Pré-étude Compatibilité : Pré-étude, Modèle, résultat Stage compatibilité

Référence IRT Saint Exupéry: NT-S085L01-054
Référence IRT System X: : ISX-S2C-DOC-478
Version : V1
Date : 31/03/2023

Author(s)	*Fonction(s) & name(s)*	*IRTs Team*	*S. Creff*
Checker(s)	*Fonction(s) & name(s)*	*Pilote IRT SystemX*	*A. Dubois*
Approver	*Fonction & name*	*Chef de projet IRT Saint Exupéry*	*J. Perrin*

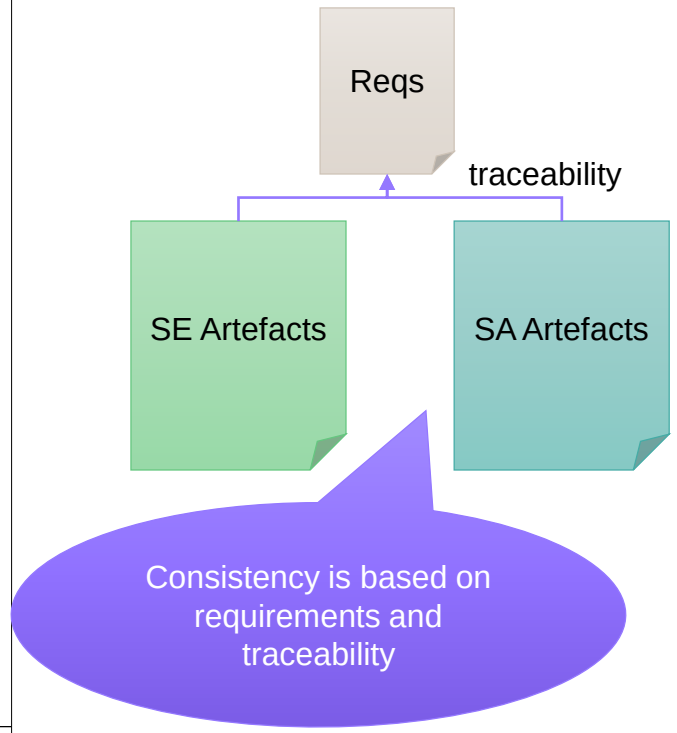
Pre-study on Compatibility

Pre-study, Model, Result
Compatibility internship

Summary

1. Context - two "main streams" of implementation of consistency in systems engineering

- « Traditional » Requirement Driven Approach

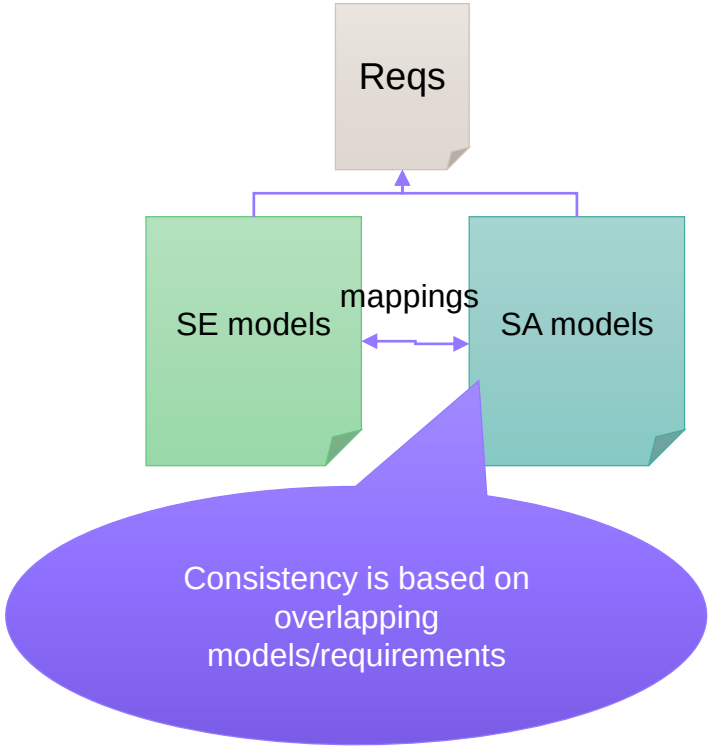


Difficulties: expression of verification rules applicable on unstructured information

Construction of rules, checklists, ... on artifacts (with their level of abstraction and details)

Idea: add a *complementary* new abstraction to reconcile engineering domains

- Model driven approach



Challenges: identification and formalization of exhaustive overlap between models

2. Compatibility : a new abstraction level that supports global consistency

Consistency *: are the data / points of view used by the different domains well aligned?*

vs

Compatibility *: are the different domains constraints or solutions compatible*

3. Exploratory approach - summary

- Philosophy: formalize the abstraction and then bring out the rules of incoherence (incompatibility in a given logical space)
- **Question** : which abstraction to formalize ?
  - Covering critical cases of inconsistency,
  - Presenting a "high level" view of architecture and design decisions,
  - Declined on levels: functional, logical, physical...

Engineering continuity, cost reduction

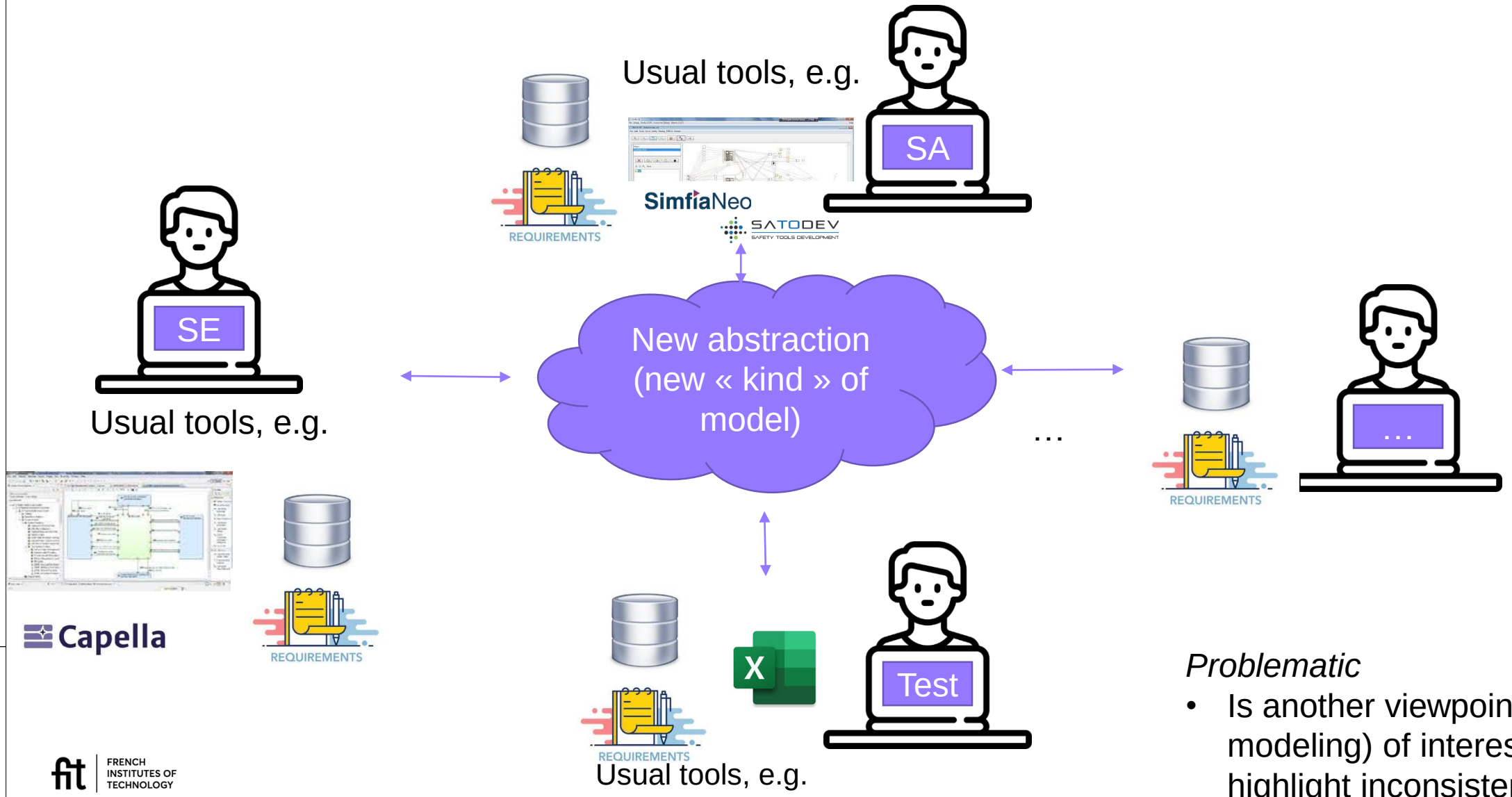
- **Objectives**:

1. Avoid redundancy of measures (solutions) proposed by the business Identification of needs / solution patterns
2. Avoid inconsistencies (incompatibilities) from this abstraction/modeling of a domain of possibilities and incompatibilities at the need and solution level.

Reduce the risk of inconsistency

Perimeter of the domains concerned for the illustration: Archi (IS and SW) + RAMS + Test

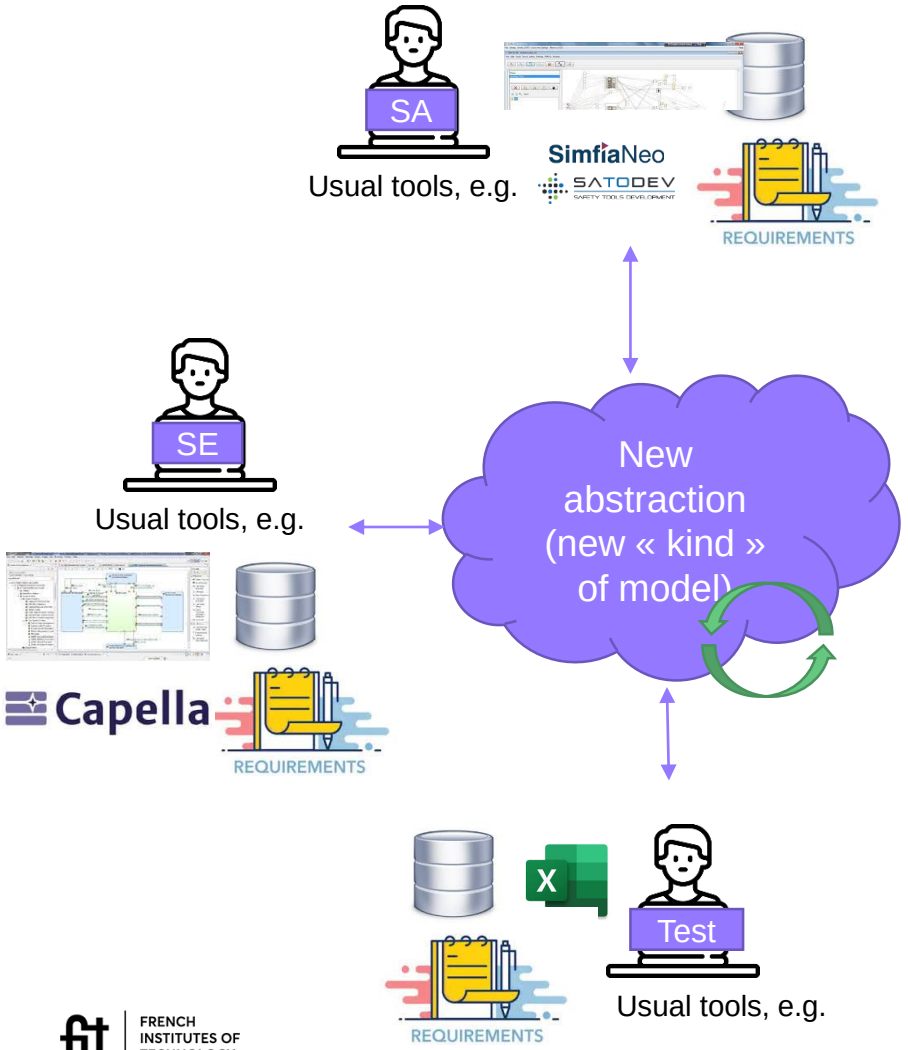
4. Goal: a new abstraction to facilitate the detection of incompatibilities



Problematic

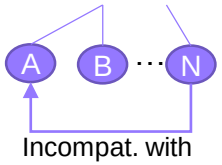
- Is another viewpoint (i.e. modeling) of interest to highlight inconsistencies?

4. Goal: a new abstraction to facilitate the detection of incompatibilities



What to capture in this new abstraction?

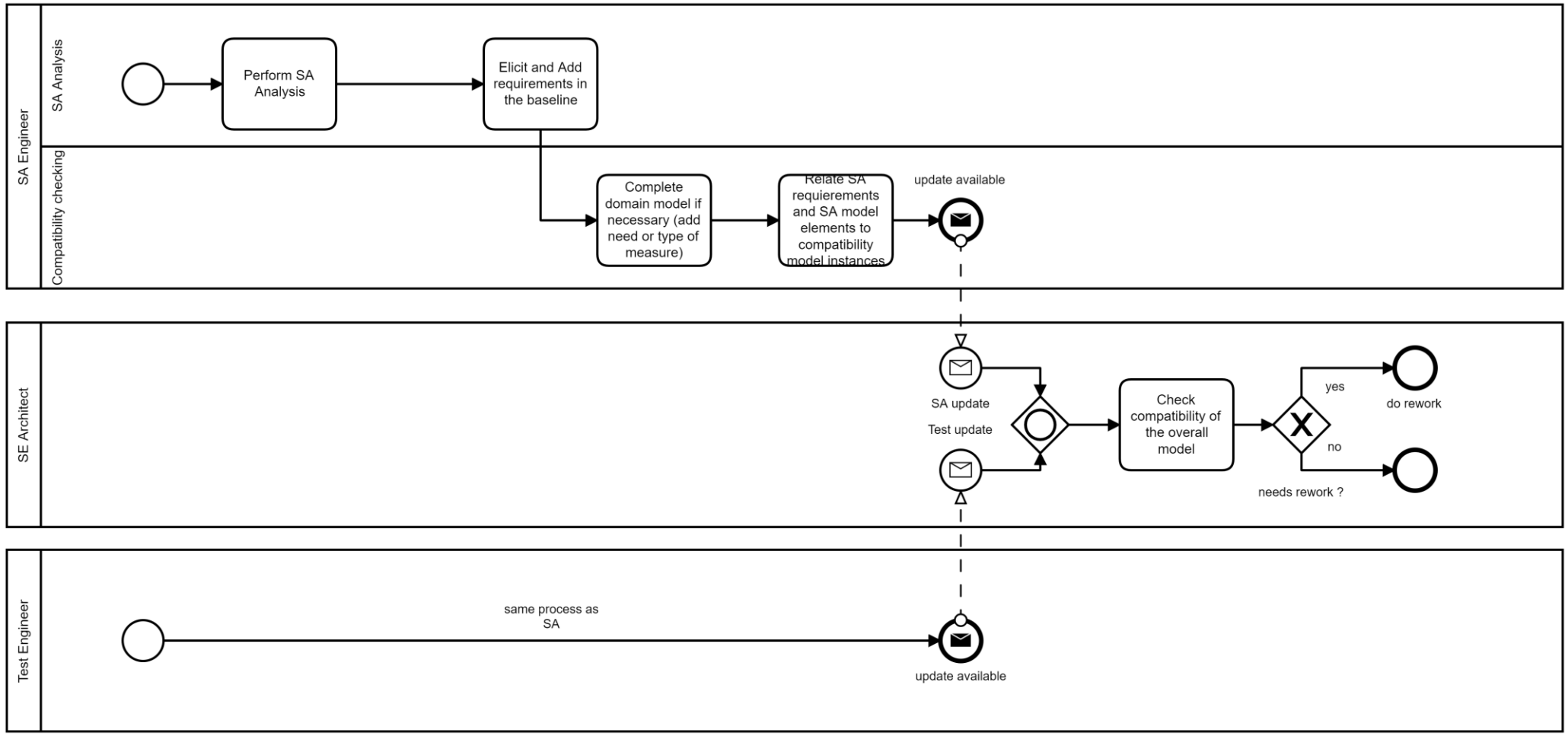
- **For each concern** (SE, SA, Test, …)
  - Explain the **objectives / needs**, e.g.
    - SA: Increase reliability, increase robustness, …
    - Test: Increase testability
    - SE: Increase performance, reduce space requirement, …
  - Explain the **measures** applied, e.g.
    - SA: redundancy, diversity, quality of components, …
    - Test: add test links, …
    - SE: NF constraints…
  - **Related them to existing engineering artefacts** (instances), e.g.
    - Model elements in Capella, in SimfiaNeo, Requirements….
- **Between concerns** (architecting)
  - **Capture high level incompatibilities** between measures, e.g.
    - Redundancy != reduce space requirement,
    - Independence != testability
  - **Check if contradictory measures are applied** on a same set of artefacts (instance) to raise warnings!



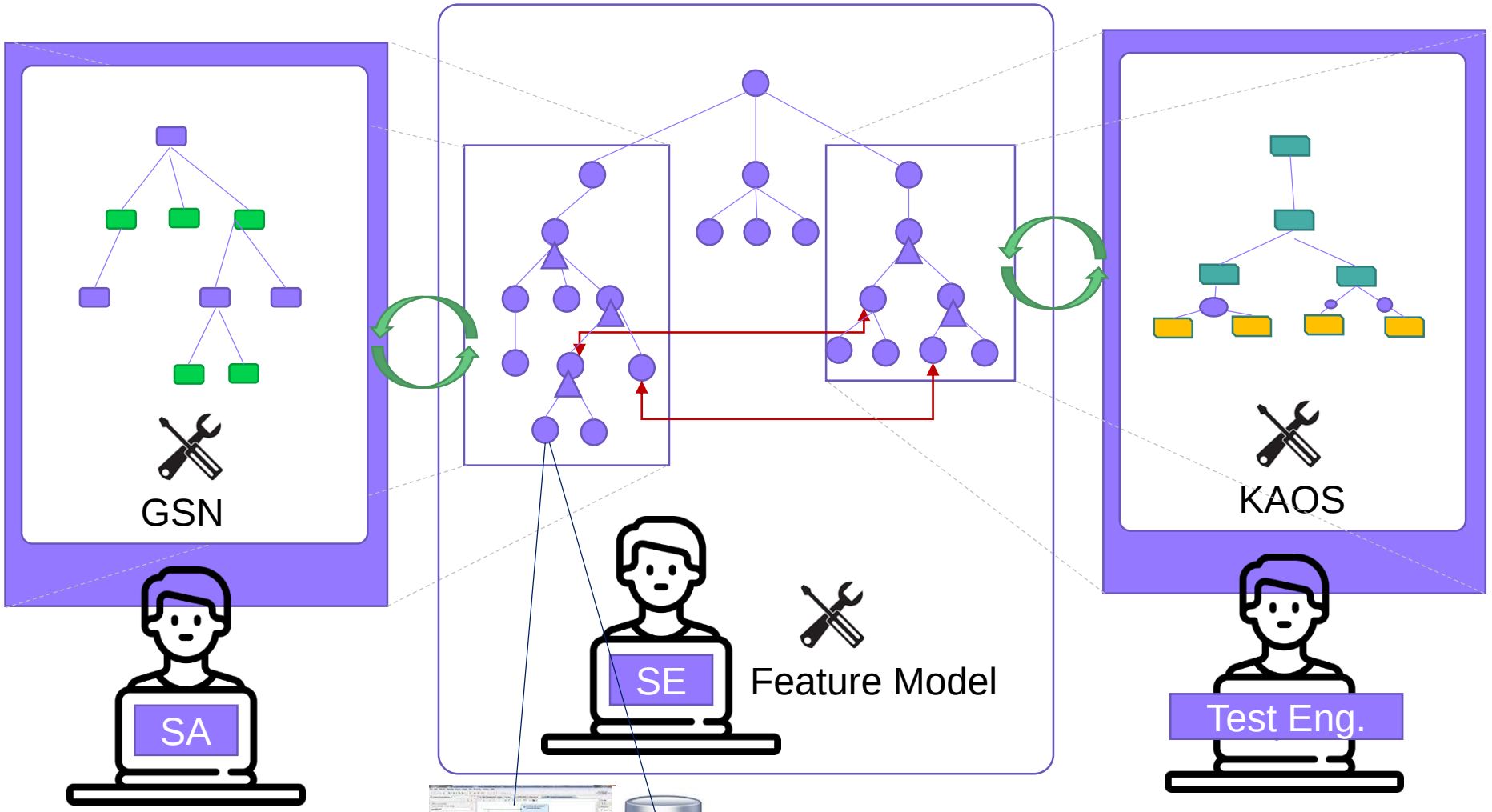
➢ Capture a domain knowledge, that will be iteratively enhanced

FRENCH INSTITUTES OF TECHNOLOGY

5. Application in an engineering process

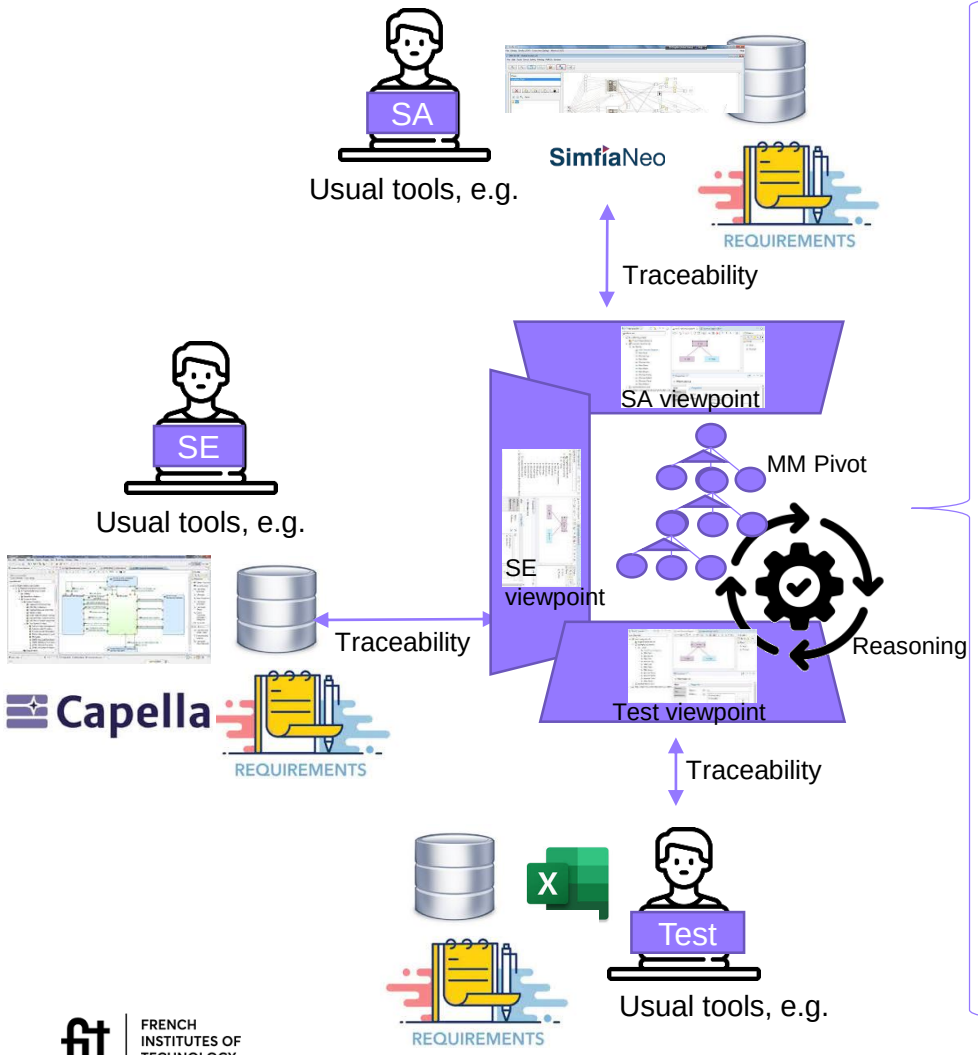


6. Goal: a new abstraction to facilitate the detection of incompatibilities



How to capture in this new abstraction?
-> tooling perspective addressed by internship

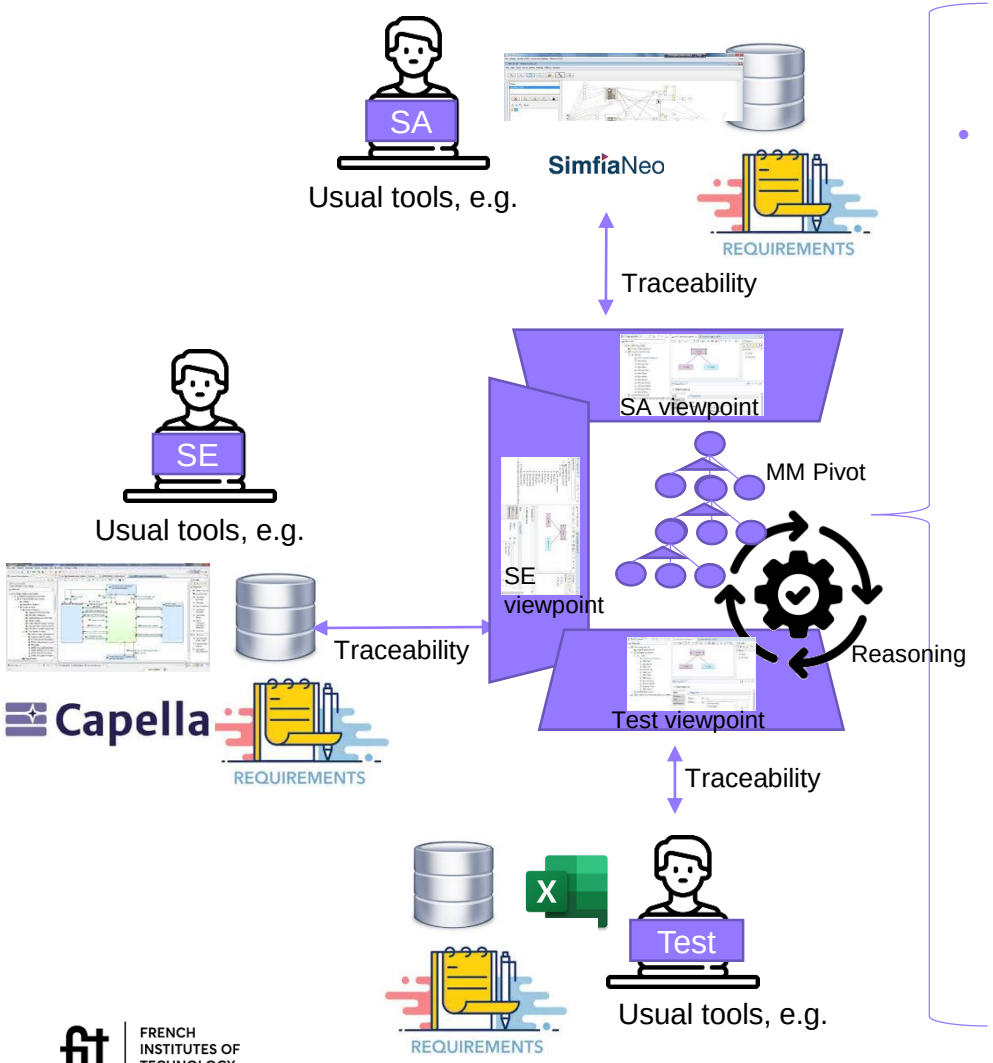
6. Tooling, what are the issues ?



Tooling, what are the problematics?

- Provide a consistent high-level view of the system
- Allow different views of the system for different teams
- Ensure all views are synchronized
- Detect and resolve incompatibilities in the overall view
- Extract a model from existing work (engineering artefacts, in Capella, Simfia, …)

6. progress report on the tooling



How to capture this new abstraction?

- **Provide a new tool** to engineers to capture this knowledge
  - A **new viewpoint** (i.e. language) for each concern, e.g.
    - GSN for SA,
    - KAOS for Test Eng.,
    - Generic Feature Model Notation for SE.
  - **A pivot MM to align these viewpoints**
    - Generic (abstract) language for compatibility
      - Operator needed: variability, compatibility/incompatibility
    - Synchronisation between viewpoints
      - Logical
      - Language (abstract / concrete syntax, semantics)
  - **Reasoning capability** to assist the engineers
    - In *detecting potential inconsistencies*,
    - In *extracting « features »* to feed this new abstraction from existing tools.
  - **Traceability/link support** to existing tools

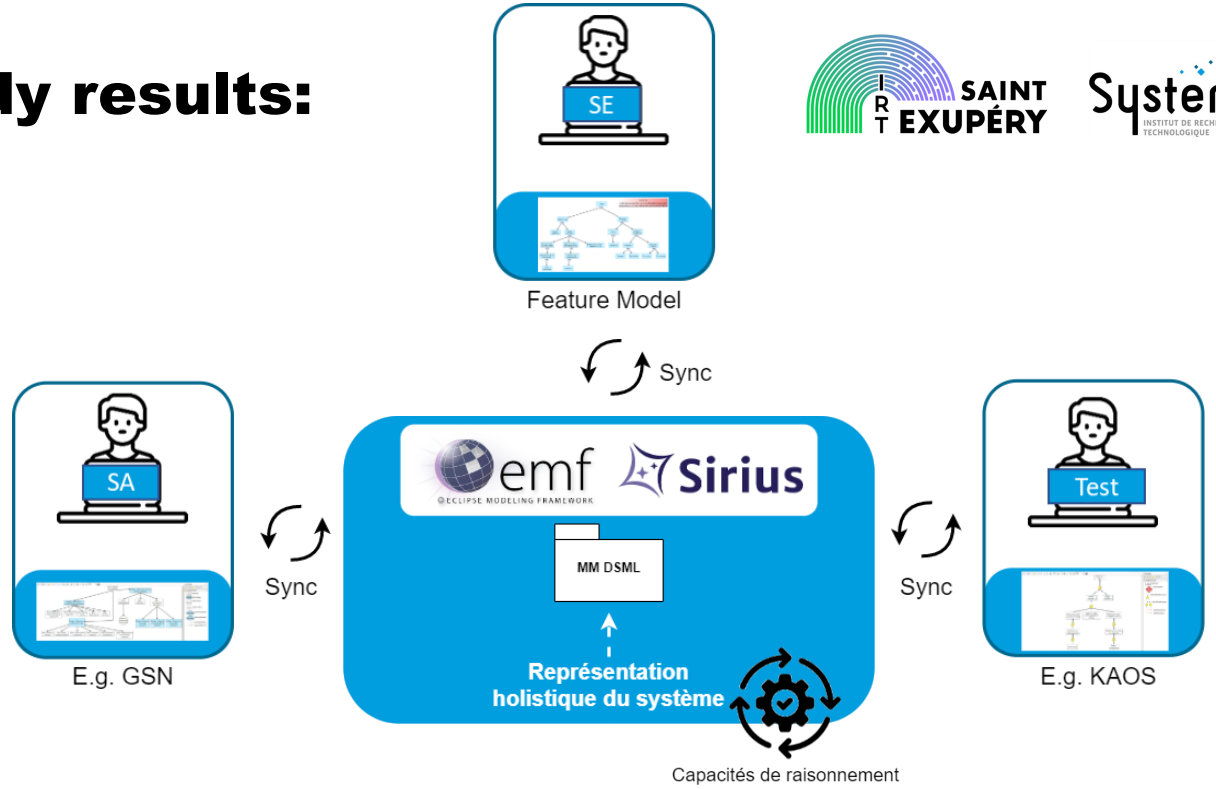
WIP

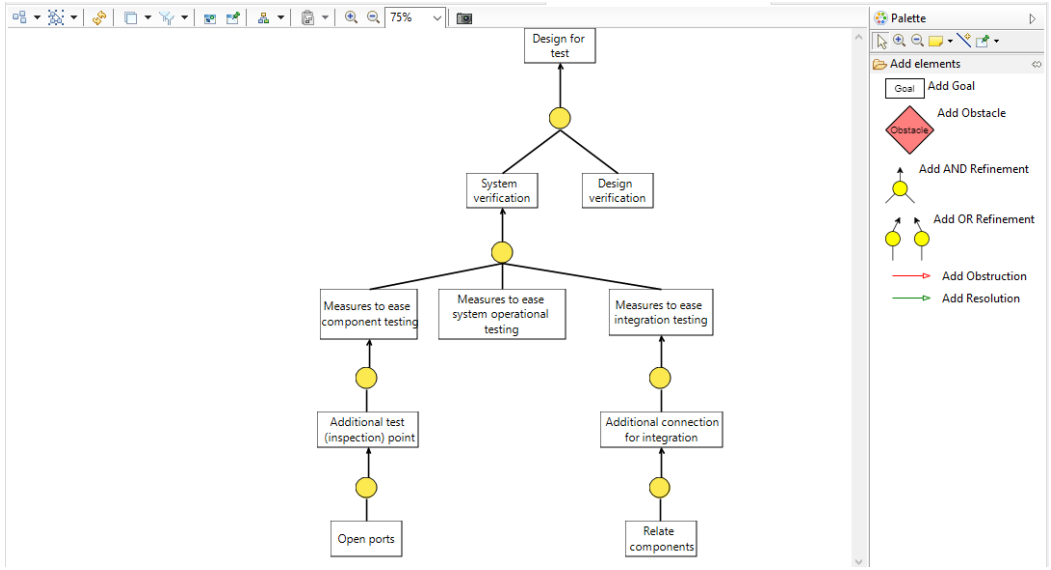
FRENCH INSTITUTES OF TECHNOLOGY

6. Tooling Incompatibility Pre-study results:

- **What has been produced:**
  - A framework based on Eclipse to
    - Model the domain
    - Represent synchronized viewpoints (feature model, GSN, KAOS),
    - Analyze incompatibilities between models, with a propositional logic (SAT solver).
  - An application on
    - An academic case study (e-shop)
    - A partial representation of 3 engineering domains (SE, SA, and Test)

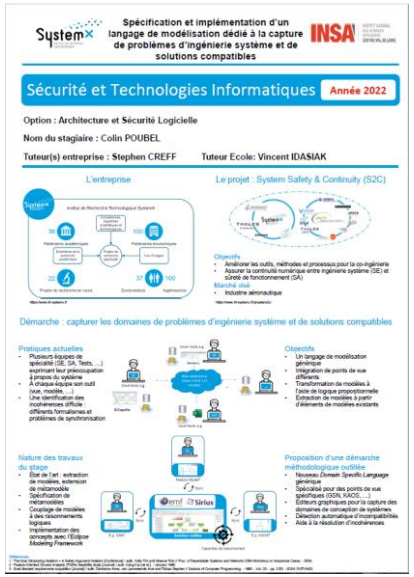
- **What is missing:**
  - Connection to engineering domain tools
    - And more over Feature extraction from existing domain artifacts
  - Application to the AIDA case study





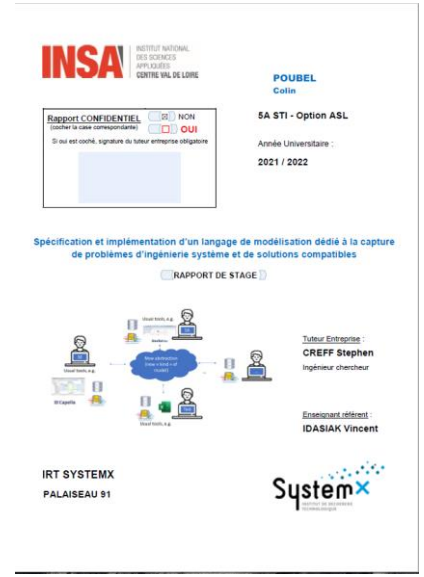
7. Internship result materials

- Poster



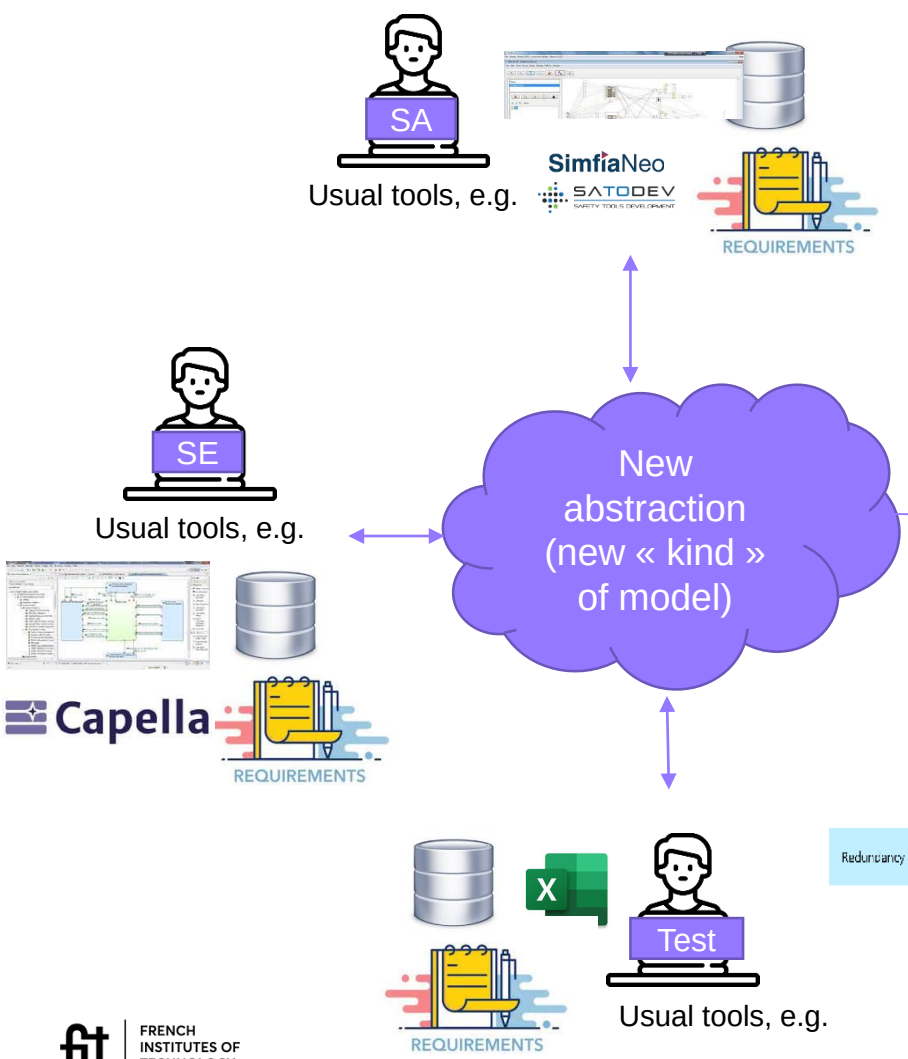
- Presentation support

- Report



FRENCH INSTITUTES OF TECHNOLOGY

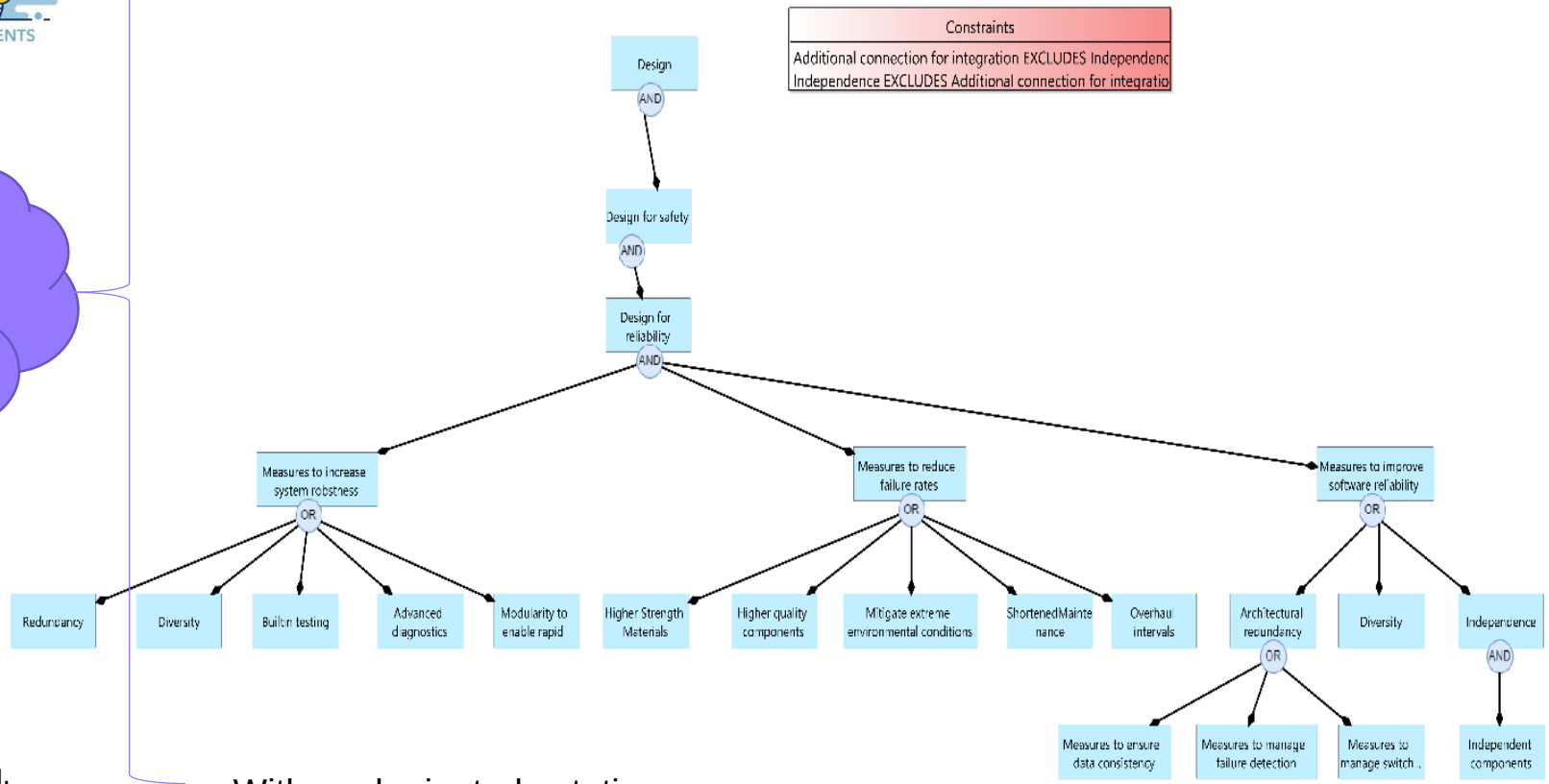
8. Example of domain models



What to capture in this new abstraction?

Domain specific intentions: « Design for X »

e.g. Design for Safety (feature model representation)

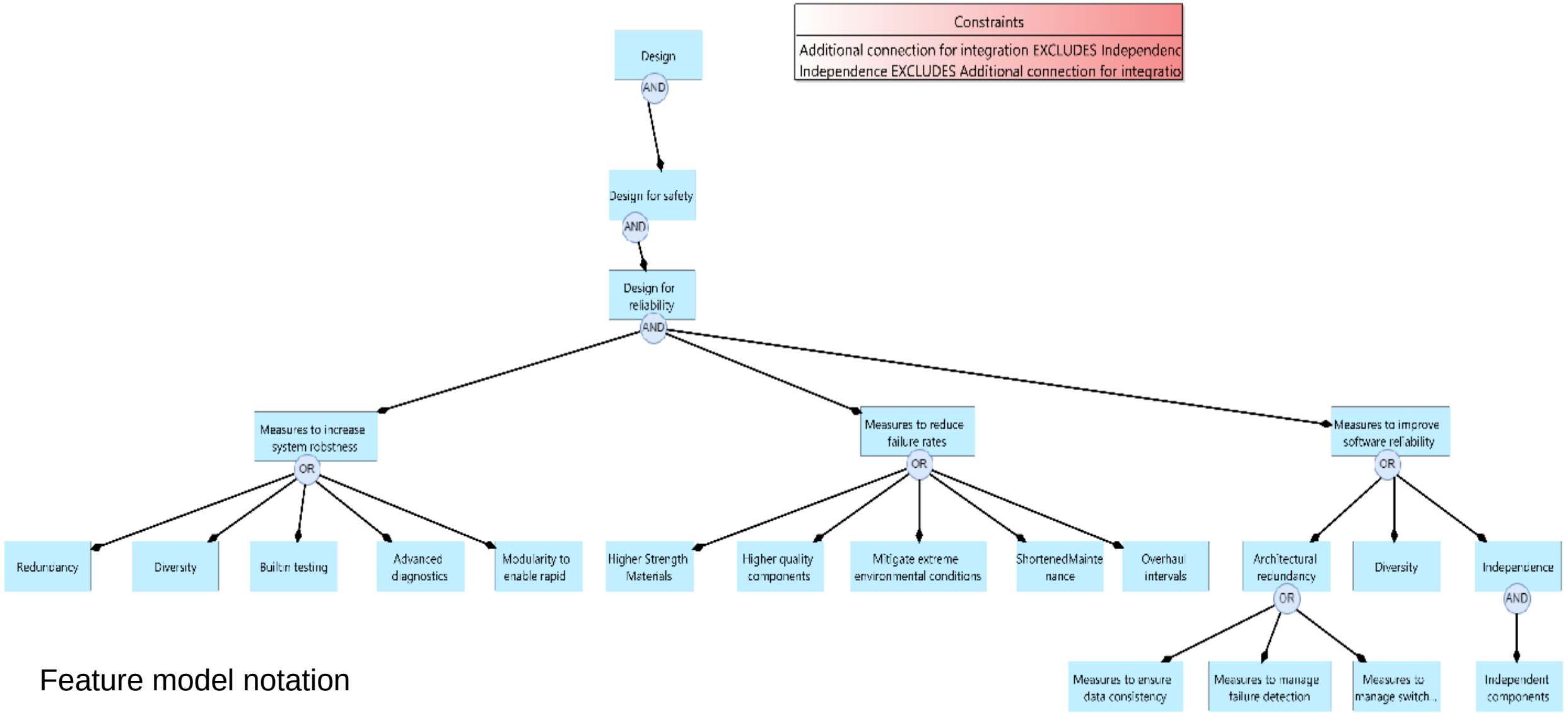


With goal oriented notations:

- GSN (Goal Structuring Notation), or KAOS for example

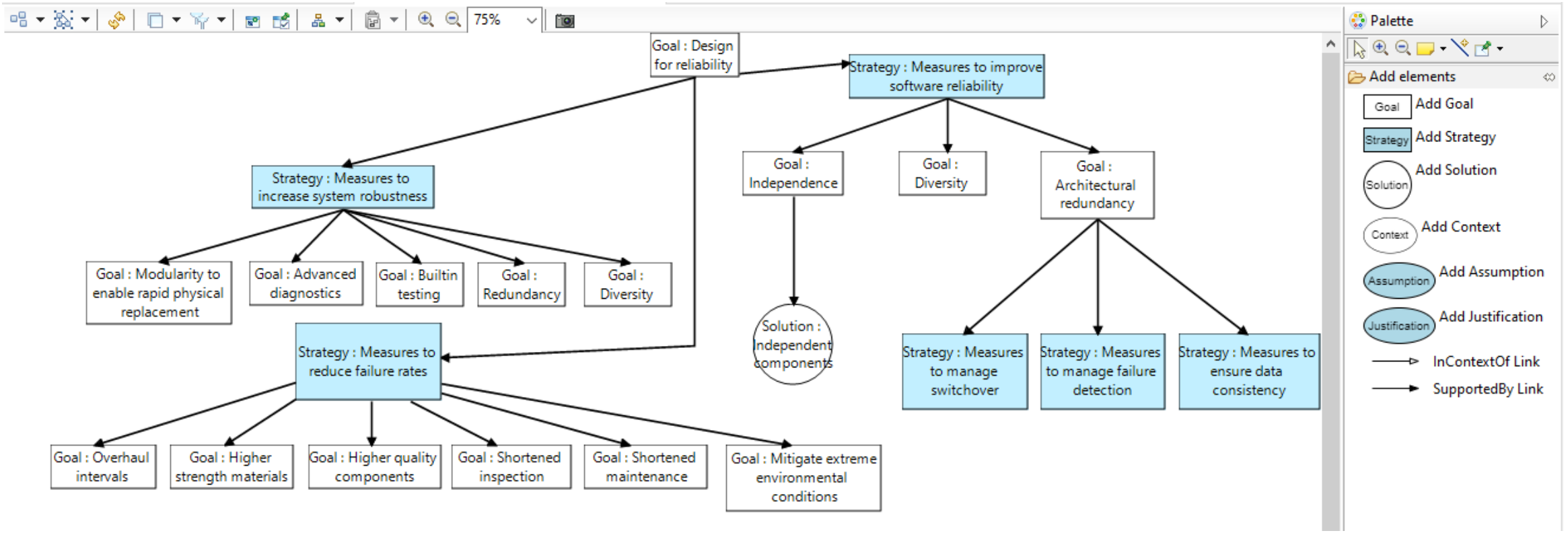
FRENCH INSTITUTES OF TECHNOLOGY

8. Example of domain models



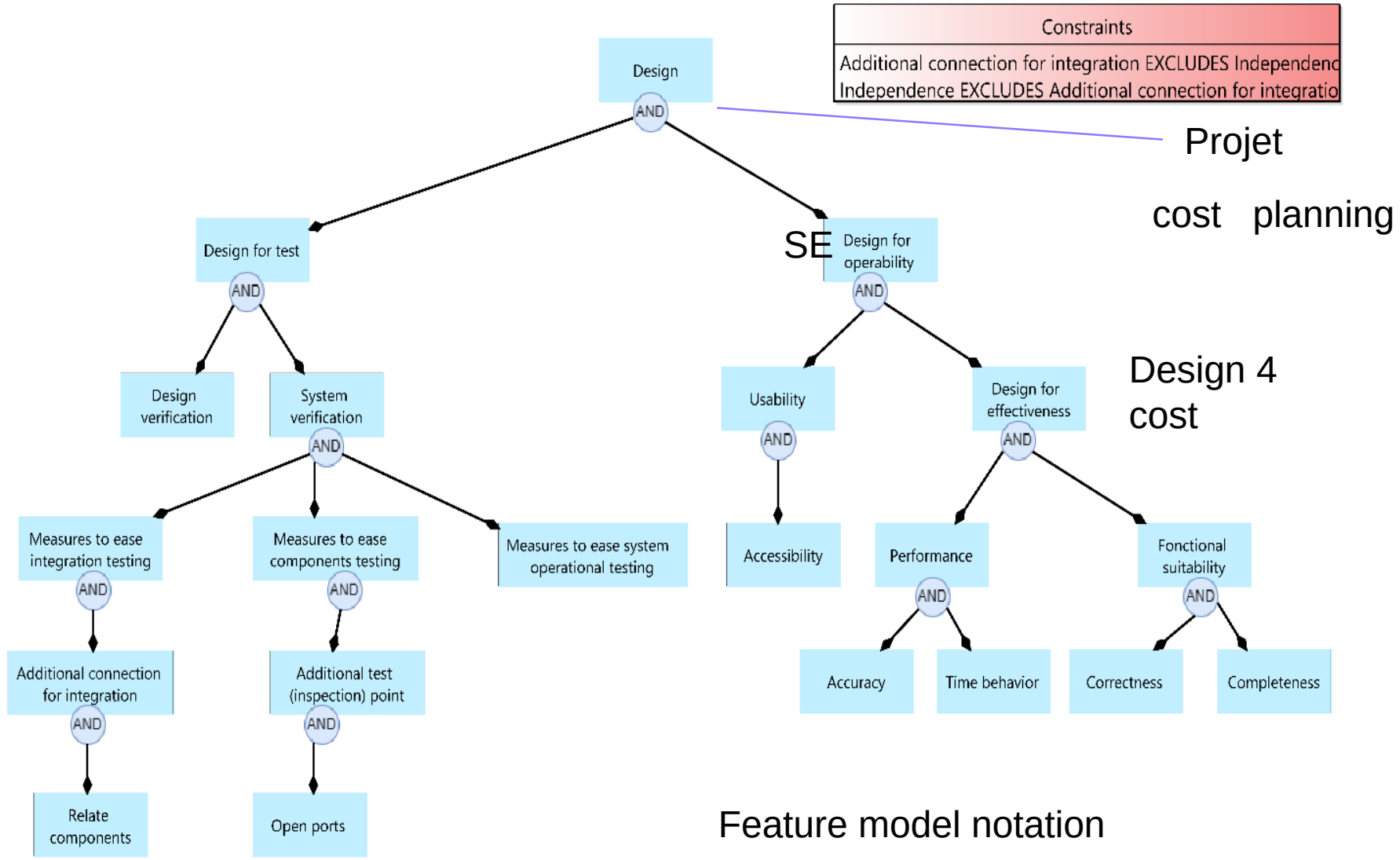
Feature model notation

8. Example of domain models



GSN notation

8. Example of domain models



Projet

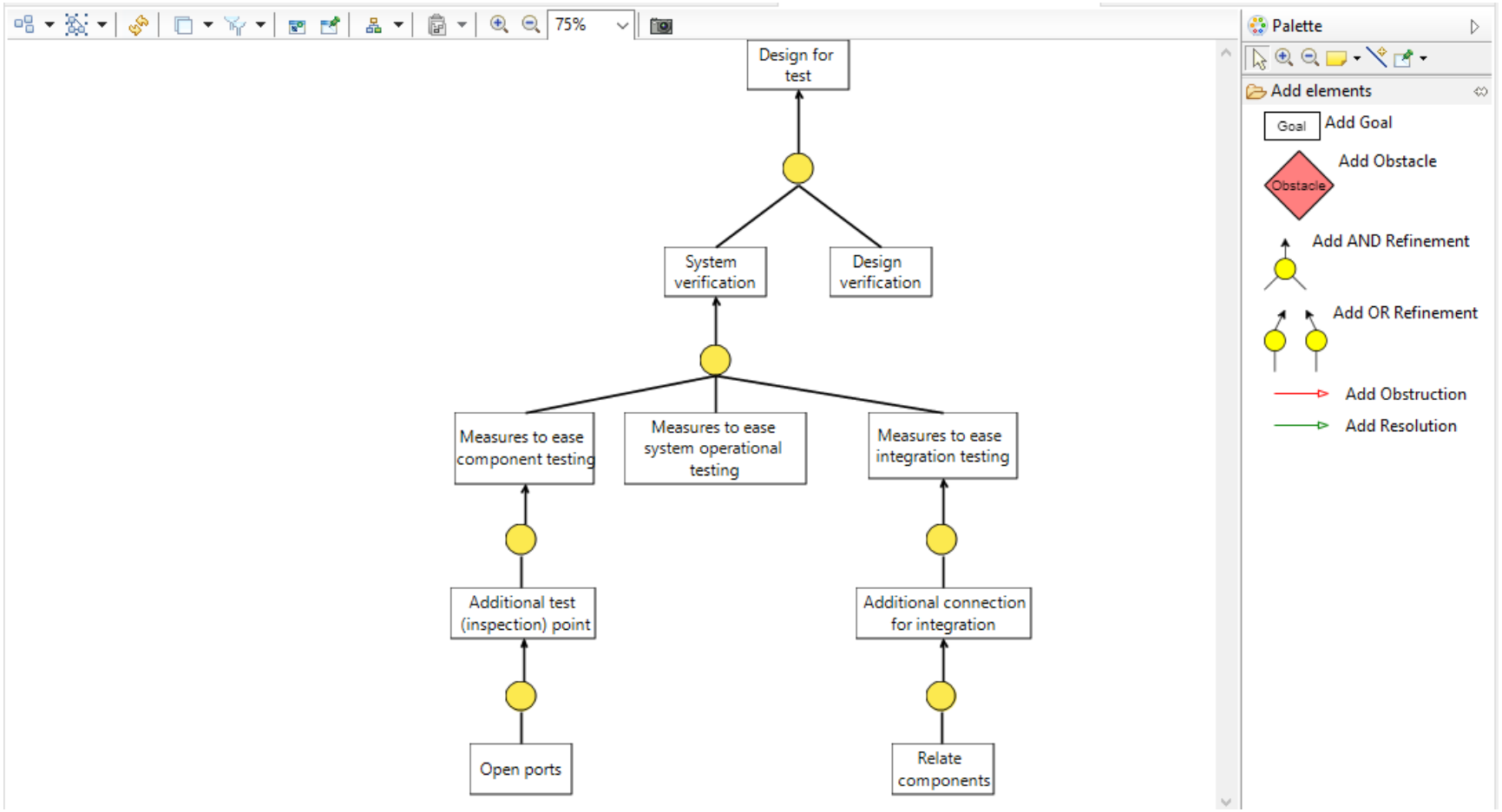
cost planning

SE

Design 4 cost

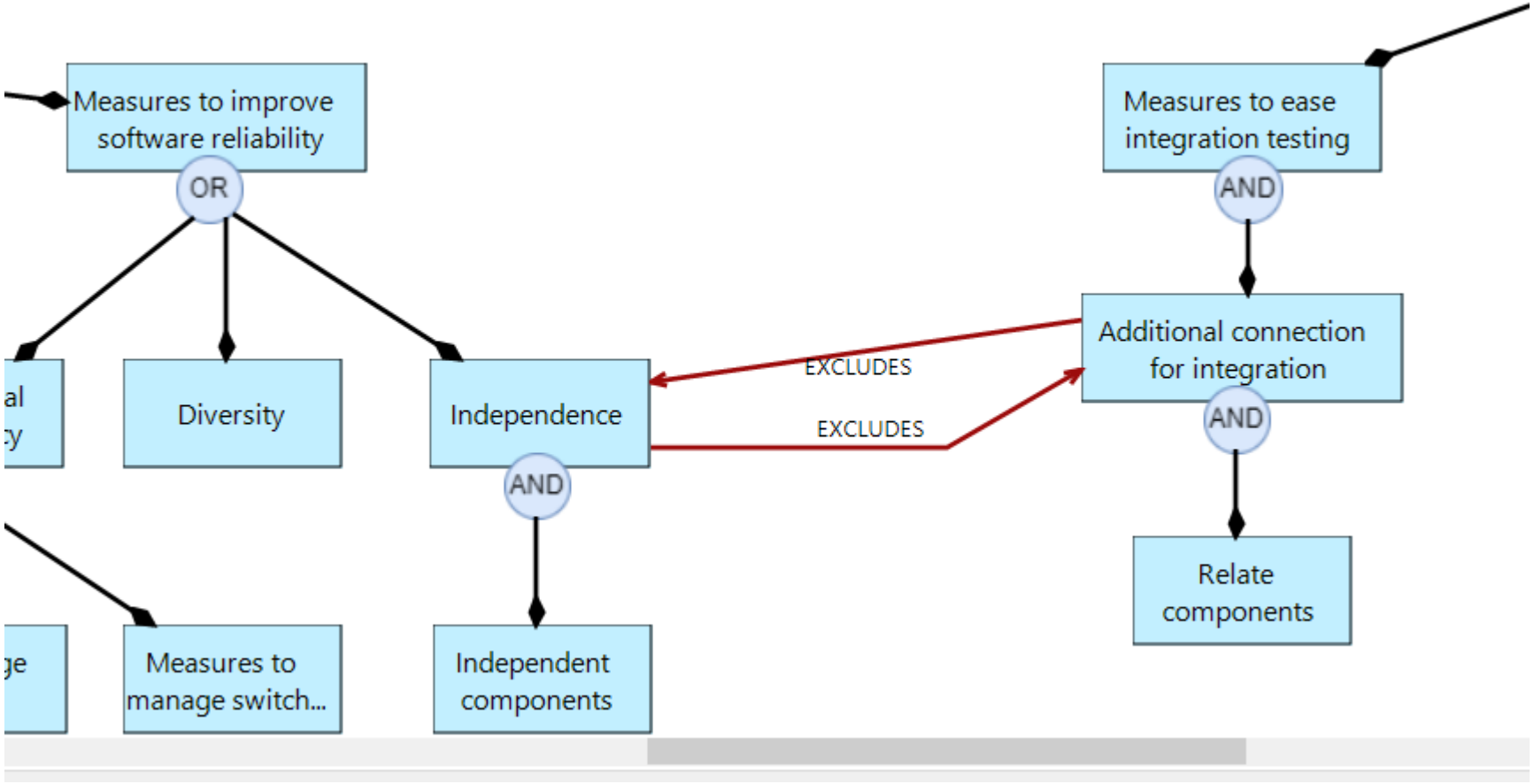
Feature model notation

8. Example of domain models



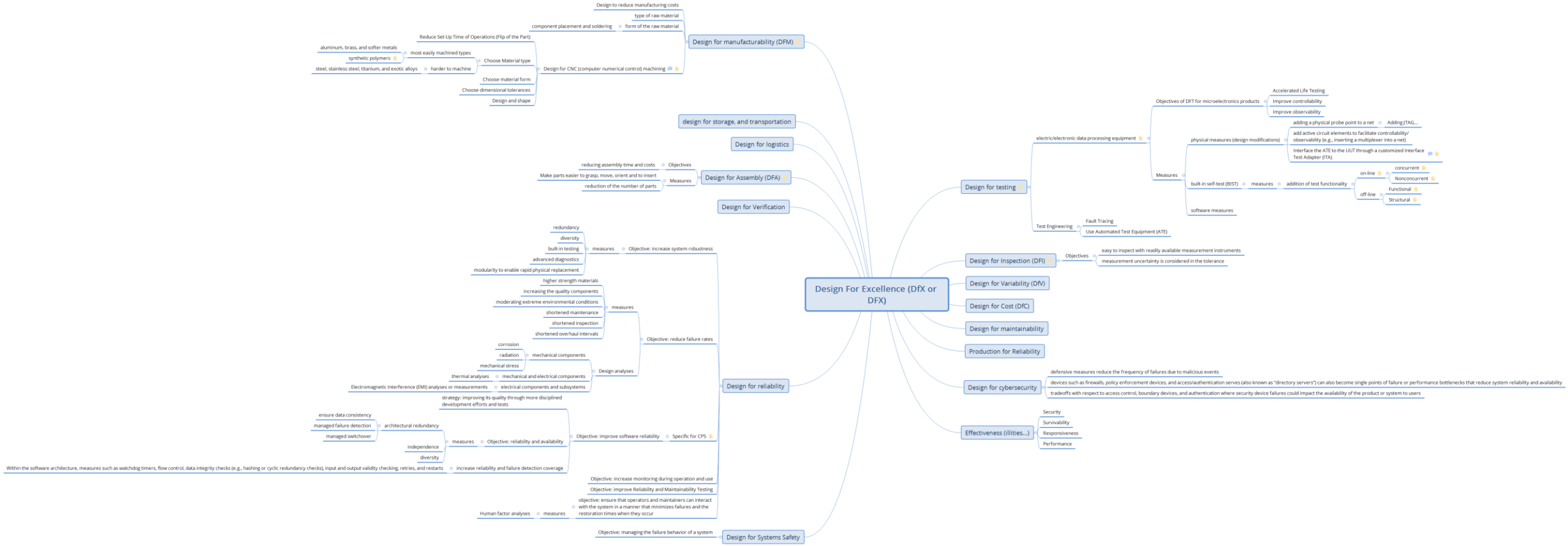
KAOS notation

8. Example of domain models



Feature model notation

8. Example of domain models



Example of Design for X domain model