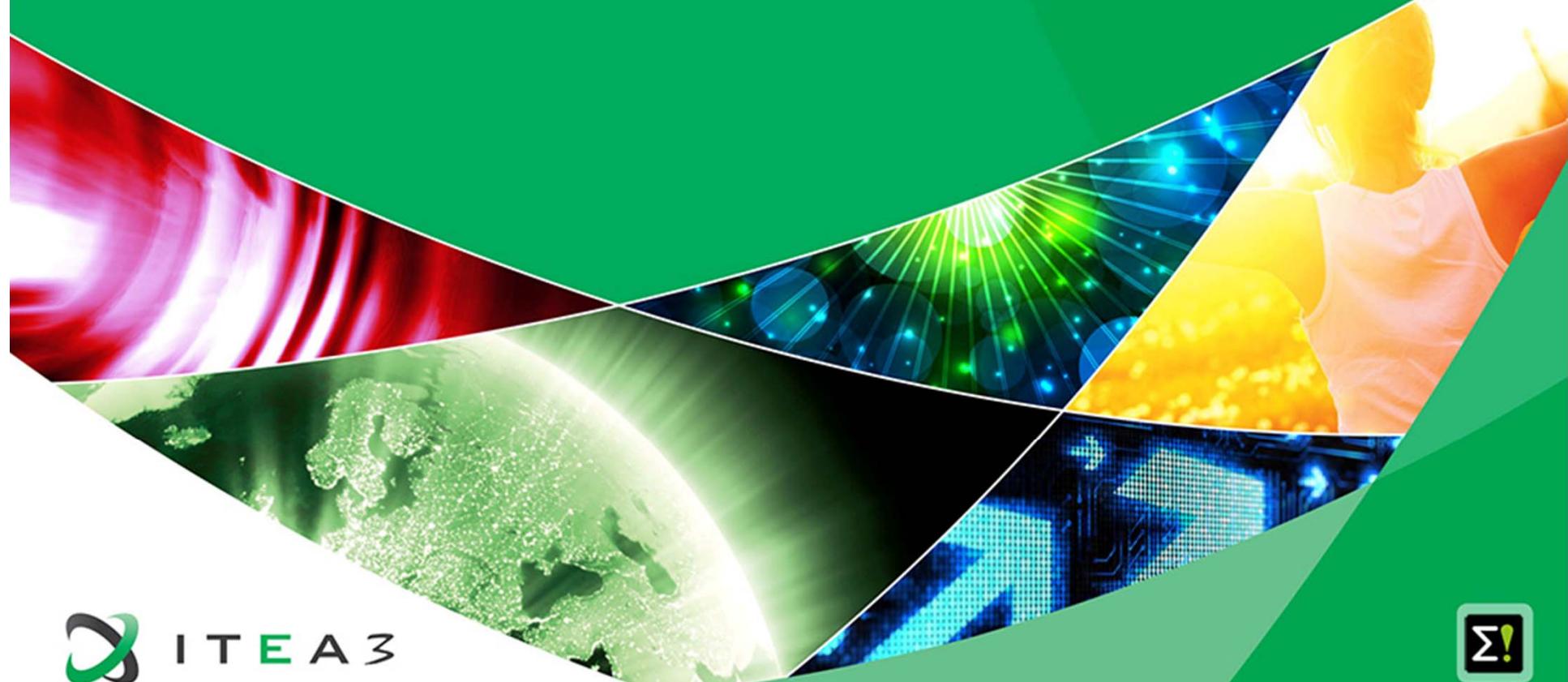


Model Writer

Progress on the Industrial Use Cases



UC-FR-01 - Synchronization between models and documents

*Yvan lussaud
OBEO*



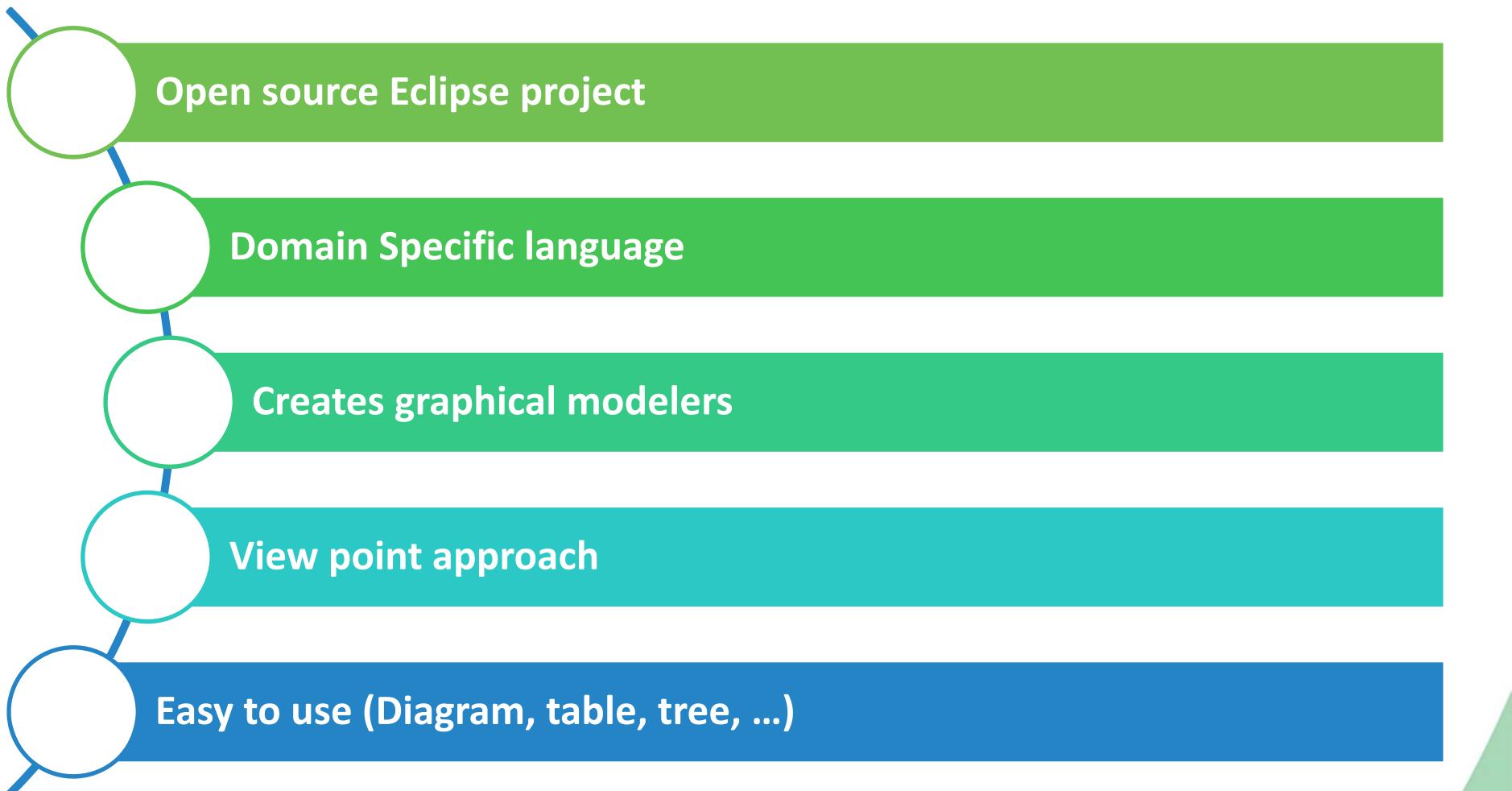
UC-FR-01

Synchronization between models and documents



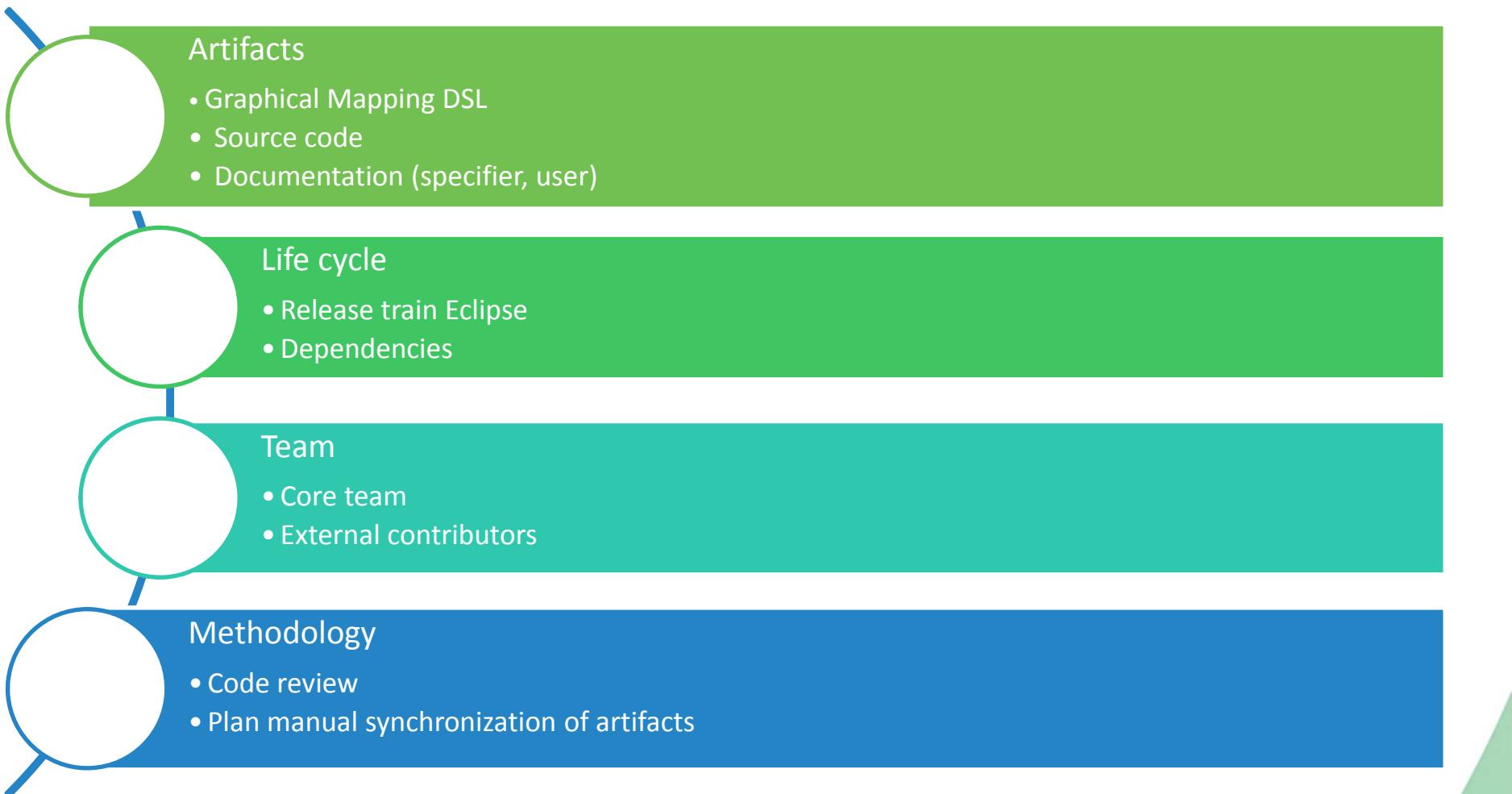
UC-FR-01

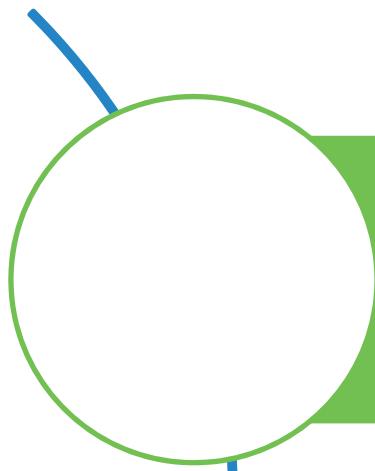
Sirius



UC-FR-01

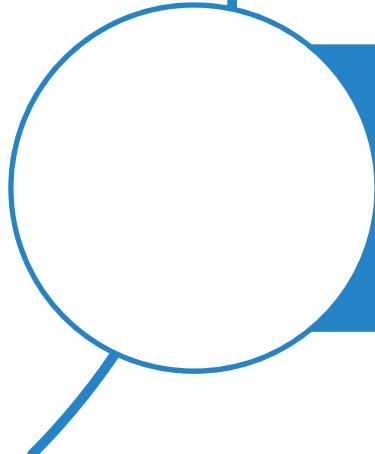
Development of Sirius





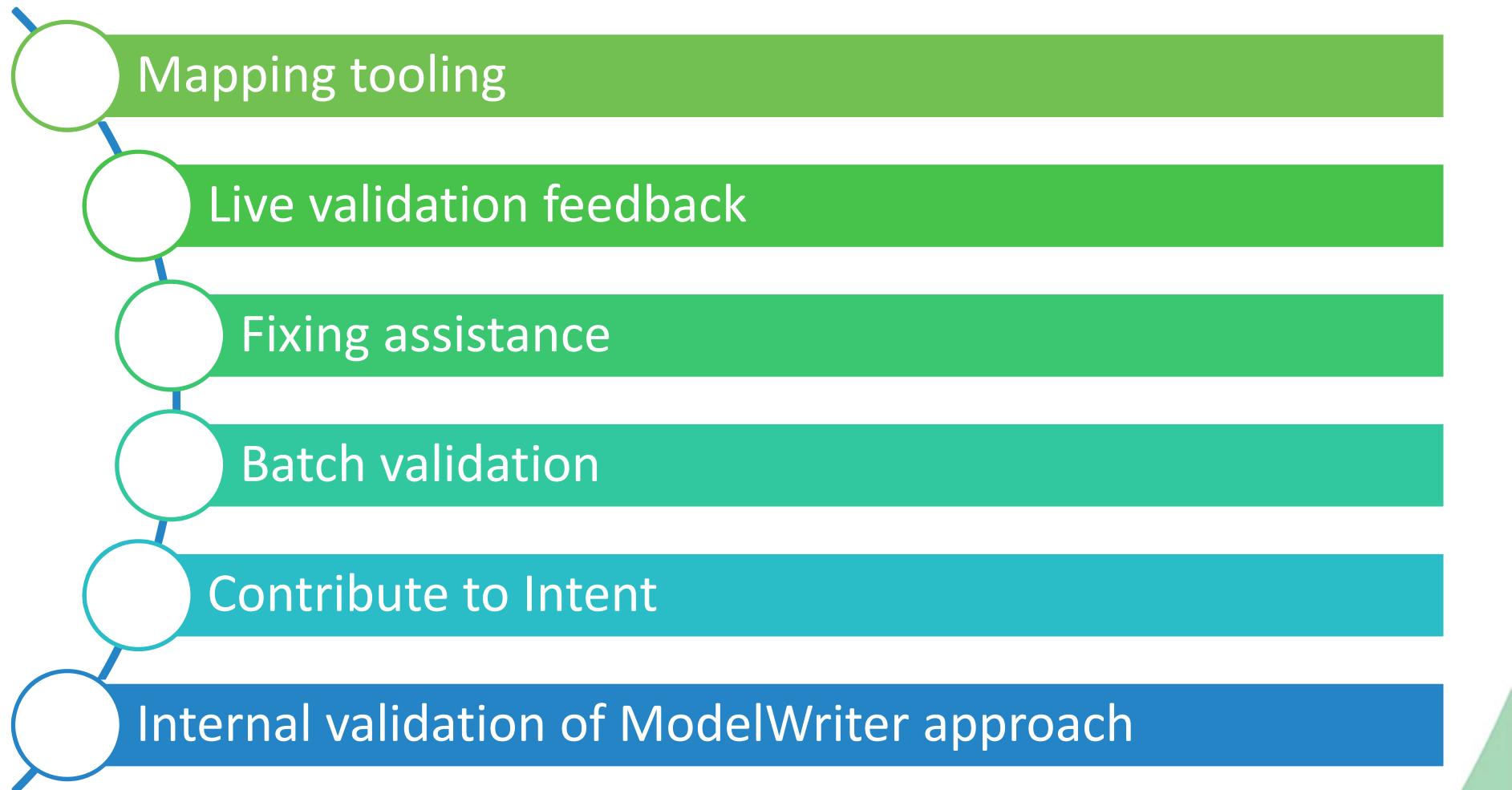
Synchronization gap

- Development
- Artifacts knowledge is volatile



Validation

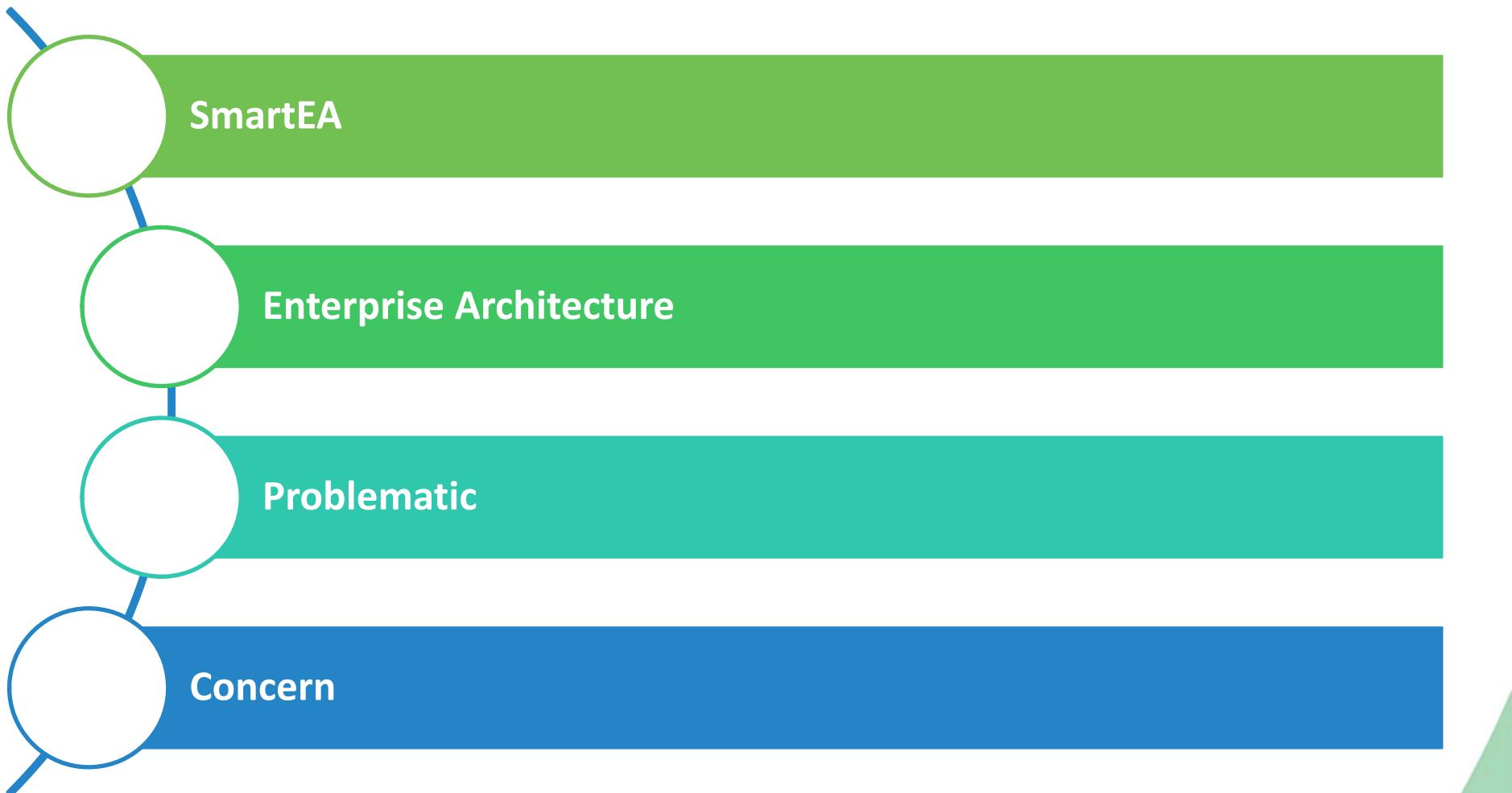
- Release
- Quality

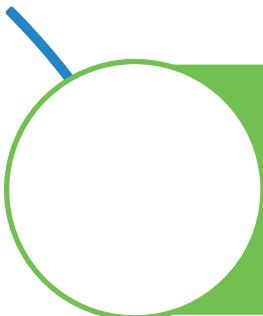


UC-FR-02 - Enterprise Architecture

*Yvan lussaud
OBEO*

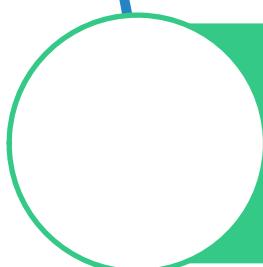






Model the company

- Centralize information
- Ease communication in the company
- View point approach



Support changes in the company

- Business opportunities, business changes, interoperability, new technologies
- Gap analysis
- Impact analysis

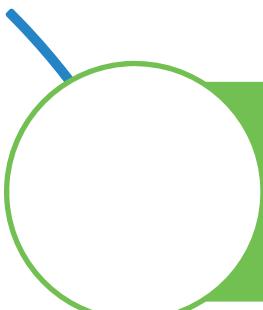


Technical information

- Web portal
- Eclipse and Sirius editors
- Internal model can be changed (TOGAF9, BPMN, ...)

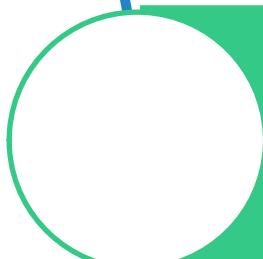
UC-FR-02

Enterprise Architecture



Define As-Is state

- Import documents
- Consolidate the company model



Define To-Be state

- Identify possible scenario
- Evaluate possible scenario
- Modify the company model



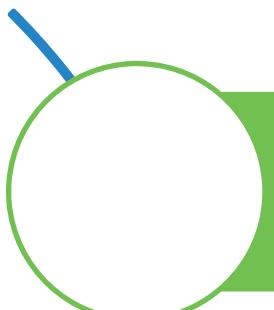
Define trajectories

- Gap analysis
- Impact analysis
- Define milestones



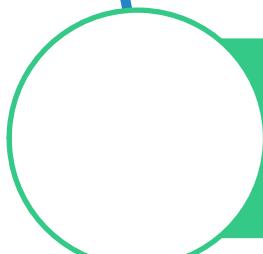
UC-FR-02

Problematic



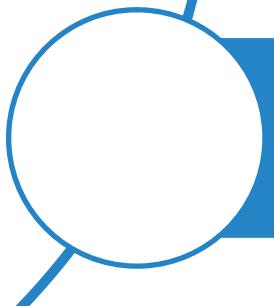
Documentation

- Keep track of sourced documents
- Keep representation and portal up-to-date



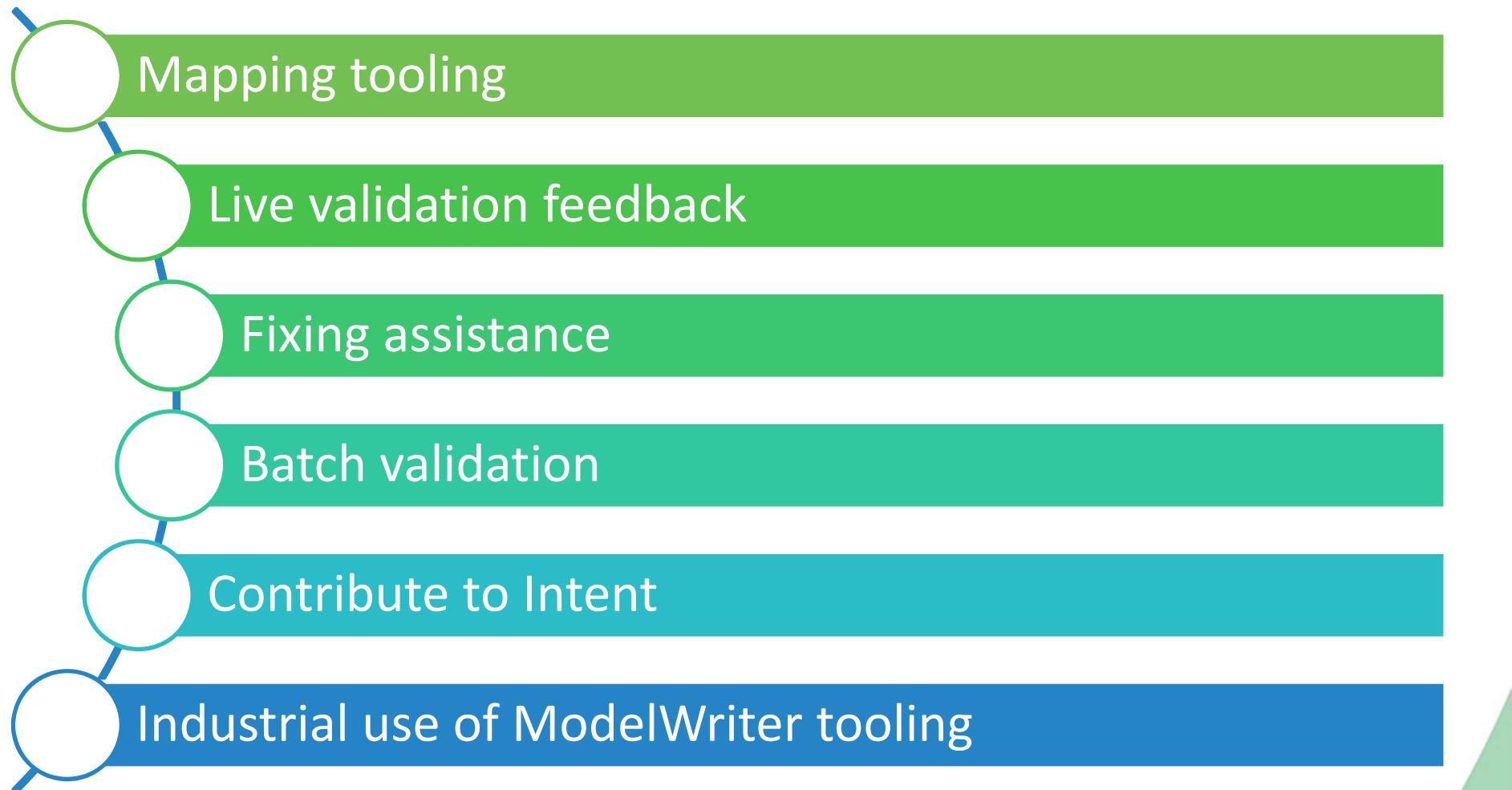
Synchronization gap

- Organizational changes
- Artifacts knowledge is volatile



Validation

- Produce documentation of organizational changes
- Quality



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

Anne MONCEAUX

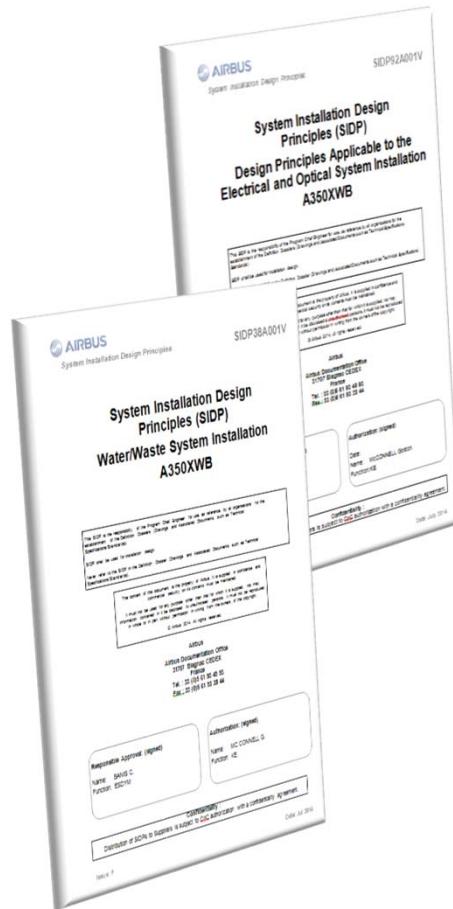
Airbus Group Innovation

Louis ROUCH, Ayhan MOHOVIC

Airbus SAS

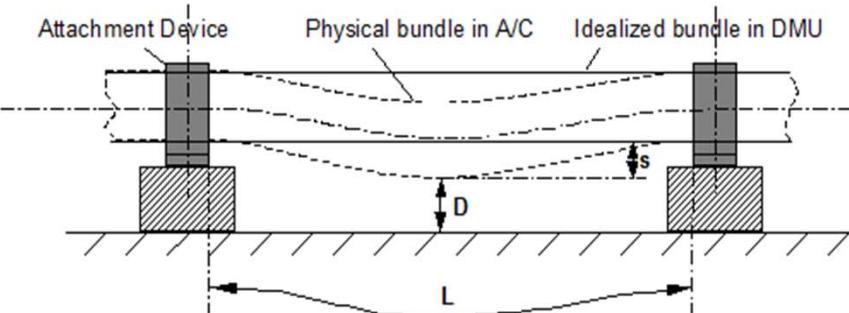


SIDP: “System Installation Design Principles”



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

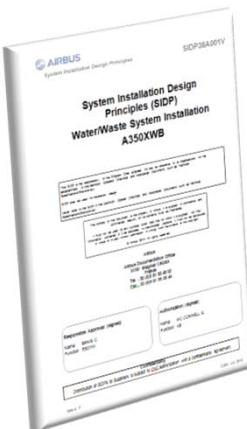
L... Actual length of a bundle segment between two Attachment Points (as designed in DMU)

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.

Context and problem

- SIDP documents explain how to install the aircraft systems and attach them to the structure. They capitalize the best practices & proven technical solutions.
- SIDP are defined per [ATA chapter](#) (~functional domain) to be applied for [each given A/C project](#): for example “A350 Electrical installation”
- SIDP are [open to Extended Enterprise](#): installation tasks are performed by risk sharing partners.
- SIDP are [living documents](#): during the aircraft development any new DP allowing to satisfy all targets/constraints can be added, assuming it is validated by Airbus dedicated committee.



Industrial high level needs

To improve SIDP creation, maintenance and consultation in order to:

- save costs by supporting DP consultation / retrieval in accordance to the needs
- avoid non-compliance of design caused by DP updating lead times
- keep traceability with upstream regulations and requirements and downstream design models

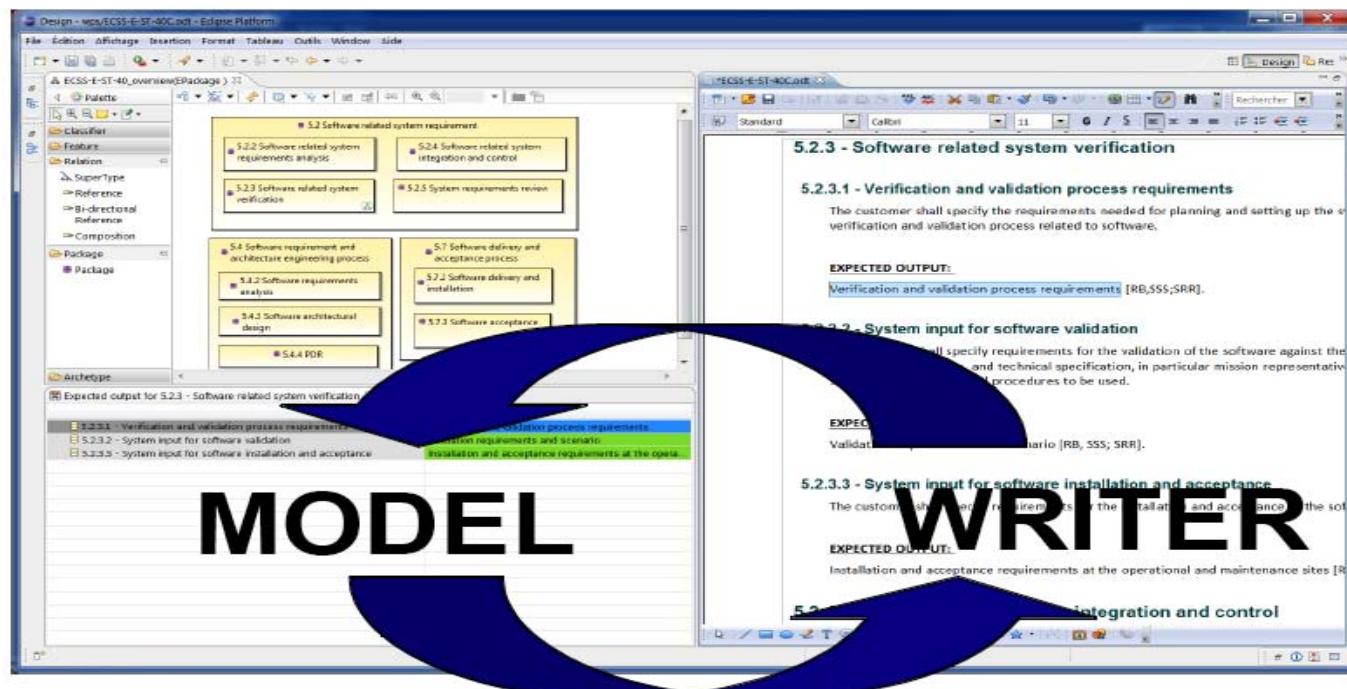
Synchronization of regulation documentation with a design rule repository

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



Goal: synchronizing the SIDP database content with documentation

- Create links between text fragments & model elements (manual annotation?, semi-automatic?...)
- Manage consistent synchronization (manage changes)



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



- Approach: limited case on Electrical Installation functional domain
- (Confidential Data - Non Disclosure Agreement finalized in June 2015).
 - Text
 - 1 document: SIDP ATA 92
 - In our industrial context SIDP are edited using MSWord
 - Models
 - An OWL model is built that reflects the DB schema: “Rule ontology” (30 classes, 35 object properties and 54 data properties)
 - Automatic population mechanism of the model from a csv BD export produces the KB (45781 triples)

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



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

The screenshot shows the Edgeware Platform interface with two main windows. The left window, titled 'Design - ngn/ECSS-E-ST-40C.nct - Edgeware Platform', displays a 'Model Overview' tree with various components like 'Architecture', 'Requirements', 'Design', 'Implementation', and 'Verification'. The right window, titled 'Design - Res - ECSS-E-ST-40C.nct - Edgeware Platform', shows a 'Text Overview' tree with similar categories. A large blue arrow points from the left window to the right window, indicating the flow of synchronization. Two green callout boxes provide detailed descriptions:

- Model part = rule KB** (points to the left window)
- Textual part = rule sentences** (points to the right window)

On the right side of the slide, there is a large green box containing the following text:

Thermal variation shall be taken into account
Structural deformation shall be taken into account
Pressurization taken into account
Component/It qualified to withstand temperature reached by the routes they are in contact with

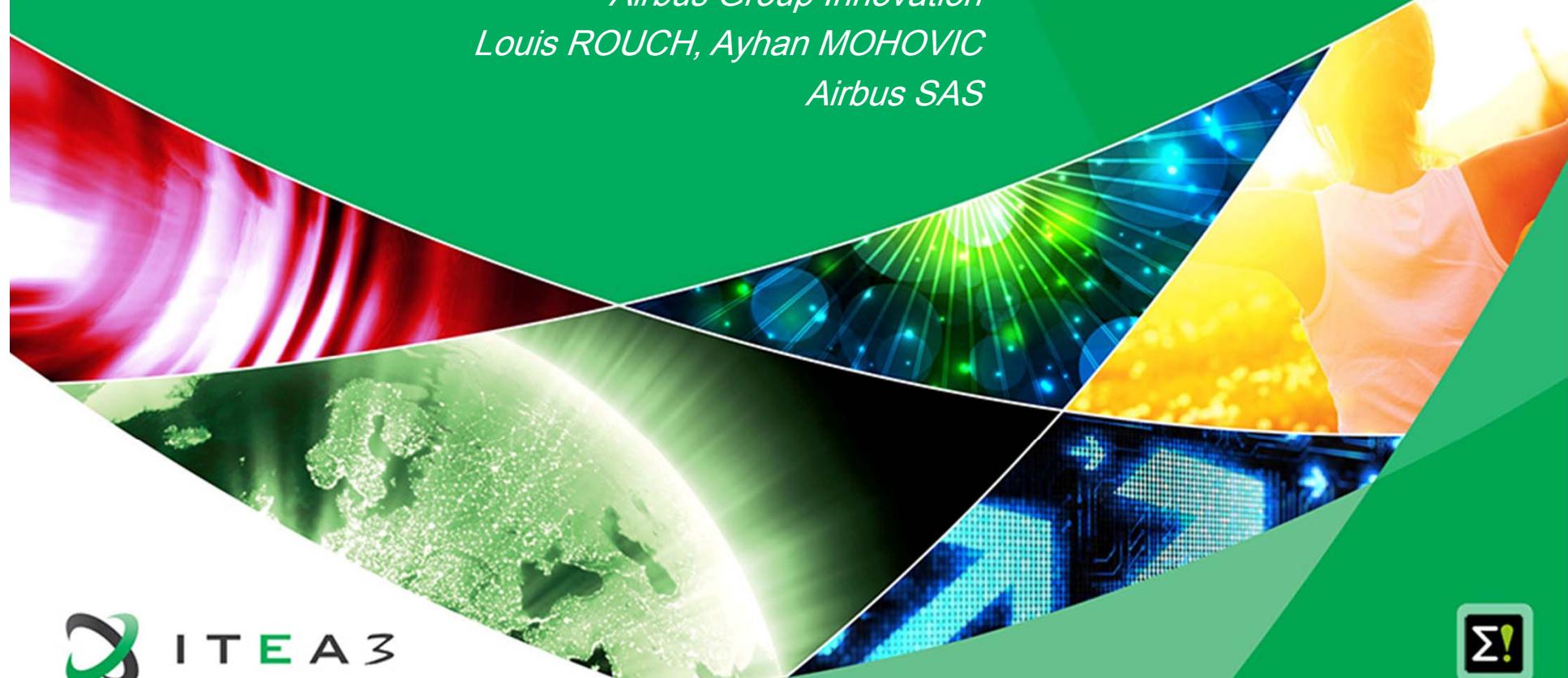
UC-FR-04 - Production of a Context Specific Design Document

Anne MONCEAUX

Airbus Group Innovation

Louis ROUCH, Ayhan MOHOVIC

Airbus SAS

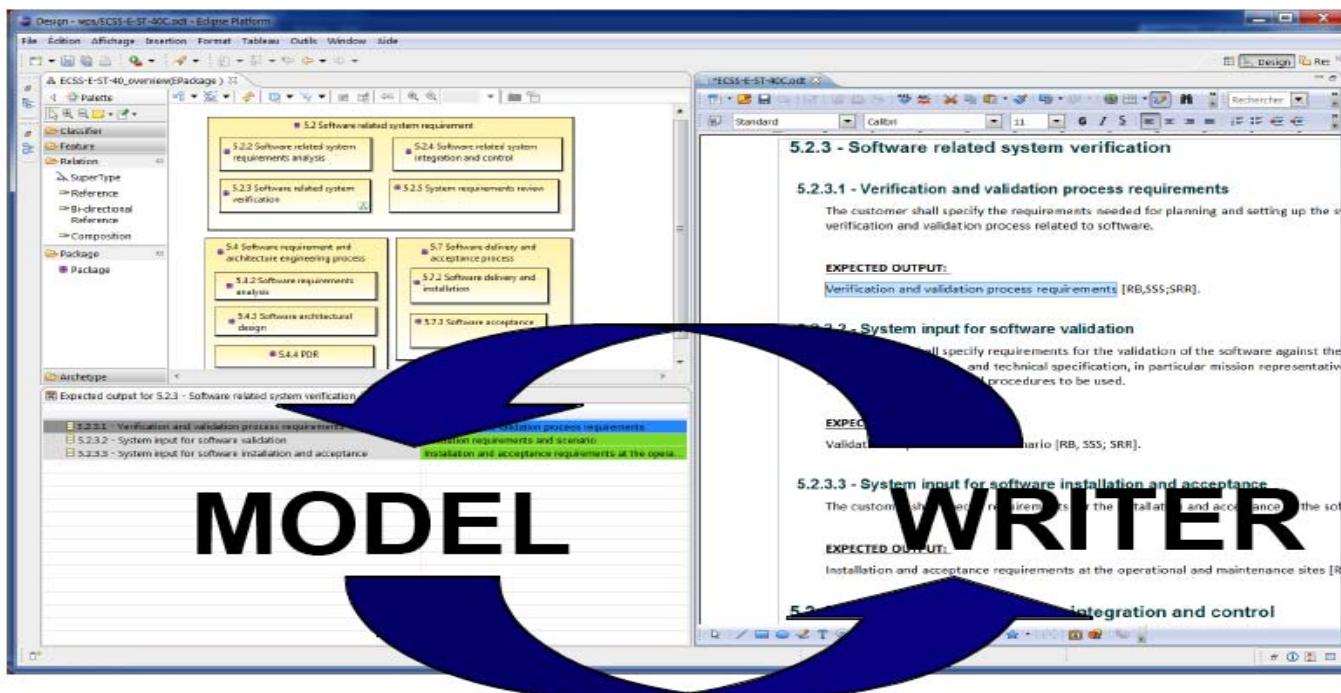


UC-FR-04 Production of a context specific design document



Goal: producing document according to usage “needs”

- Produce “filtered” document with subset of Design Principle textual elements according to usage “needs”

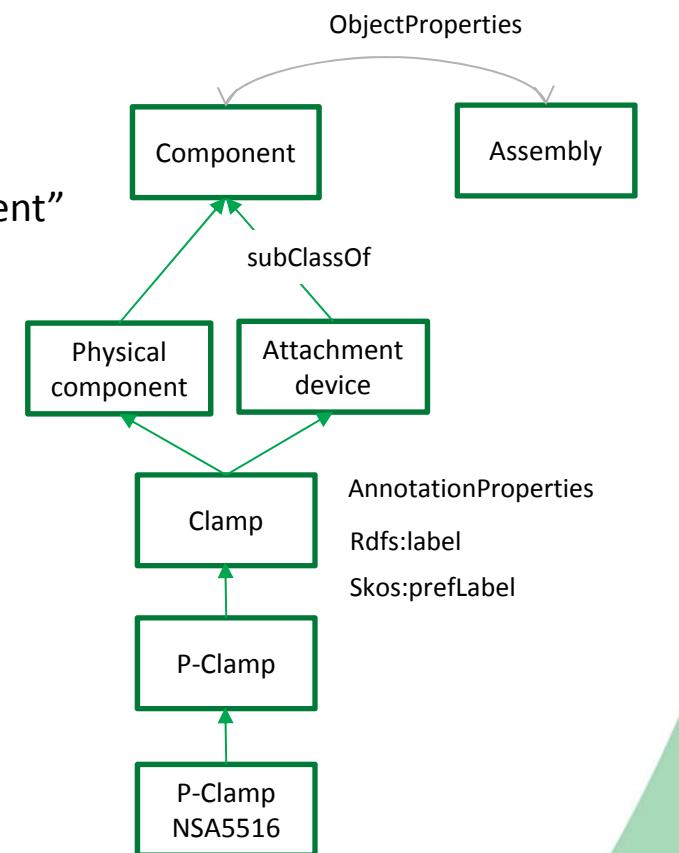
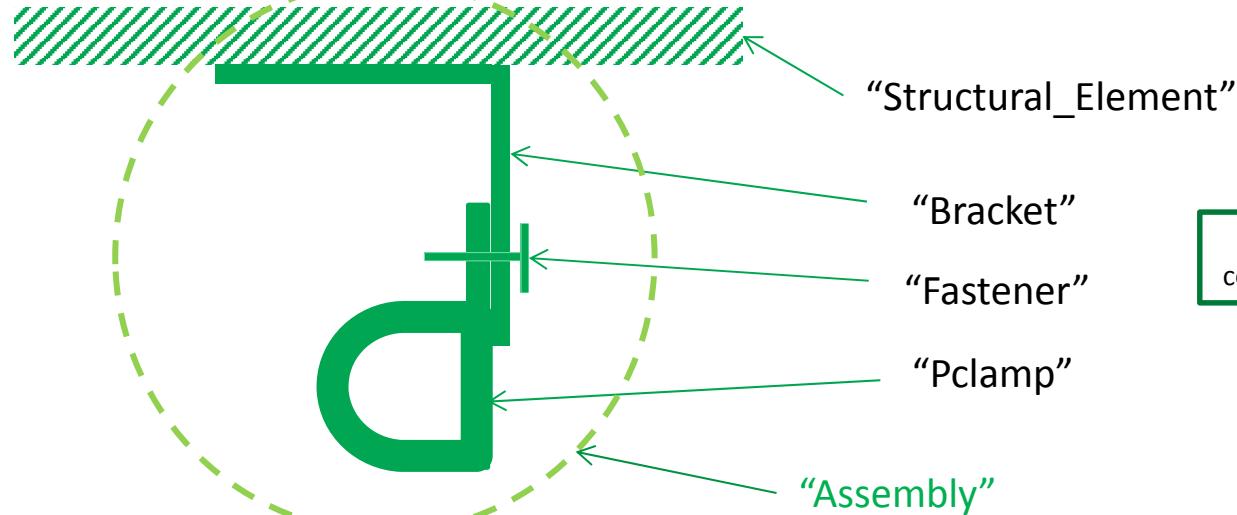


UC-FR-04 Production of a context specific design document



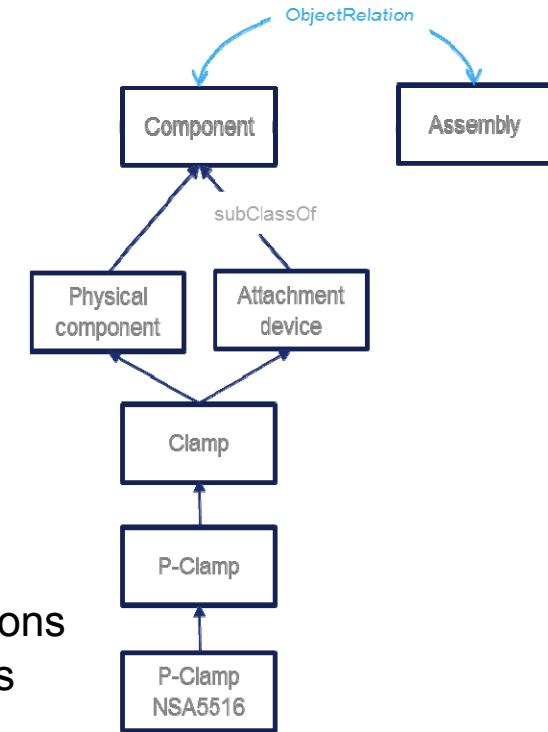
- Approach:
 - use case Electrical Installation functional domain
 - Confidential Data - Non Disclosure Agreement finalized in June 2015
 - Use model elements to retrieve relevant Design Principles
- Text
 - 1 document: SIDP ATA 92
 - In our industrial context SIDP are edited using MSWord
- Models
 - The previous Rule KB (populated Rule ontology)
 - **Component ontology** (476 classes, 21 ObjectProperties and 35 DataProperties)

Component classes taxonomy



Component classes taxonomy

“P-clamp NSA5516 can be fixed on X with Y”
 “Physical component” “Standard reference”

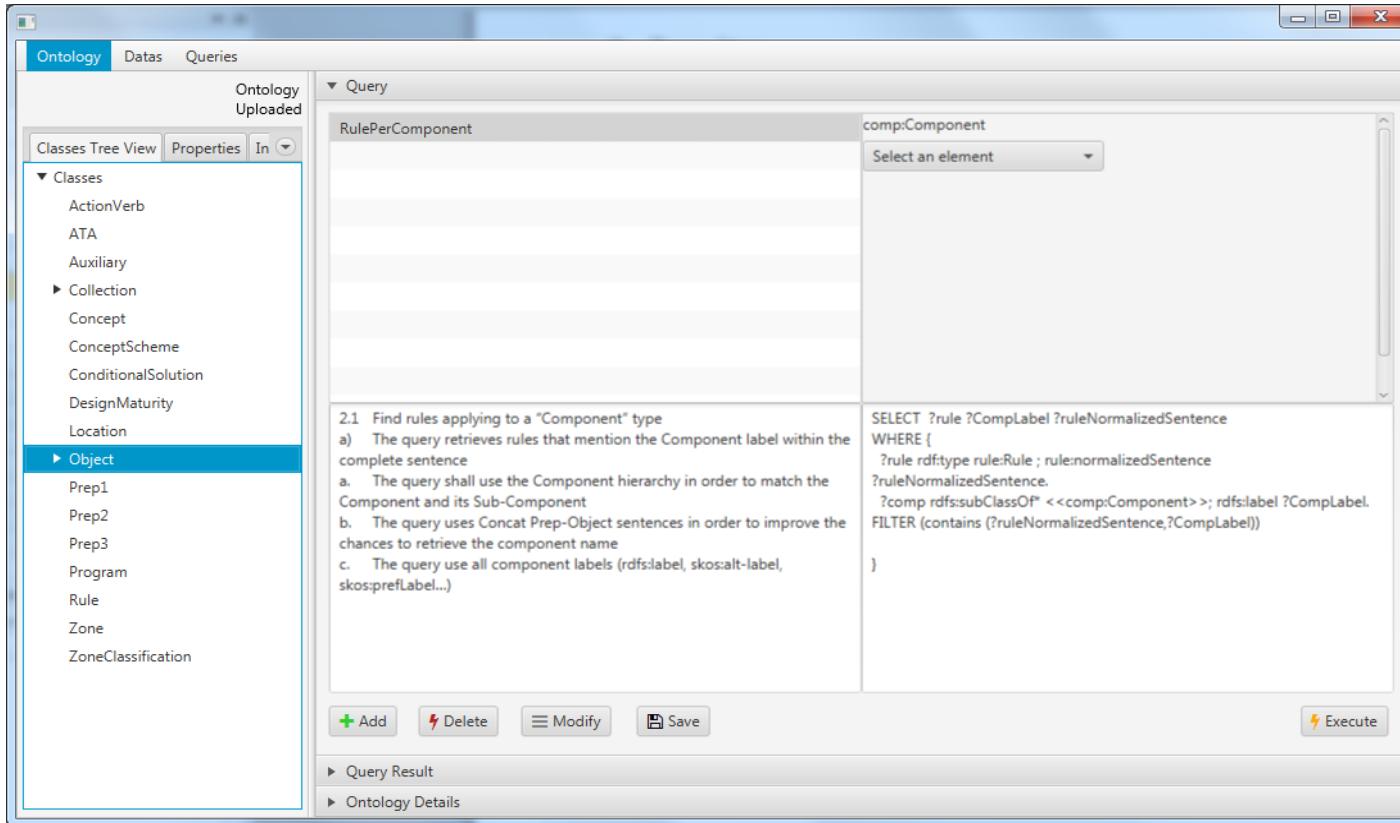


- **NLP Parsing** uses this taxonomy. Labels + assumptions such as a physical component may be referred to using its name or its reference or both concatenated
- **Inference rule:** a rule applying to a component type (Attachment device) applies to its subtypes (P-clamp)
- **Document display :** when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



The screenshot shows the RDFizer application interface. On the left, there's an ontology browser with tabs for Ontology, Data, and Queries. Under the Ontology tab, there's a tree view of classes like ActionVerb, ATA, Auxiliary, Collection, Concept, ConceptScheme, ConditionalSolution, DesignMaturity, Location, and Object. The 'Object' node is currently selected. On the right, there's a 'Query' panel with a sub-panel titled 'RulePerComponent'. Below it, a detailed description of a query is shown:

2.1 Find rules applying to a "Component" type

a) The query retrieves rules that mention the Component label within the complete sentence
 a. The query shall use the Component hierarchy in order to match the Component and its Sub-Component
 b. The query uses Concat Prep-Object sentences in order to improve the chances to retrieve the component name
 c. The query use all component labels (rdfs:label, skos:alt-label, skos:prefLabel...)

The query itself is:

```

SELECT ?rule ?CompLabel ?ruleNormalizedSentence
WHERE {
  ?rule rdf:type rule:Rule ; rule:normalizedSentence
  ?ruleNormalizedSentence.
  ?comp rdfs:subClassOf* <<comp:Component>>; rdfs:label ?CompLabel.
  FILTER (contains (?ruleNormalizedSentence,?CompLabel))
}
  
```

At the bottom of the query panel, there are buttons for Add, Delete, Modify, Save, and Execute.

UC-TR-01 - Documents of Quality Assurance Department

Ersan GÜRDOĞAN
HISBİM



UC-TR-01 Documents of Quality Assurance Department



UC-TR-01	Documents of Quality Assurance Department
Version	V1.0.0 dated 15-Nov-2014
Description	To create faster and more accurate the forms that are used in quality control progress and trigger related forms (re-work form triggering, revision needed, approved, rejected etc.)
Actors	Quality Managers, Quality Measurement Specialists, Quality Control Personnel, Quality Auditors, Production Crews
Assumptions	Quality control measurement units are standard Rejected and Approved products forms are standard Quality Certification standards are always applied
Steps	Products information forms are created according to product information comes from ModelWriter system Products are measured by quality control department authorized personnel's according to standards To decide product is standard or not according to measurement report Product would be sent rejected products section if its measures are out of standards ModelWriter system is created Rejected Product Quality Form Else Approved Product Quality Form is created by ModelWriter System Approved products are sent to warehouse as accepted products
Alternatives	If rejected products' measurement out of range in rework standards, it means cannot be applied rework on this product, it returns to scrap. Then it should be sent wasteland.
Issues	Standardizing form designs is hard because of all customers have their own unique reporting tools/formats



UC-TR-02 - Non-Disclosure Agreements

Ersan GÜRDOĞAN
HISBİM



UC-TR-01 Non-Disclosure Agreements



UC-TR-02	Non-Disclosure Agreements
Version	V1.0.0 dated 15-Nov-2014
Description	<p>Non-Disclosure Agreement (NDA), also known as a confidentiality agreement (CA), confidential disclosure agreement (CDA), proprietary information agreement (PIA), or secrecy agreement (SA), is a legal contract between at least two parties that outlines confidential material, knowledge, or information that the parties wish to share with one another for certain purposes, but wish to restrict access to or by third parties. It is a contract through which the parties agree not to disclose information covered by the agreement. An NDA creates a confidential relationship between the parties to protect any type of confidential and proprietary information or trade secrets. As such, an NDA protects nonpublic business information.</p> <p>NDAs are commonly signed when two companies, individuals, or other entities (such as partnerships, societies, etc.) are considering doing business and need to understand the processes used in each other's business for the purpose of evaluating the potential business relationship. NDAs can be "mutual", meaning both parties are restricted in their use of the materials provided, or they can restrict the use of material by a single party. It is also possible for an employee to sign an NDA or NDA-like agreement with an employer. In fact, some employment agreements will include a clause restricting employees' use and dissemination of company-owned confidential information.</p>
Actors	Responsible/Authorized personnel in both parties.
Assumptions	Agreements are prepared and written according to European Business Law.
Steps	To define both parties who would sign the NDA To write items of agreement according to scope of NDA Reviewing agreement by decision maker then getting approval Sharing NDA each other Send feedback to ModelWriter system if any change apply on NDA Sharpen final version of NDA then signing by both parties
Alternatives	After sharing NDA, both parties sign without any change and no feedback. Cancellation of NDA
Issues	Different laws could be applied in out of European Union



UC-TR-03 - Synchronization of ReqIF models from requirement specifications

Ferhat Erata
UNIT



UC-TR-03 - Synchronization of ReqIF models from requirement specifications



- Tech doc to ReqIF model
 - e.g Req doc => ReqIF

Examples:

- Airbus SIDP template=> ReqIF
- Havelsan Req doc template=> ReqIF
- Univ Man. System Use cases screenshot as the corpora doc to ReqIF
- Eclipse RMF use case specification ReqIF

- Snapshot



UC-TR-04 - Integration with Application Lifecycle Management (ALM) Tools

*Eray Tüzün
Yagup MACİT*

HAVELSAN



UC-TR-04 - Integration with Application Lifecycle Management (ALM) Tools



UC-TR-03	Integration with ALM tools
Versioning Info	V1.0.0 dated 28-Apr-2015
Description	<p>Show that the ModelWriter can extract required elements from structured requirement objects in ALM tool to generate automatic design model.</p> <p>Show that the ModelWriter's can generate requirements specification document from structured requirement objects in ALM tool.</p>
Actors	Requirements Engineer/Manager, System/Software Architect
Assumptions	<ul style="list-style-type: none"> All the traceability information would continue to be followed from the structured requirement object via ALM tool. (ModelWriter does not need to have any traceability information) Textual representation of requirements are stored in MS Word documents. Design models are stored in Sparx Systems Enterprise Architect. Structured requirement objects are stored in Microsoft Team Foundation Server.
Steps	<p><u>Scenario1:</u></p> <ul style="list-style-type: none"> From a set of structured requirement objects in the ALM tool, ModelWriter would create a natural-language text requirement document. <p><u>Scenario2:</u></p> <ul style="list-style-type: none"> From a set of structured requirement objects in the ALM tool, ModelWriter would create/synch a design model. From a design model, ModelWriter would create/synch to a set of structured requirement objects.
Variations (optional)	Scenario1: From a natural-language text requirement document, ModelWriter would create/synch to a set of requirement objects.
Non-functional (optional)	<p>The system should have a Word plugin for natural-language text transformation.</p> <p>The natural-language support should be in both English and Turkish</p>
Issues	A common format between ALM platforms and ModelWriter may need to be implemented.

Requirement Work Item

Customized Form

Attributes

WorkItem Number Patterns

History

Discussion

Requirement 4878*: Üniversitenin tüm akademik birimlerini hiyerarşik olarak görebileceklerdir.

Code: YGB_BLG_001 **Title:** Üniversitenin tüm akademik birimlerini hiyerarşik olarak görebileceklerdir.

DESCRIPTION ANALYSIS STORYBOARDS ALLOCATIONS LINKS ATTACHMENTS HISTORY PLAN OTHER **CLASSIFICATION**

New → Proposed
Yagup MACİT 7/26/2012

DISCUSSION ONLY ALL CHANGES

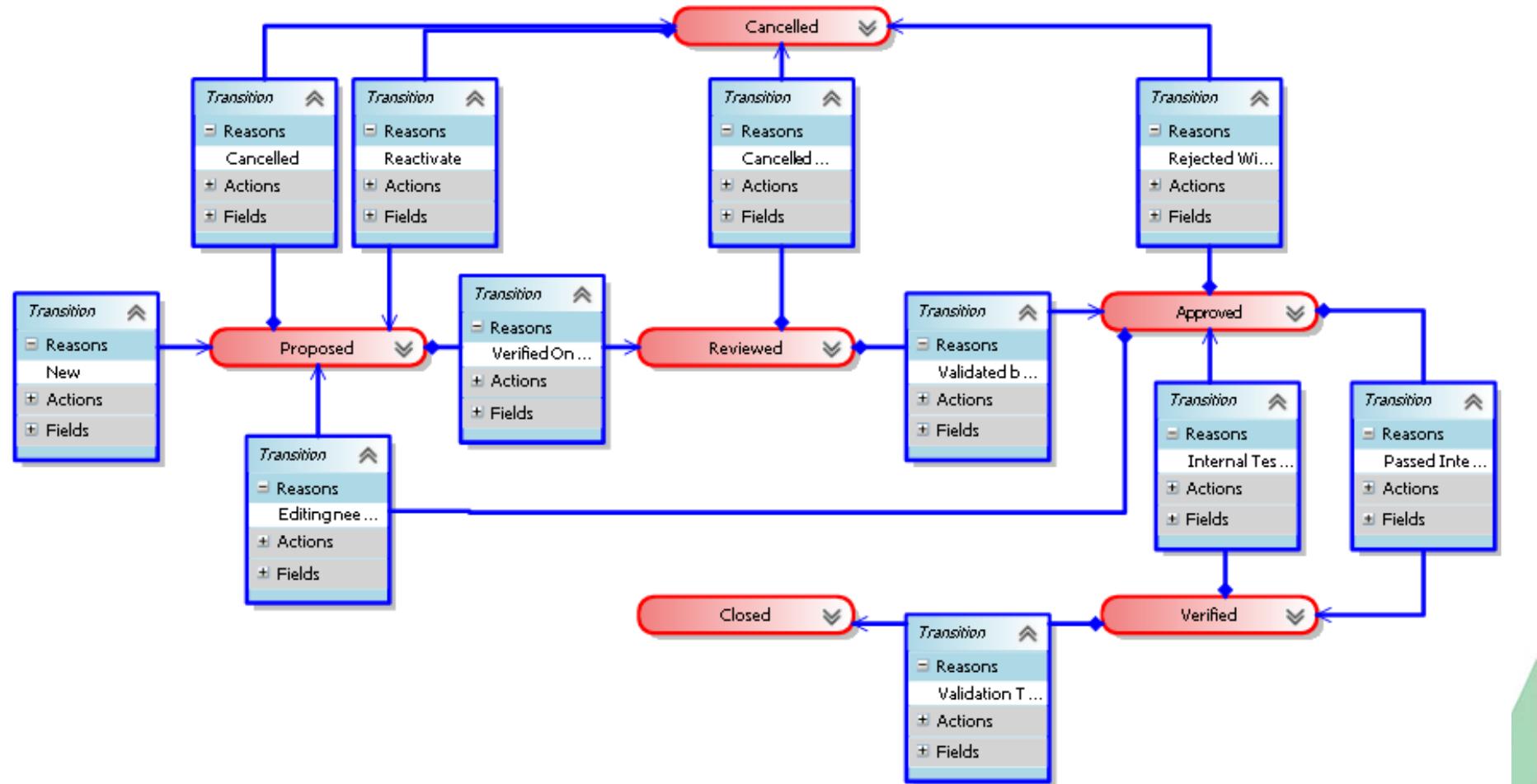
- ▷  Kürşat İNCE made field changes (2 weeks ago)
- ▷  Kürşat İNCE made field changes (2 weeks ago)
- ▷  Nahit Fırat DOĞAN added link (3 months ago)
-  Yagup MACİT created the Requirement (8 months ago)
 - ◀ Fields

Field	New Value
Iteration Path	UYY\Sürüm 2\Tur 4
Iteration ID	20
Team Project	UYY
Node Name	II. Faz - Gereksinim Yönetimi (A-H91201.02.10)
Area Path	UYY\YGO (A-H91201.02)\II. Faz - Gereksinim Yönetimi (A-H91201.02.10)
Area ID	84

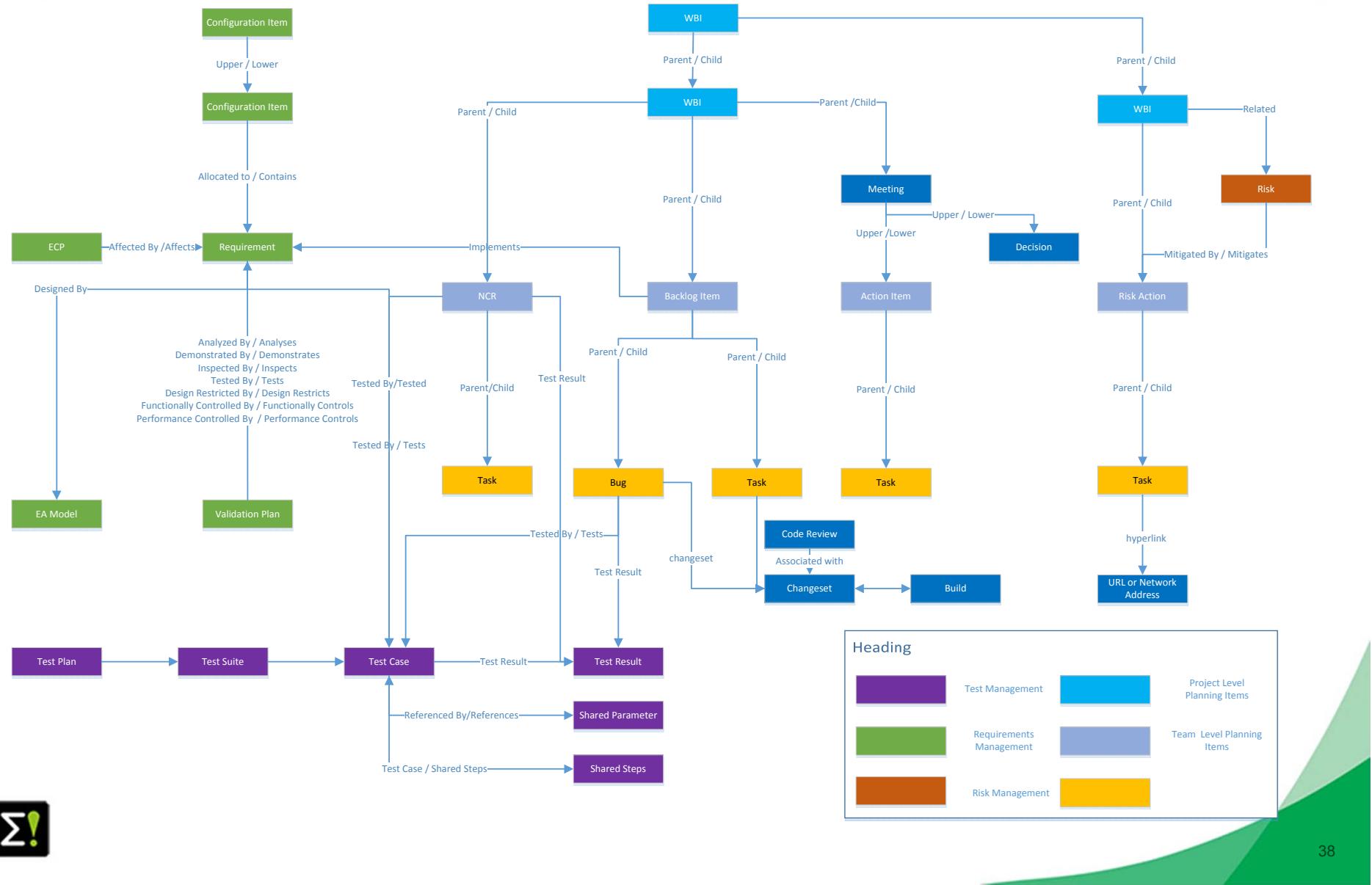
21.9.2015



Requirement LifeCycle



Modeling Artifacts 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
 - ...

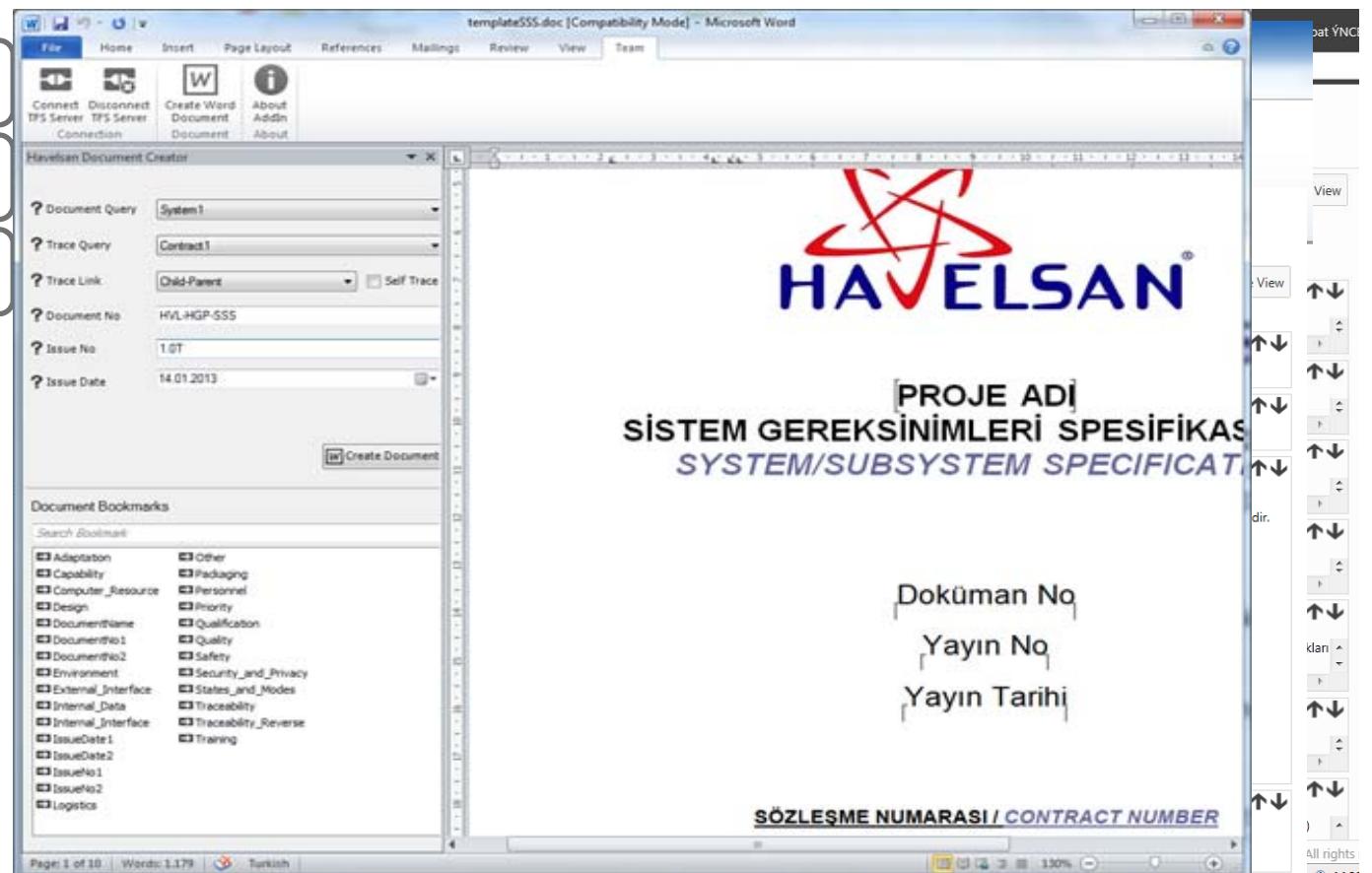


Havelsan Ext - Document Generation

Preview

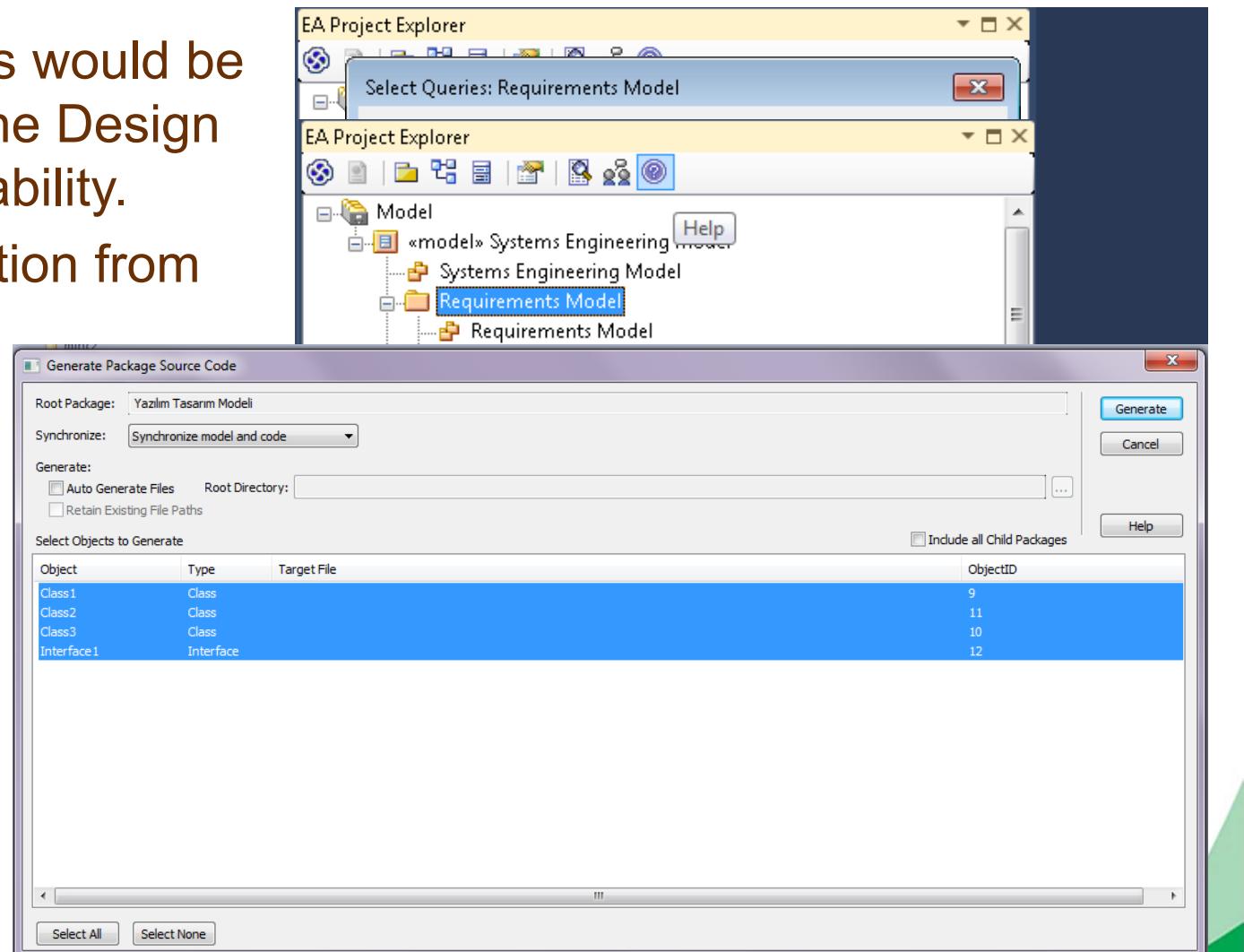
Word Extension

Support for Templates

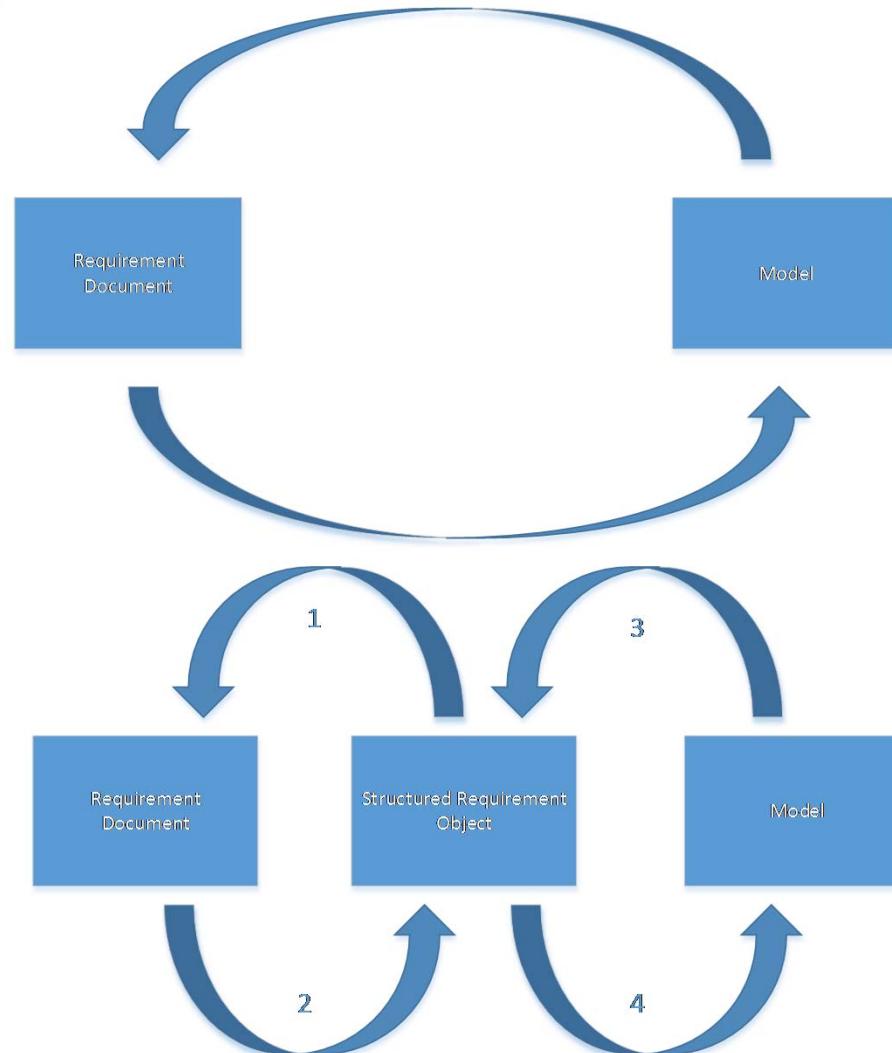


Requirements - Design Model Traceability

- Requirements would be imported to the Design tool for traceability.
- Code generation from design



HAVELSAN Use case for ModelWriter



Currently we support scenario #1 and #4, and interested in Scenario #3, #2

UC-TR-05 - Synchronous Business Process Design with Use Cases

Geylani Kardaş

KoçSistem



UC-TR-05 - Synchronous Business Process Design with Use Cases



- Use case document to BPMN model transformation
 - Univ Man. System Use cases screenshot as the corpora doc to BPMN
 - Eclipse RMF use case specification BPMN
- (Planned) BPMN model transformation to Use case document (screenshot of the corpora model)
- Next step: using technique documents of Ford-Otosan;

- screenshots



**Thank you for your attention!
Any question ?**

