



“Towards a Smart Feature Model Evolution”

Olfa Ferchichi
Lamia Labeled Jilani

Raoudha Beltaifa
Raúl Mazo

RIADI Lab, ENSI, Tunis, Tunisia

Lab-STICC, ENSTA Bretagne, Brest, France,

olfaferchi@ yahoo.fr
lamia.labeled@isg.rnu.tn

raoudha.beltaifa@ensi.rnu.tn
raul.mazo@ensta-bretagne.fr



Who am I?

2

Olfa FERCHICHI is a doctoral student from ENSI, the national school of computer sciences in the RIADI lab under the direction of Mrs. Lamia Labeled Jilani and the co-supervision of Mrs. Raoudha Beltaifa Ismail.

I work in the field of:
Software Product Line Engineering (SPLE)



My paper is entitled
”Towards a Smart Feature
Model Evolution”



OUTLINE

3

- INTRODUCTION
- VARIABILITY MODELING IN SOFTWARE PRODUCT LINE
- VARIABILITY MODELING WITH FEATURE MODELS EVOLUTION OF FEATURE MODELS
- MODEL DRIVEN APPROACH FOR EVO-FM CONSTRUCTION
- EVO-FM: EXTENDED FM META-MODEL
- MODEL TRANSFORMATION
- IMPLEMENTATION
- CONCLUSION AND FUTURE WORK

INTRODUCTION-1

4

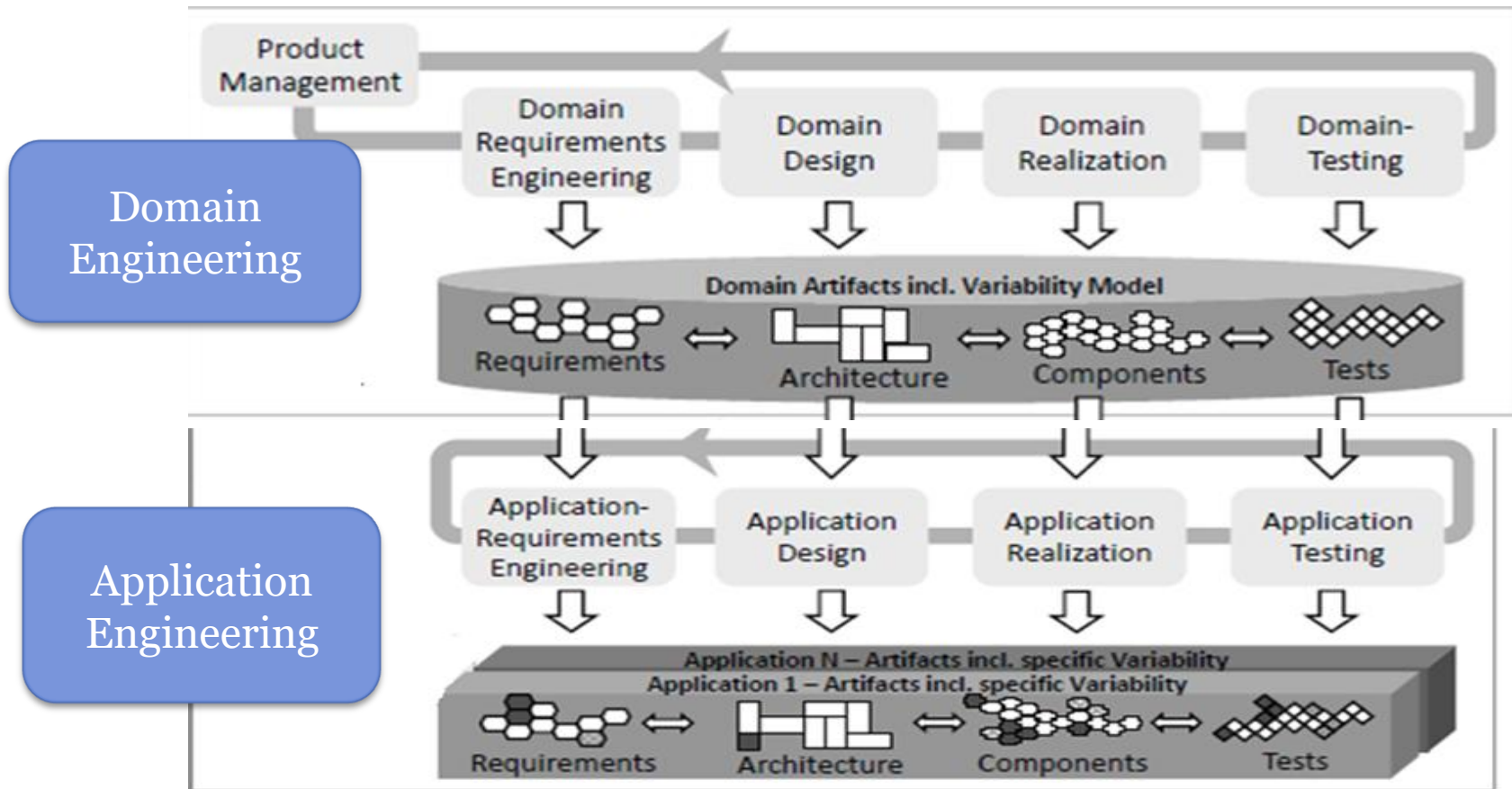
A Software Product Line (SPL) : set of systems

- Share a common, managed set of features;
- Satisfy the specific needs for particular domain;
- Applications developed in a controlled manner from a common set of reusable elements.

INTRODUCTION-2

5

SOFTWARE PRODUCT LINE ENGINEERING (SPLE)



VARIABILITY MODELING IN SOFTWARE PRODUCT LINE-1

6

- Modeling variability is a Key Activity in the SPLE approach it:
 - ✓ Comes to identify the features of the SPL
 - ✓ Used to identify the relationships and constraints that exist between these FEATURES
 - ✓ Reduces the complexity of managing variability during the implementation phase

- There are several techniques for modeling variability such as Feature Models

VARIABILITY MODELING WITH FEATURE MODELS-2

7

- Domain analysis activity creates a Feature Model (FM) as a core asset to model domain requirements;
- A FM represents common and variable features of Product Line
- It consists in a compact representation of all possible products of a Line.

EVOLUTION OF FEATURE MODELS

9

- Feature model evolution refers to the process of timely making changes in it for diverse reasons such as:
 - The apparent freedom of feature choices at the level of a tree hides the reality of "inter-feature" dependencies;
 - Features such as quality attributes are rarely specified;
 - Evolutions of features in time and in space are not expressed;
 - Distinguishing the nature of each feature; for specifying software concerns ;
 - Lack of distinction between behavioral and structural features.

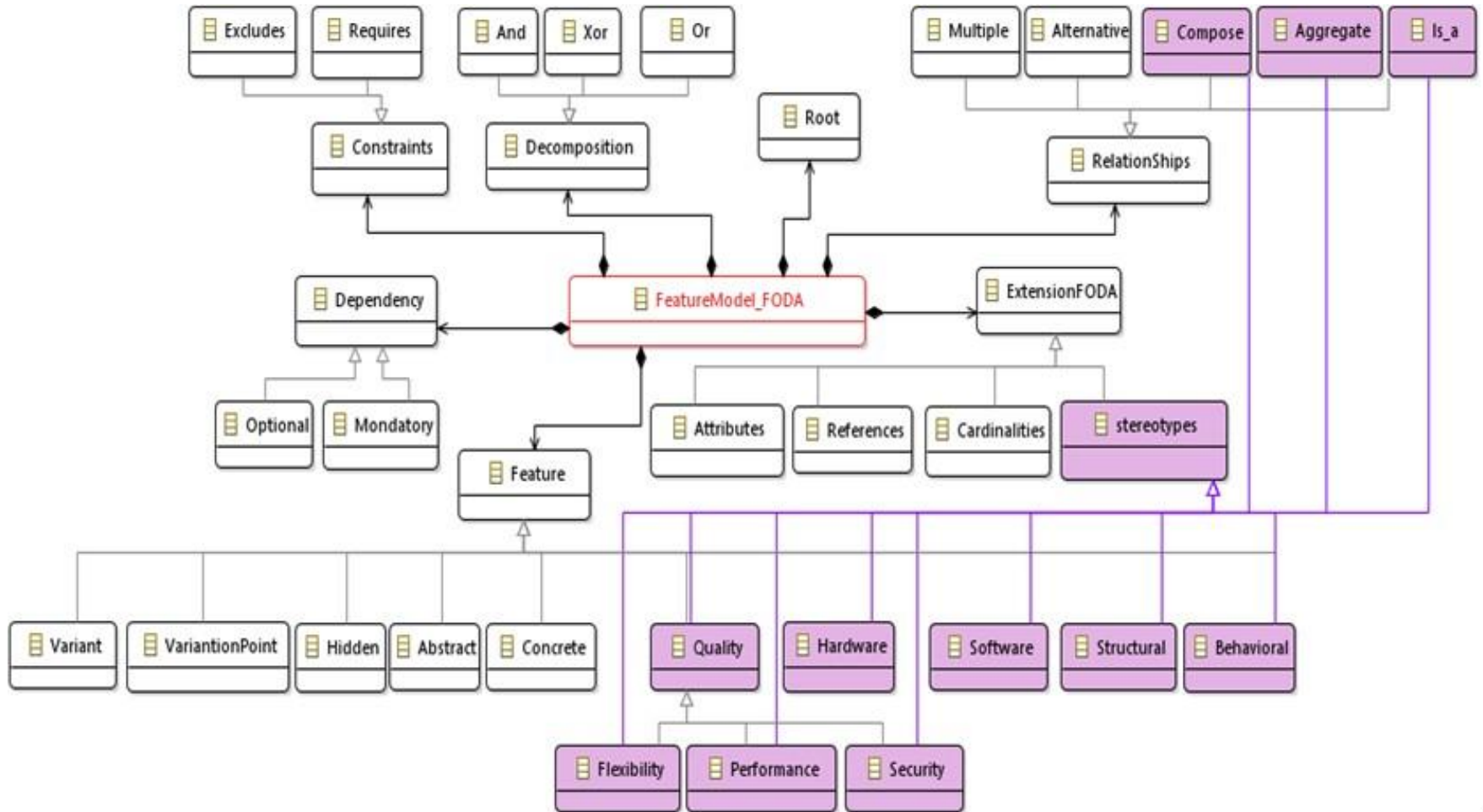
MODEL DRIVEN APPROACH FOR EVO-FM CONSTRUCTION

10

- Feature models evolution EVO-FM is proposed for:
 - a. Enriching the FM semantic with (Applies a FM that represents supprimer) knowledge and information for its evolution;
 - b. Transforming the EVO-FM metamodel to an ontology metamodel by means of model-driven transformation
 - c. Allowing intelligent reasoning by inferring knowledge via the ontology

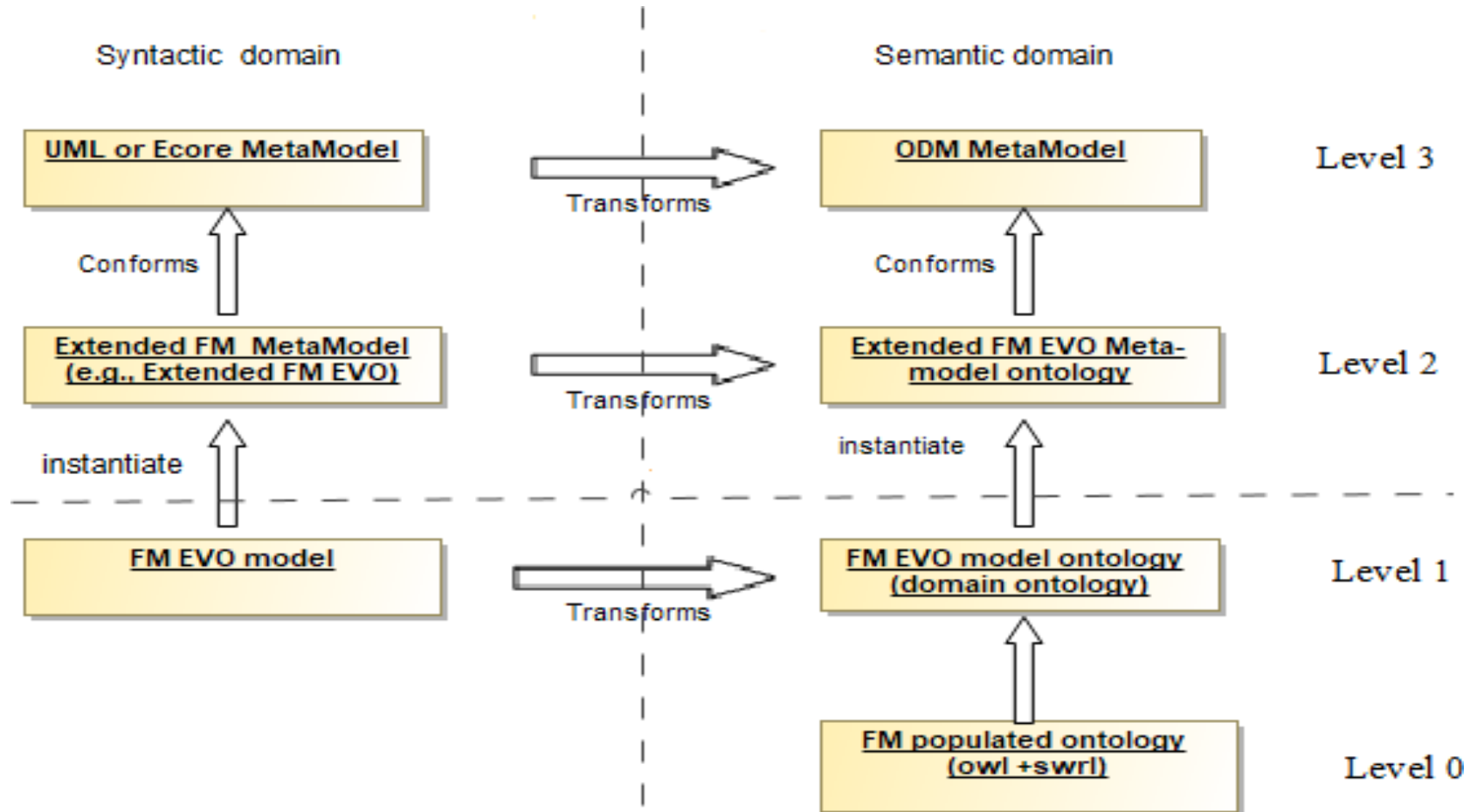
EVO-FM: EXTENDED FM META-MODEL

11



MODEL TRANSFORMATION-1

12



MODEL TRANSFORMATION-2

13

```
|- @path MM=/ATL_FM/Papier.ecore
-- @path MM1=file:/C:/Users/HP/Desktop/OCTA_2019/onto_FM/Ontology_FM.owl

module BD;

create OUT : MM1 from IN : MM;

rule Concept_Feature {
  from f : Feature!Feature
  to out : Feature!Feature (
    name <- f.name,
    FeatureID <- f.FeatureID()
  )}

rule Concept_Hardware_Feature{
  from b : Feature!Feature
  to out : Hardware_Feature!Hardware_Feature ()}

rule Concept_Software_Feature{
  from b : Feature!Feature
  to out : Software_Feature!Software_Feature ()}
```

IMPLEMENTATION-1

-Eclipse family of integrated development environments:

- ❑ The Eclipse Modeling Framework (EMF) [6]: modeling and code generation framework;

- ❑ Textual Modeling Framework (TMF) [26];

- ❑ Xtext is a framework for development of programming languages and domain-specific languages;

- ❑ FeatureIDE [24] is an Eclipse-based IDE that supports all phases of feature-oriented software development for the development of SPLs;

- ❑ Feature Model Ontology using Protégé OWL [22] and SWRL [18] to write rules.

IMPLEMENTATION-2

15



Properties Individuals Forms → SWRL Rules

Expression

- $Quality(?x) \wedge Behavioral(?y) \rightarrow swrlb:notEqual(?x, ?y)$
- $Mandatory(?x, ?y) \wedge Optional(?x, ?z) \rightarrow swrlb:notEqual(?y, ?z)$
- $Hardware(?x) \wedge Software(?y) \rightarrow swrlb:notEqual(?x, ?y)$
- $Structural(?x) \wedge Behavioral(?y) \rightarrow swrlb:notEqual(?x, ?y)$
- $Structural(?x) \wedge Quality(?y) \rightarrow swrlb:notEqual(?x, ?y)$
- $VariationPoint(?x) \wedge Variant(?y) \wedge Mandatory(?x, ?y) \wedge NotSelected(?x) \rightarrow NotSelected(?y)$
- $VariationPoint(?x) \wedge Variant(?y) \wedge Optional(?x, ?y) \wedge Selected(?y) \rightarrow Selected(?x)$

CONCLUSION

16

- EVO-FM, is an extension of the FODA metamodel
- Enriches it with knowledge and information to support the evolution;
- Enrich the feature models with quality and semantic features;
- Add support for new types of feature relationships and extensions with stereotypes;
- Add semantic features in the form of stereotypes such as «software», «hardware», «structural» and «Quality»;
- Import a textual specification into grammars using the Xtext framework to process it and transform it into XML;

FUTURE WORK

17

- Apply our model driven approach to different system families ;
- Validate the evolution rules that are behind the EVO-FM ontology
- Enable a smarter evolution of feature models by using different versions of EVO-FM feature model .
 - Used as a learning base of a learning algorithm
 - EVO-FM, a new version can be predicted as being a new FM evolution version

Thank you for your attention

