (Simple)	143	117	69
Partial (Complex)	396	91	9
All Parses	540 (56%)	208 (22%)	209 (22%)

Regenerating from the DL formula derived through parsing from an SIDP:

- Produces a sentence identical to the input SIDP for complete parses
- Produces a sentence highly similar to the input SIDP in 44% of of the cases for partial parses

Perspectives and Future Work

- Improve lexicon construction using chunking
- Improve coverage on complex sentences including conditions
- Querying the KB (support for AIRBUS engineers)
- Improve robustness and genericity (experiment with deep learning approaches using data expansion techniques and sequence to sequence models)

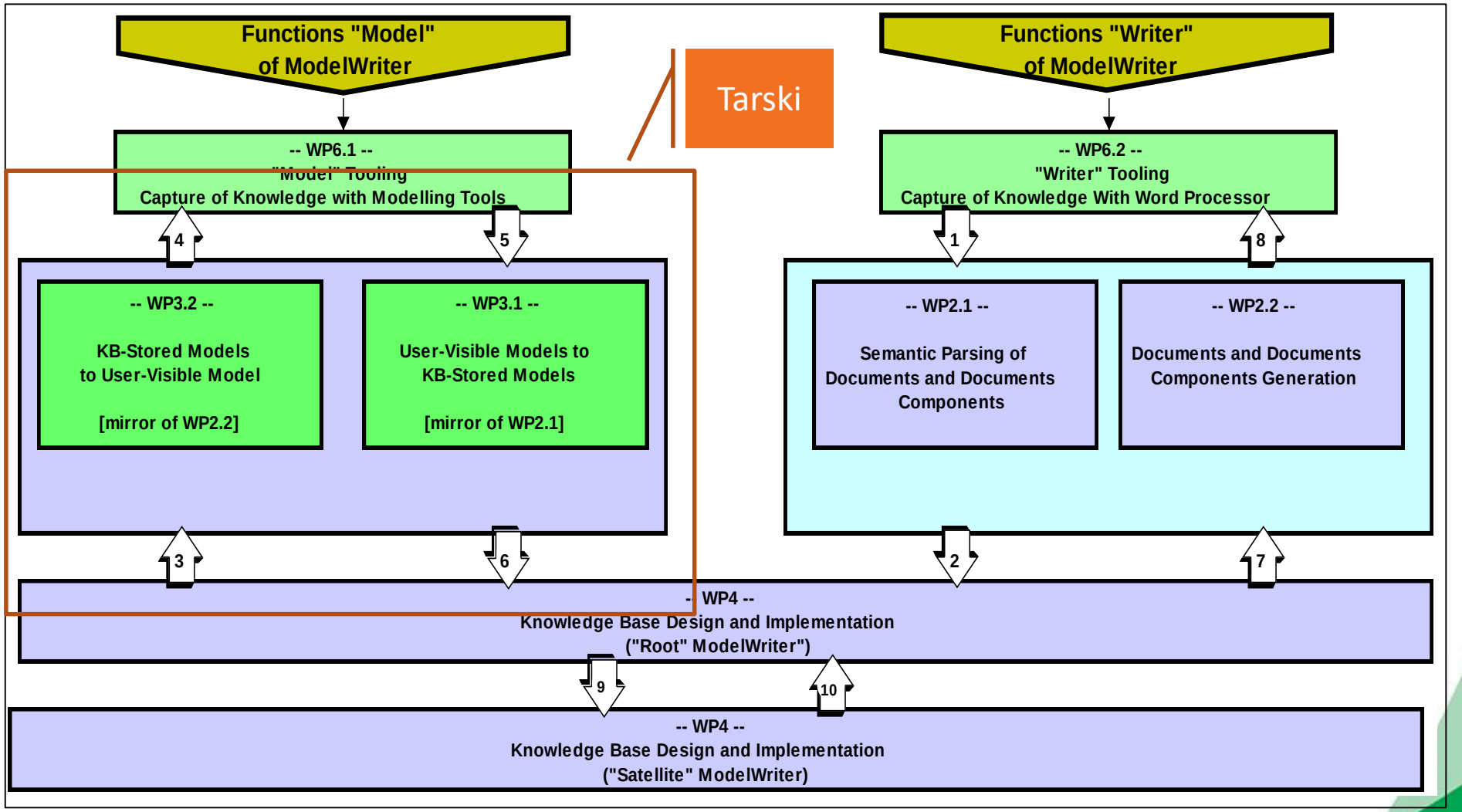
WP3 - Model to/from Knowledge Base Synchronization Mechanism

Ferhat Erata, Work Package Leader
UNIT Information Technologies R&D Ltd.

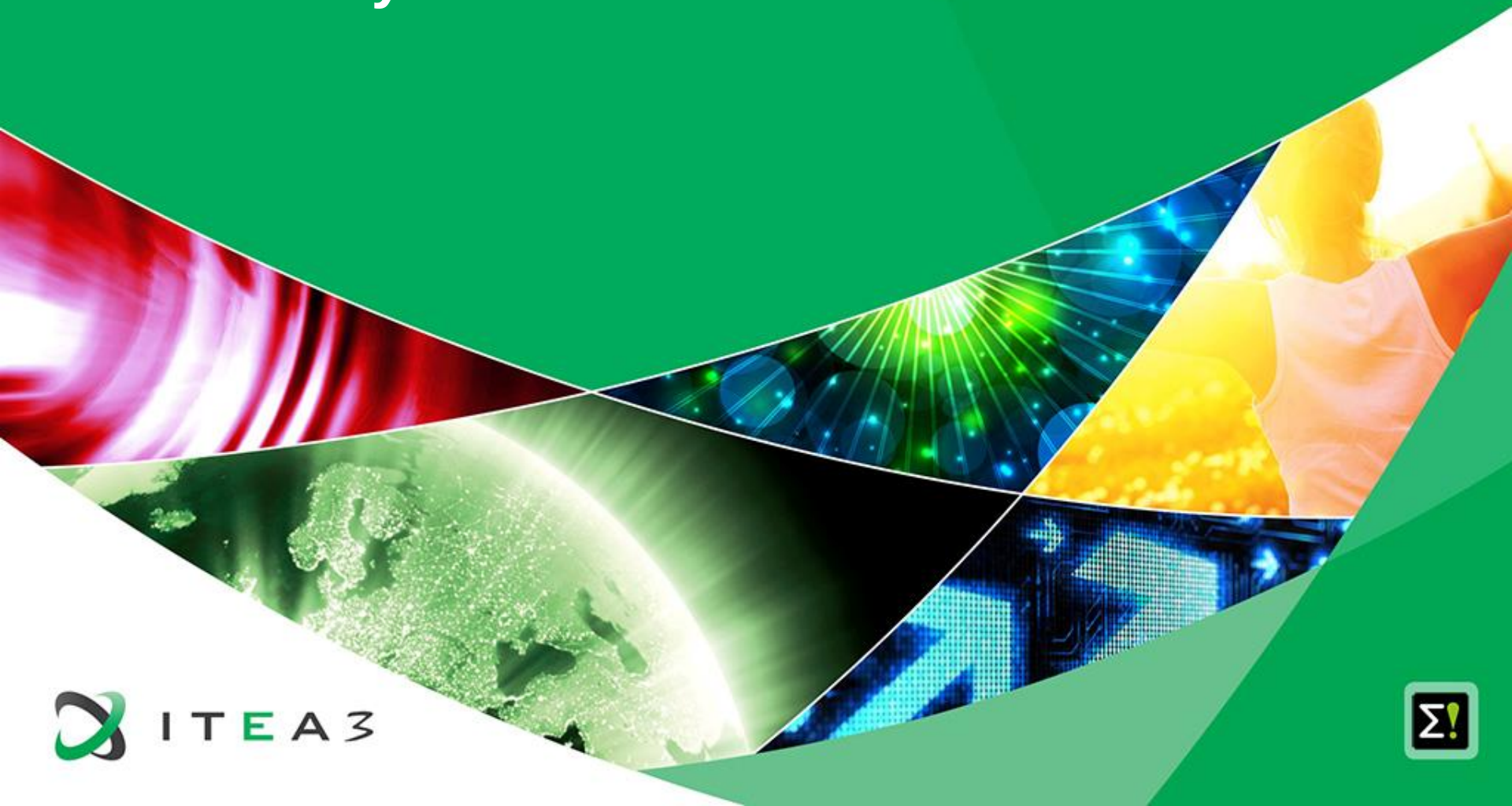
ITEA 3

Technological components & interactions

Colloboration by WP interactions



Tarski: A Platform for Automated Analysis of Dynamically Configurable Semantics of Traceability



Challenges of Traceability in Industry

Semantically meaningful traceability

- traceability relations should have a rich semantic meaning instead of being simple bi-directional referential relation

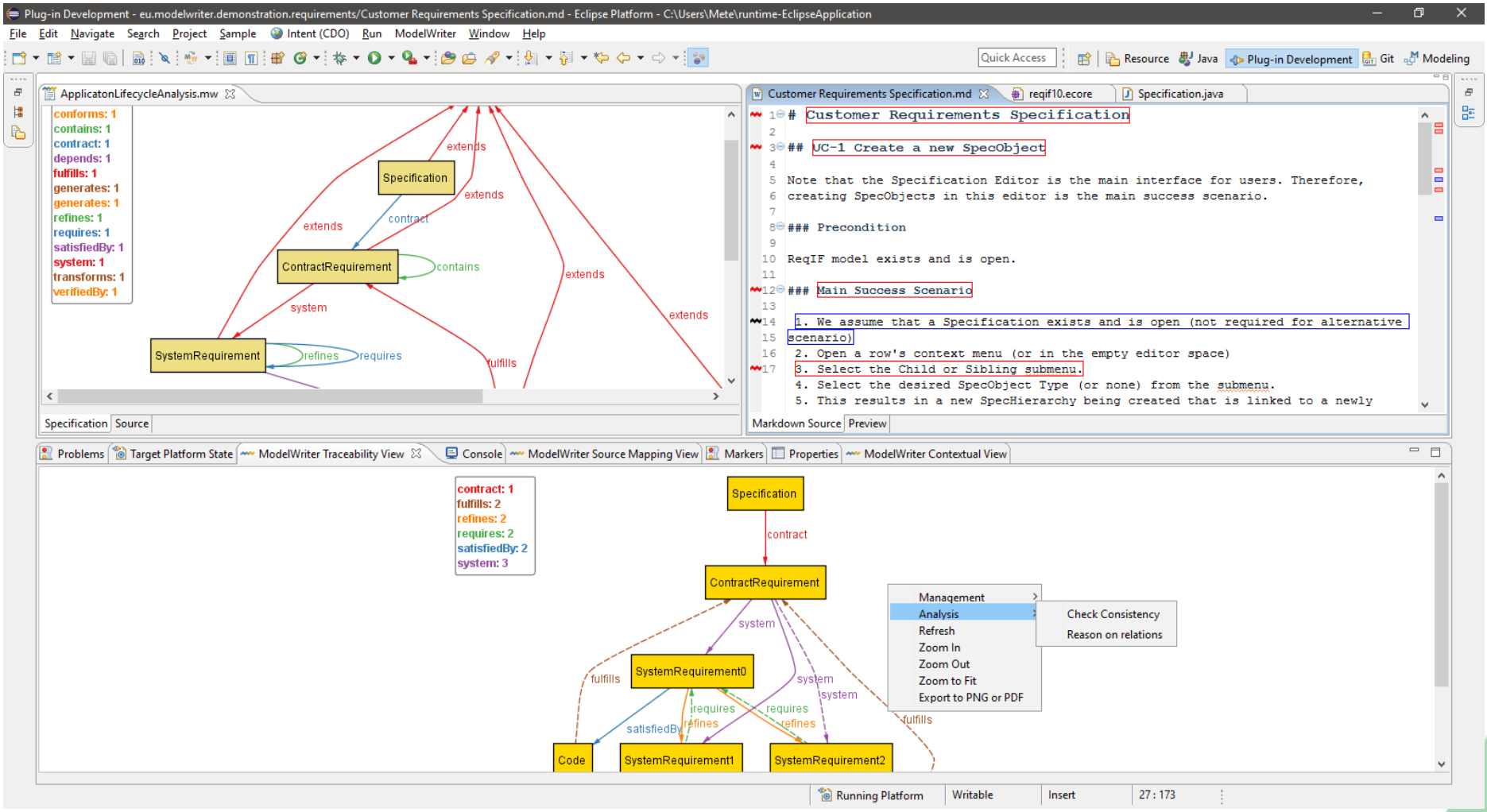
Configurability of traceability (possibly dynamically)

- the semantics of traceability is often statically defined
- the semantics cannot be easily adapted for the needs of different projects.
- different traceable elements and the types of relations exist in industrial settings.

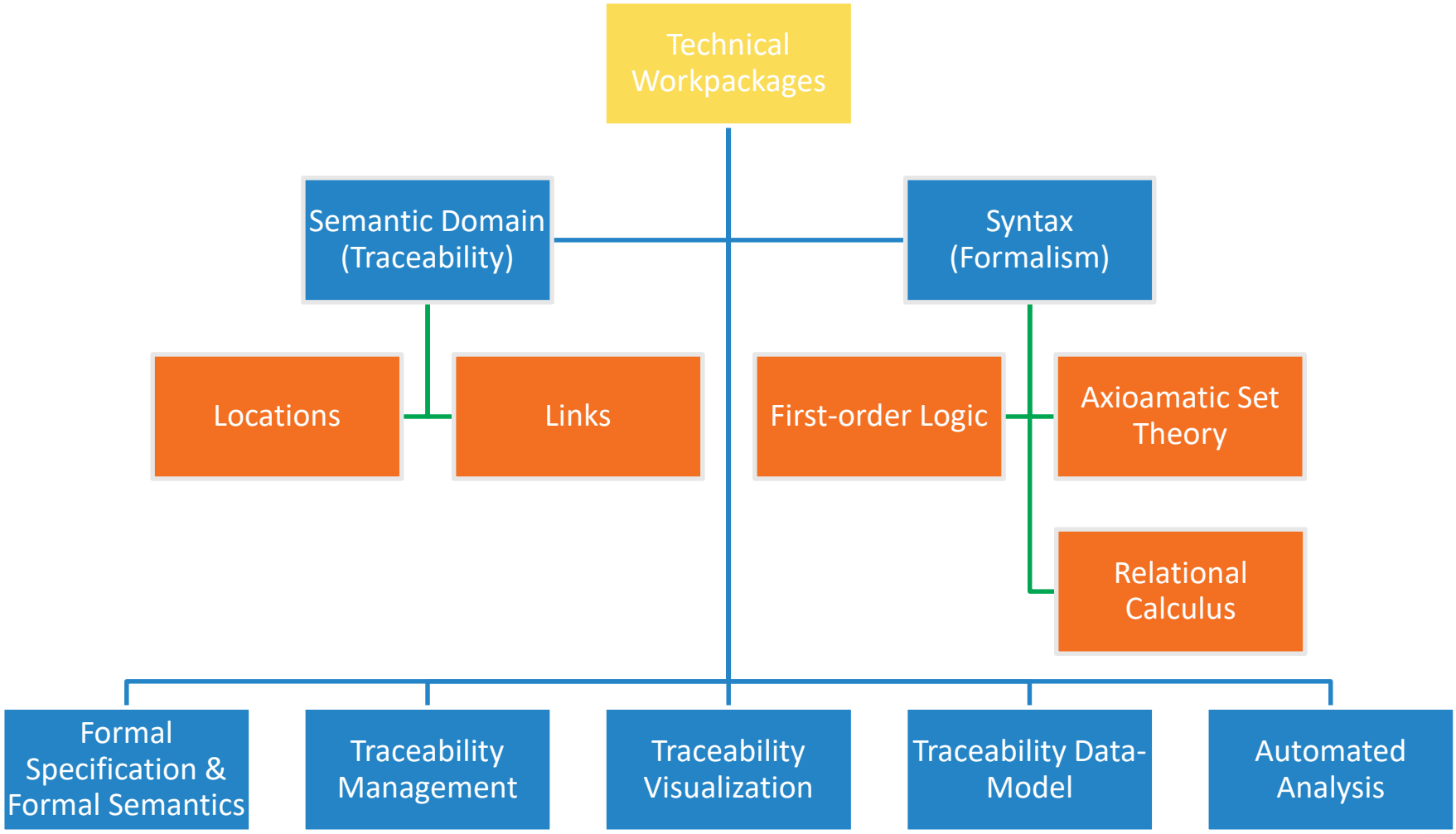
Several industries demands formal proofs of traceability

Consistency checking and repairing broken trace links

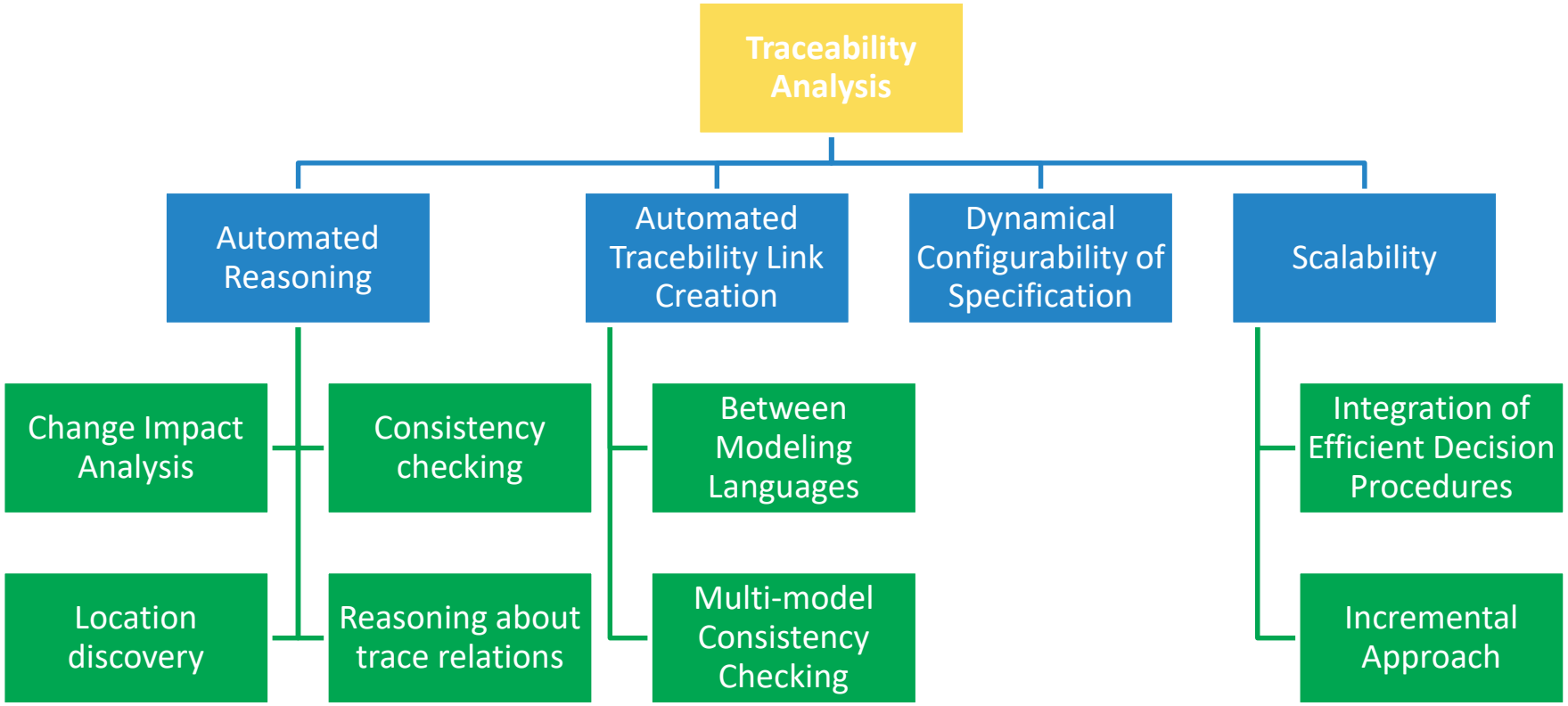
Technical Contributions @Tarski



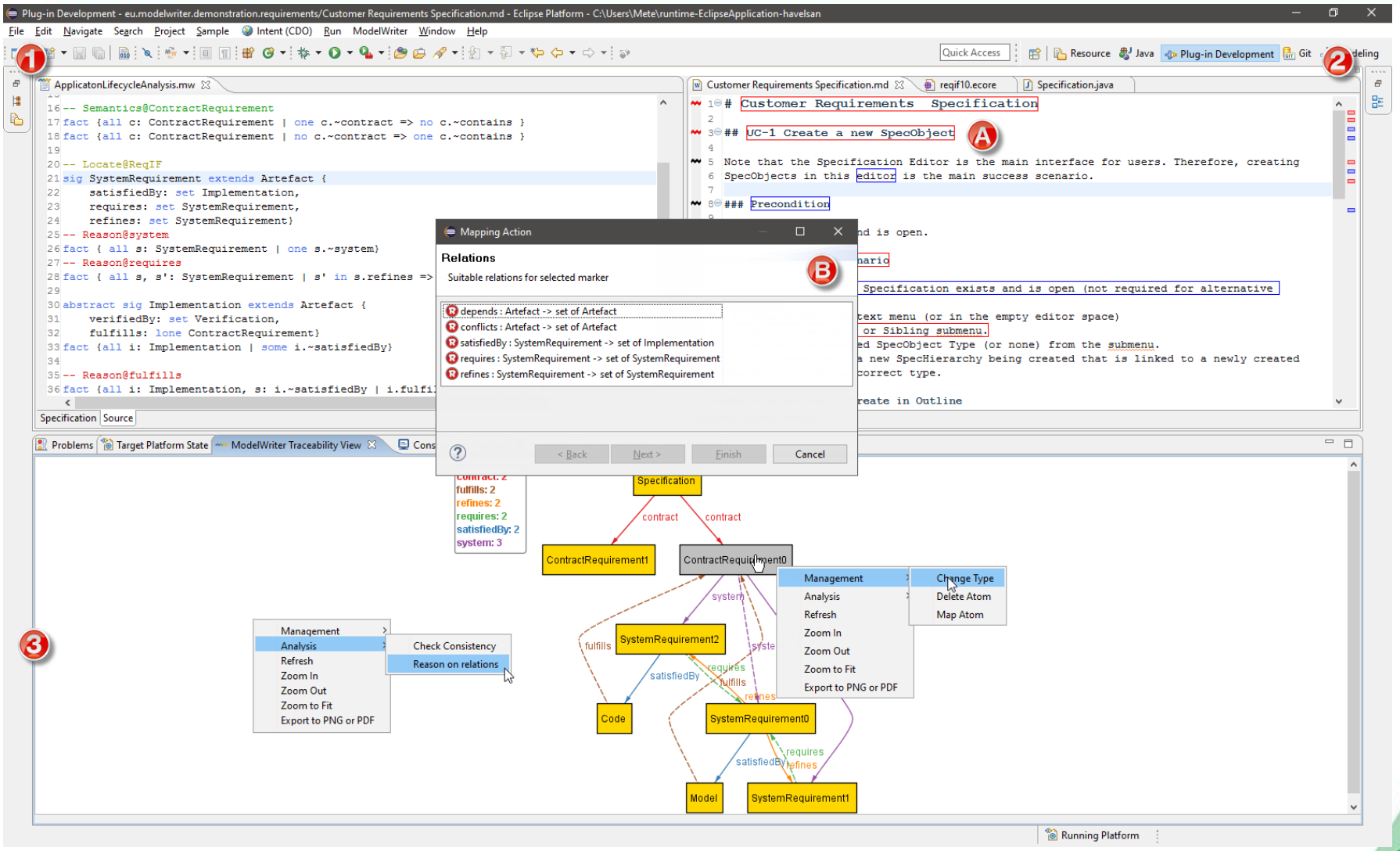
Overview of Technical Contributions @Tarski



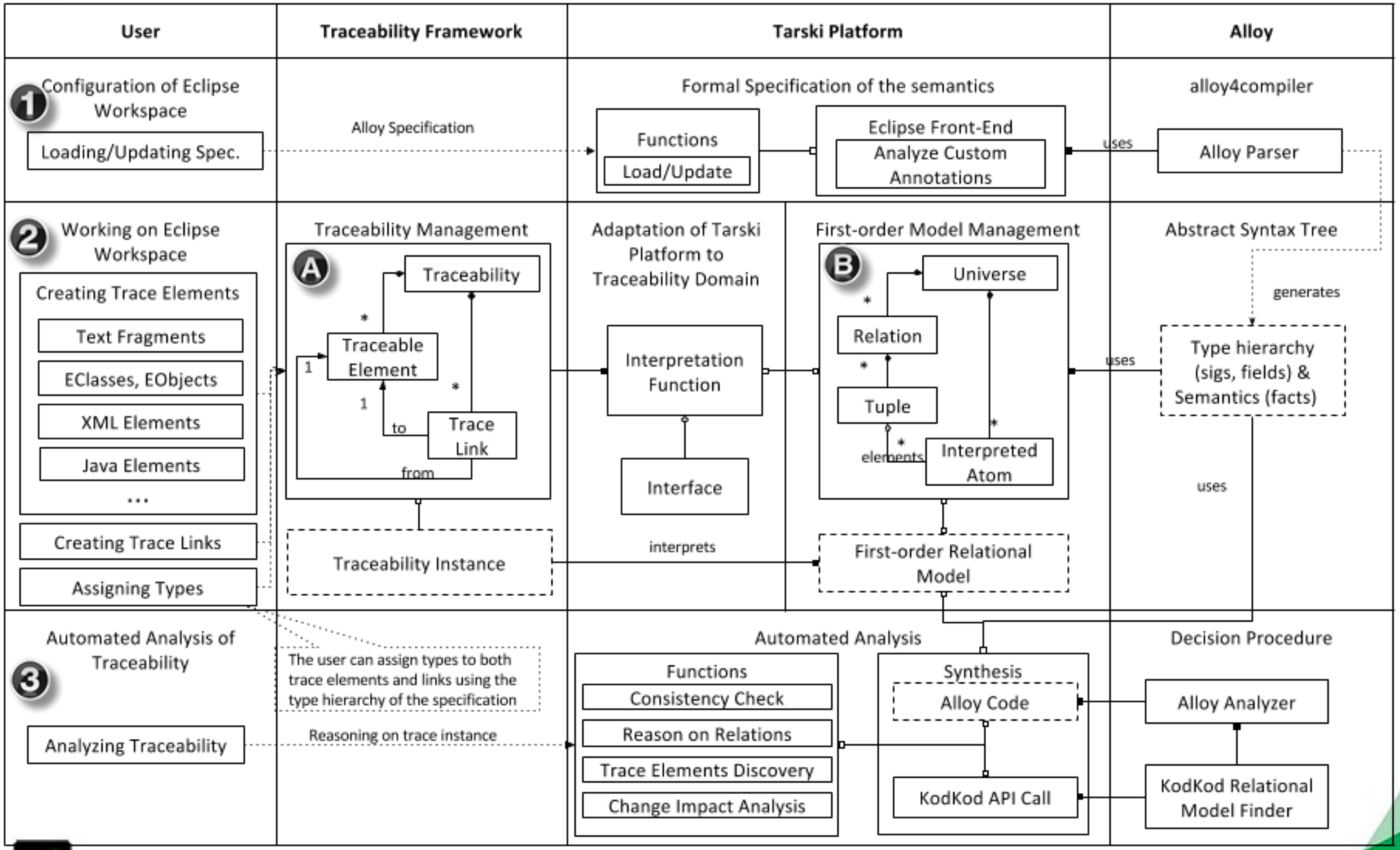
Traceability Analysis



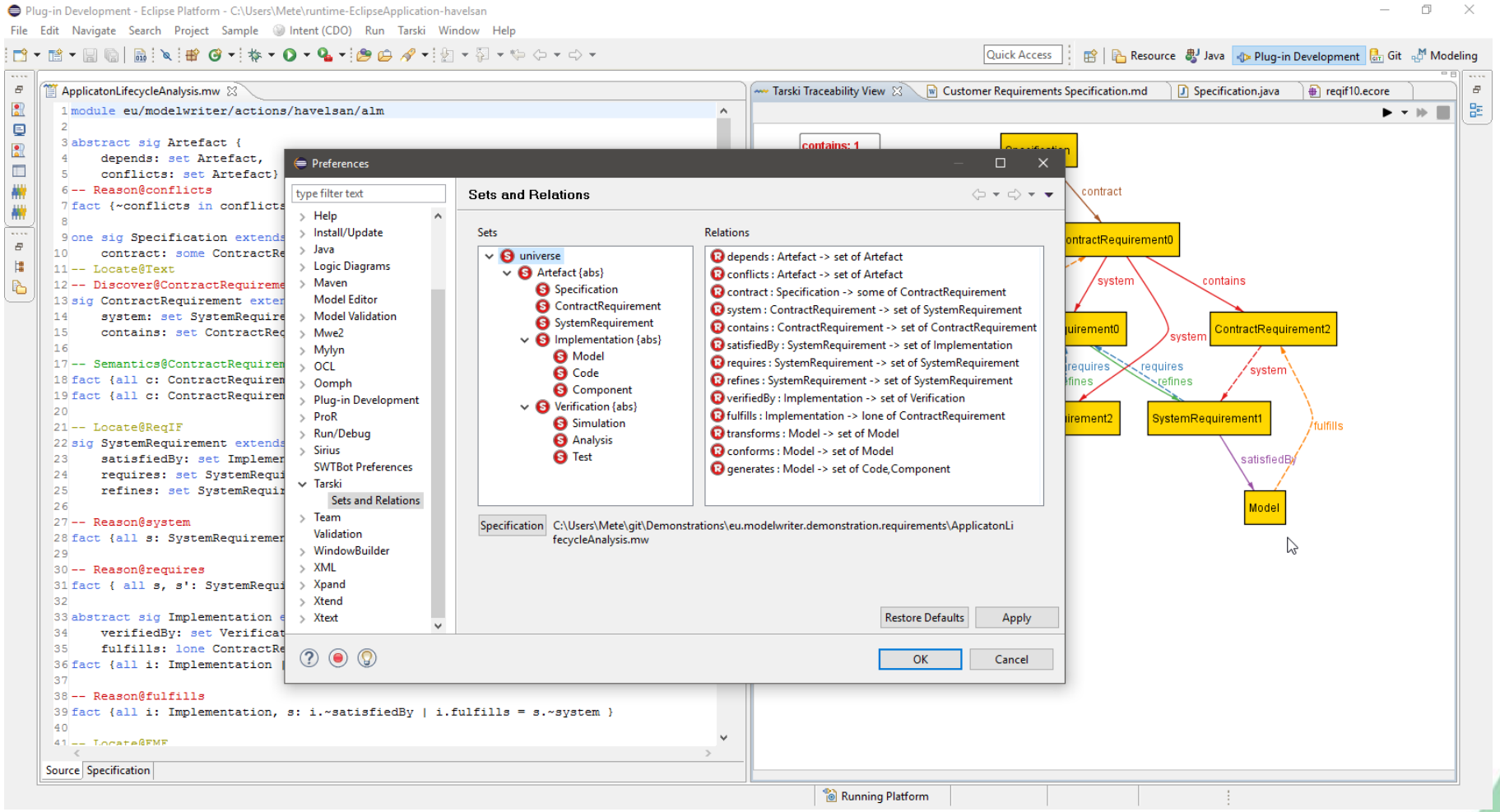
Tarski
Approach



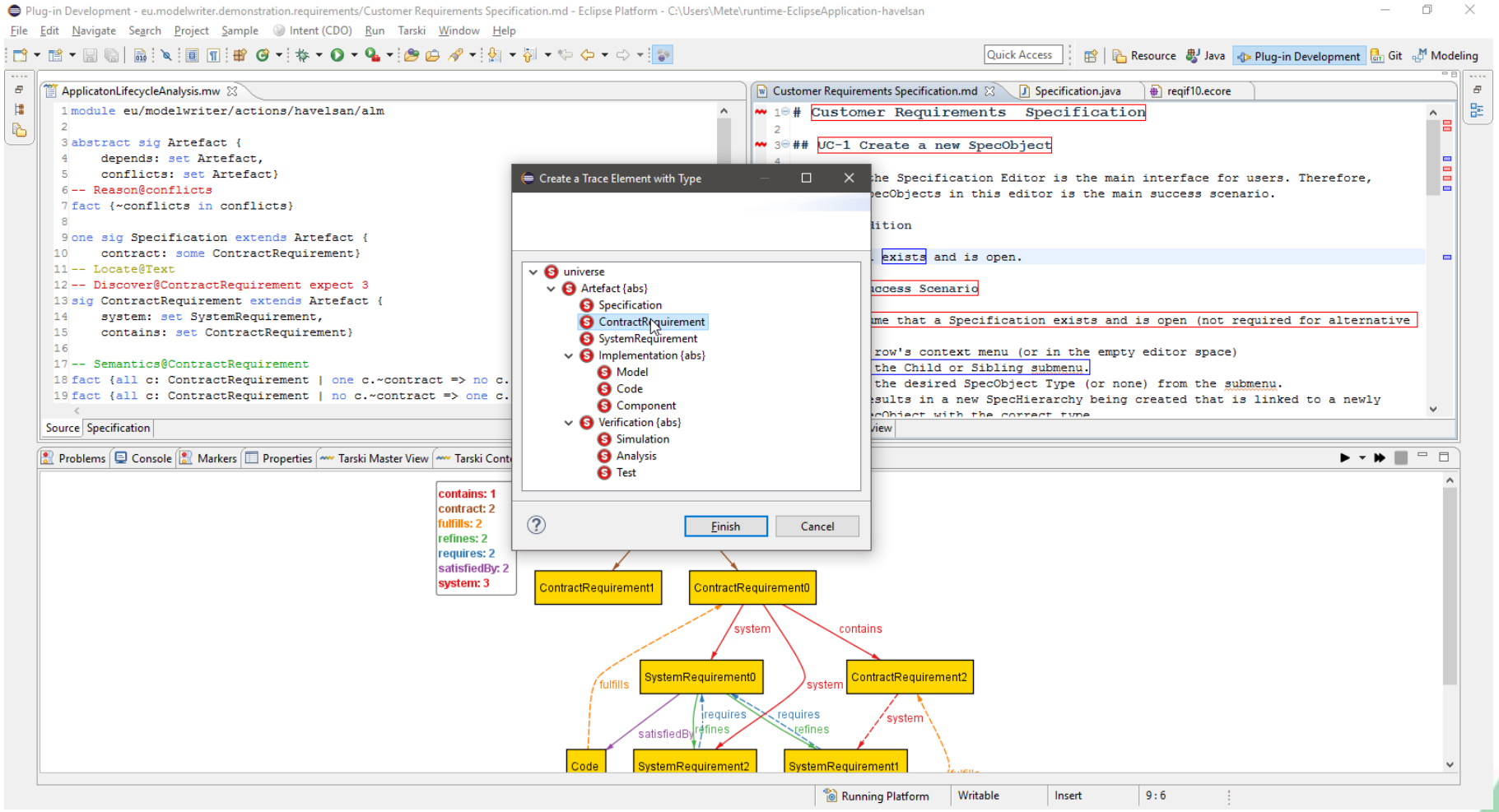
Tarski
Approach



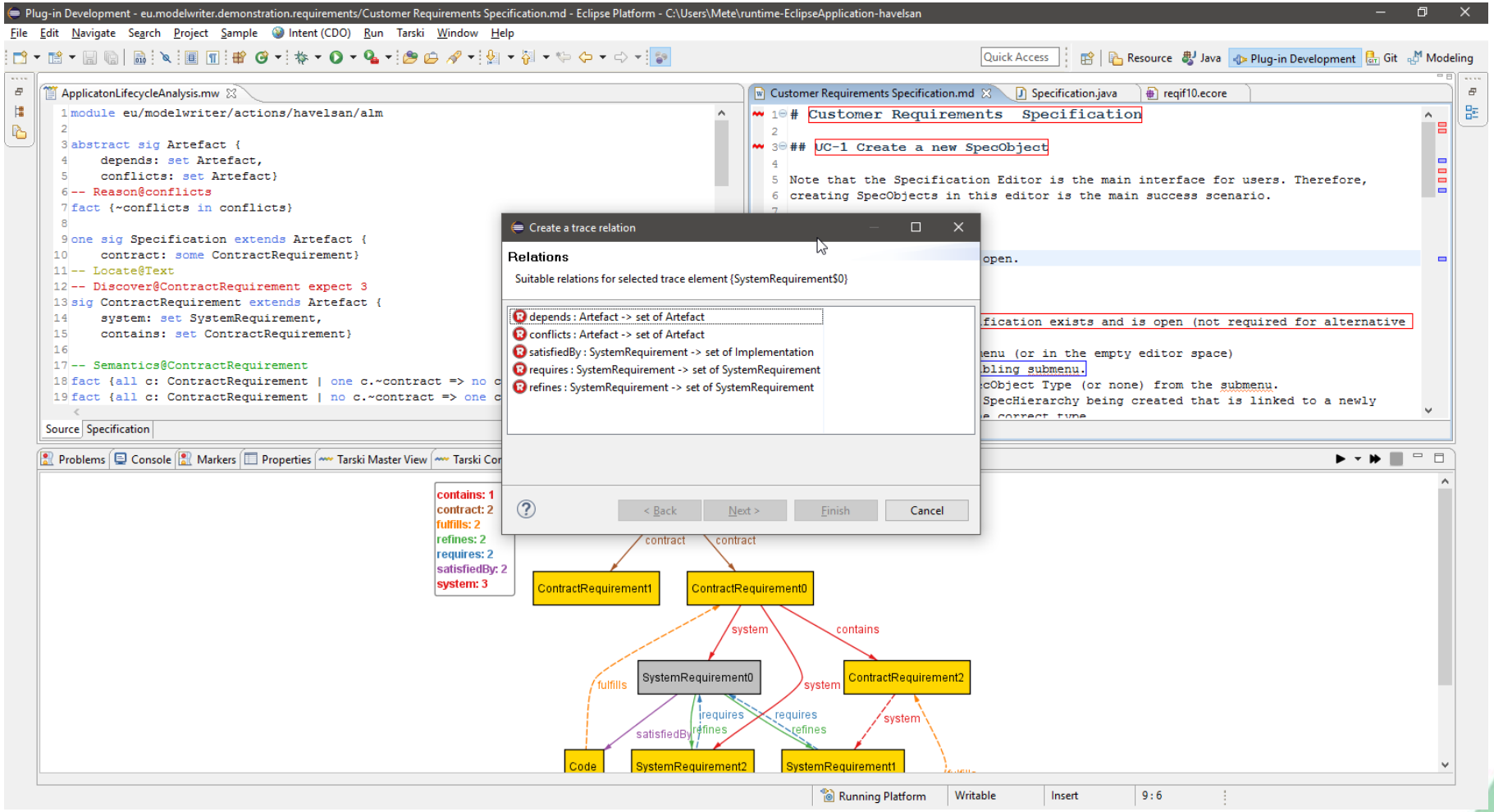
Types/Component Ontology derived from the specification



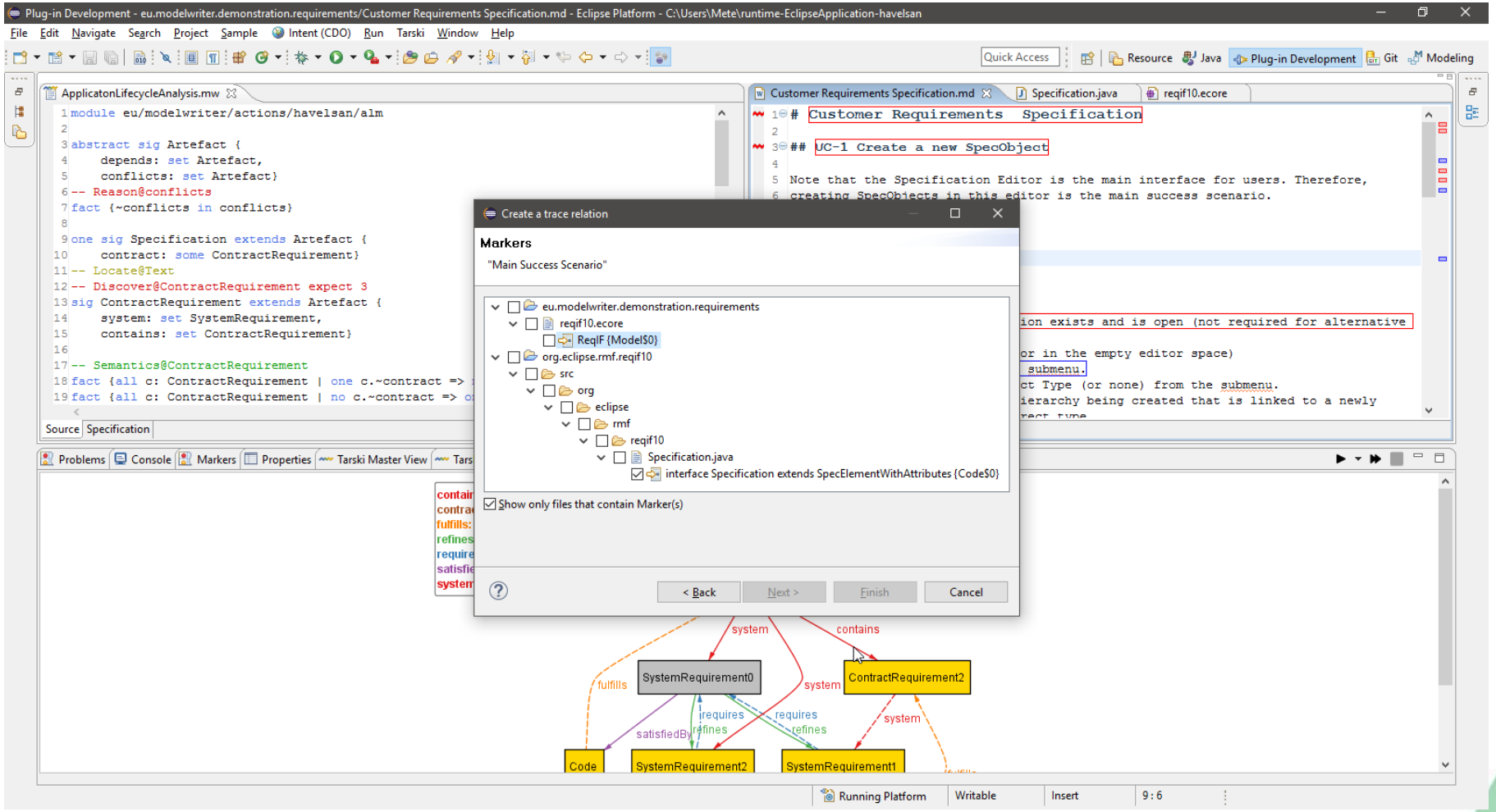
Assigning Unary Relations to a Traceable Elements



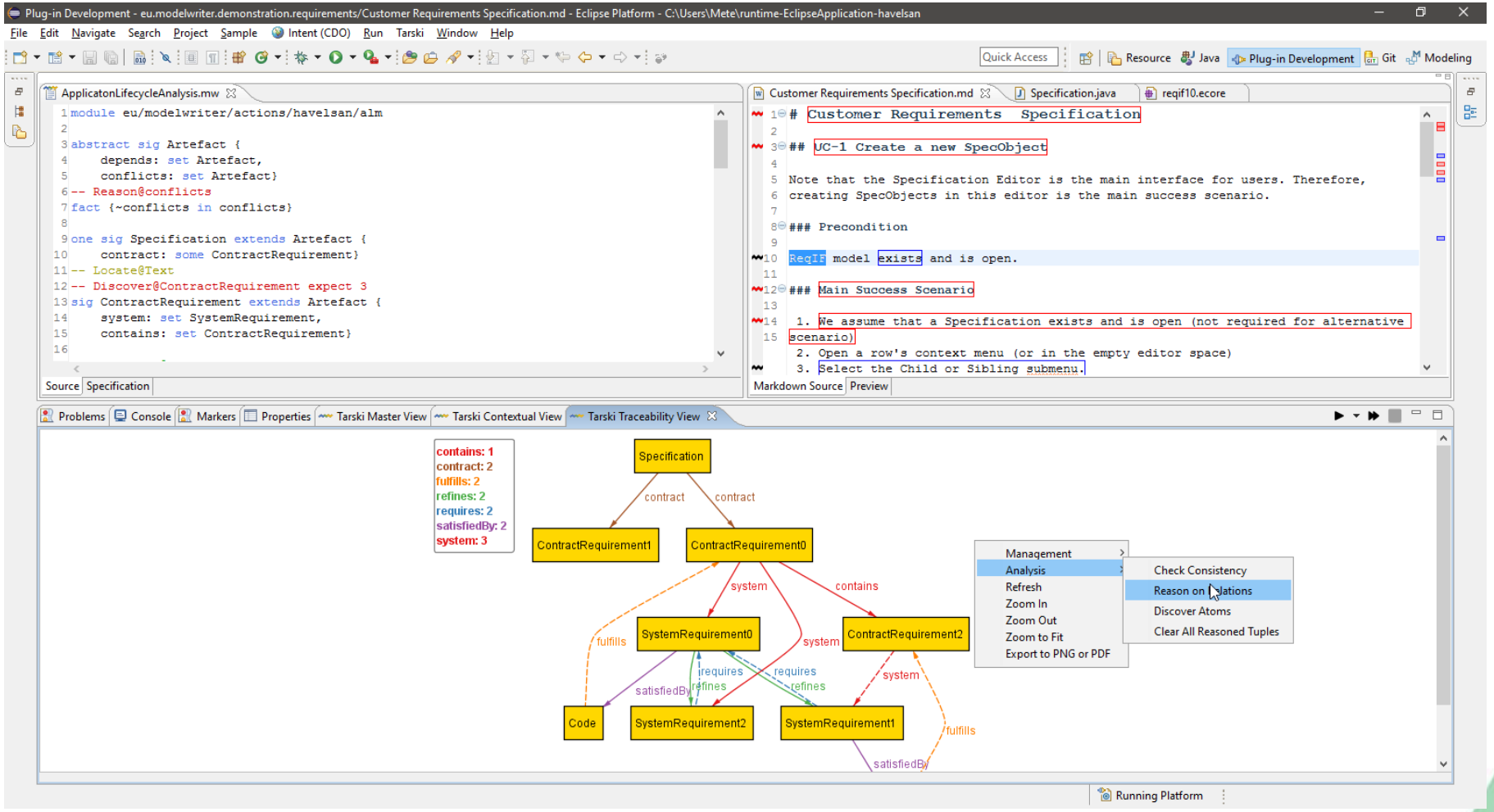
Assigning Binary Relations to a Trace Link



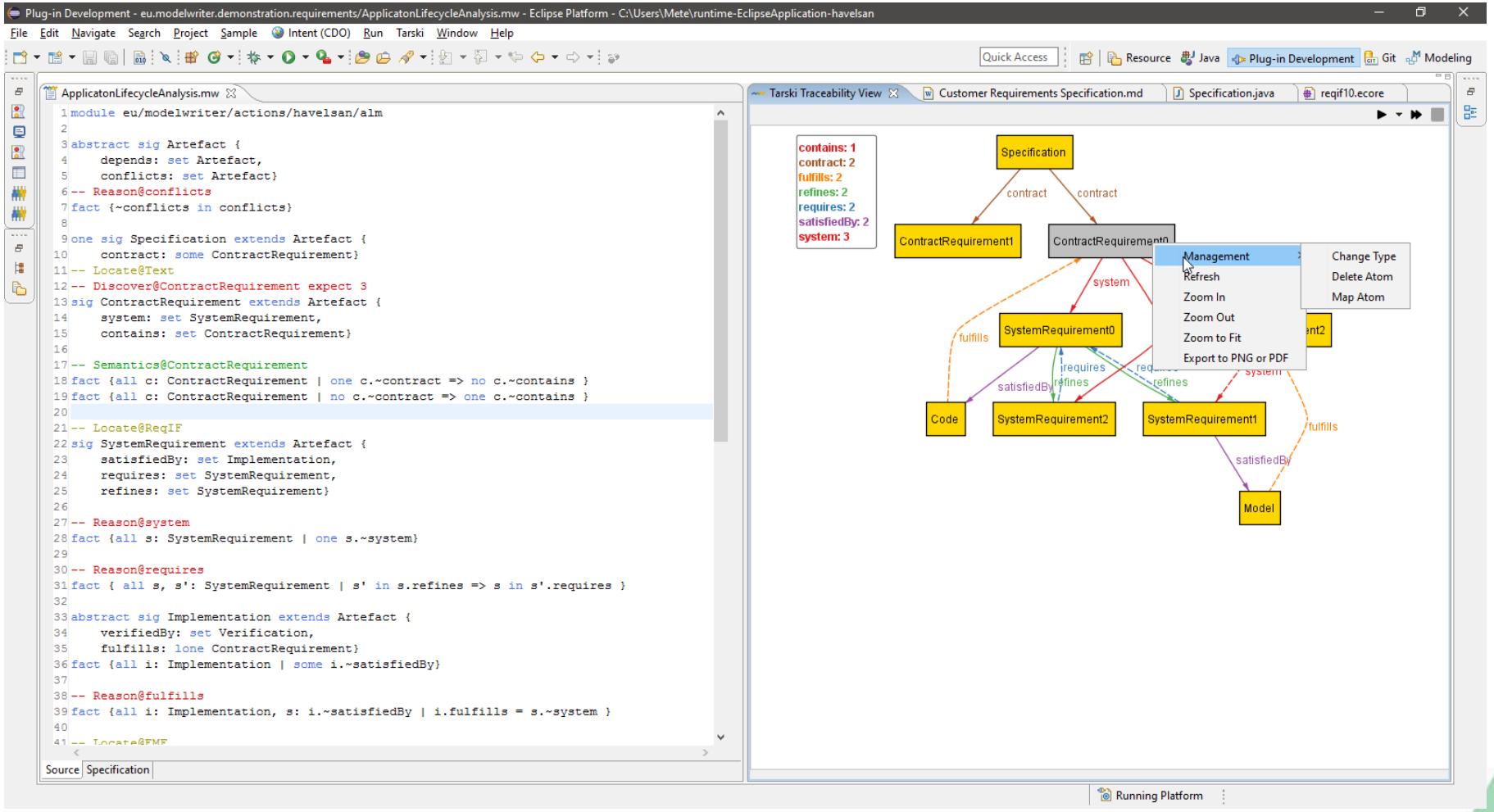
Selecting a range for a binary relation from an existing traceable elements



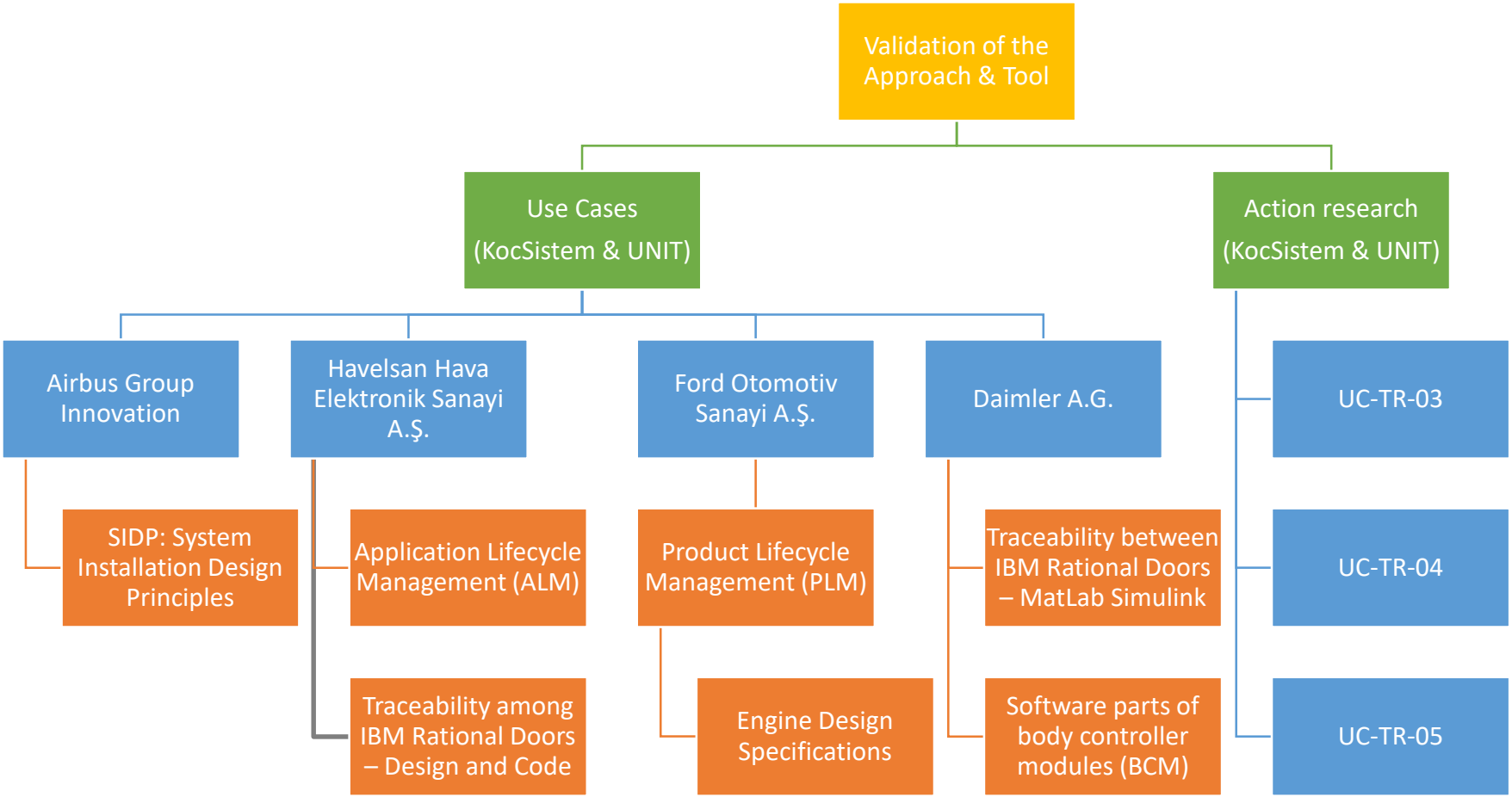
Automated Analysis of Traceability



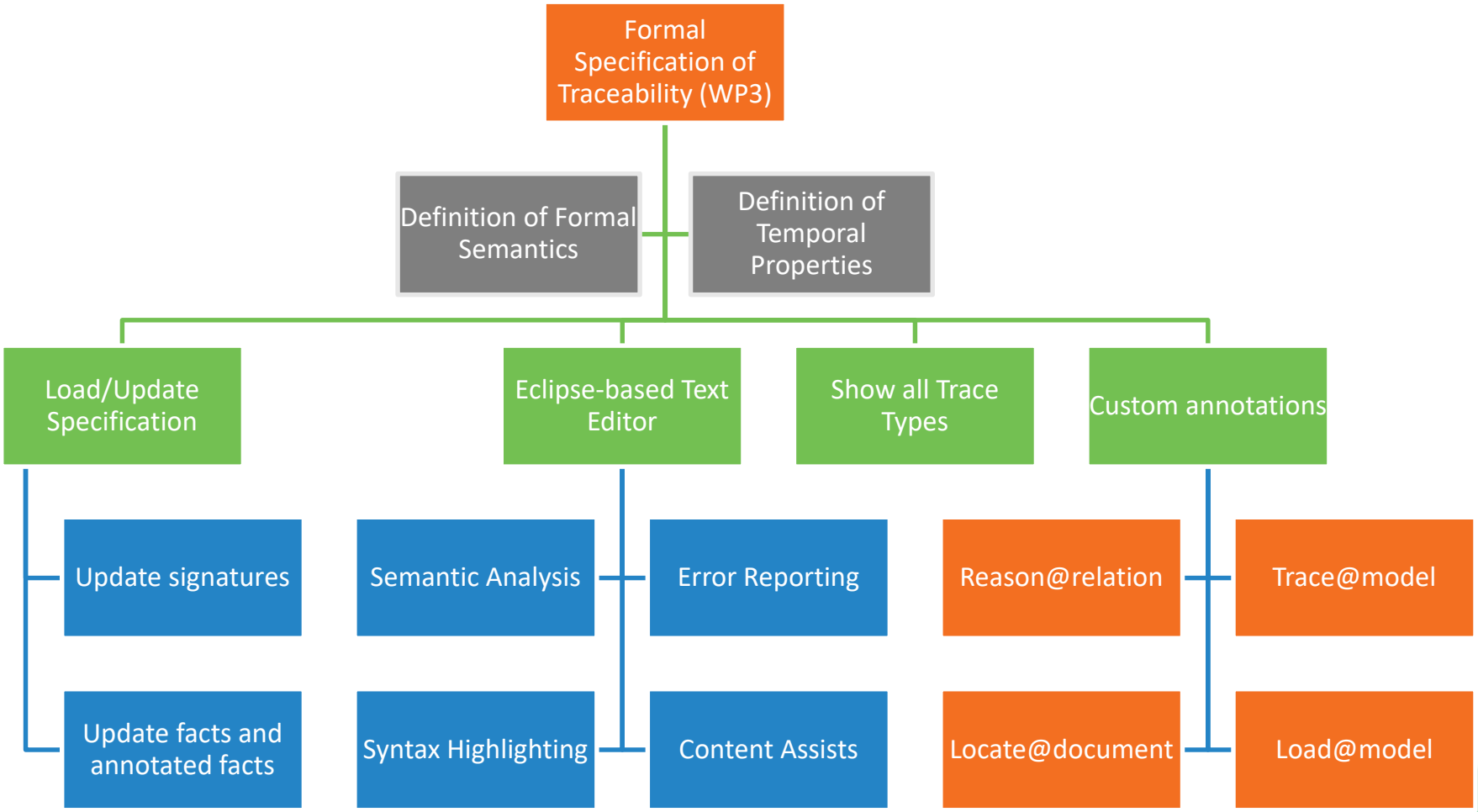
Dynamical Configuration & Model Management



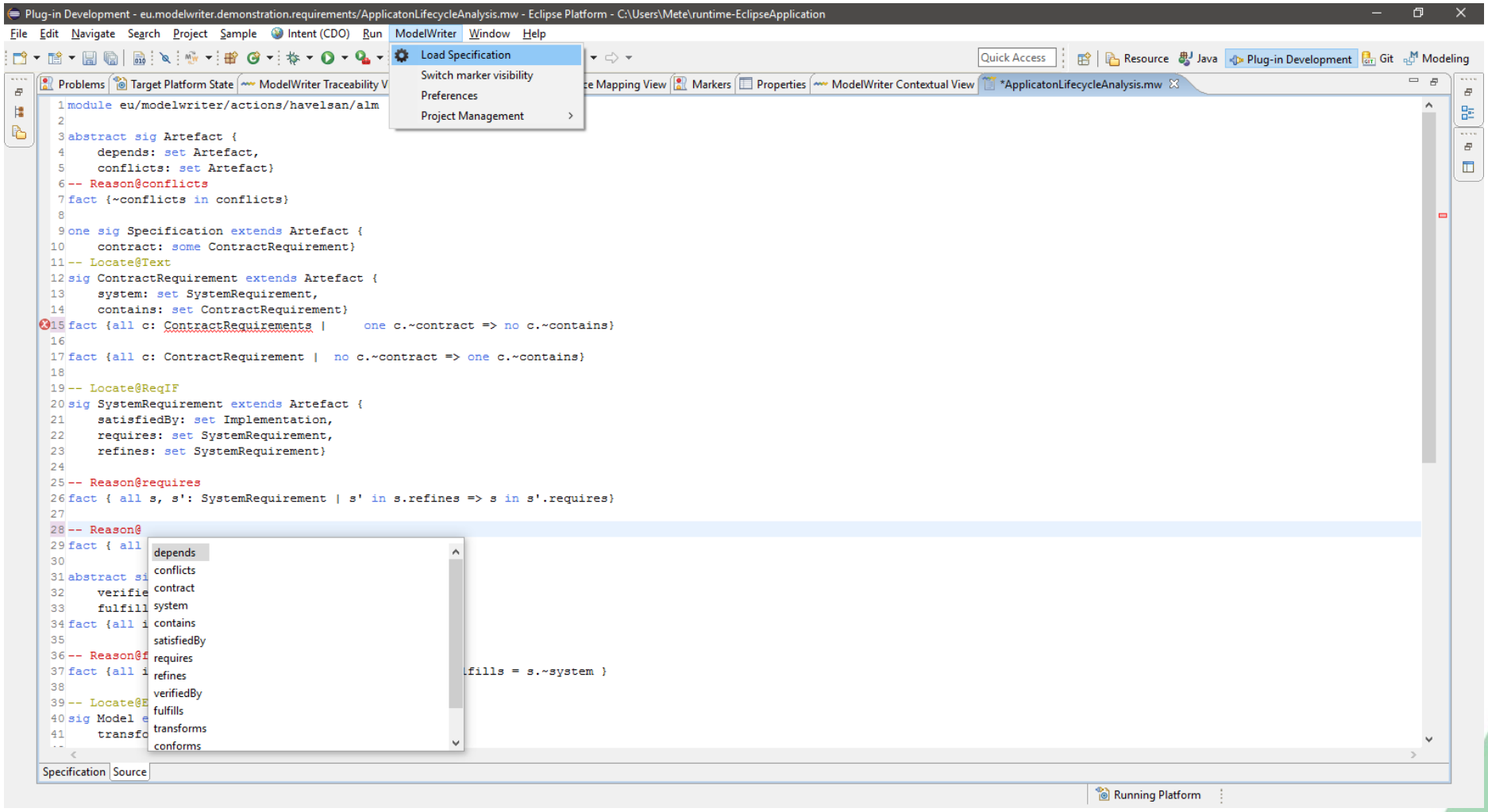
Validation of the Approach and Tool



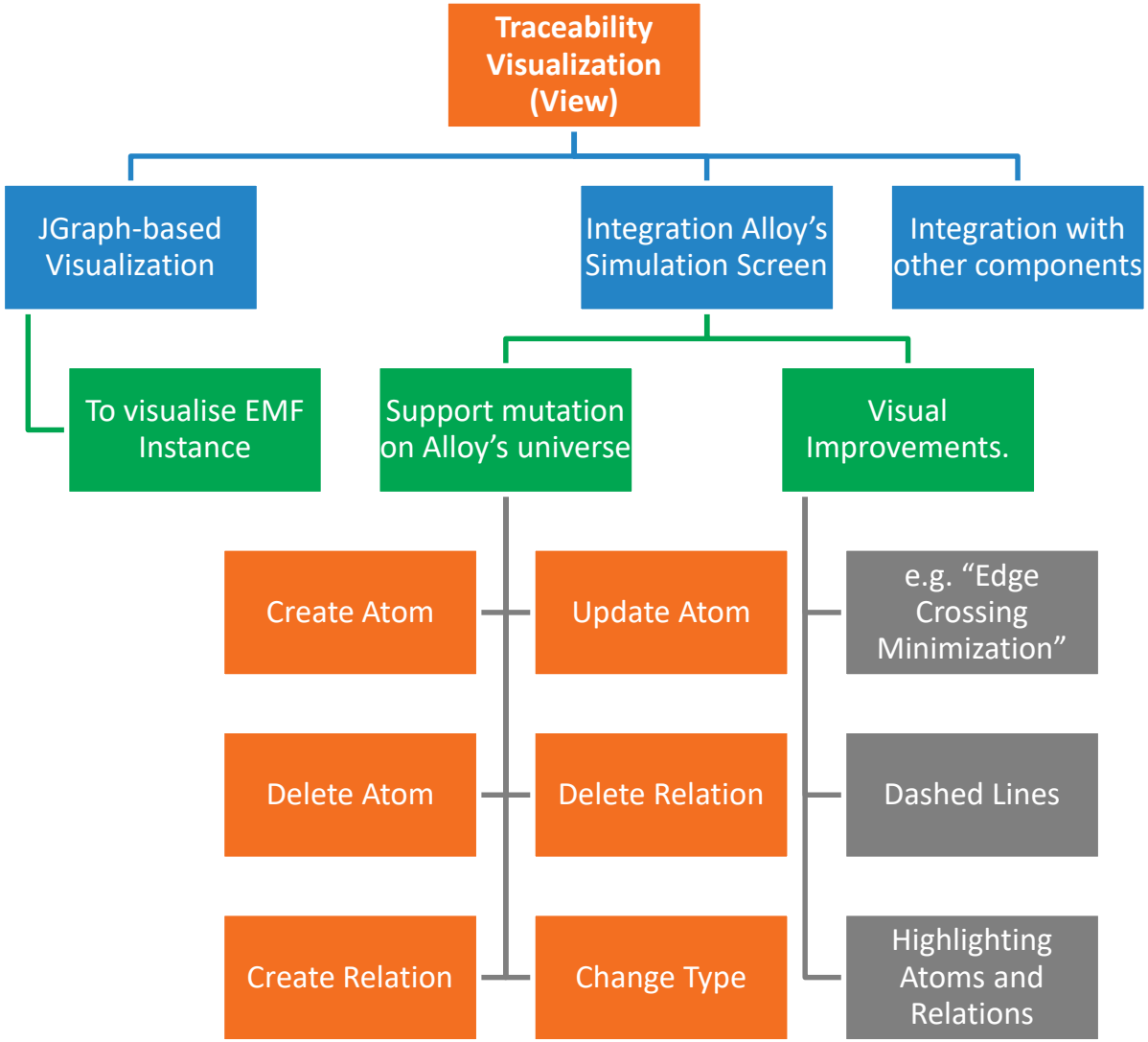
Formal Specification of Traceability (WP3) [UNIT]



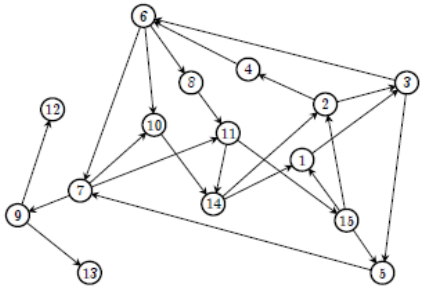
Demonstration
Textual Editor in Action



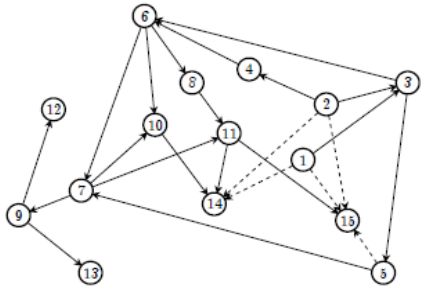
Traceability Visualization/View (WP3) [KoçSistem]



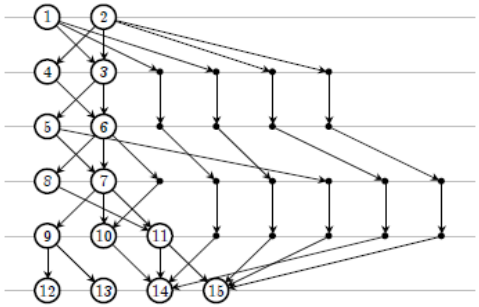
Layered/Hierarchical Graph Drawing Sugiyama Framework



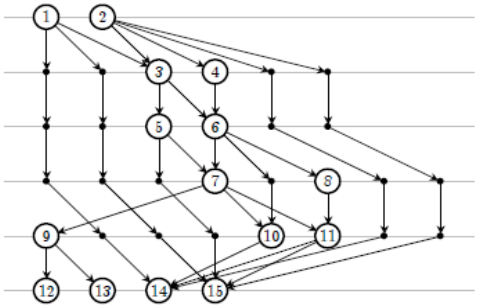
(a) Input digraph, G.



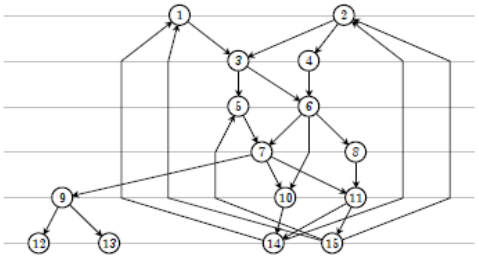
(b) Cycles removed.



(c) After leveling.

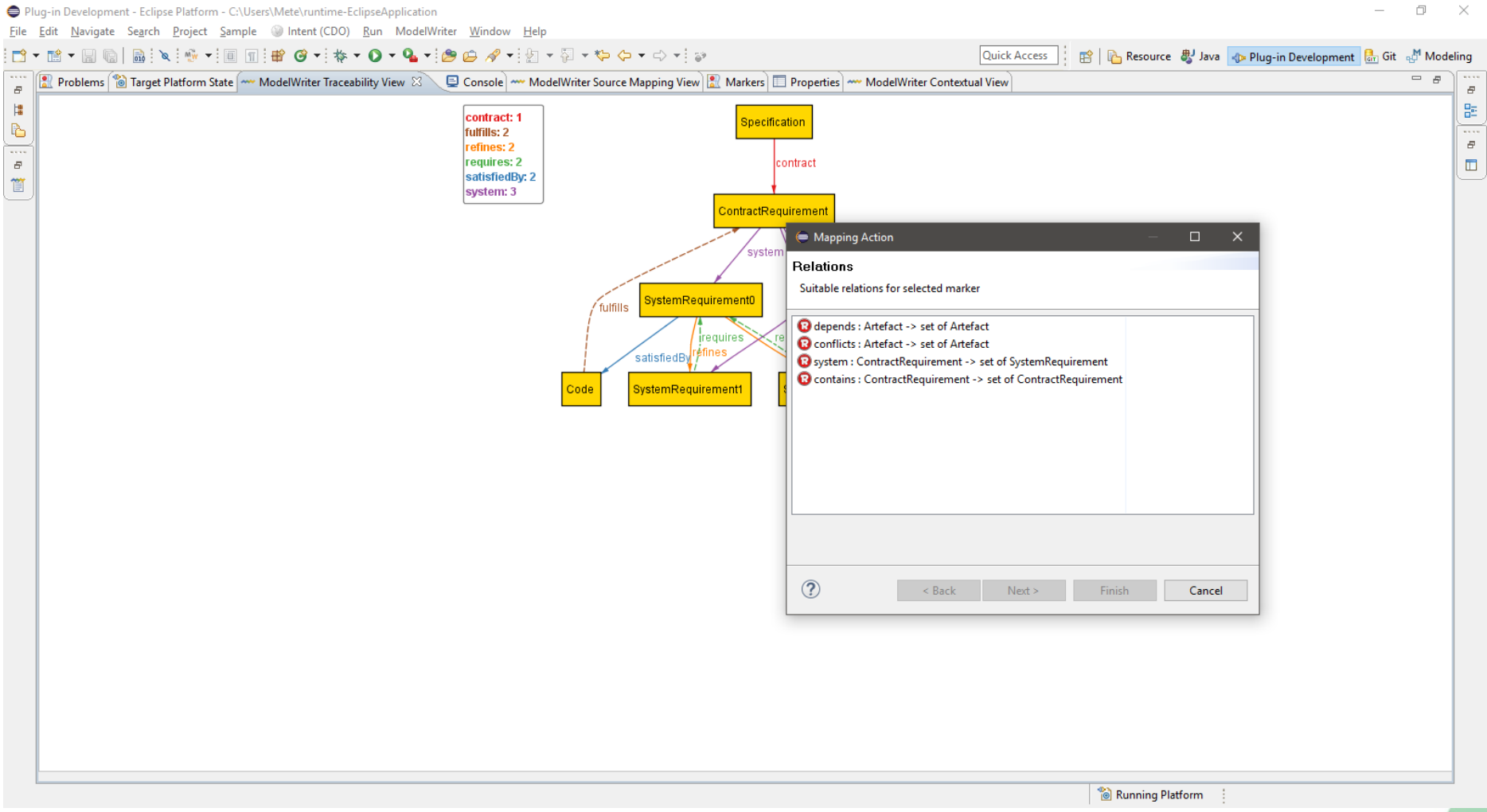


(d) Edge crossings minimized.

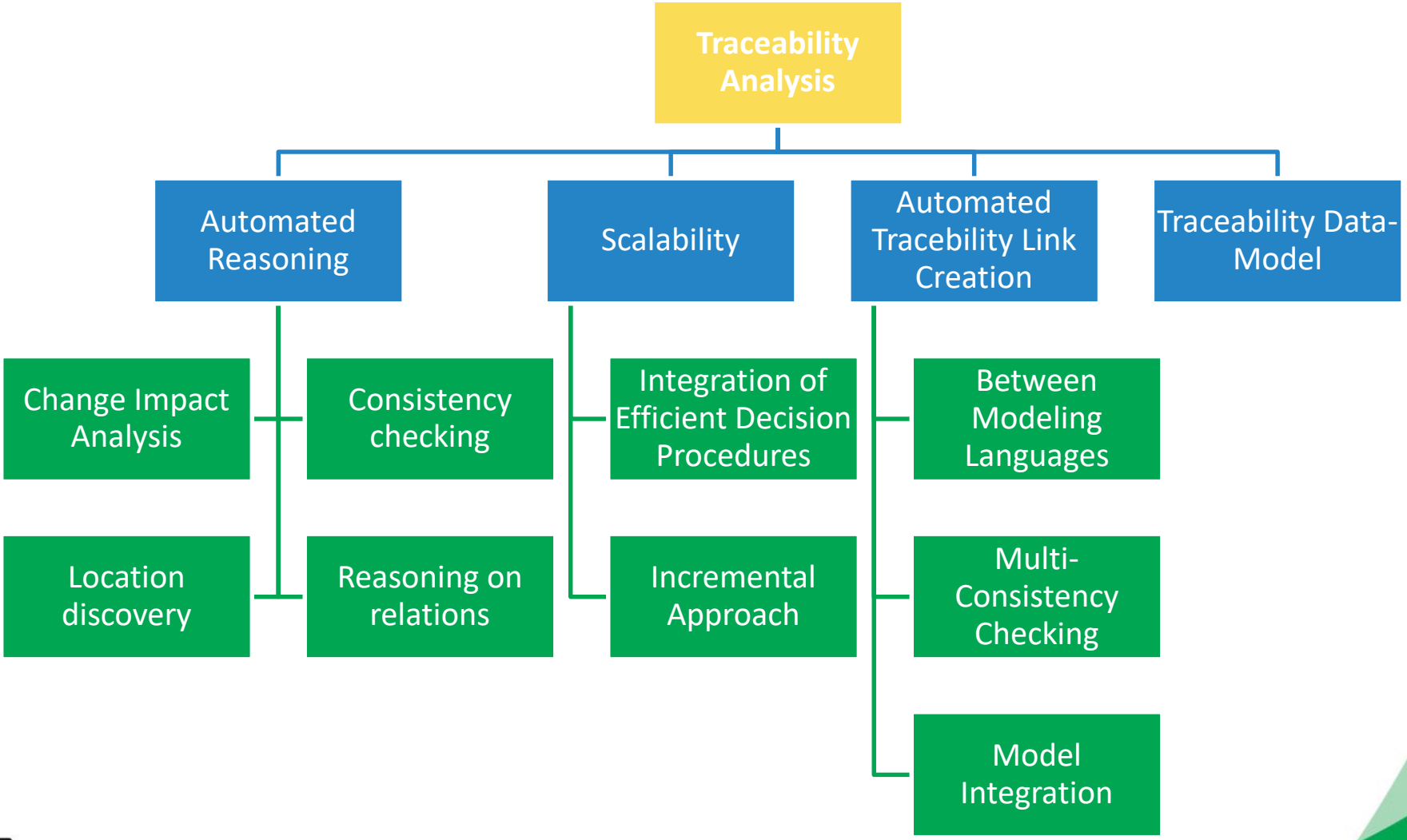


(e) Edges straightened.

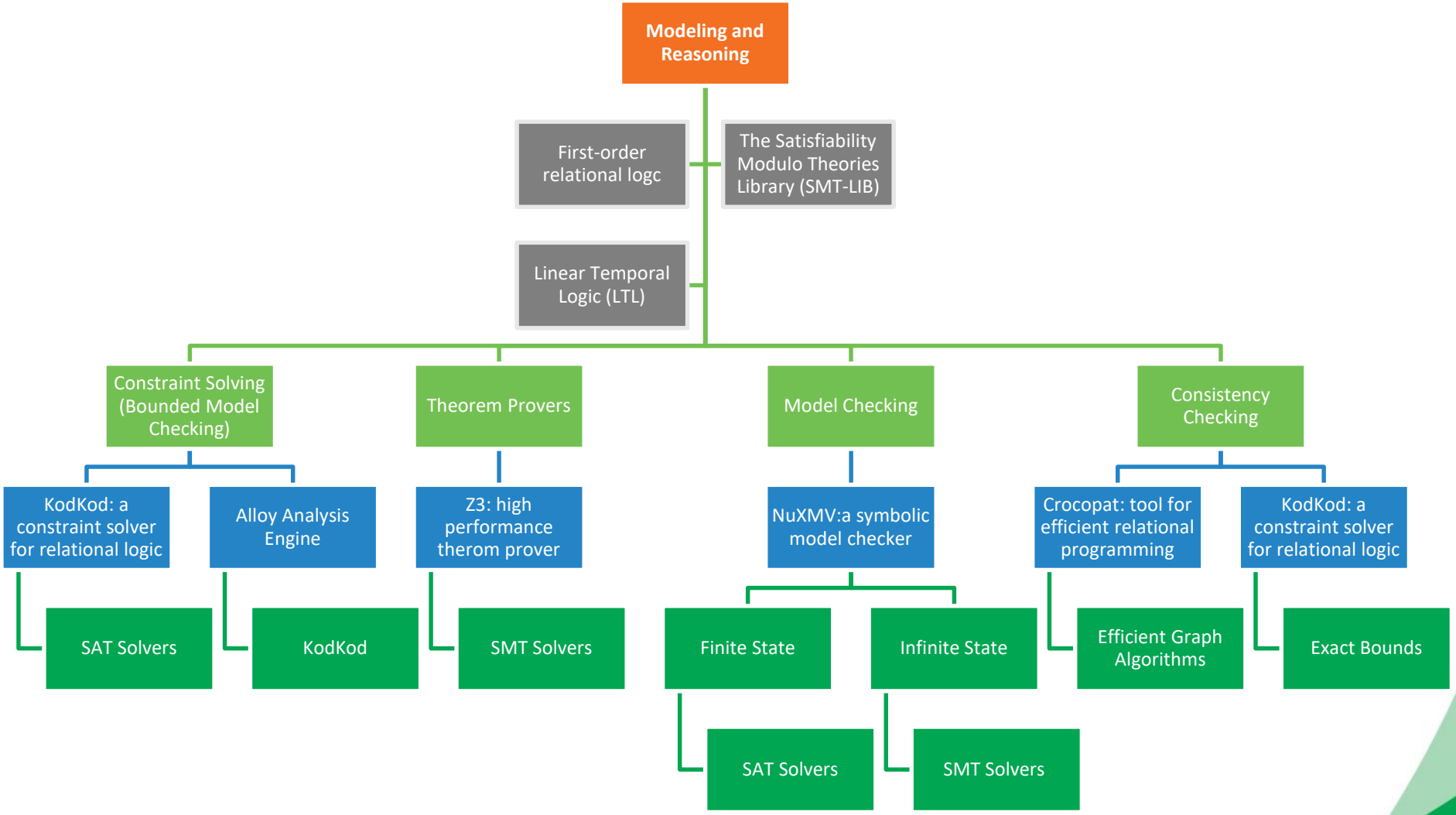
Demonstration
Visualization in Action



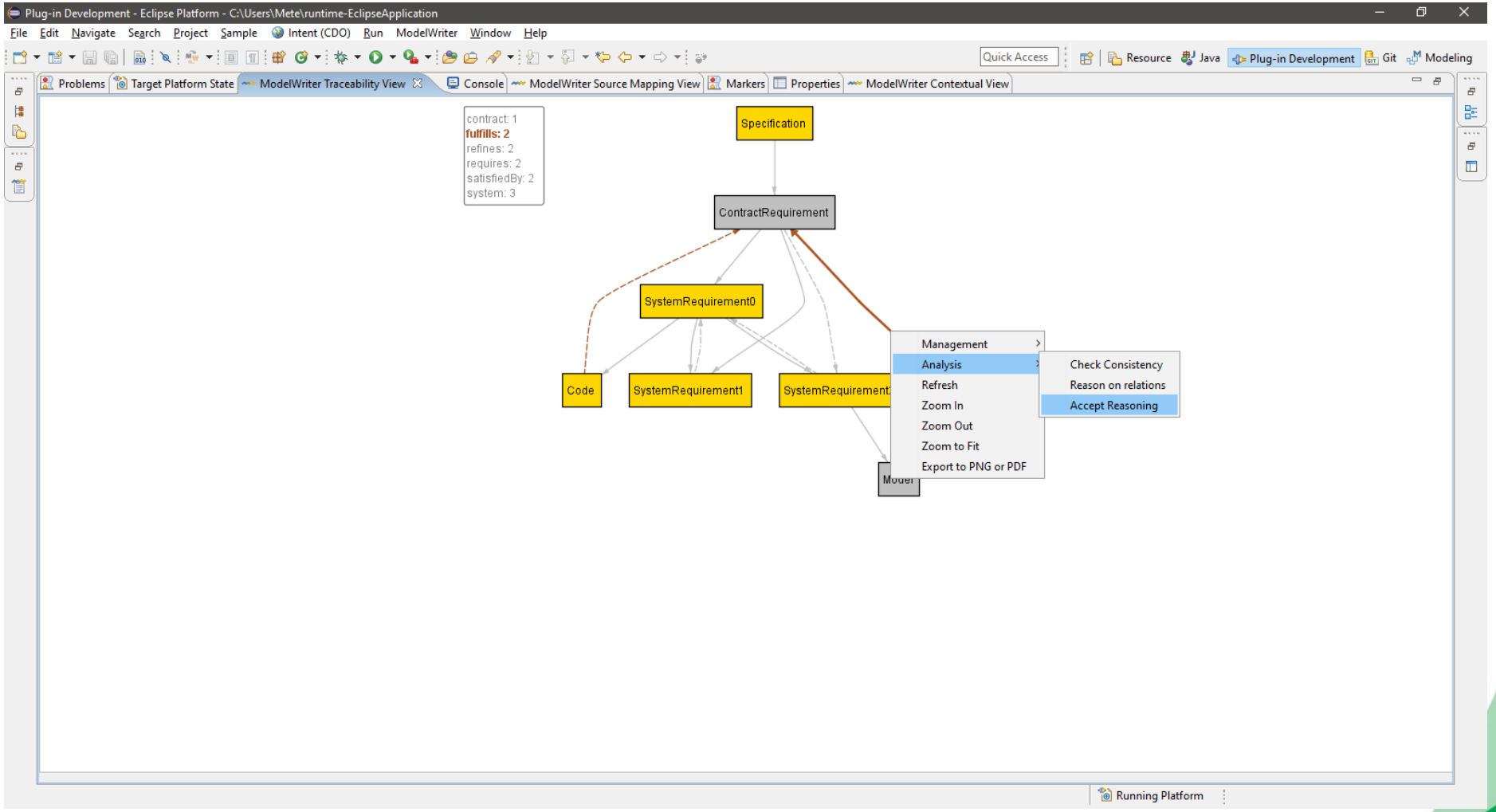
Traceability Analysis (WP4 & WP3) [UNIT]



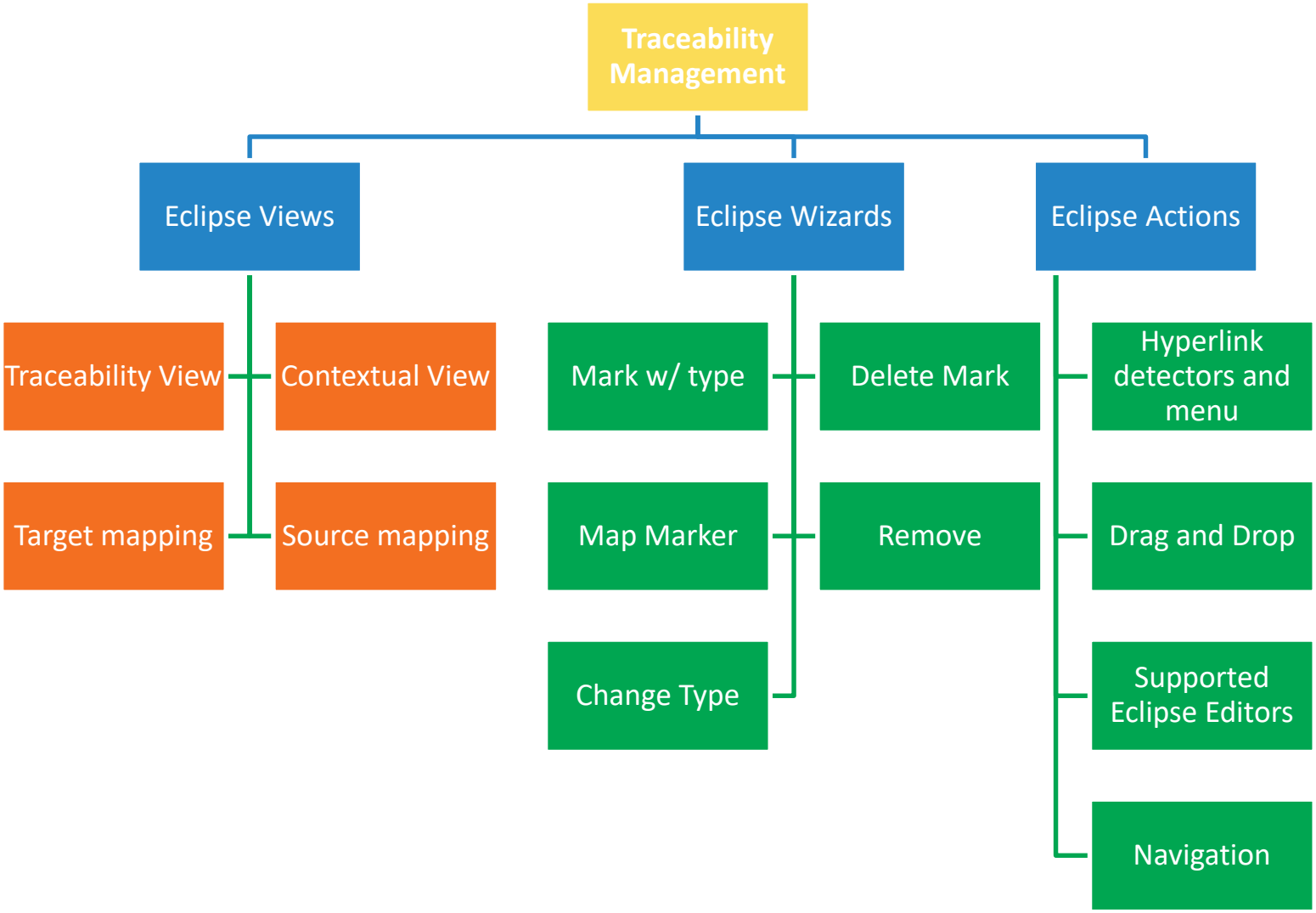
Modeling and Reasoning Approaches (WP3) [UNIT]



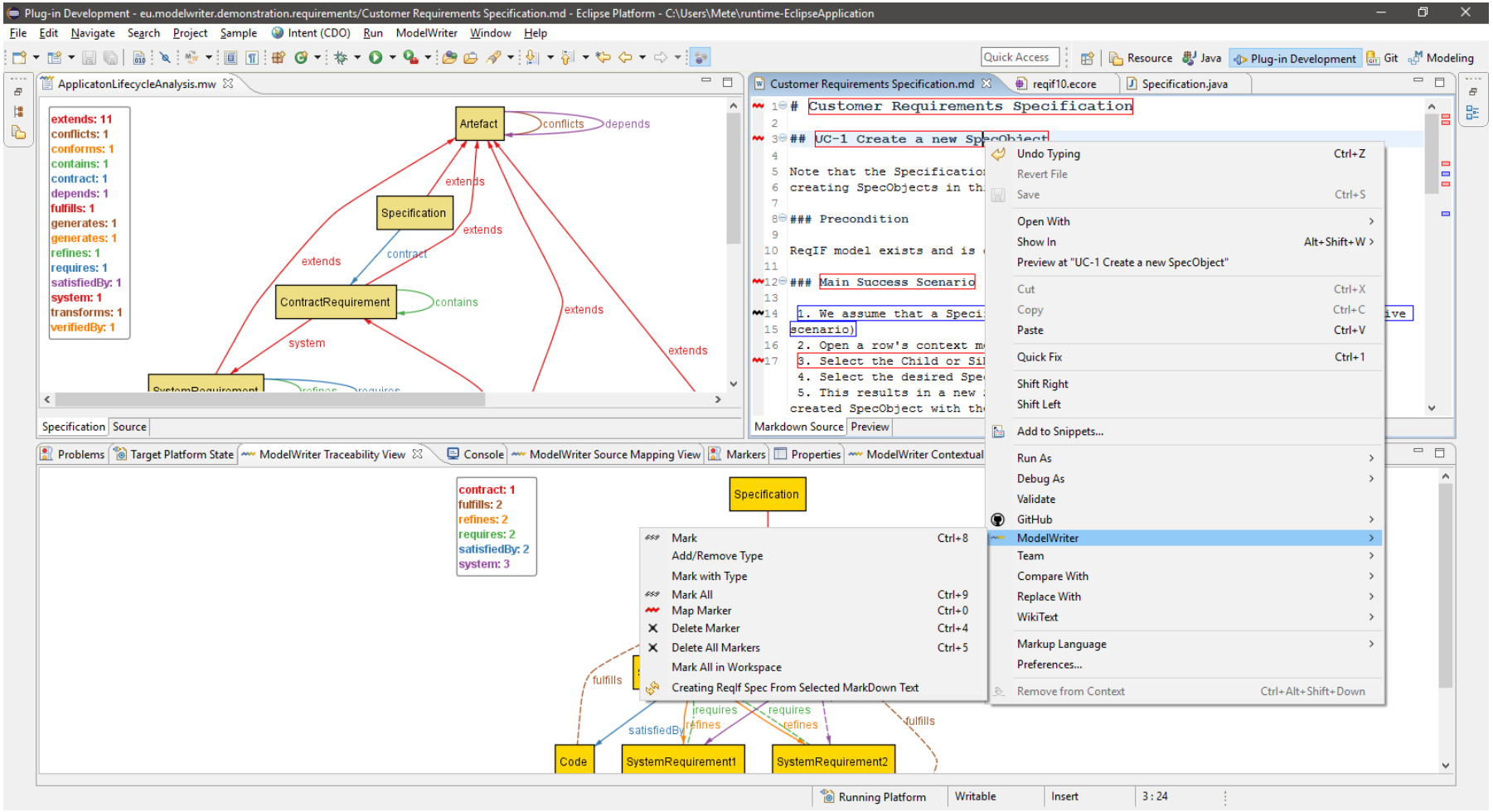
Demonstration
Traceability Analysis in Action



Traceability Management (WP6) [KoçSistem]



Demonstration
Traceability Management in Action



Thank you for your attention
We value your opinion and questions.

WP4 – Knowledge base Design and Implementation

Prof. Dr. Geylani Kardas
(KocSistem)

WP4 Knowledge Base Design and Implementation

- Design and implement the ModelWriter's:
  - federated Knowledge Base itself, hosting multiple formalisms.
  - bi-directional text-model synchronization mechanism.
  - required API
  - a set of specialised modules (plug-ins) that exploit the Knowledge Base in ways that make the tasks of Technical Authors much more productive, e.g. consistency checks.
  - the collaborative functions linking and hierarchically organizing multiple ModelWriter KBs used by different Technical Authors on different sites.

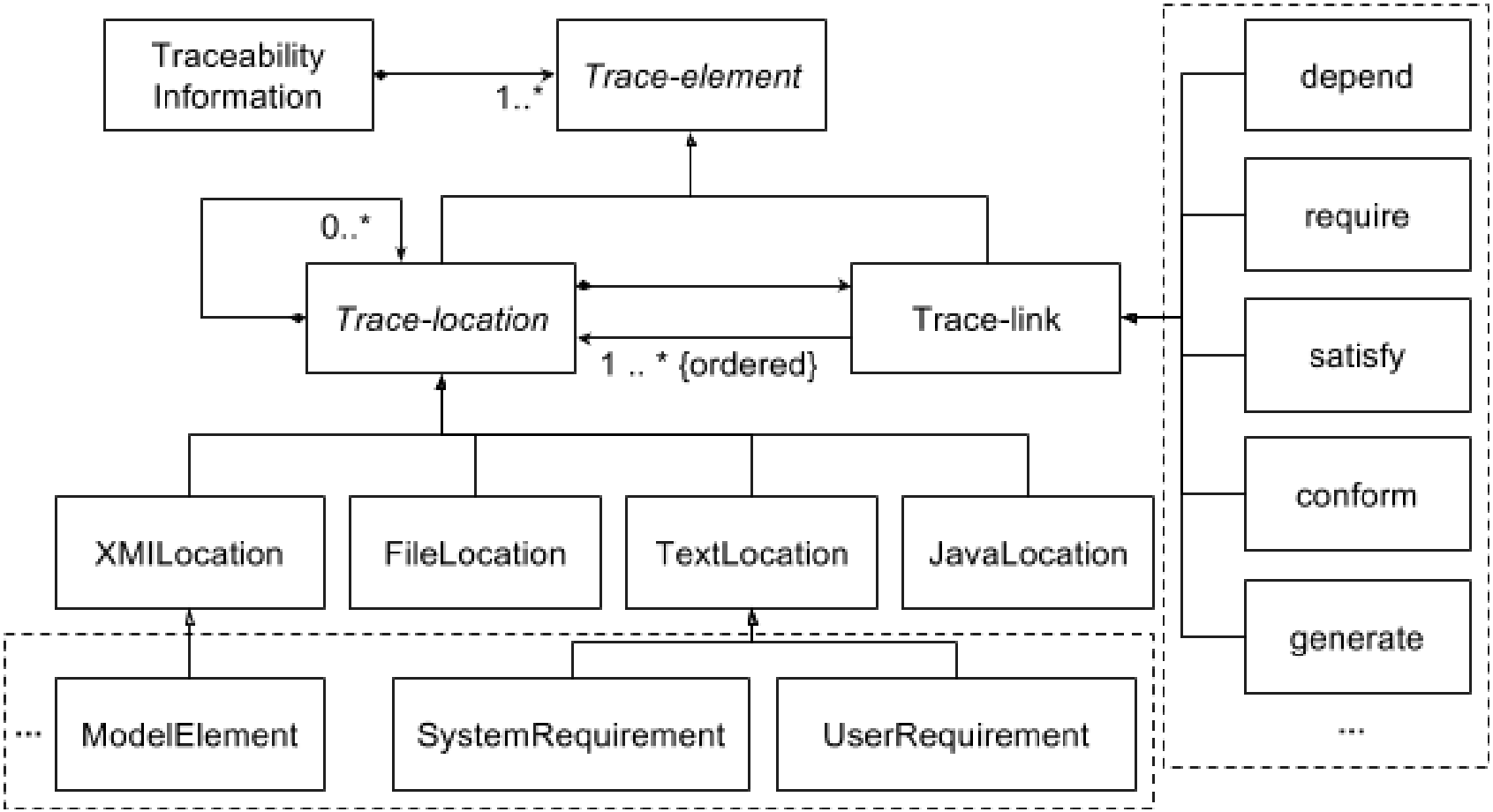
WP4 Knowledge Base Design and Implementation

- Plug-in #1 – This provides consistency and completeness checks within the same software lifecycle document, allowing automatic quality review of the content (meaning).

- Plug-in #2 – This provides consistency and completeness checks between related set of documents.

- Plug-in #3 – This provides semantic comparison between two versions of the same software lifecycle document (i.e. what conceptual changes have happened).

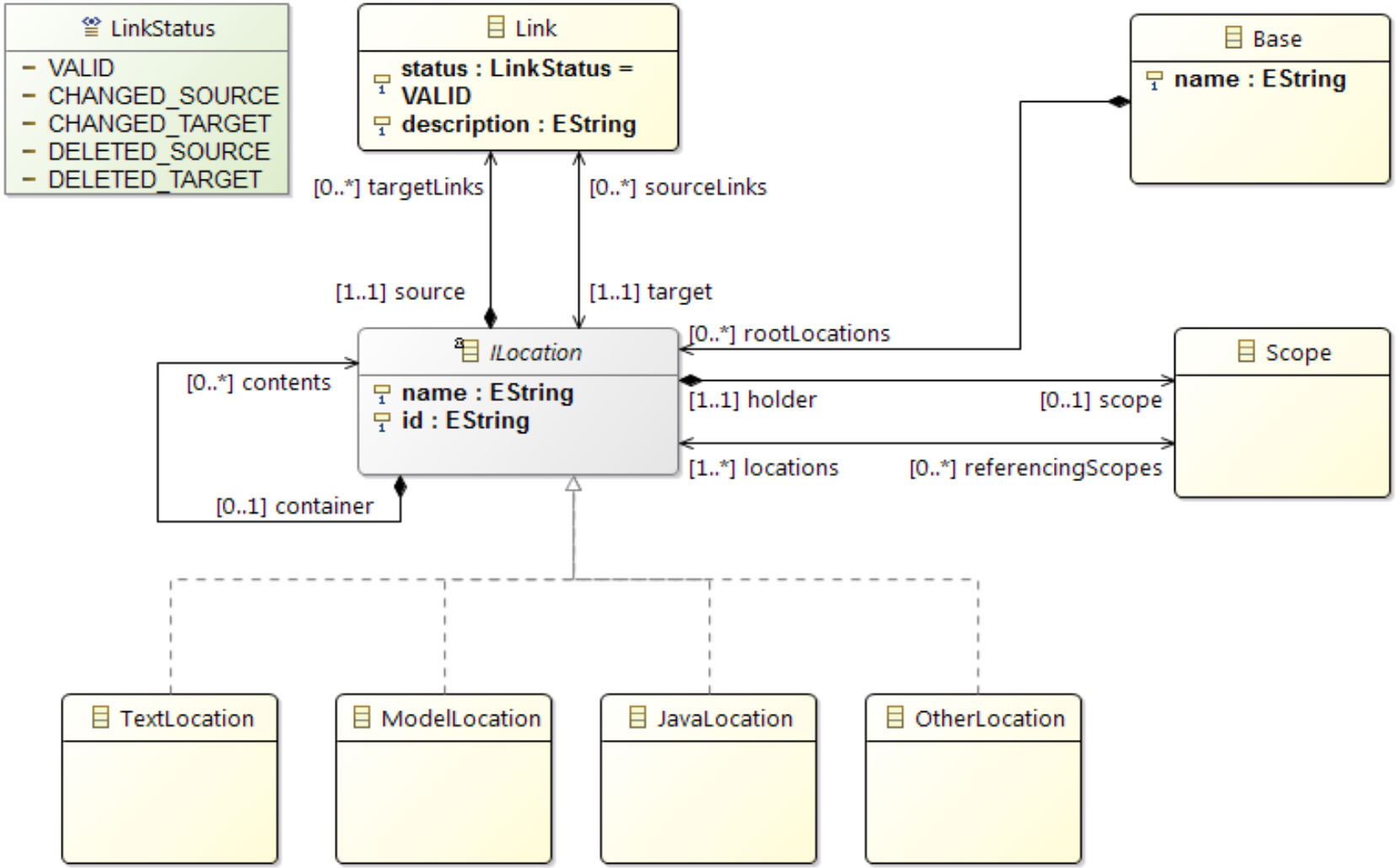
ModelWriter Core Model Approach



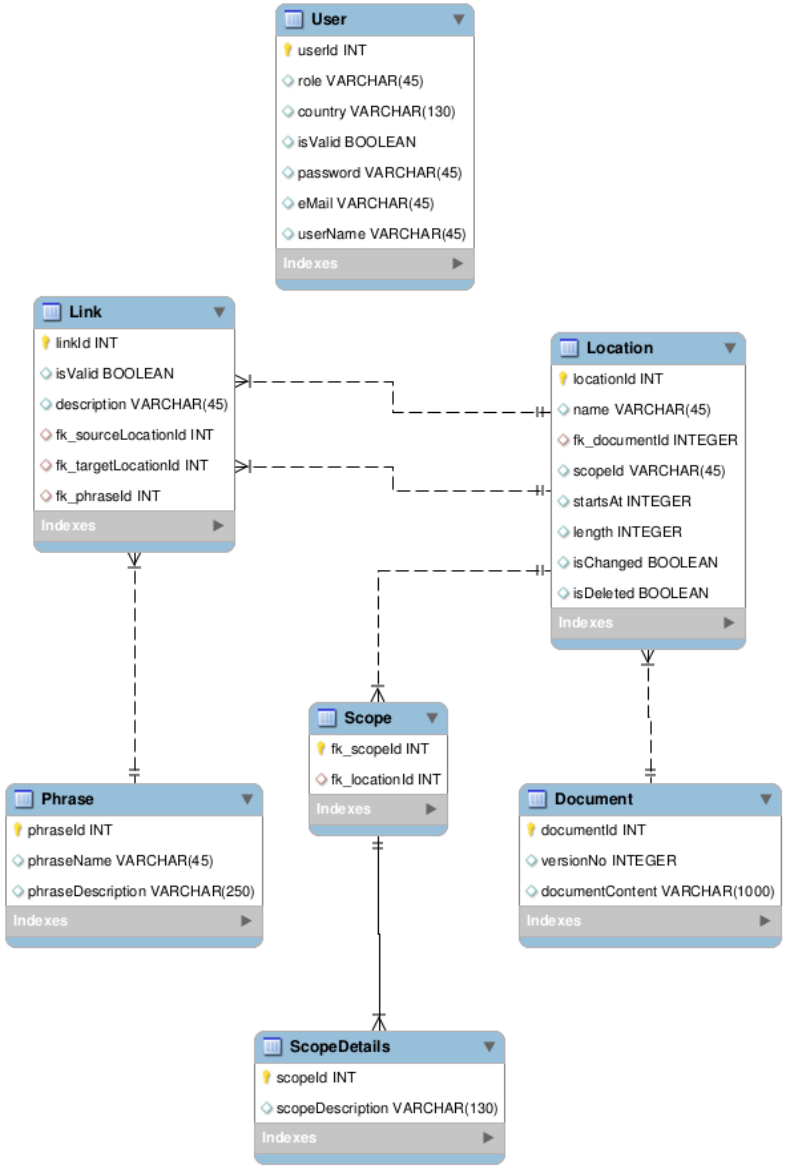
ModelWriter Core Model
State-of-the-art

	SotA Tools & Approaches	Consideration of Different Artefacts/ Heterogeneity of artefacts (internal or external models)	Traceability Management		
			Approach (Management of Traceability)	Definition of Formal Semantic for trace-links	Dynamic Configuration of Semantics of trace-links
Industrial Tools, Methods & Standards	SysML[1]	UML Elements	UML Profiling mechanism	-	-
	ReqIF[2]	Textual Requirements	Definition of XML Schema and extending its Data-Model	-	-
	IBM Doors[3]	Arbitrary between model elements	Creation of Relation Types	Transitivity of relations	-
	ModelWriter	Arbitrary between any model element or textual requirements	Basic Traceability Model extended with a Formal Specification	FORL (First-order Relational Logic)	+

ModelWriter Core Model Implemented by OBEO



Knowledge-base Design

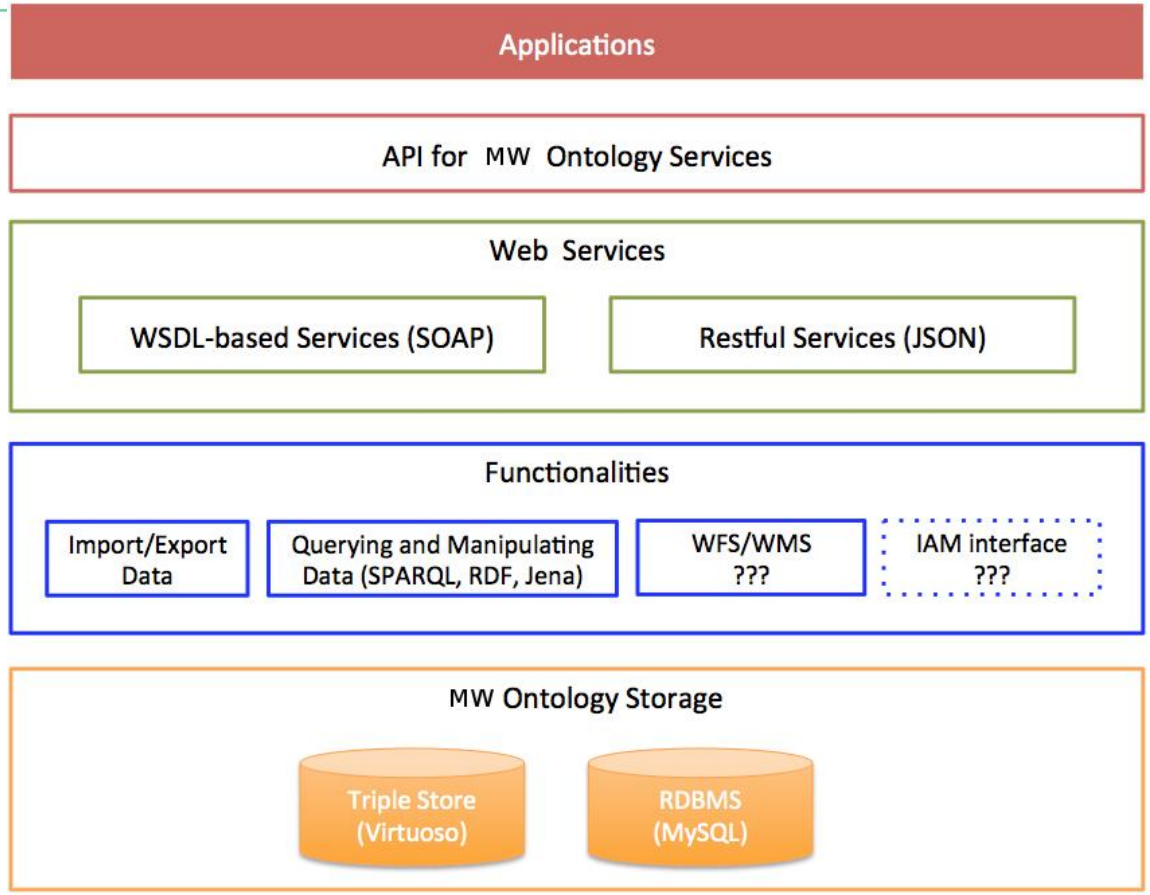


The class diagram of the knowledge base data structure representation.

MANTIS has prepared D4.1.1 Knowledge Base Design Document.

This document still is open to edit and to add contributions.

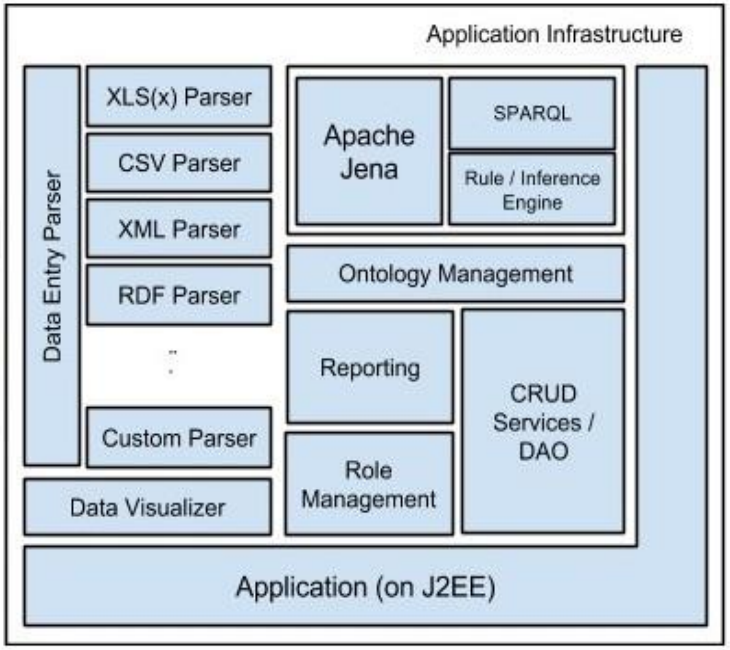
Knowledge-base Implementation
Ontology Infrastructure

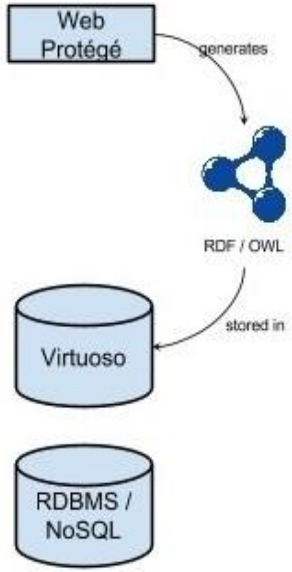


WMS/WFS services: standardized by Open Geospatial Consortium to use geographical attributes. **IAM interface** for identification and access management. Current platform uses OAuth2 authentication system (http://oauth.net/2/)

Web Feature Services (WFS)/Web Map Services (WMS) may be partially implemented. Identity and access management (IAM) services may not be implemented at all.

Knowledge-base Implementation
Ontology Infrastructure





The requests that arrives via web services are stored as triple in the Virtuoso Ontology Database and can be queried.
All processes that are based on ontology are done in Ontology Management Module (OMM). OMM is set up on parent Jena framework and inference engine can be utilized efficiently with SPARQL queries.
Also by using Web Protégé ontology can be generated in RDF/OWL formats and that ontology can be stored in Virtuoso.

Ontology Issues and Services

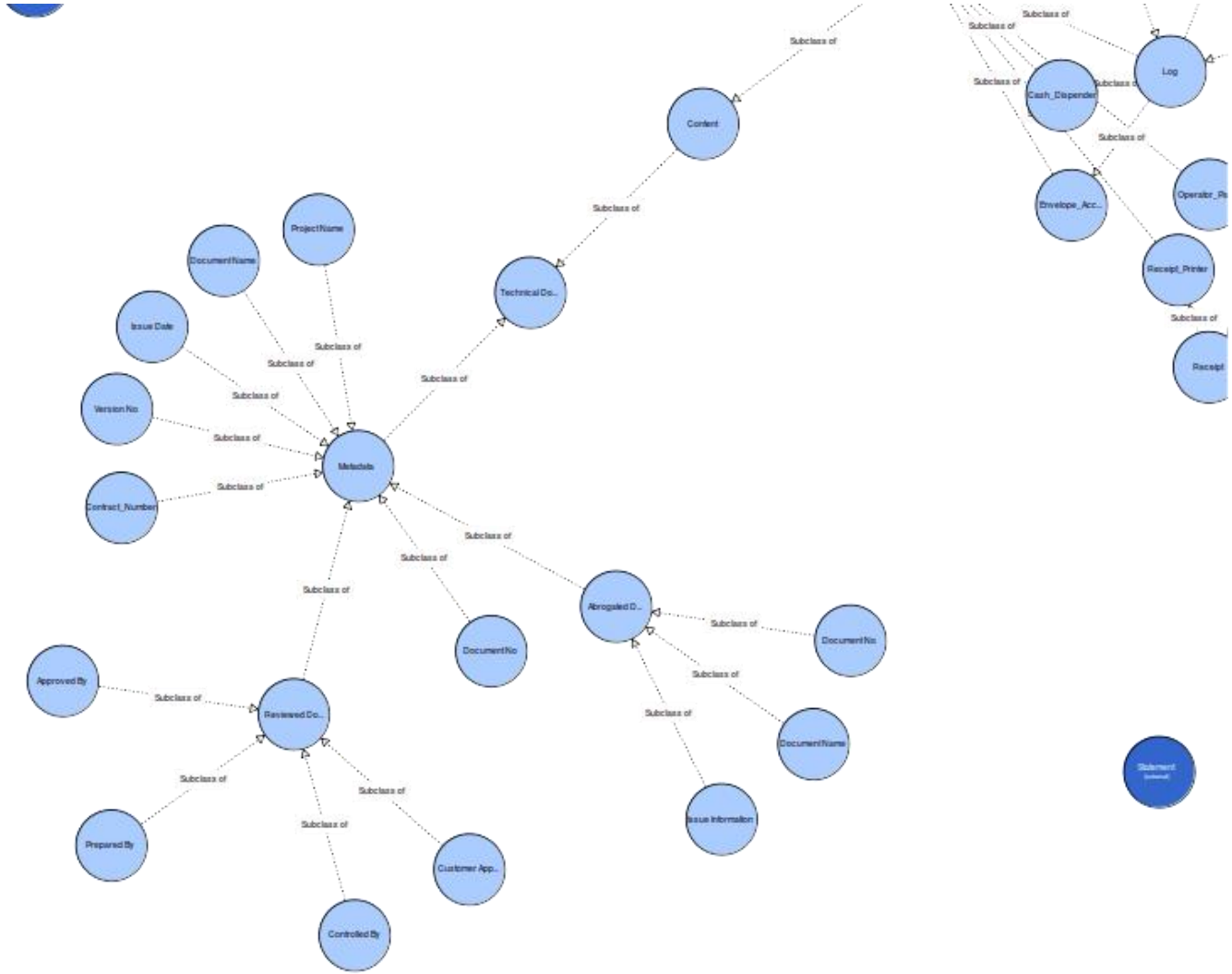
Ontology Issues

CRUD operations on ontologies as RESTFUL services
Using a sample design document, Mantis designed a document ontology
Ontology is hosted on Mantis servers
Manual RDF export
Automatically RDF export (working on)

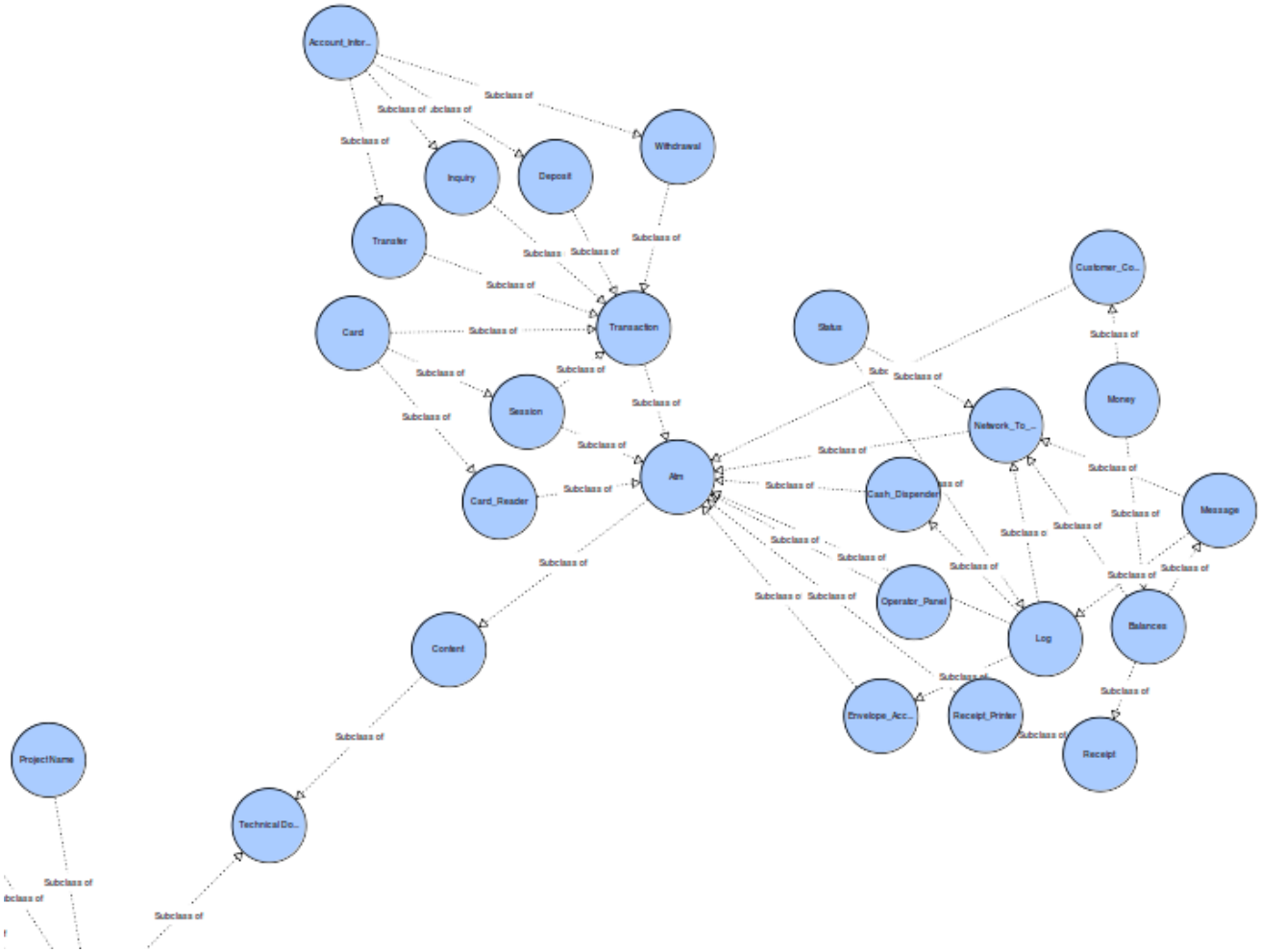
Ontology Services

insertTriple: This method inserts a new triple into an ontology.
ImportIntoOntology: This method imports triples into an ontology
exportOntology: This method exports a specific ontology
executeQuery: This method executes specific query
dropOntology: This method drops a specific ontology
removeTriple: This method removes specific triple(s)

Sample Document Ontology Model



Sample Document Ontology Model Instance

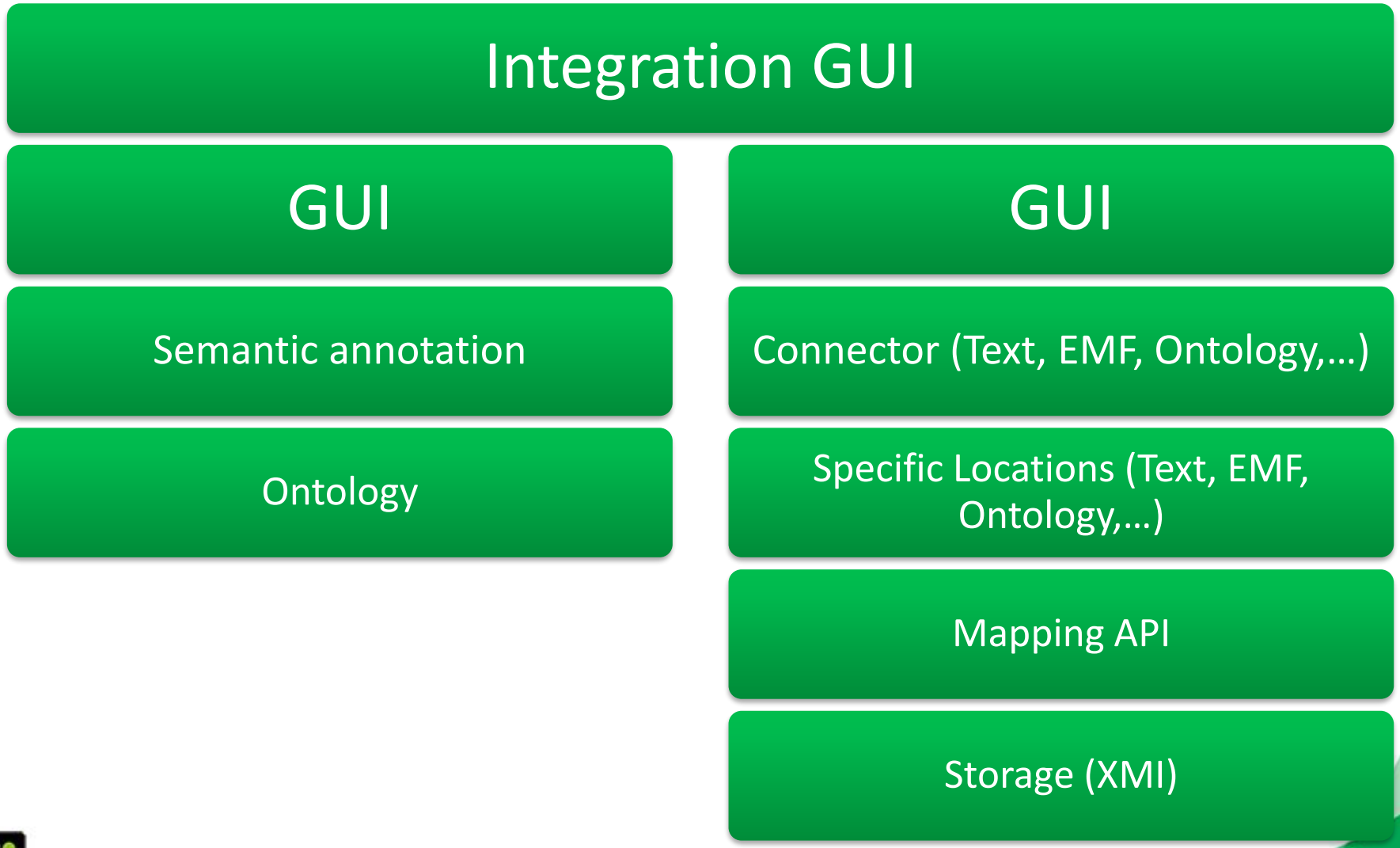


Thank you for your attention
We value your opinion and questions.

User Interfaces and Integration (WP6)

Yvan Lussaud, OBEO

Software stack



WP6
Integration

Components

- Mapping API and implementation (EMF/XMI)
- Text and model (EMF and ontology) connectors
- Semantic annotation

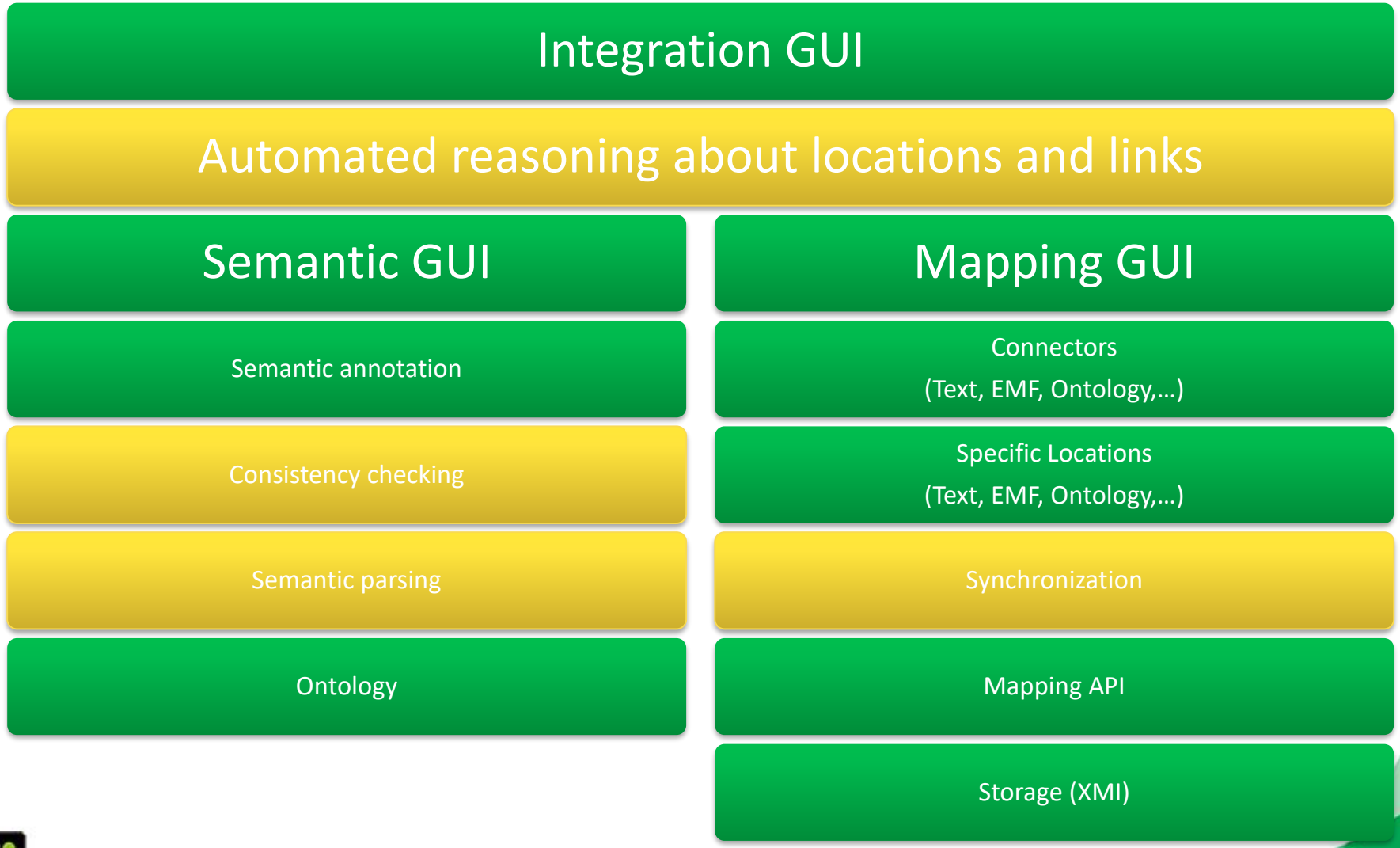
Continuous integration

- Travis https://travis-ci.org/ModelWriter/Source
- Nightly and release builds
- Eclipse update sites for each build

Next steps

- Develop synchronization API and GUI
- Integrate semantic parsing and consistency checking

Software stack



WP6
Evaluation

Code

- Release 1: 19K line of codes and 13K lines of comment
- Release 2: 30K line of codes and 20K lines of comment

Commits

- Release 1: 60 commits
- Release 2: 88 commits

Tests

- Release 1: 403 unit tests
- Release 2: 688 unit tests + some RCPTT tests

WP6
Release 2

Eclipse update sites

- Two update sites one for EPL features and one for GPL features
- http://modelwriter.obeo.fr/download
- Eclipse: Luna, Mars, Neon

Mapping features

- Text, EMF Model full support and ontology partial support
- Mapping base saved to XMI
- EMF/XMI mapping base creation wizard

Semantic features

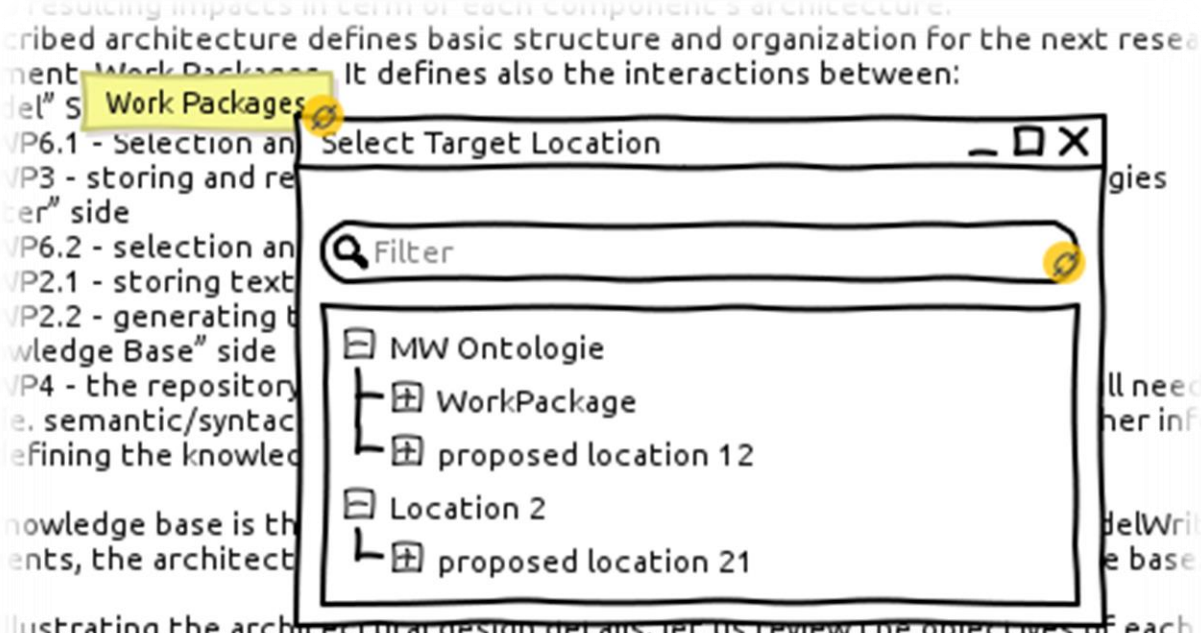
- Main ontology format supported (RDF, OWL, Turtle, …)
- Semantic annotation and mapping creation from annotation
- Text extraction from .doc and .docx files

Deliverables for WP6

Kind	Name	Status
Software	D6.3.1-2 Writer enhancements (release 2)	Done
Software	D6.4.2-2 User Interface for the Writer part (release 2)	Done
Software	D6.4.3-2 IDE-integrated User Interface to handle Sync issues (major release 2)	Done
Document	D6.5.1-2 Acceptance Test Plan (release 2)	Merged
Software	D6.5.2-2 Automated Acceptance Tests (release 2)	Done
Document	D6.6.1-2 Acceptance Test Procedures (release 2)	Merged
Software	D6.7.1-2 ModelWriter major release (release 2)	Done
Software	D6.7.2-2 WP2 Integration in ModelWriter and Documentation (release 2)	Done
Document	D6.8.1-2 Evaluation report (release 2)	Coming soon

Completion/quick outline

- + limit interactions to a minimum
- - Need to know what can be targeted (not possible for text)
- - Project possible targets out of their editing form



WP6
Ergonomics

Goals

- No projection / get elements from the editor
- Works with text
- Works with graphics

Palette

- Eclipse view
- Add element from editors to the palette
- Link to elements in the palette

Possible improvements

- Quick outline with elements of the palette ?
- Use current selection and an action on element from the palette ?

5 Demonstrations

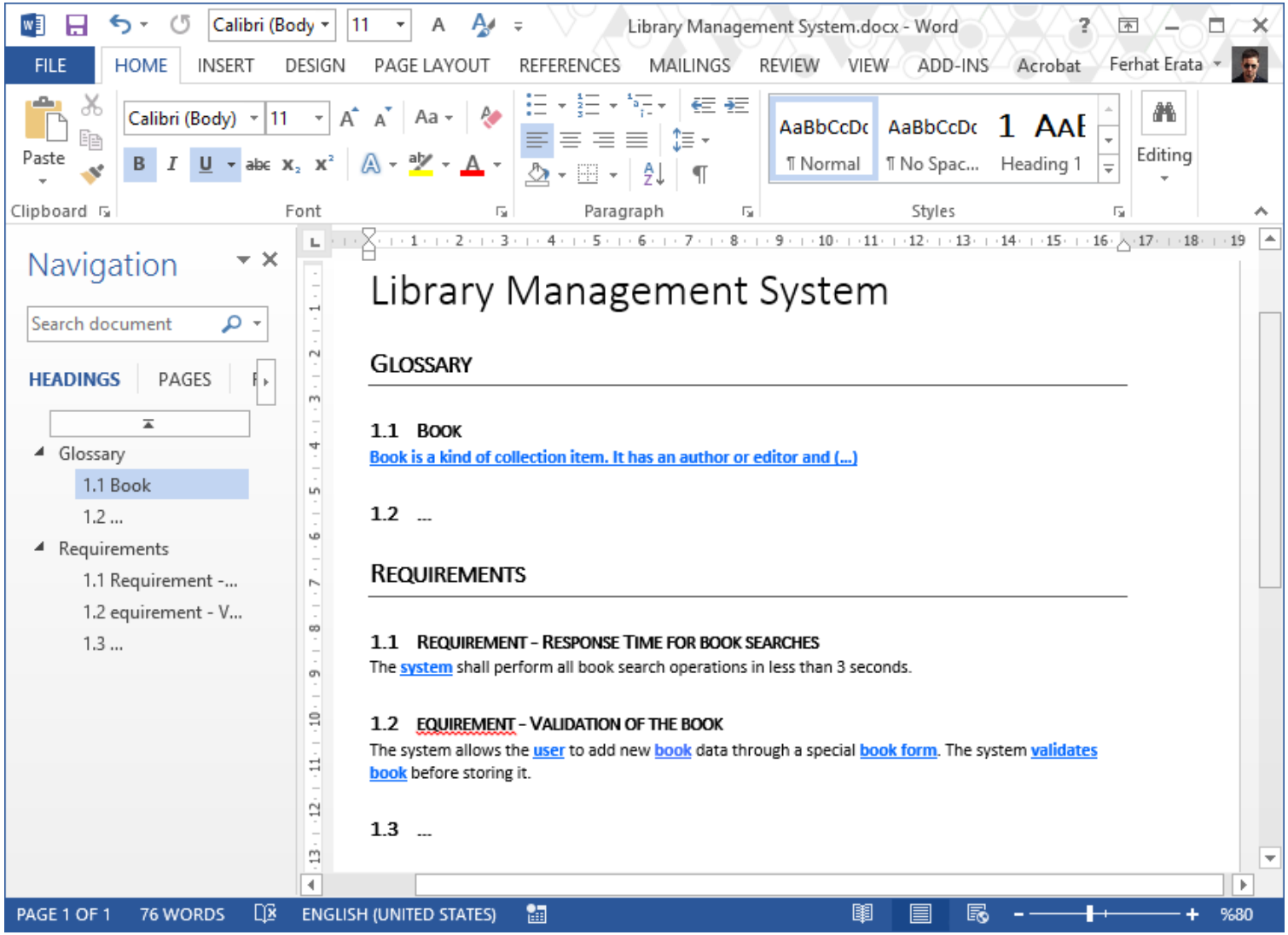
Ferhat Erata (UNIT, WP3 Leader)
Emre Kirmizi (UNIT, Technical Contact)
Dr. Claire Gardent (CNRS/LORIA, WP2 Leader)
Bikash Gyawali (CNRS/LORIA, Technical Contact)
Anastasia Shimorina (CNRS/LORIA, Technical Contact)
Dr. Geylani Kardas (KOCSISTEM, Consultant)

ITEA 3

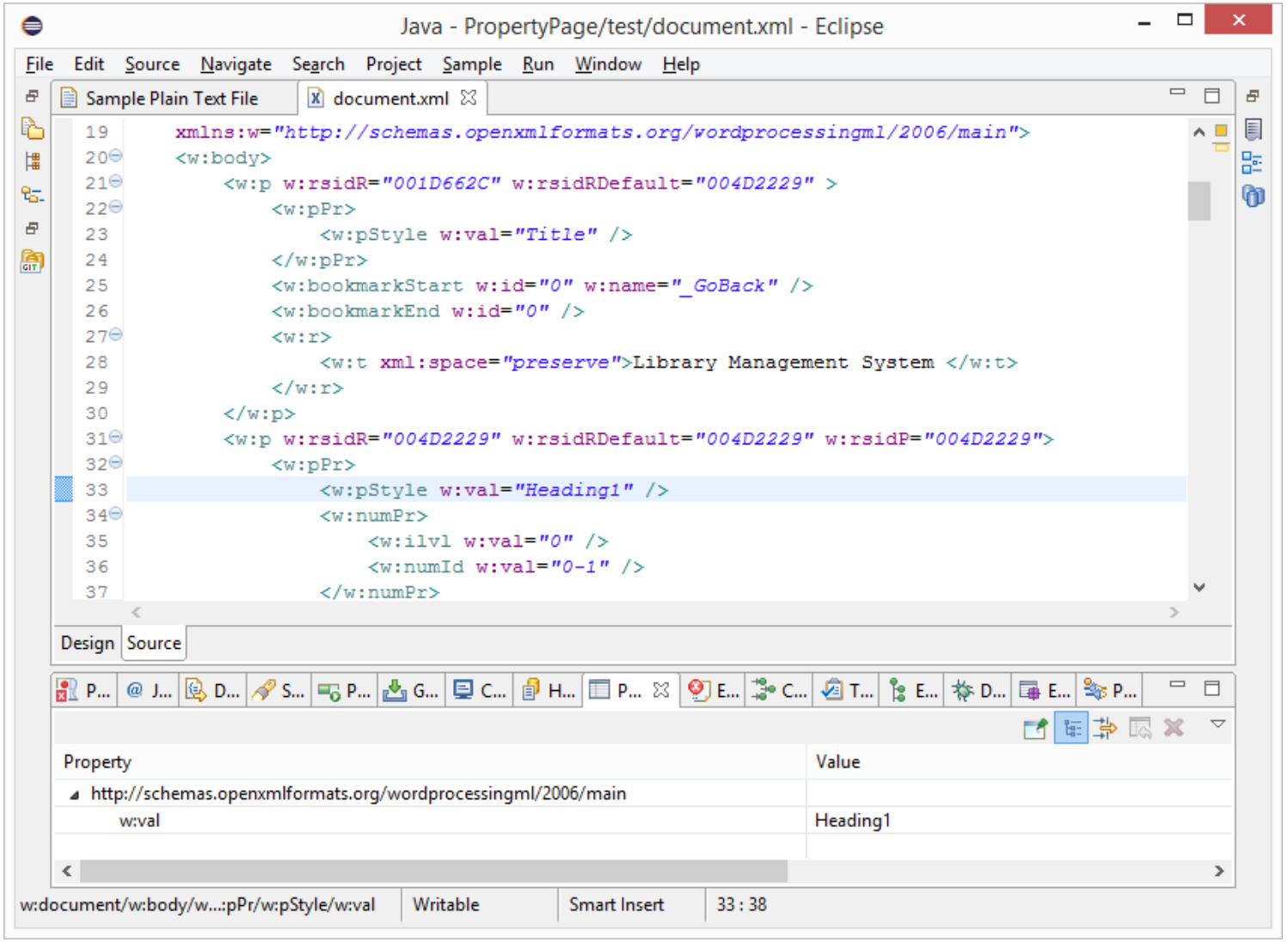
What is a text?

What is a text? (document file formats)

Office Open XML (.docx) (ISO/IEC 29500)

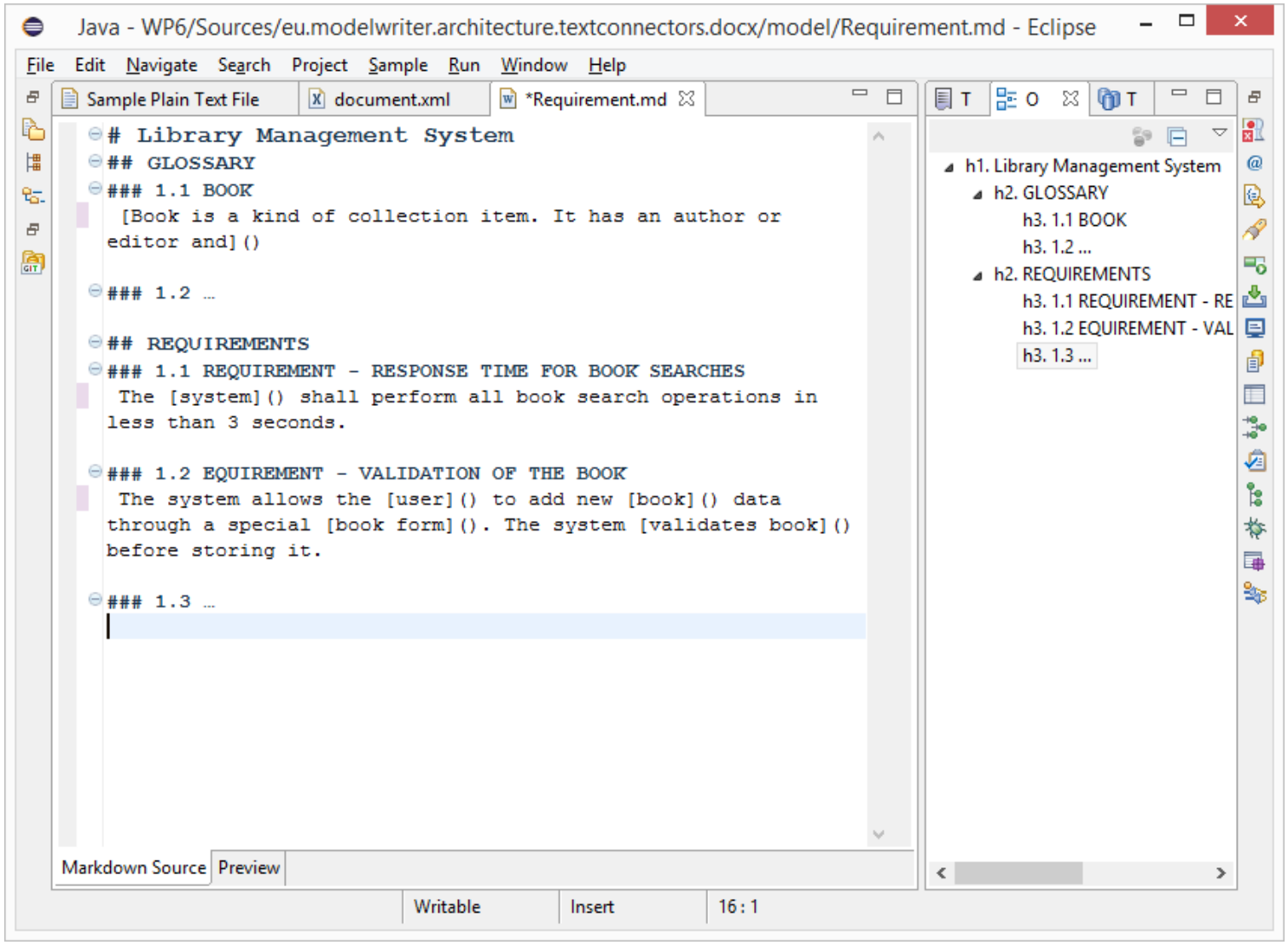


What is a text? (document file formats)
Office Open XML (.docx) (ISO/IEC 29500)

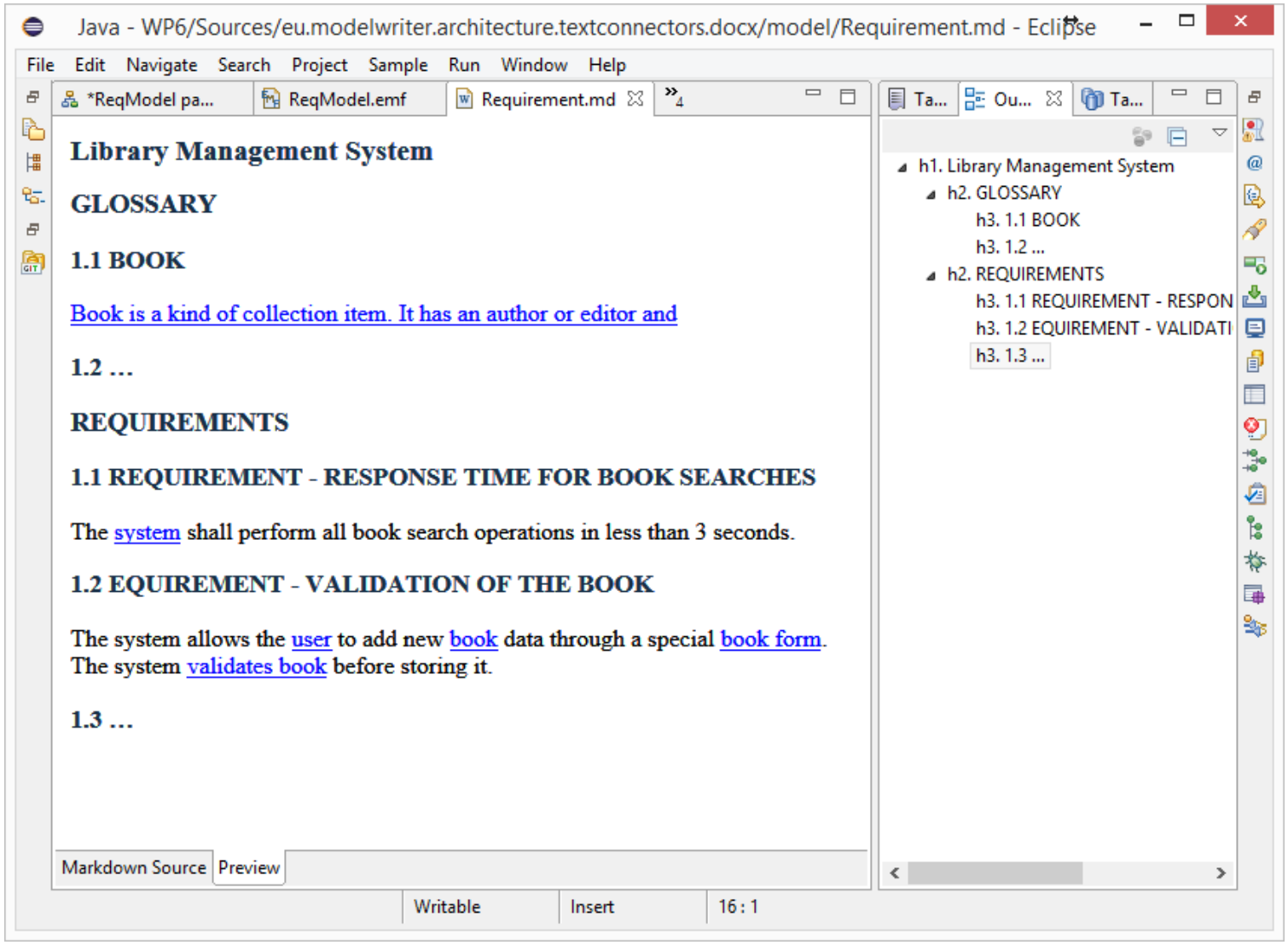


What is a text? (.md source file)

text/markdown (ICANN Standard)

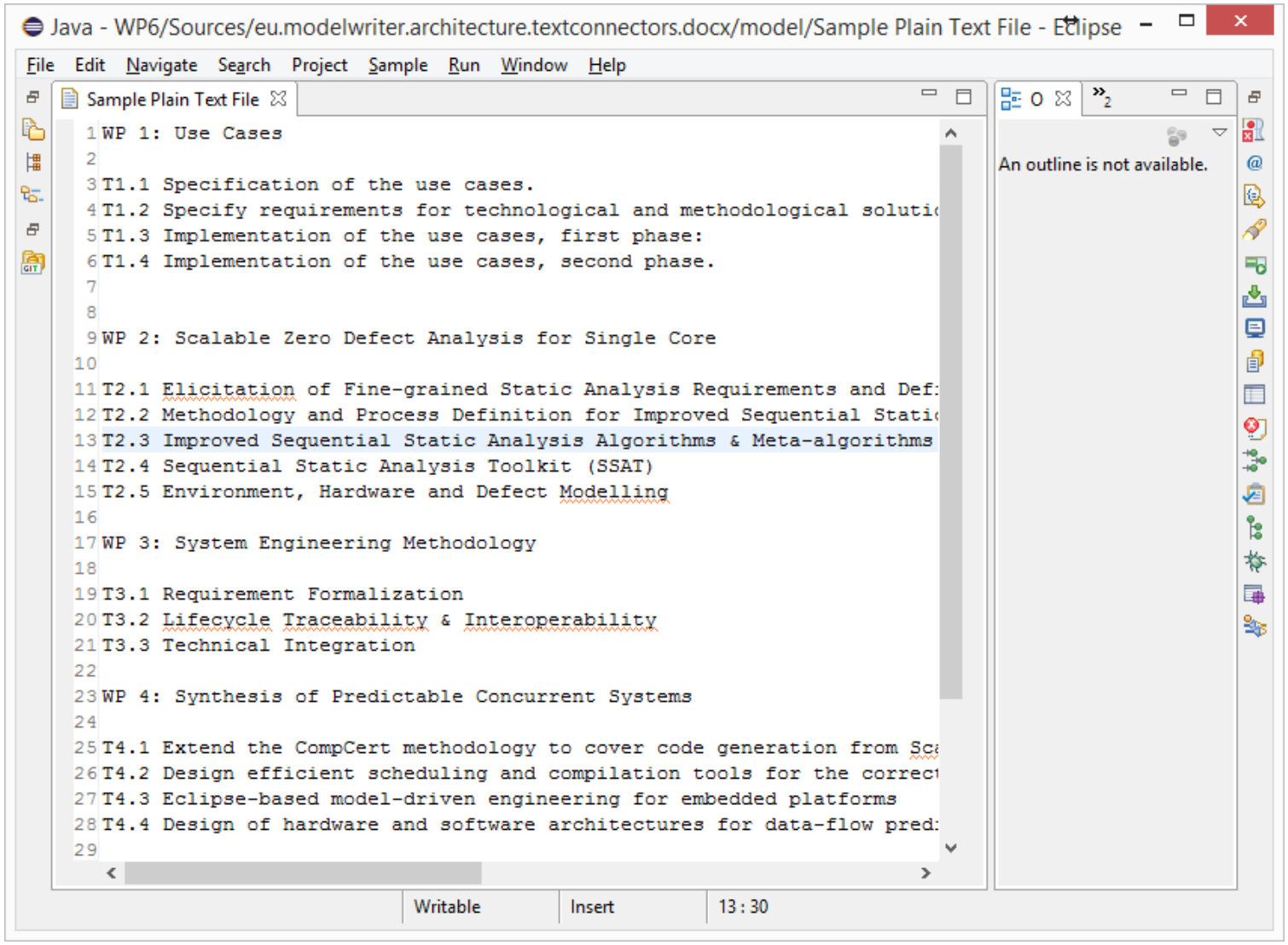


What is a text? (HTML Preview)
text/markdown (ICANN Standard)



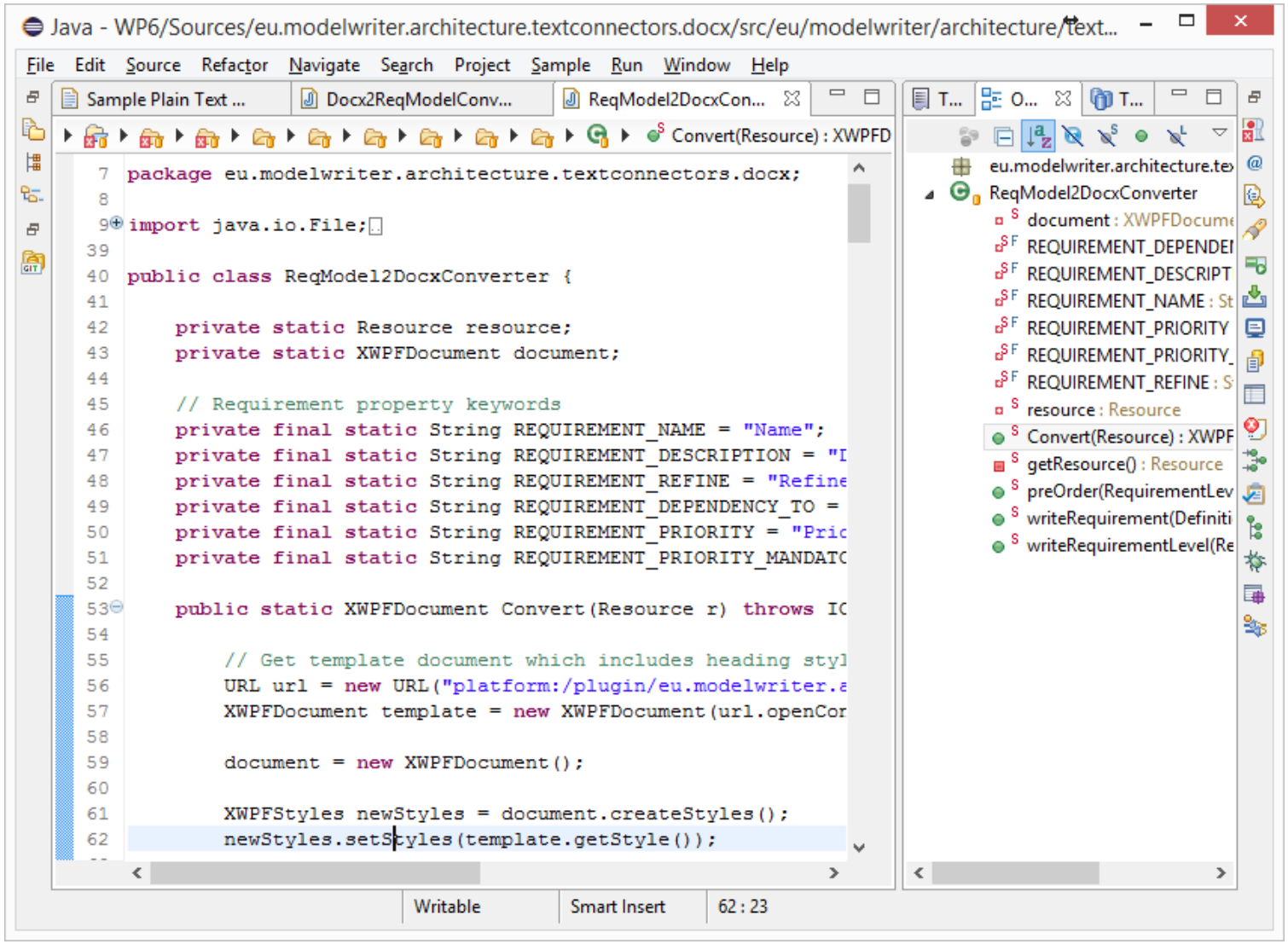
What is a text? (unformatted text)

text/plain (ICANN Standard)

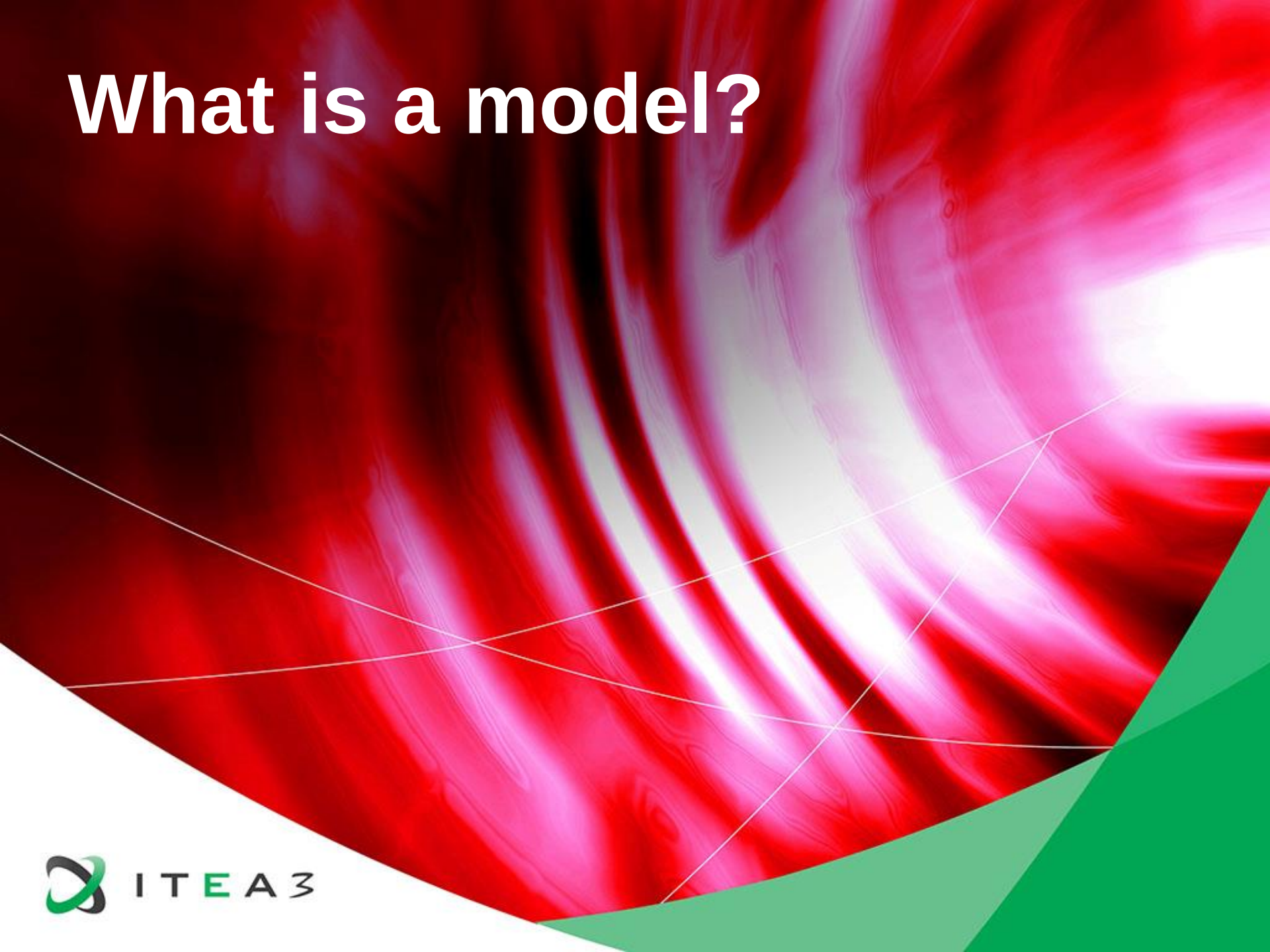


What is a text? (code files)

Java, C++ … Programing Languages

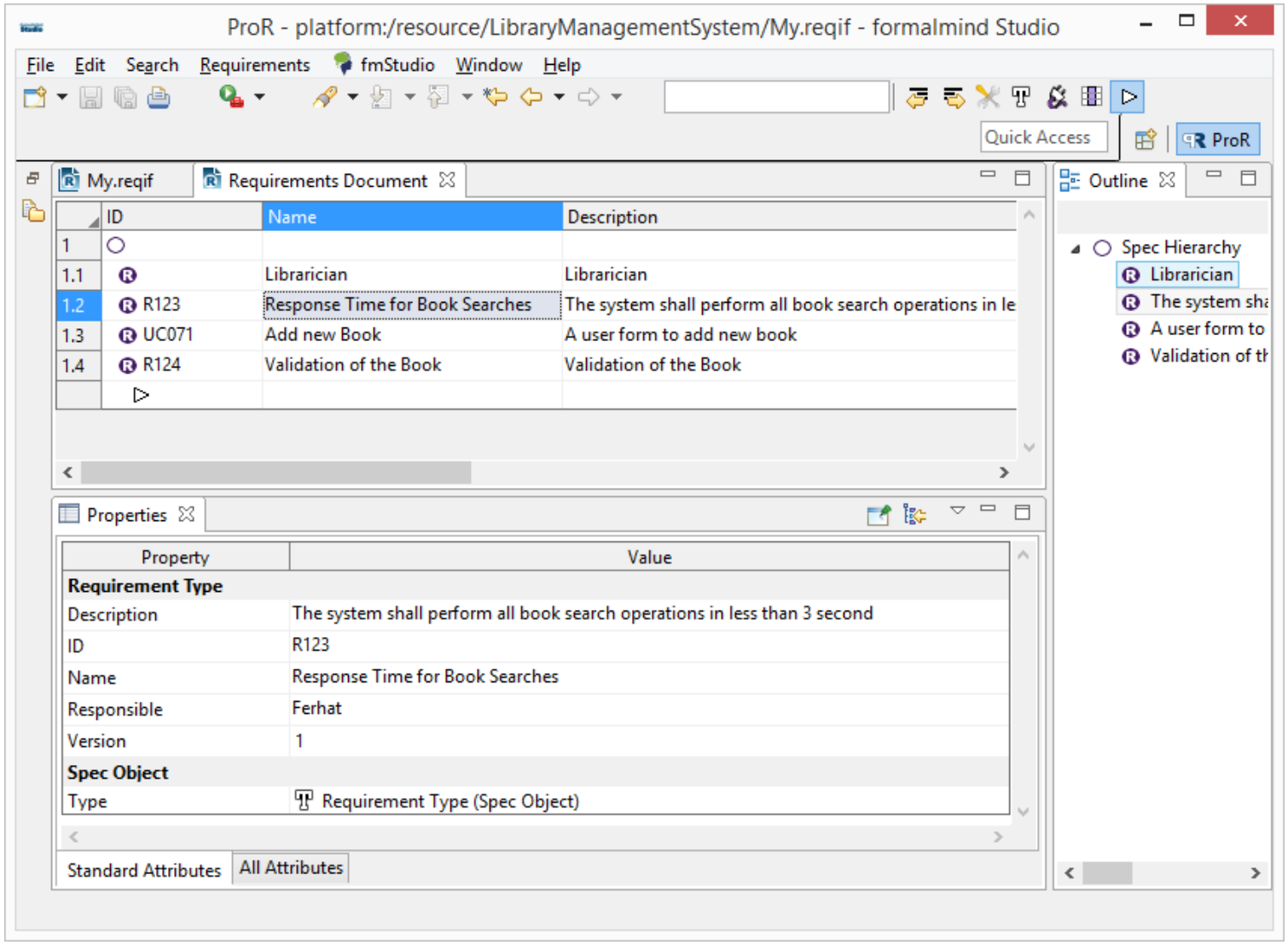


What is a model?



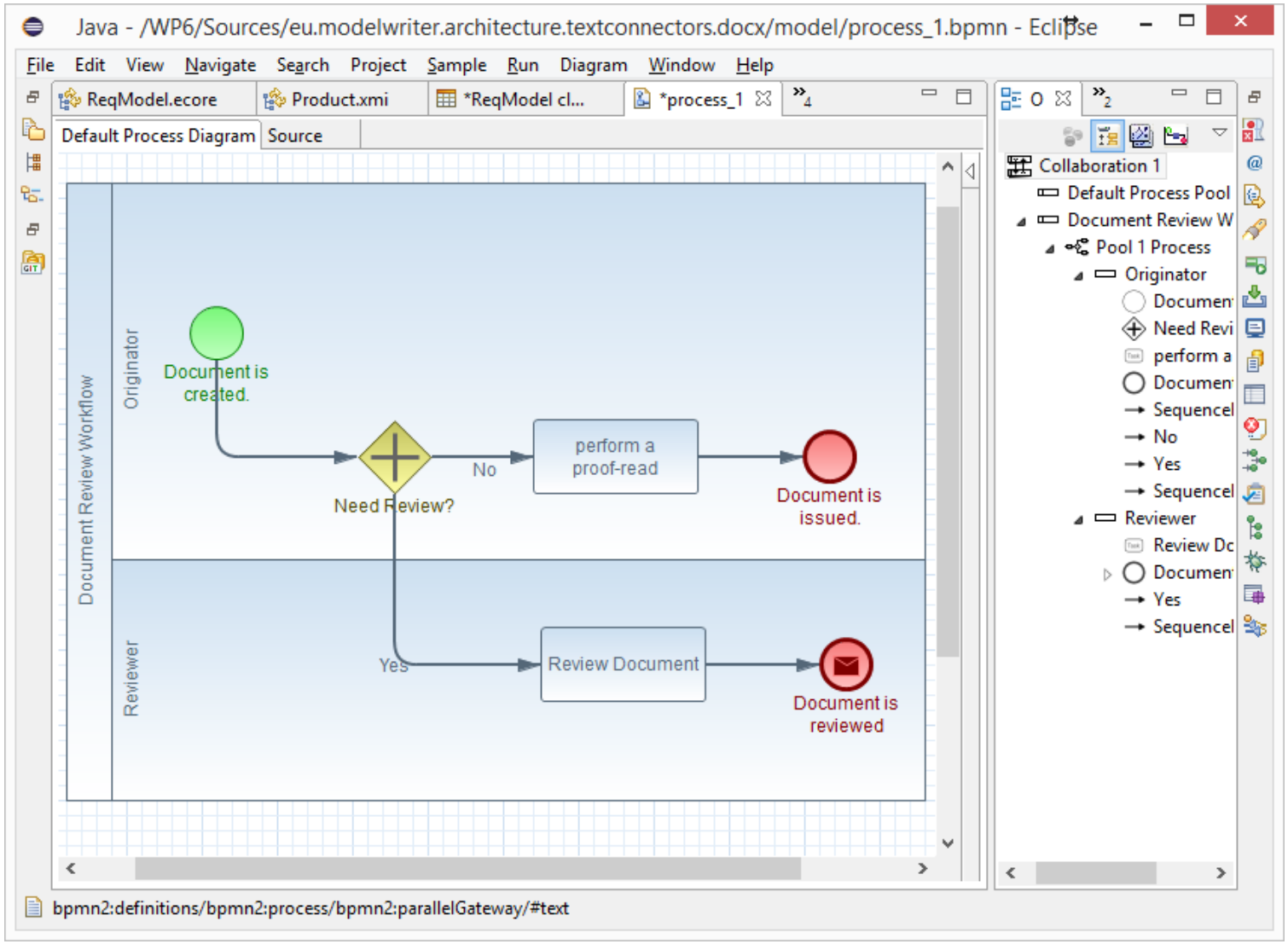
Everything is a model! (ReqIF Standard)

Requirements Interchange Format



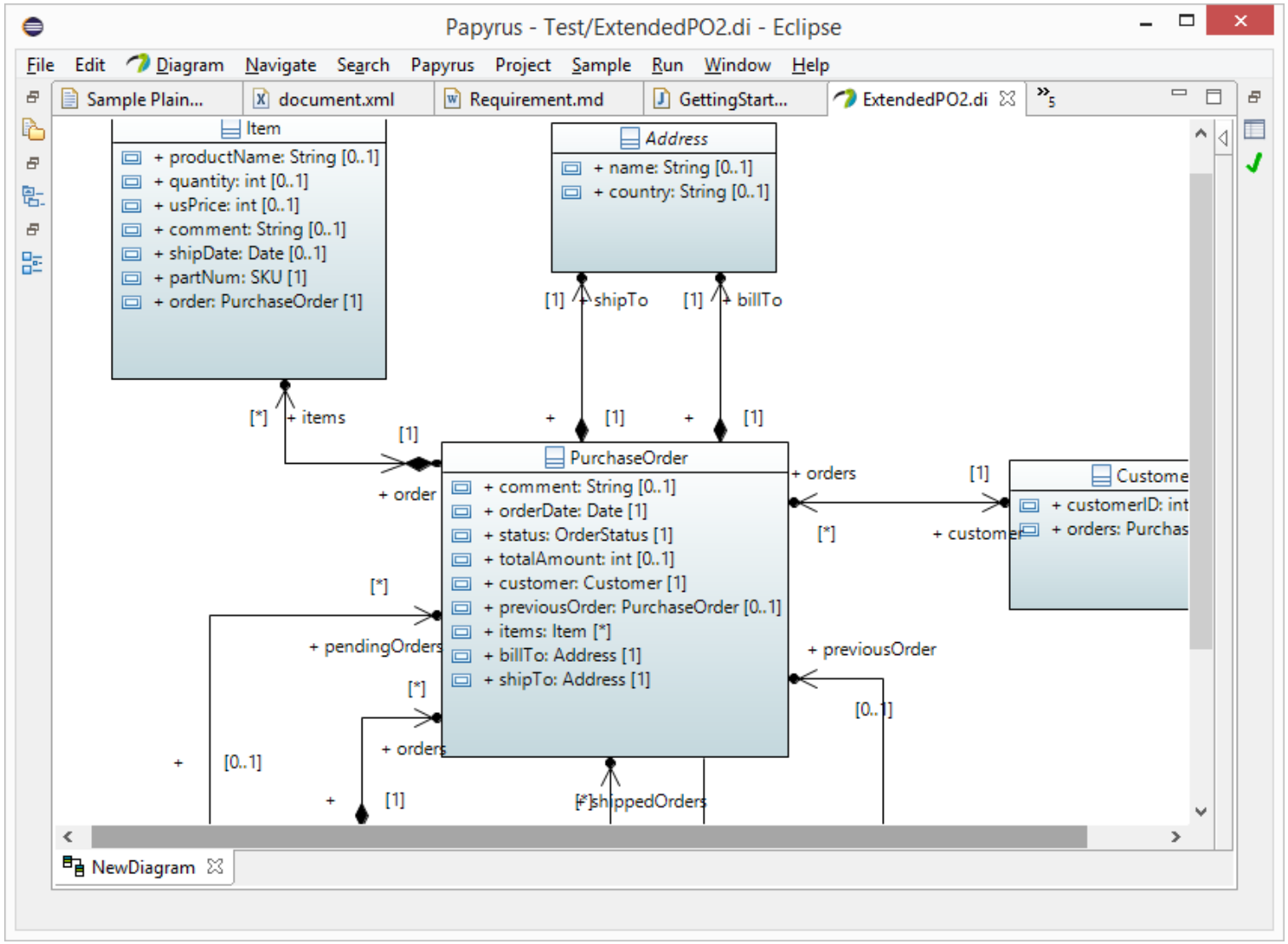
Everything is a model! (BPMN Standard)

Business Process Model & Notation

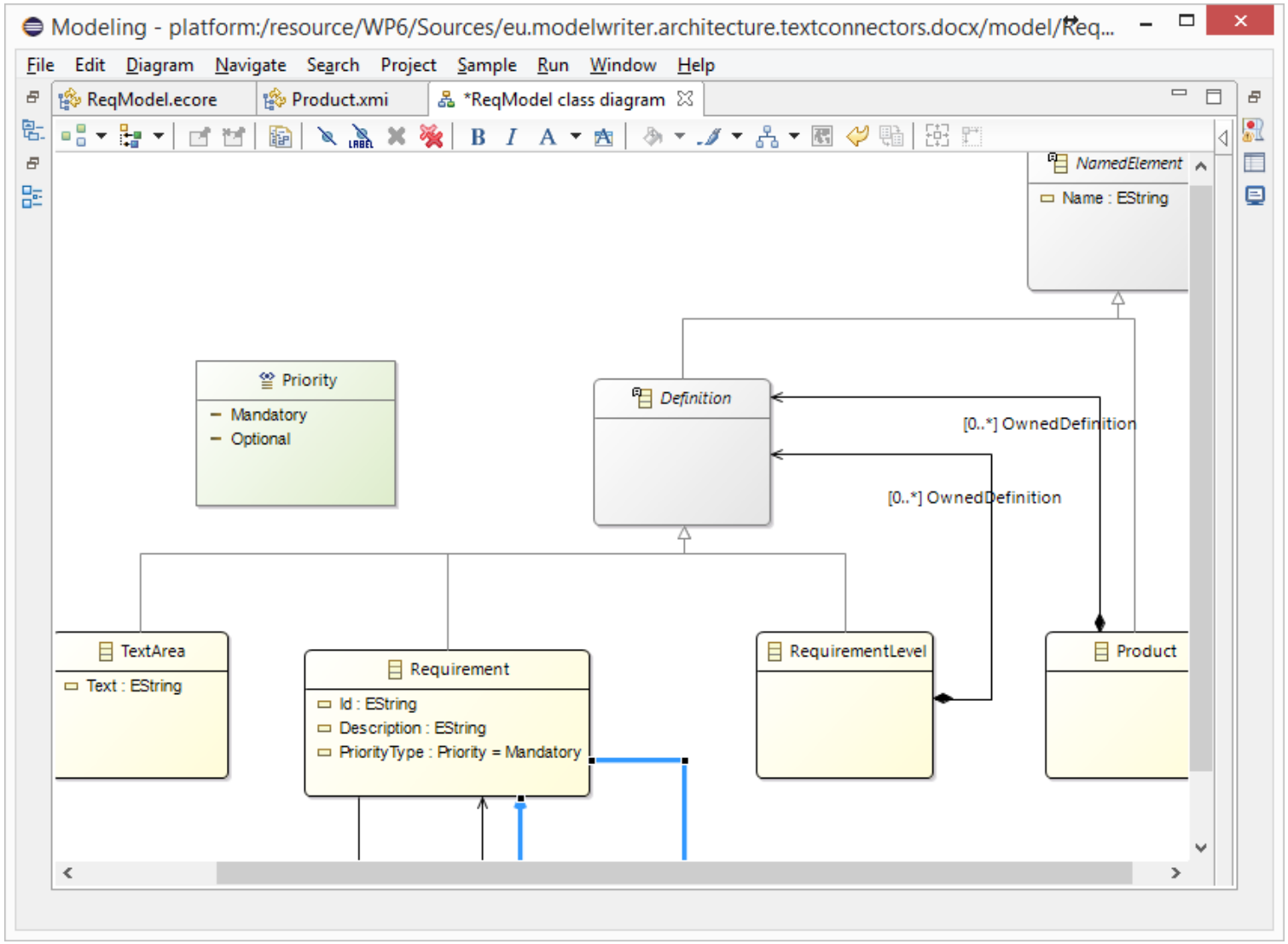


Everything is a model! (UML Standard)

UML Modeling Languages

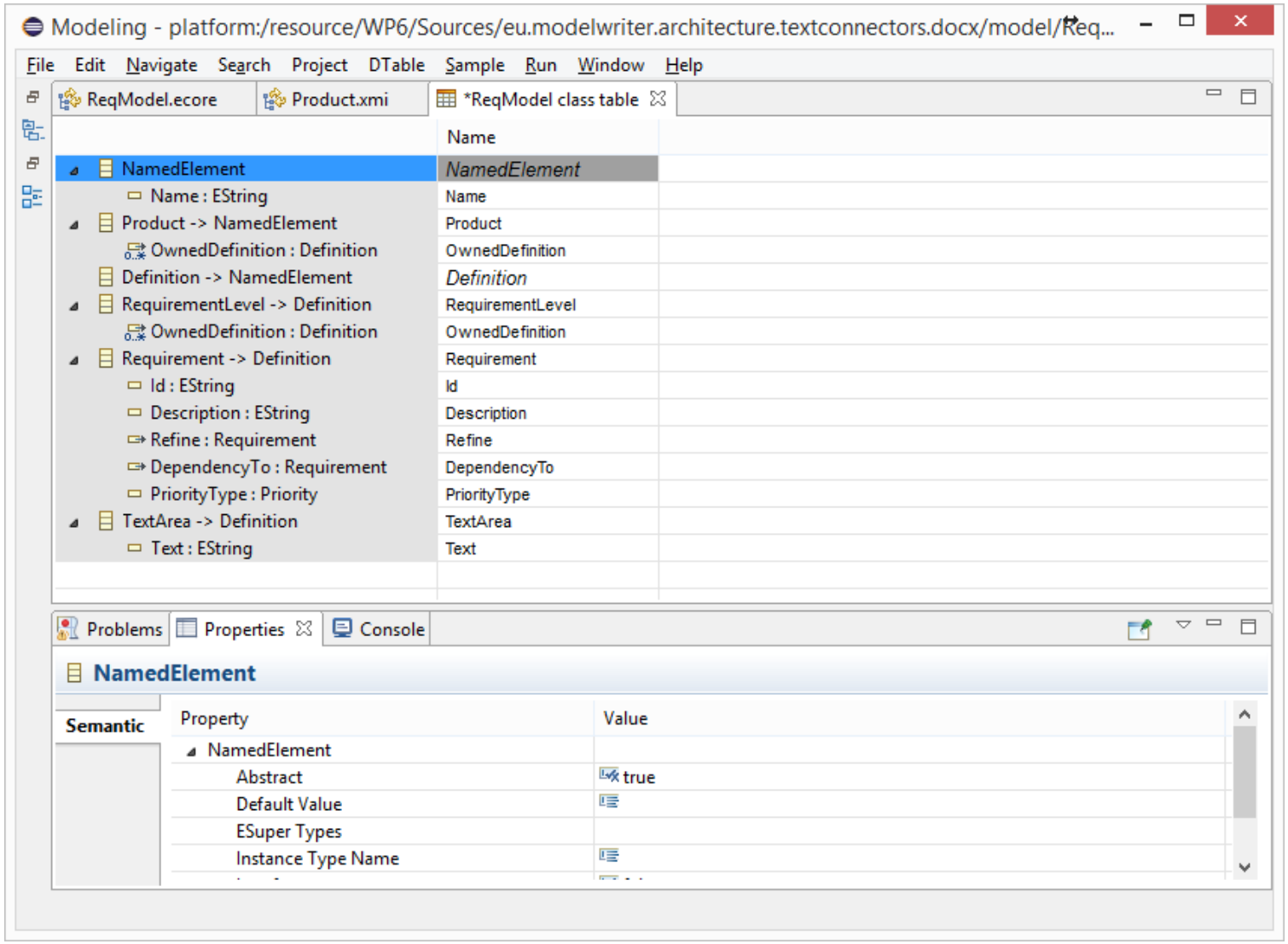


Everything is a model!
Eclipse Modeling Framework (EMF)

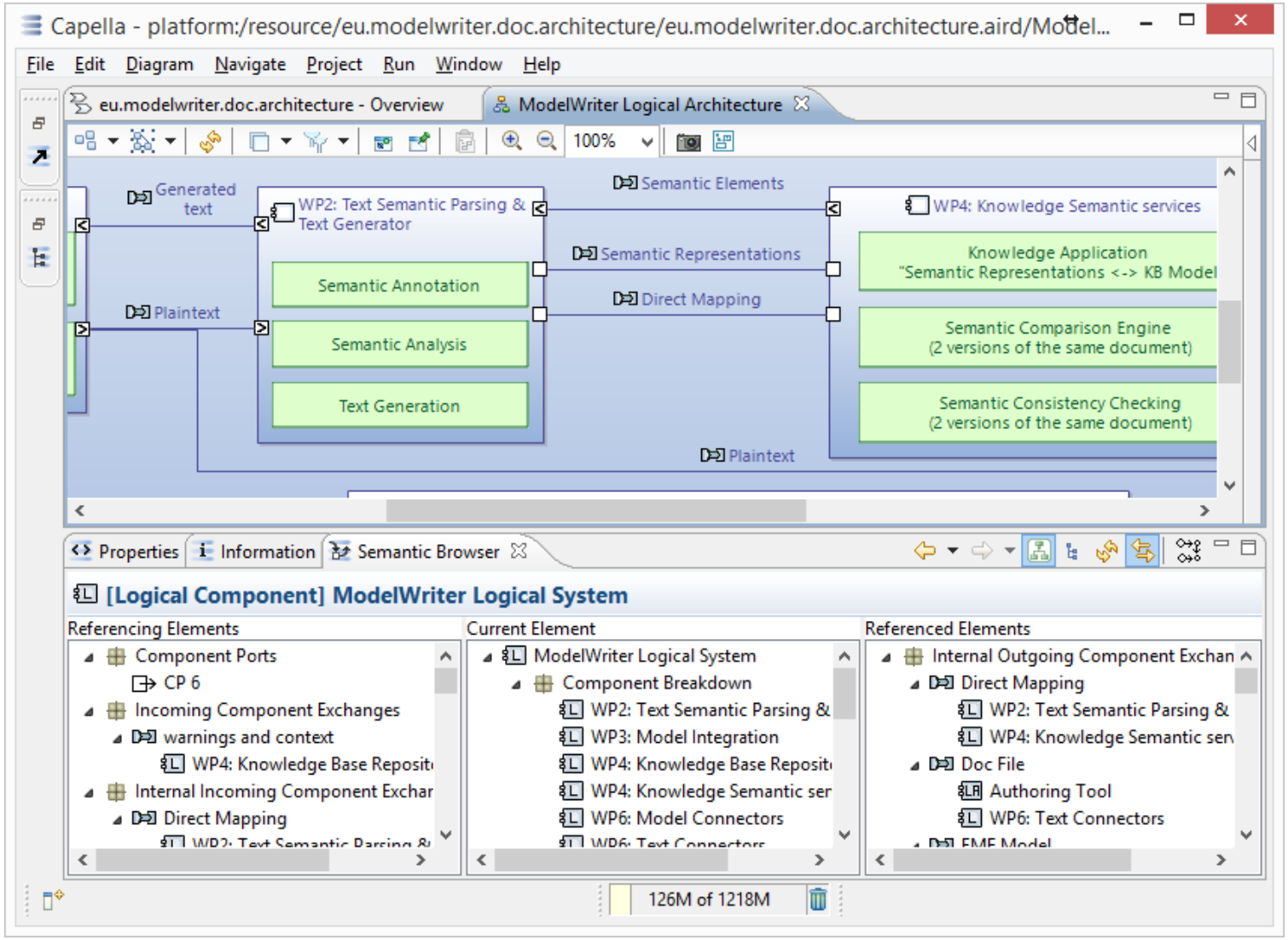


Everything is a model!

Tree-based or Tabular Representations

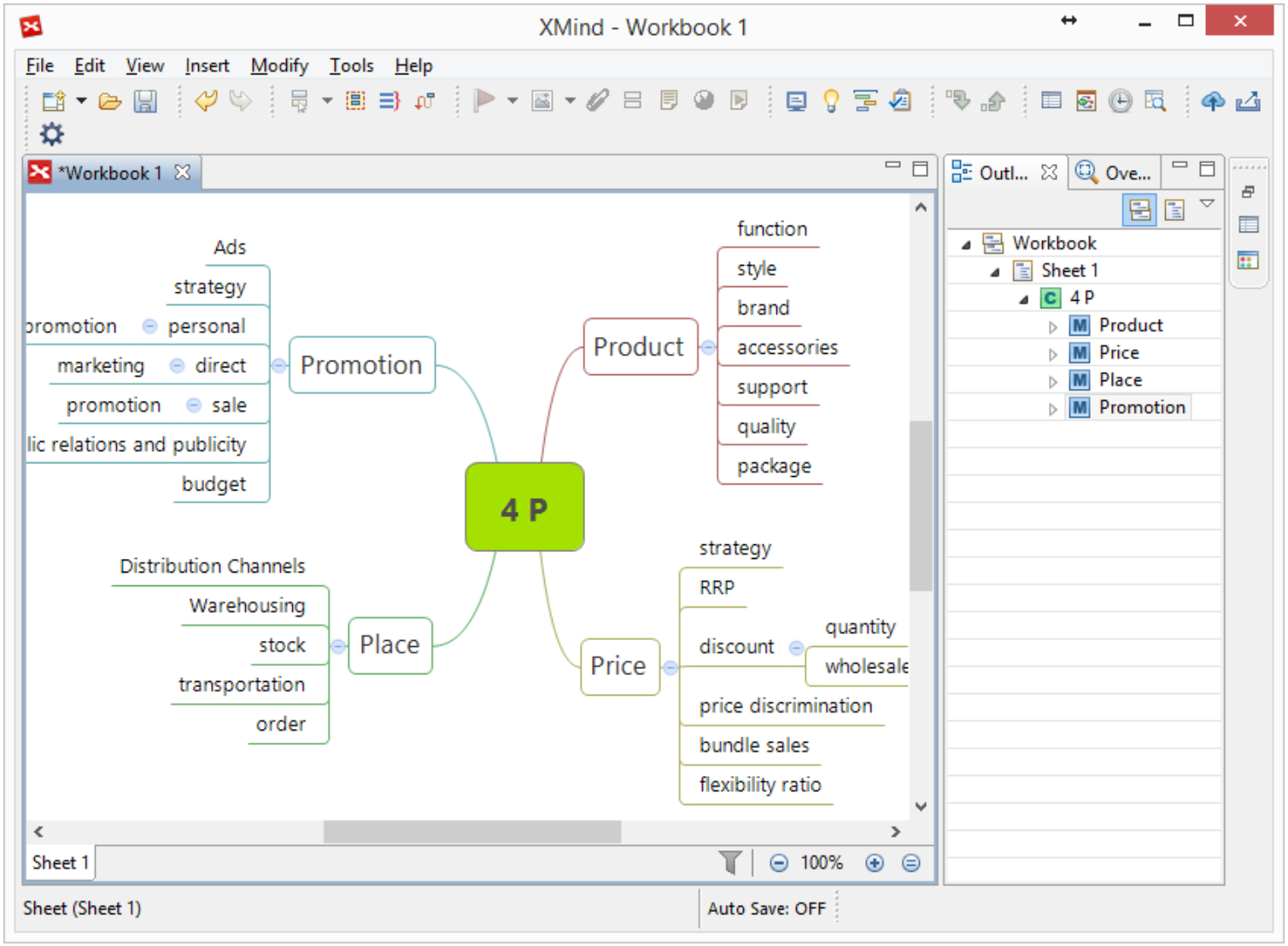


Everything is a model!
Software/System Architecture Design

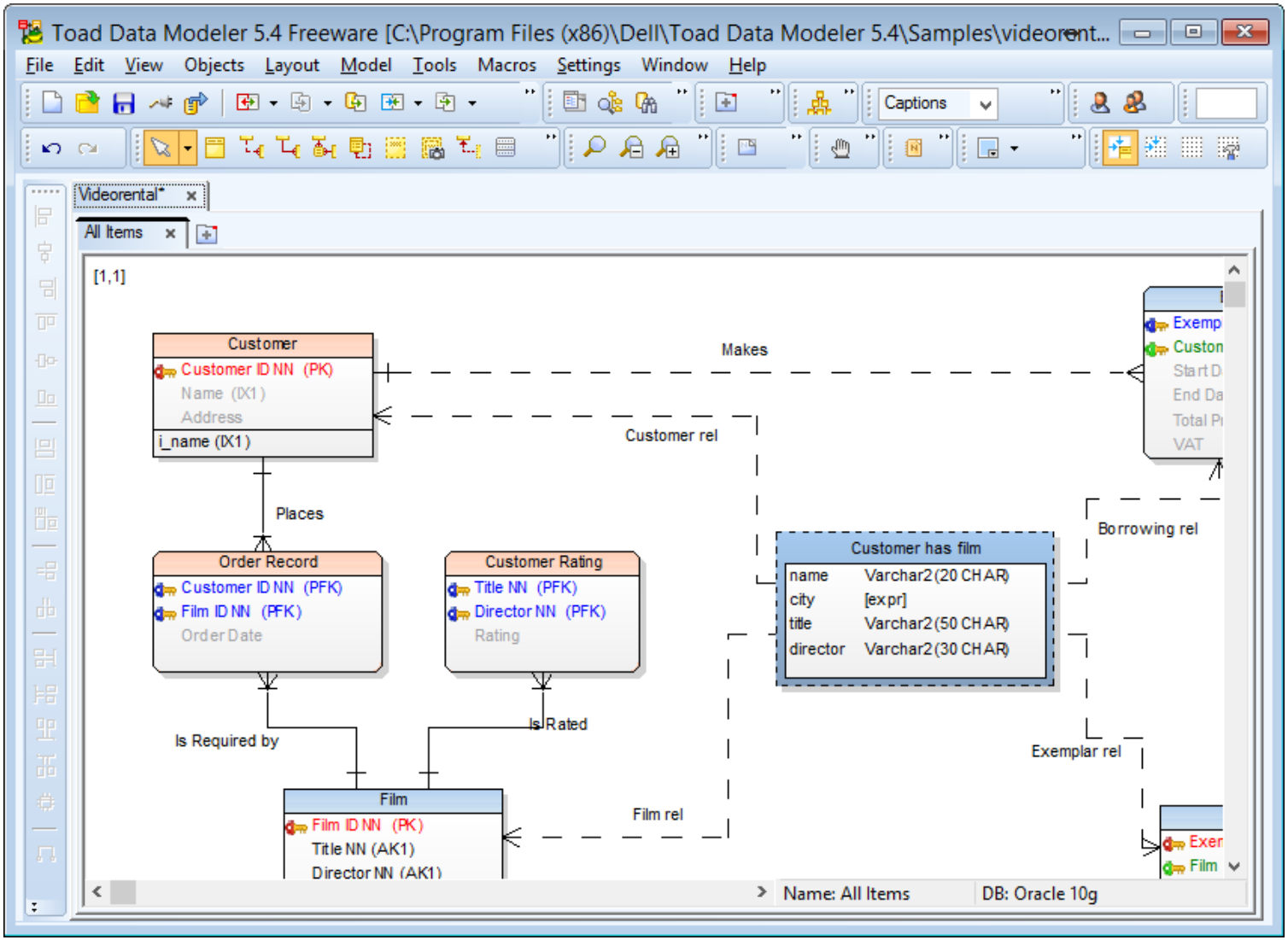


Everything is a model!

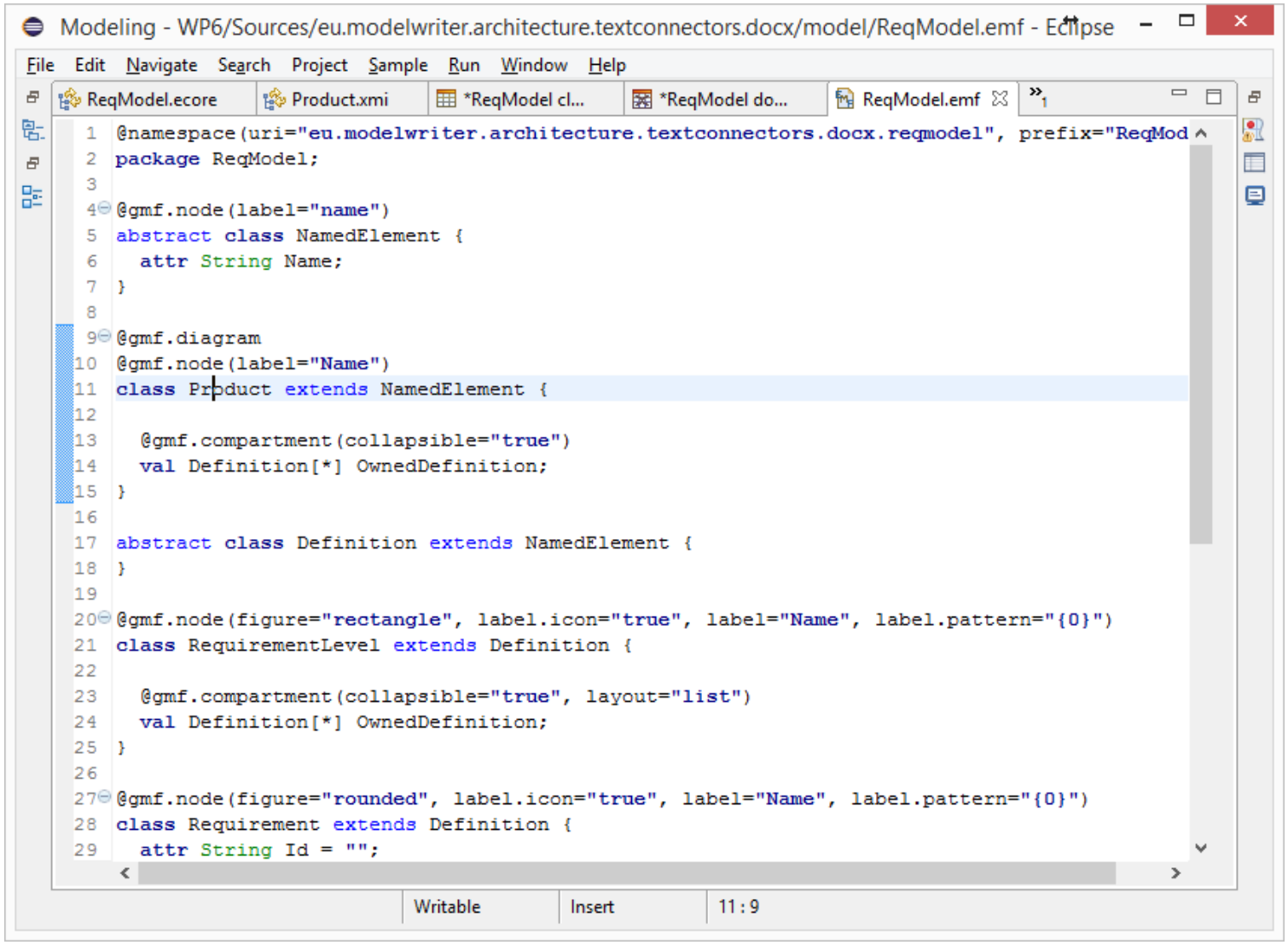
Topic Maps, Mind Maps, Vocabularies …



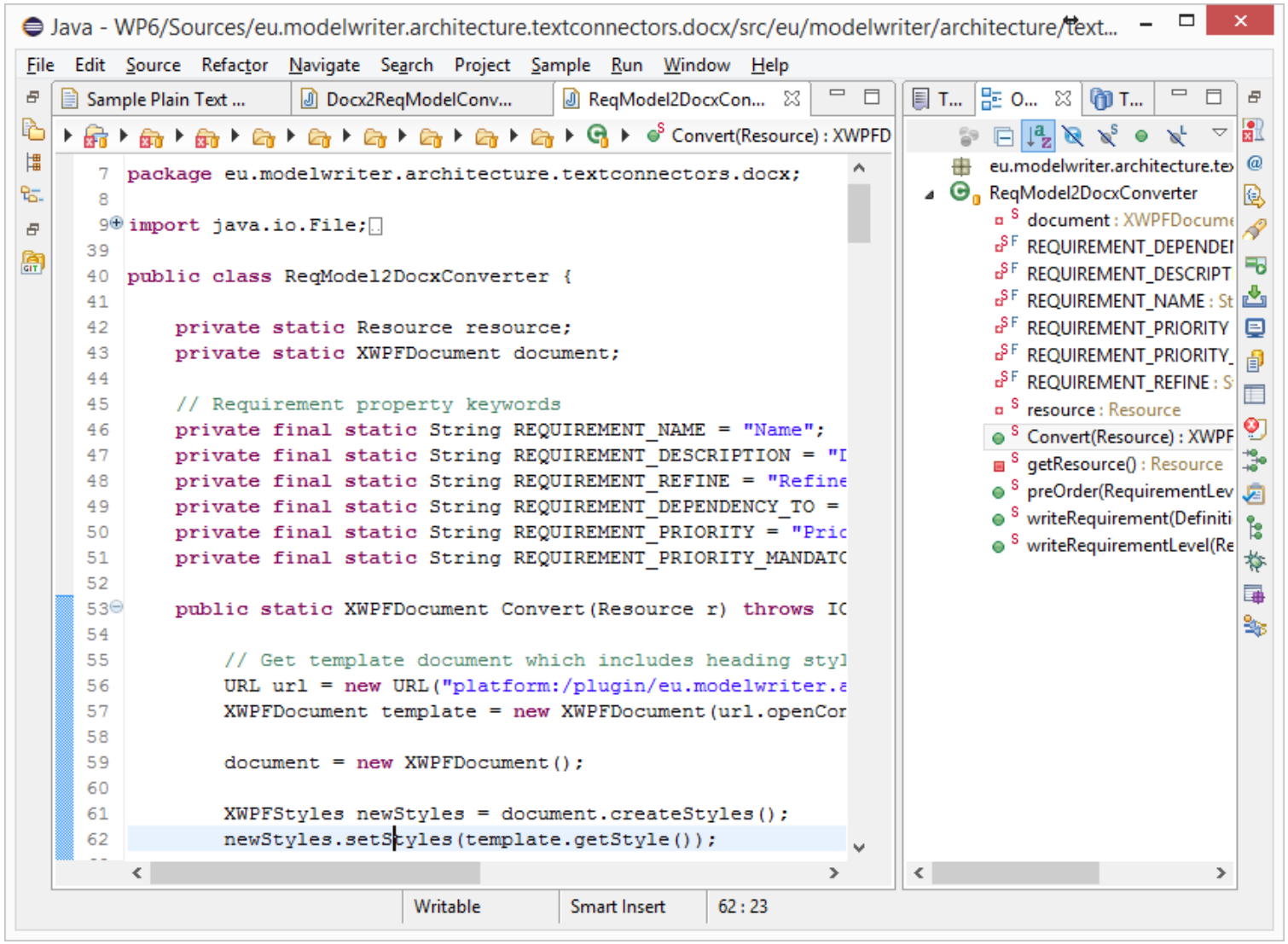
Everything is a model!
Databases (ER, IDEF1.x)



Everything is a model! (Textual Lang.)
Domain Specific Languages



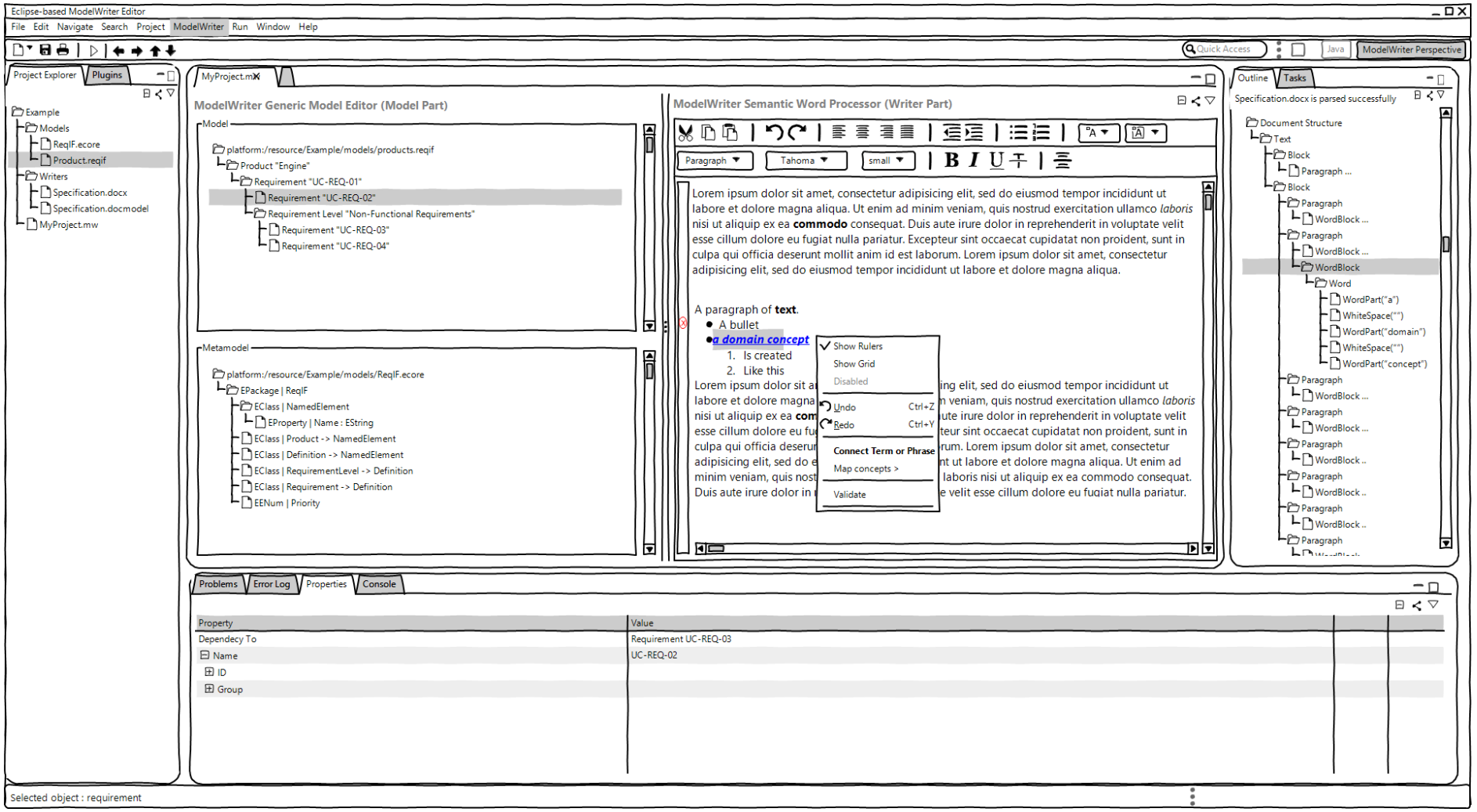
Everything is a model! (Java, C++, etc.)
Even Programing Languages (ASTs)



Is it possible to connect and keep arbitrary software/system engineering artifacts synchronized ?

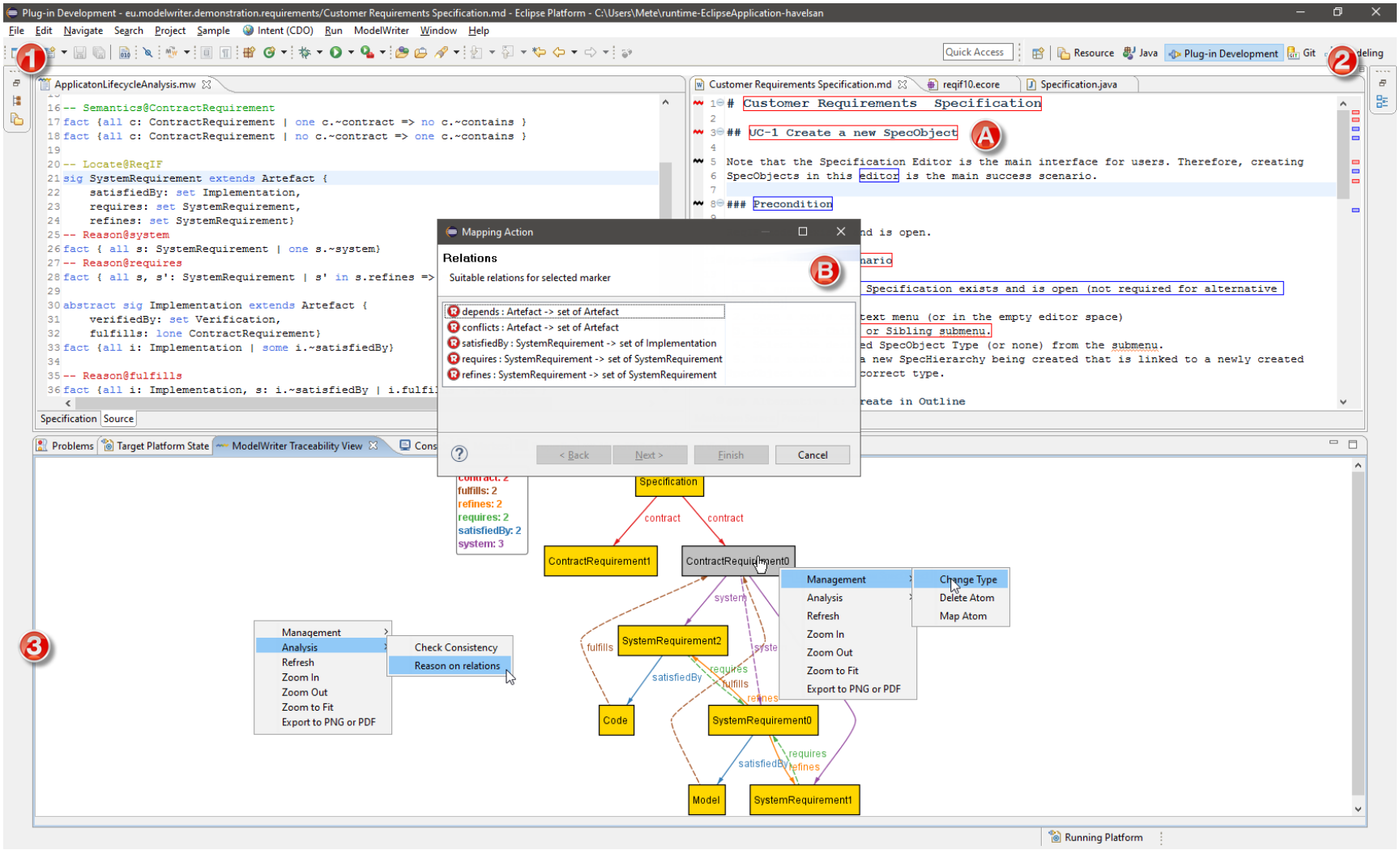
ITEA 3

ModelWriter – The Solution



Text & Model-Synchronized Document Engineering Platform

Solution – Knowledge Capture

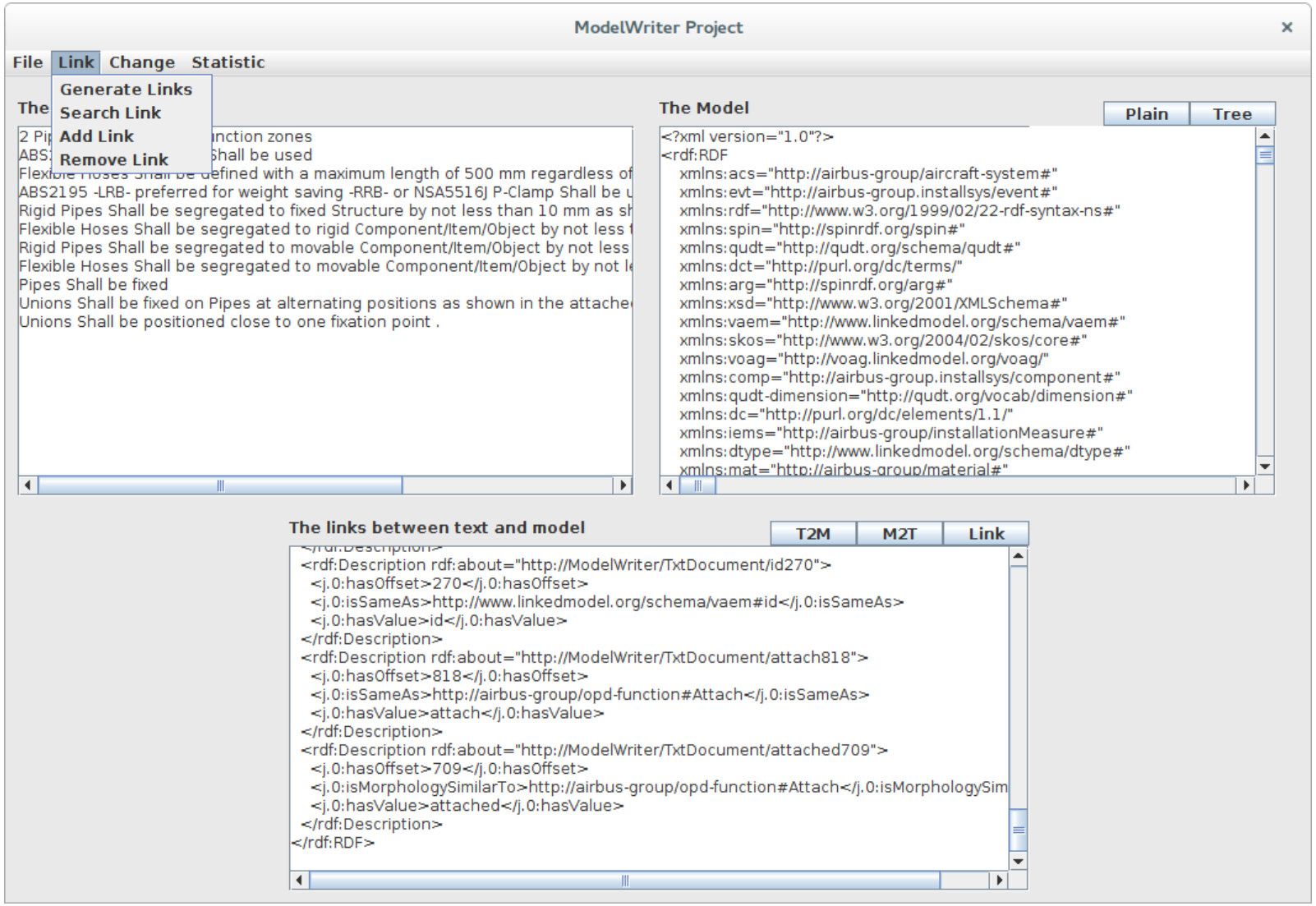


Text & Model-Synchronized Document Engineering Platform

Is it possible to extract knowledge from texts fragments based on a given ontology (model) ?

ITEA 3

Solution – Knowledge Extraction



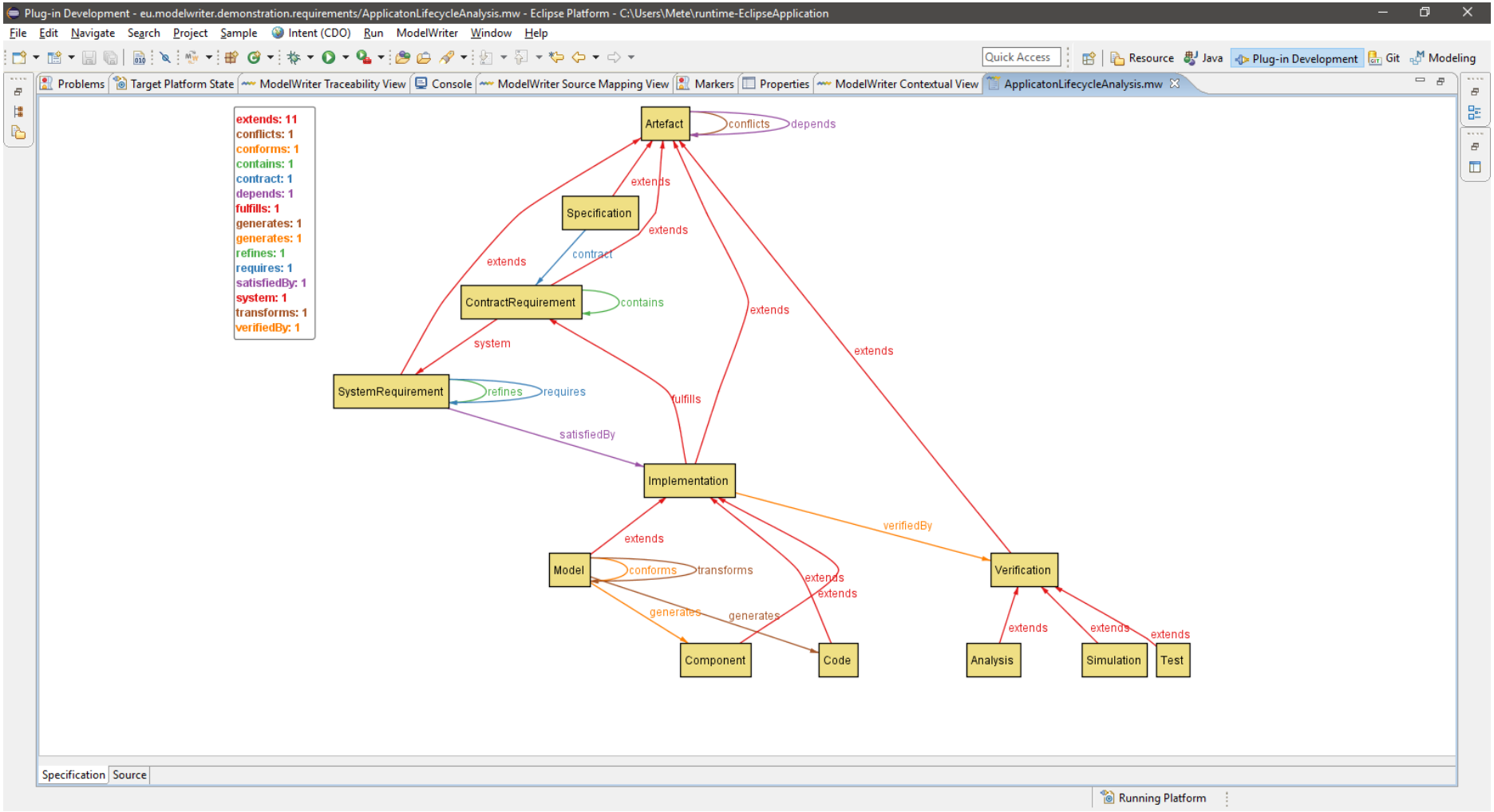
Text & Model-Synchronized Document Engineering Platform


Synchronization is maintained!

What about configuration/formalization of the platform?

ITEA 3

Configuration: Havelsan example

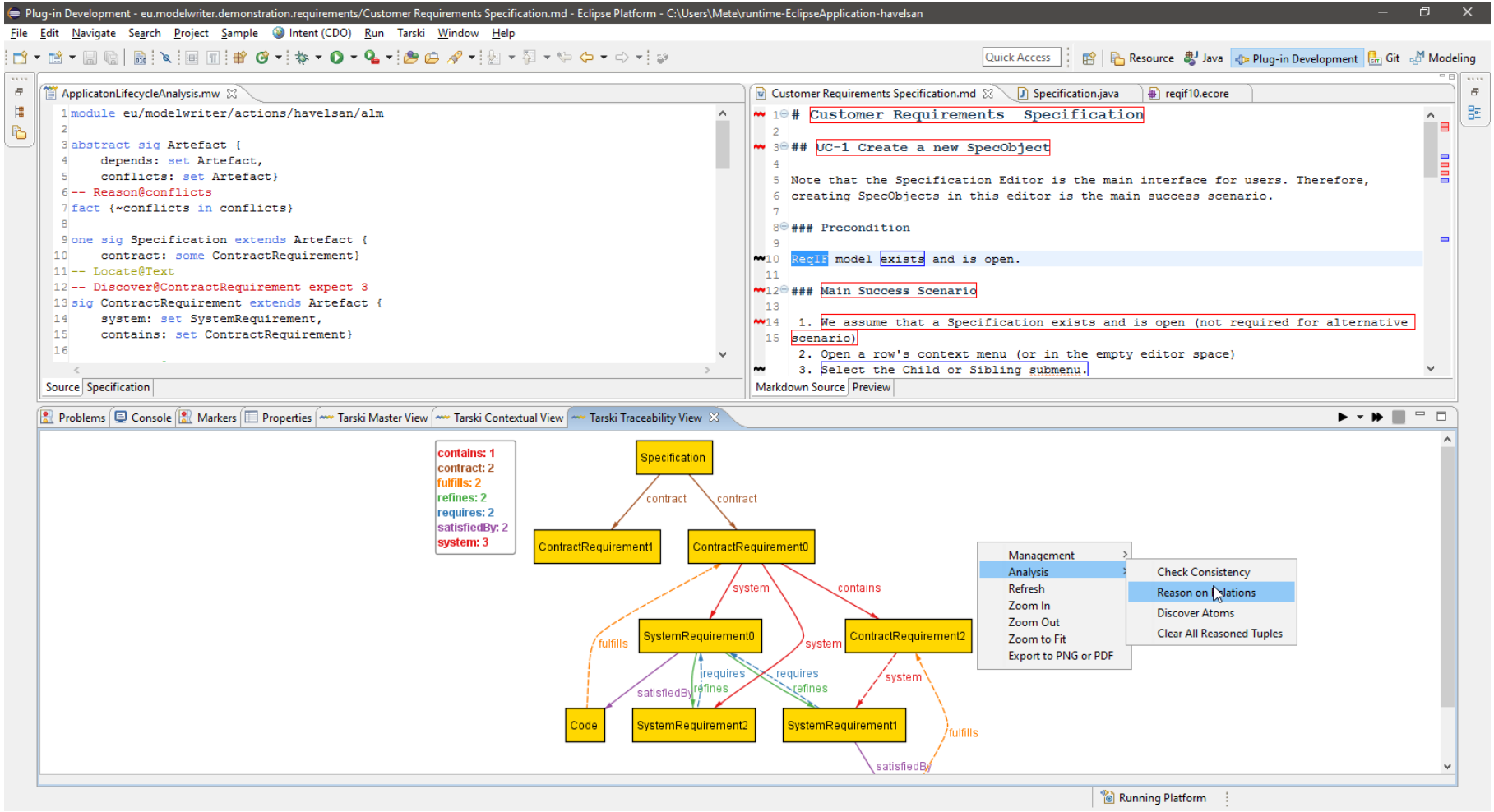


A Formal Specification Model to configure the ModelWriter

Is it possible to vizualize the trace links?

ITEA 3

Traceability: Havelsan example



A Formal Specification Model to configure the ModelWriter

Airbus Use Cases

Ford-Otosan Use Case

Havelsan Use Case