First International Workshop on Requirements Engineering
on Software Product Line Engineering

Generating Feature Model from Creative Requirements using Model Driven Design

Fernando Wanderley
Universidade de Pernambuco
Recife - Brasil

Denis Silva da Silveira
Universidade Federal de Pernambuco
Recife - Brasil

João Araujo
Faculdade de Ciência e Tecnologia
Universidade Nova de Lisboa
Lisboa - Portugal

Maria Lencastre
Universidade de Pernambuco
Recife - Brasil

Before Starting…

My apologies for not being with you

Summary

- Brief presentation
- Introduction with the main motivation
- Transformation Approach
- Study Case
- Related Work
- Conclusion and Future Studies

Brief Presentation

- conclusive student of master's degree in Computer Engineering from the State University of Pernambuco.
- focus on research **(agile) requirements modeling**
- recent experiences (last 3 years) in deploying software development processes through the IBM Rational platform. (specifically in Analysis and Design by Rational Software Modeling Platform).
- And my apologies for possible mistakes in English.

Introduction

- In SPL development, domain analysis plays a central role where the relevant features are identified.
- Feature-Oriented Domain Analysis is a method which uses a feature model to specify variabilities and commonalities of an SPL
- Activities related to the Domain Analysis process (managing commonalities and variabilities, with users visibility and relevant stakeholders) in most cases is not simple to represent.

Introduction

- Do not seem to be a simple and easy activity, since to represent these analyzes in modeling domain tools with a **certain degree of formality**…
- that **requires a technical knowledge** that domain experts do not always have it prior to use

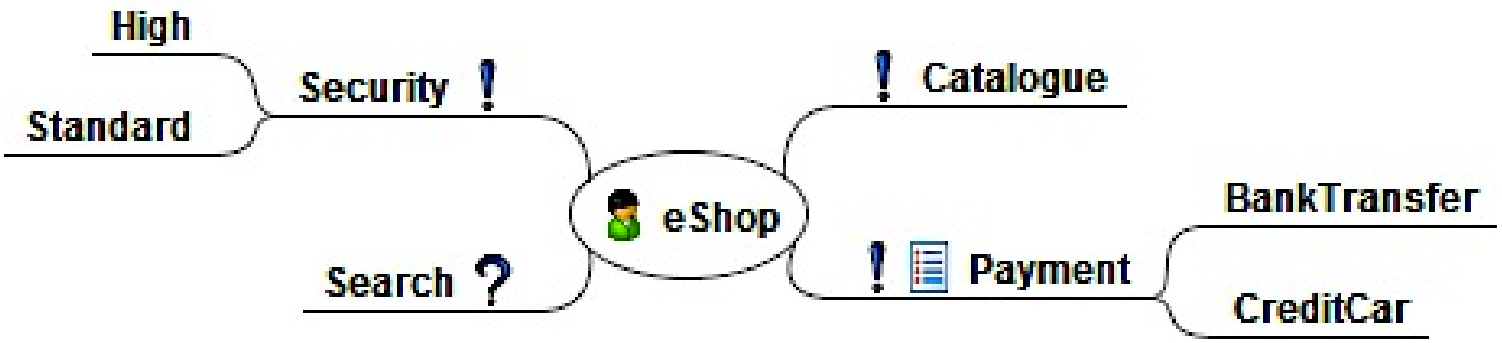
Motivation

- **Creative requirements techniques** have been suggested to facilitate the elicitation activity by **filling the gap the communications problems** between domain experts and software engineers, making the domain analysis more agile…
- So this paper seeks to set out the use of a creative and agile technique for modeling requirements by means of **mind maps** for cognitive and effective support when building feature models.

Mind Map

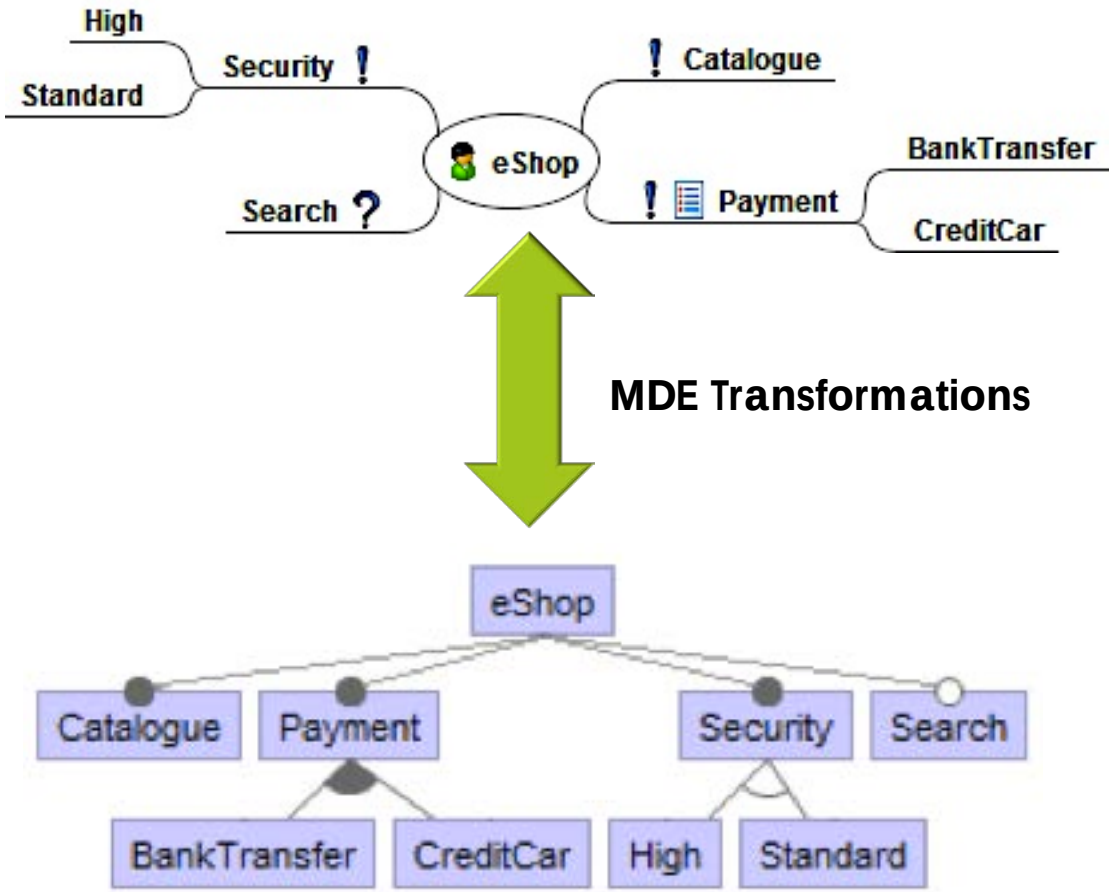
- A mind map is a diagram used to connect words, ideas and concepts to a central idea or concept; it is used to view, classify, organize concepts, and to generate new ideas (Buzan, 2003).
- In this diagram, the elements are ranked intuitively in accordance **with the importance of the concepts related to a domain**, which are then organized into groupings, branches or areas.

Mind Map

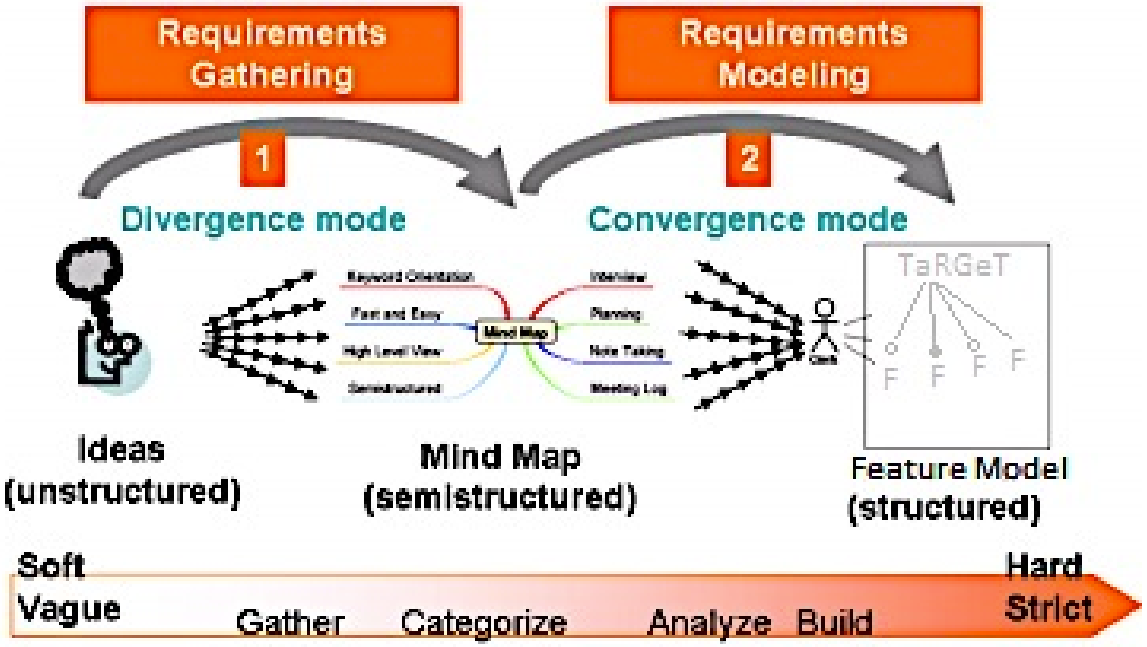


(adapted from Czarnecki, 2006)

Central Strategy



The Mind Map Role

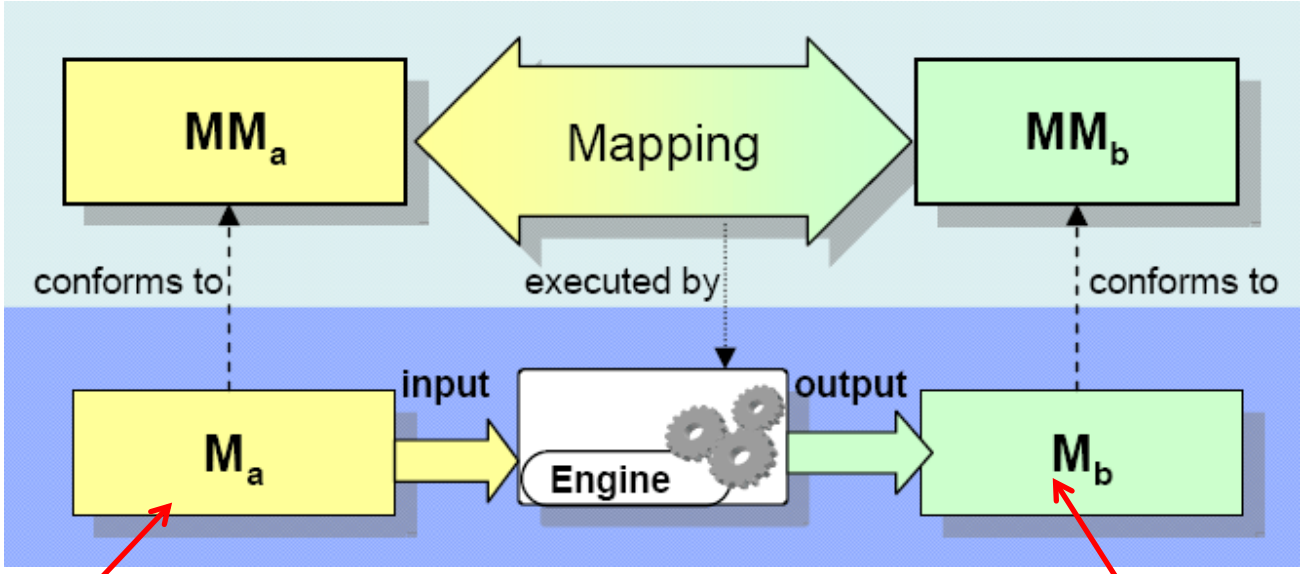


(adapted from Hiranabe, 2008)

Feature Model

- Feature modeling was proposed as part of the FODA method (Kang, 1990), and ended up being applied in other business and domain segments and field (Czarnecki, 2005).
- The feature model presented in this article refers to the **extended FODA model based on cardinalities** such as the multiplicities defined in UML models, proposed by Czarnecki.

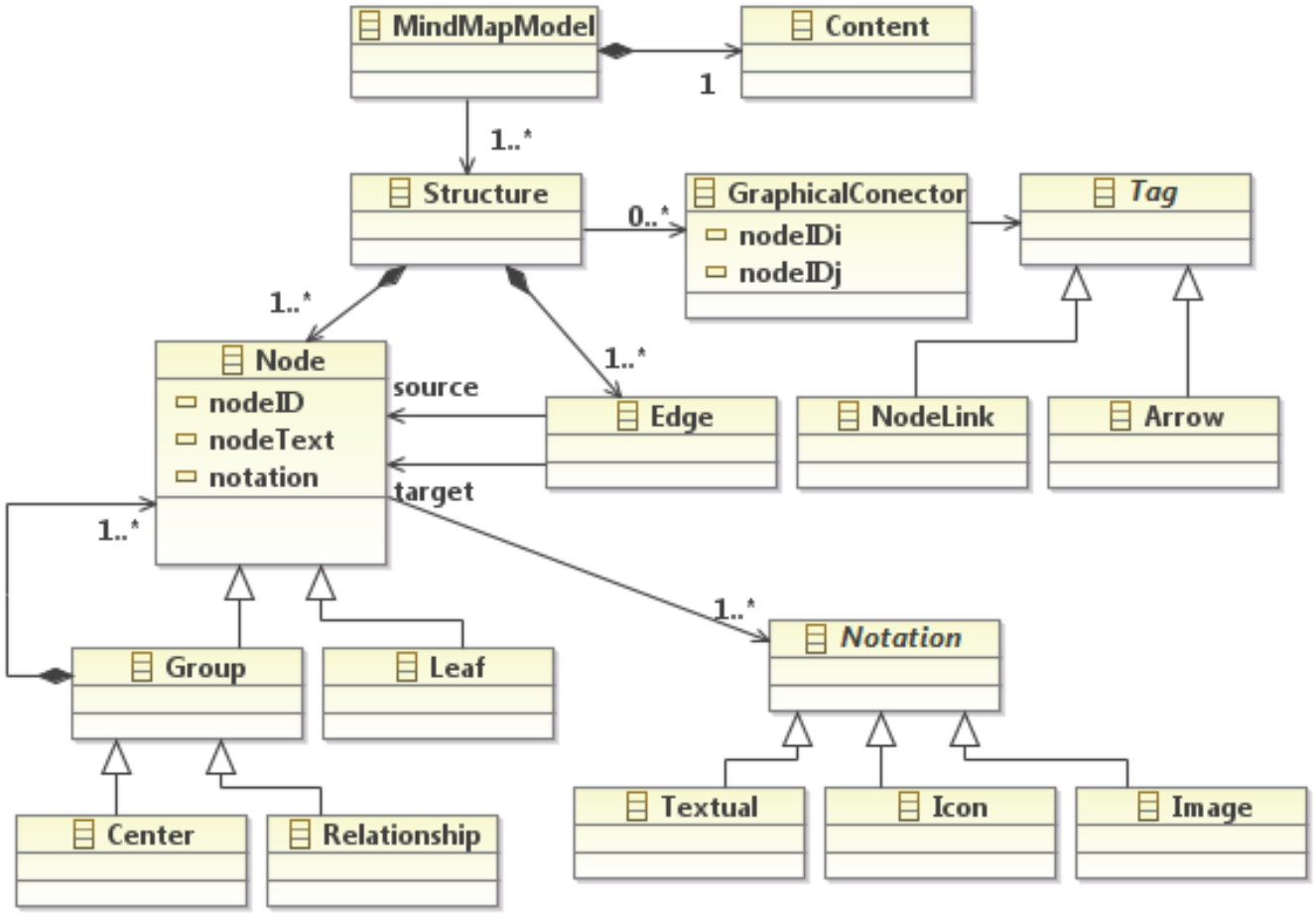
Model-Driven Engineering



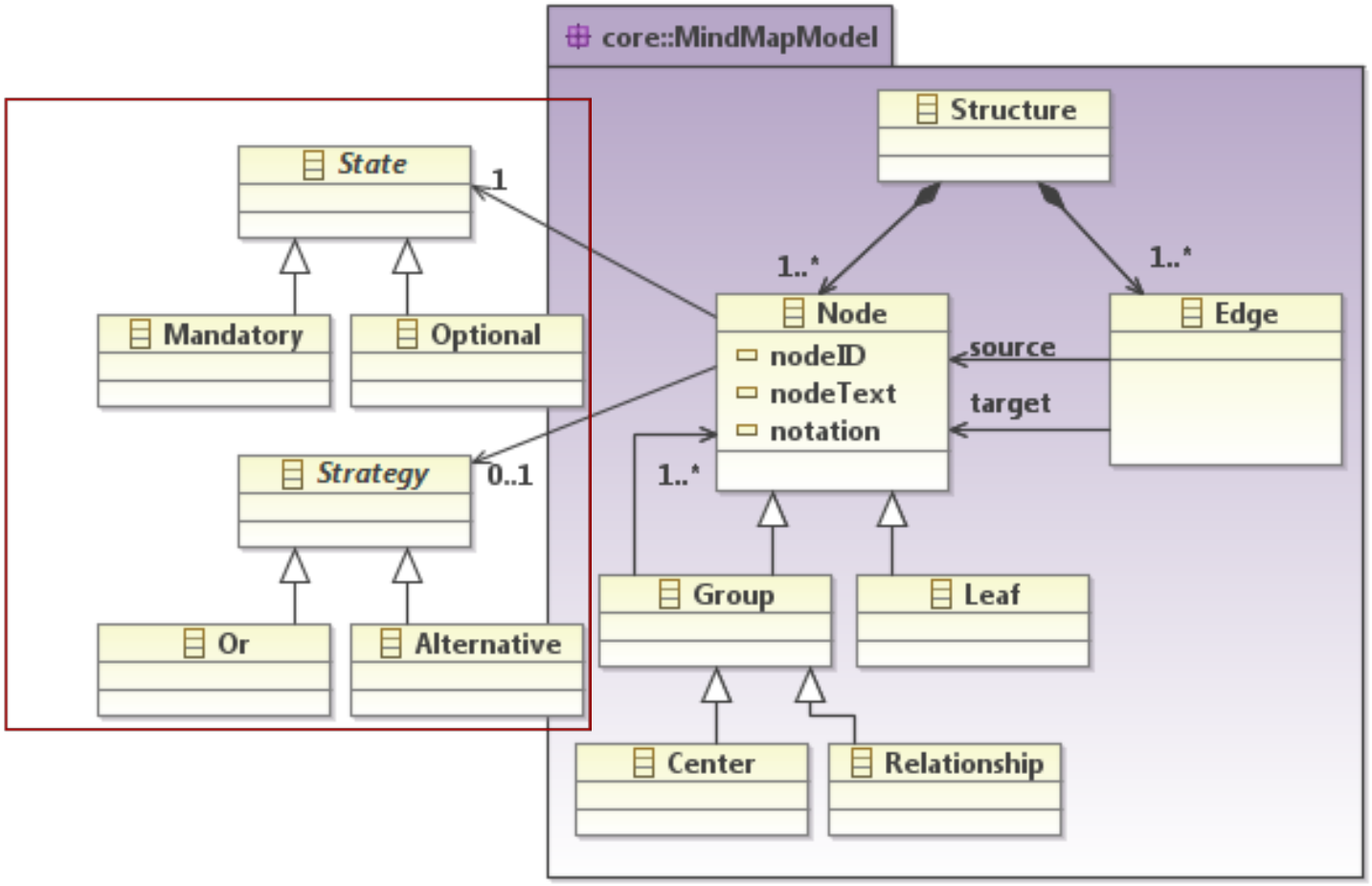
Mind Map Model

Feature Model

Mind Map Metamodel



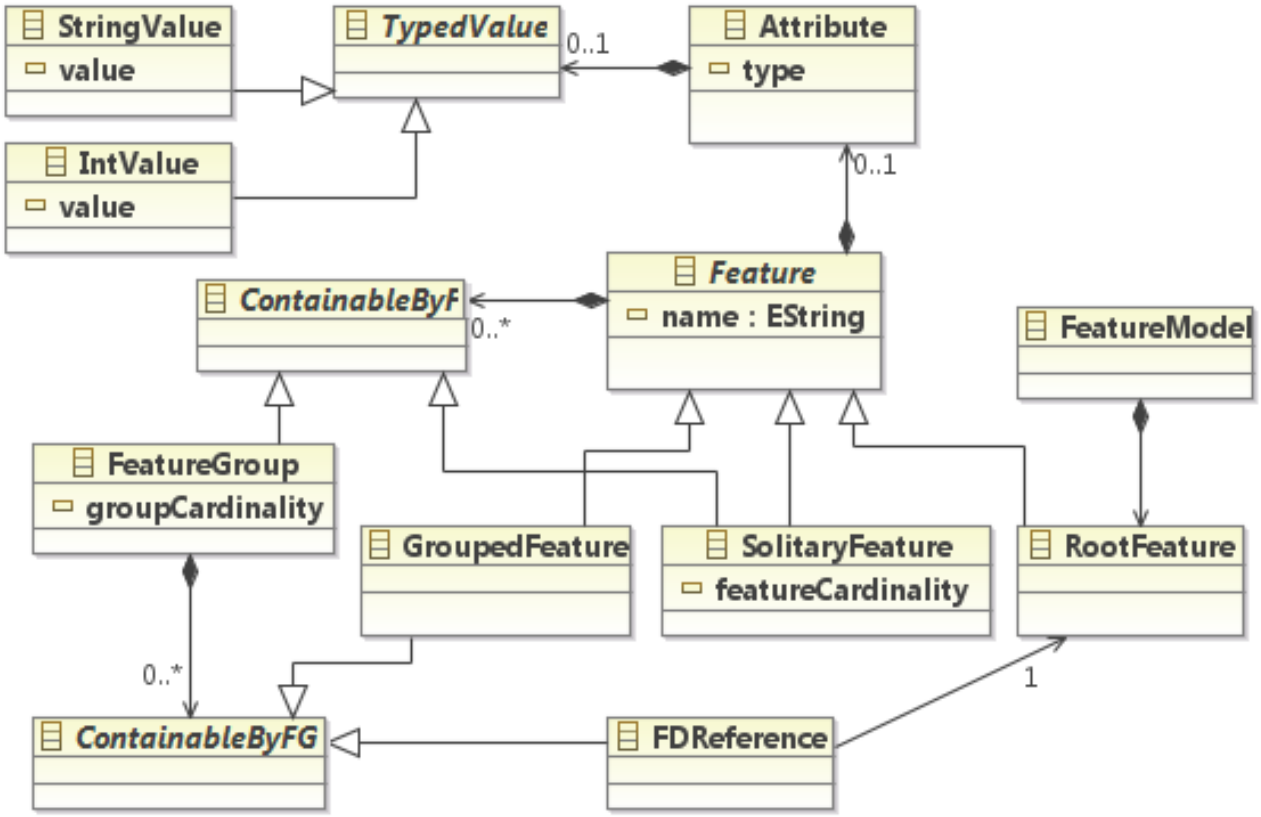
Mind Map Metamodel (Extensions)



Extension Notations

Relationship	Extension	Notation	Semantic	Category
Mandatory	Mandatory State	!	<<yes>>	Binary Relationship
Optional	Optional State	?	<<help>>	
Or	OrStrategy		<<list>>	Group Relationship
Alternative	Alternative Strategy		<<edit>>	

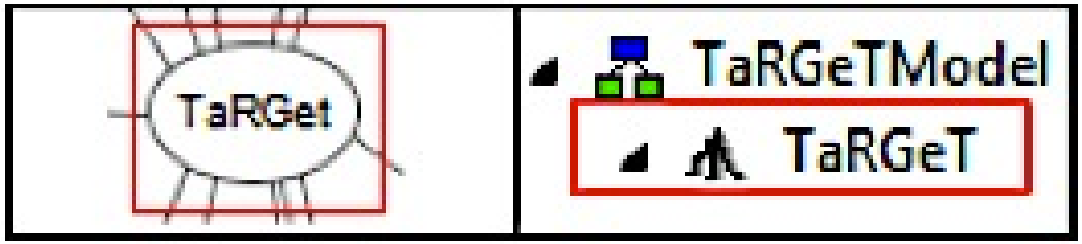
Feature Model Metamodel



Transformation Rules

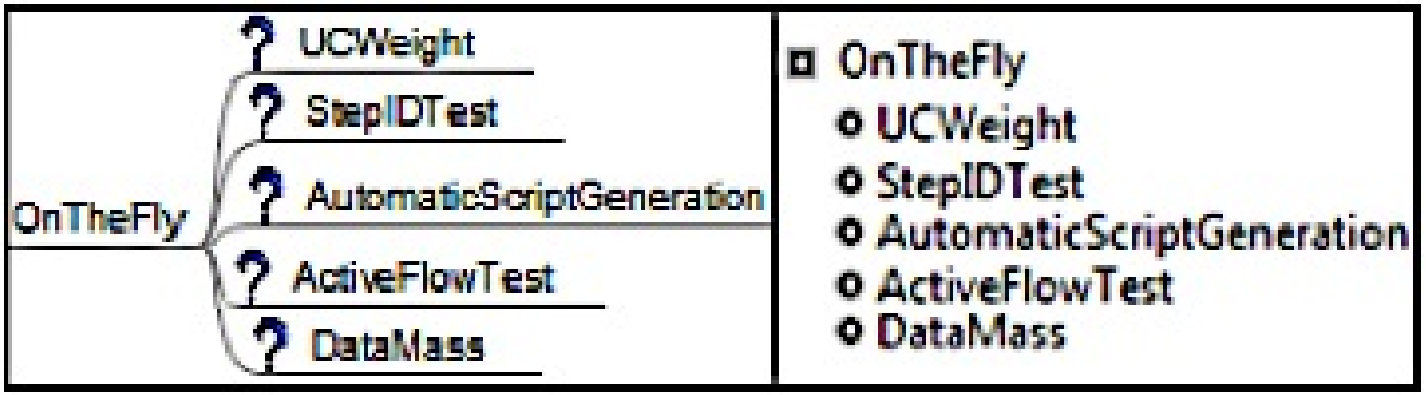
Rule 1

- This rule describes that the central node of a mind map will be related to a root feature of a feature model, with minimum cardinality equal to 1 and the maximum cardinality equal to 1.



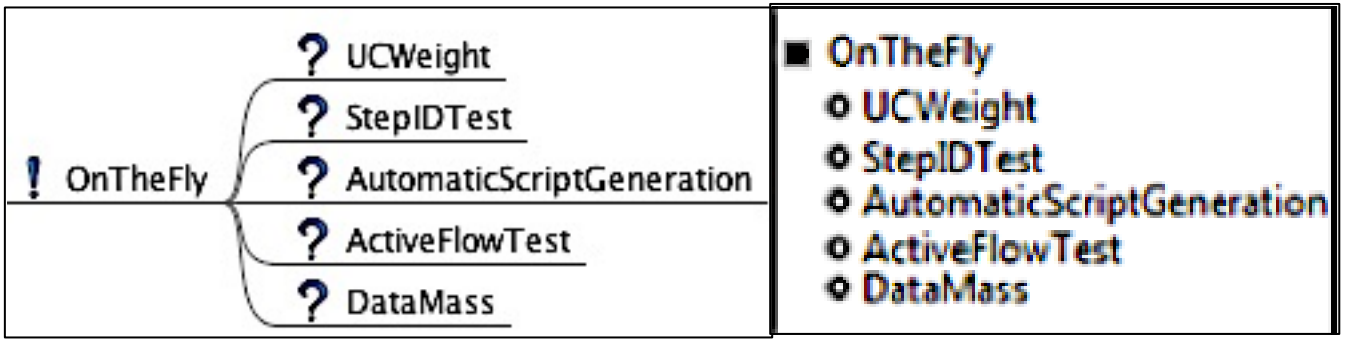
Rule 2

- This rule describes the transformation of a group node or a leaf node without the notation of a group relation, and is mapped to a **SolitaryFeature** with a minimum cardinality equals to 0 and a maximum equals to 1



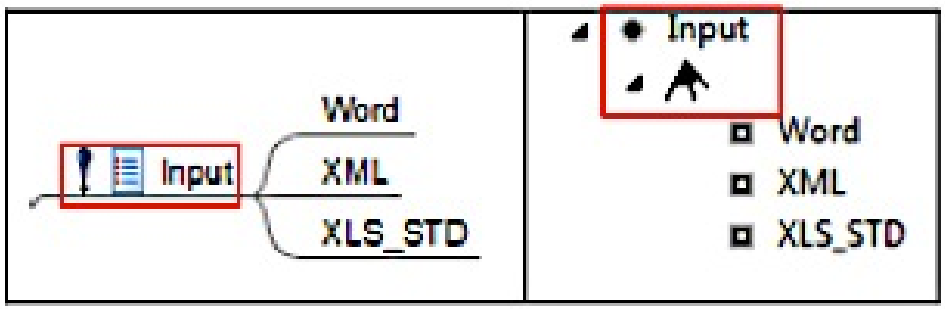
Rule 3

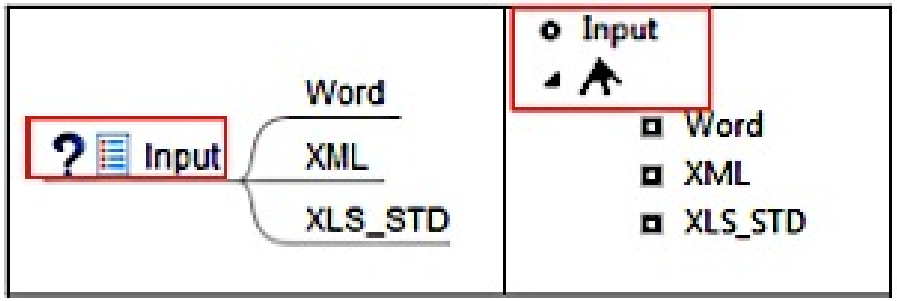
- This rule describes the transformation of a group node or a leaf node without the notation of group strategy and with the notation of a mandatory state, and is mapped for a **SolitaryFeature** with a minimum cardinality equals to 1 and a maximum equals to 1



Rule 4

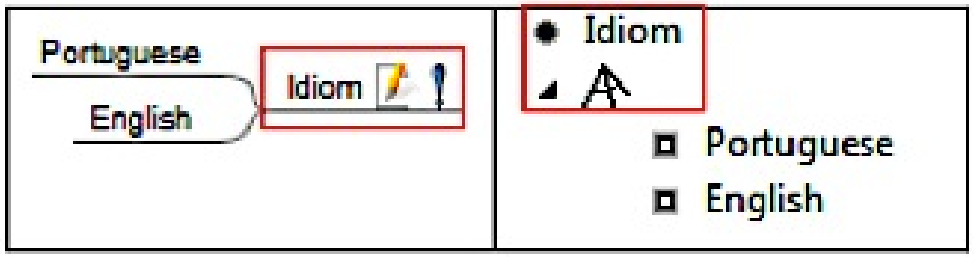
In this rule, the transformation is undertaken by identifying the node of the **OrGroup** type characterized by the notation with a semantic **<<list>>** and is mapped to a **FeatureGroup** group with a group cardinality [1, sum of features child]

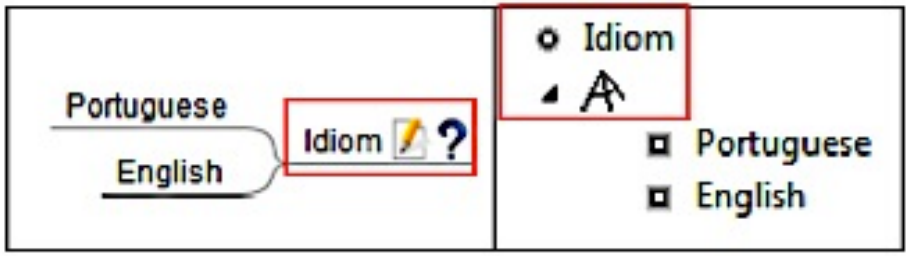




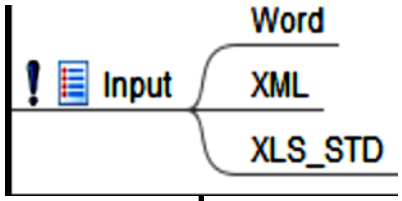
Rule 5

In this rule, the transformation performs a mapping of the node identified by the **AlternativeGroup** strategy and is characterized by the notation, with semantic **<<edit>>** for a **FeatureGroup** with a minimum cardinality equals to 1 and maximum equals to 1





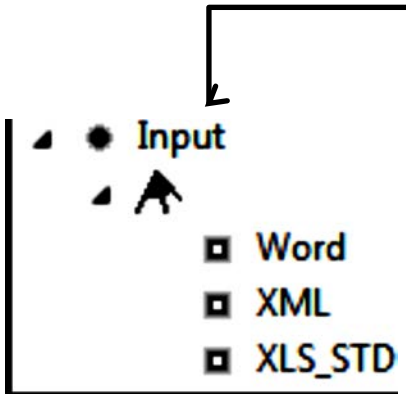
Tool Suport




```
<node ID="ID_1510020228" TEXT="Input" >
    <icon BUILTIN="yes"/>
    <icon BUILTIN="list"/>
    <node ID="ID_1719016949" TEXT="Word"/>
    <node ID="ID_1390275348" TEXT="XML"/>
    <node ID="ID_340875786"  TEXT="XLS STD"/>
</node>
```

XML Mind Map File

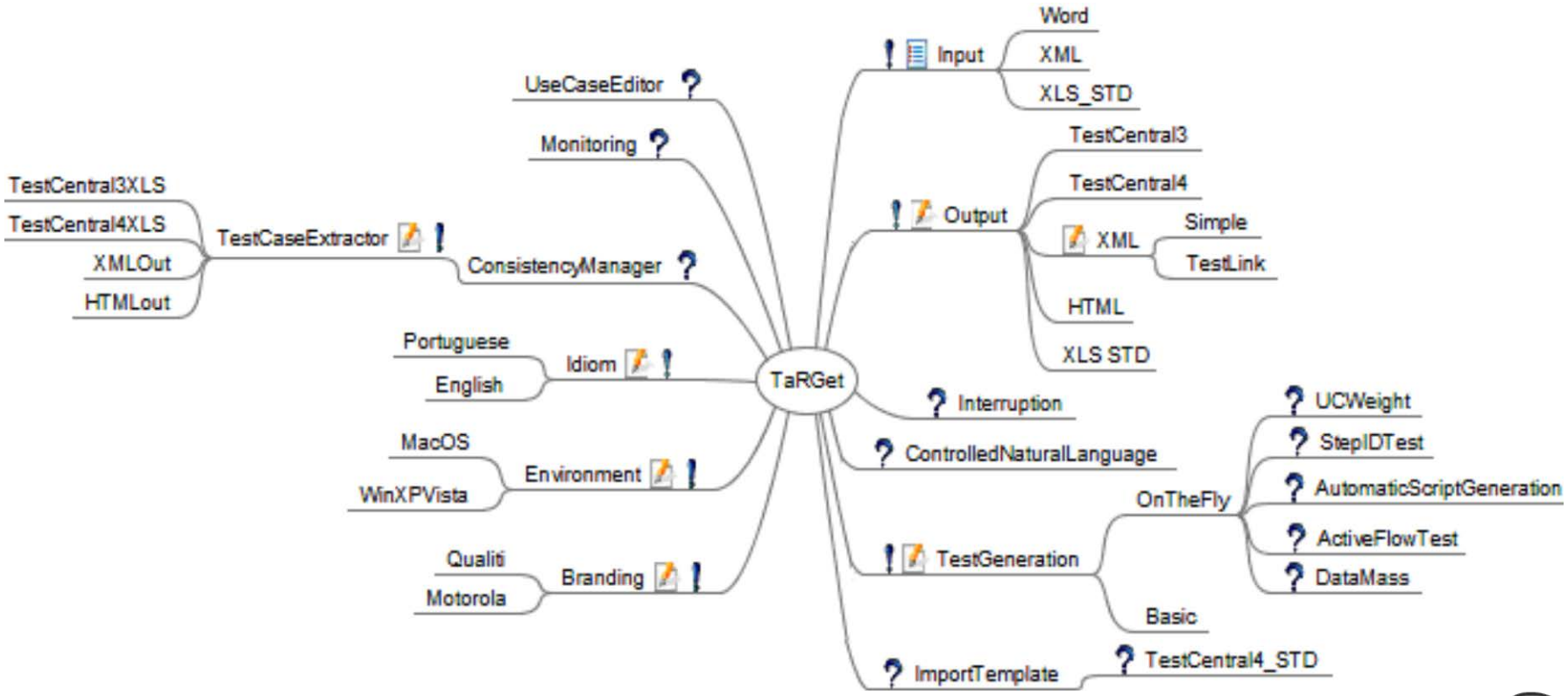
JAVA PARSER




```
<feature min="1" max="1" name="Input" type="NONE" id="inpt">
  <featureGroup min="1" max="3" id="group">
    <feature min="0" max="1" name="Word" type="NONE" id="word">
    </feature>
    <feature min="0" max="1" name="XML" type="NONE" id="xML">
    </feature>
    <feature min="0" max="1" name="XLS_STD" type="NONE" id="xLSSTD">
    </feature>
  </featureGroup>
</feature>
```

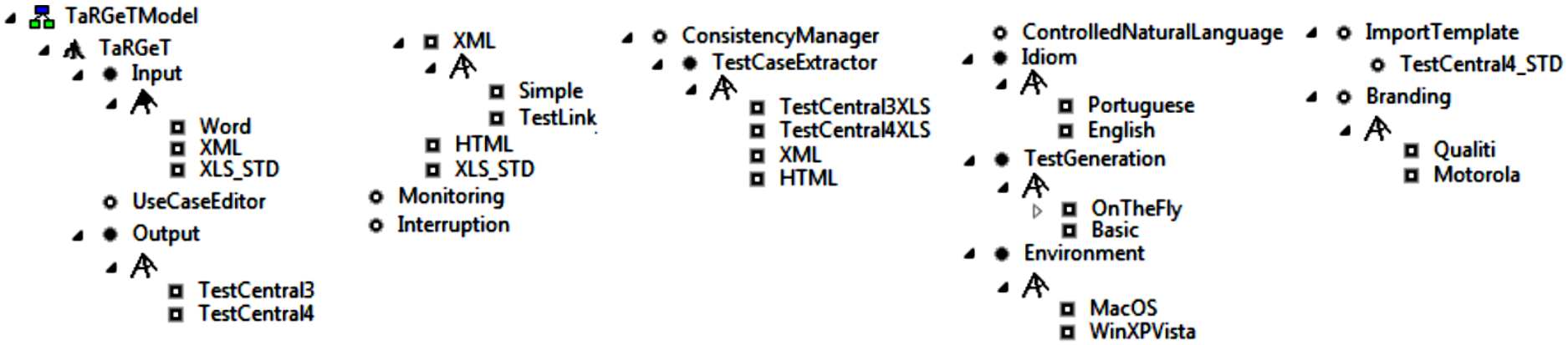
XML Feature Model File

Study Case



Target Mind Domain Model

Study Case



Target Feature Model

Study Case

Rule	No.	Description
Rule 1.	**1**	The central node *TaRGeT* was transformed into the root feature TaRGet.
Rule 2.	**12**	The node leaves – *UseCaseEditor, Monitoring, Interruption, ConsistencyManager, ControlledNaturalLanguage, UCWeight, StepIDTest, AutomaticScriptGeneration, ActiveFlowTest, DataMass, ImportTemplate and TestCentral4_STD* were transformed as SolitaryFeatures, with minimum cardinalities equal 1 and the maximum ones equal to 0.
Rule 4.	**1**	The node *Input* was transformed as FeatureGroup (Or), with a minimum group cardinality equal to 1 and a maximum e maximum equal to 3.
Rule 5.	**6**	The group nodes – *TestCaseExtractor, Idiom, Environment, Branding, Output and TestGeneration* were transformed to FeatureGroup (Alternative), with a minimum cardinality equal to 1 and a maximum equal to 1 (due to all the groups being mandatory) and their children with a minimum cardinality of 0 and a maximum of 1.

Related Works

- Aiming at verifying the relevance of the study, a brief bibliographic survey was conducted, based on the systematic review technique
- The strings were used in four of the main search portals: IEEE explore, ACM Digital Library, SpringerLink and ScienceDirect.

Related Works

- However, the few studies found were of great relevance, such as the work of Czarnecki, (2006)
  - who states that hierarchized diagrams are suitably applicable to feature models, such as the use of tables and **mind maps**
- Another very recent and relevant work that was found, also by Czarnecki, (2012)
  - reports that as domain modeling is performed in the early stage of a design, it is more appropriate that "soft tools" for modeling features such as spreadsheets or **mind maps** be used.

Conclusion and Future Studies

- The main contribution of this article was simplifying and making the process of feature modeling carried out by domain experts more agile...
- ...by means of proposing the use of a cognitive technique for modeling of requirements - the use of mind maps.
- Another contribution of this paper was the result of transforming the mind map to the feature model without the need of refinements in the final model

Conclusion and Future Studies

- For future studies are
  - The formalization of the transformation rules carried out with a specific language of ATL transformation ou MOFScript;
  - Extend the rules of transformation that are valid for the dependency relationships - requires and excludes;
  - The evolution of the proposed transformation to a set of rules that can be reused or specialized by the tool, so that, from even a mind map, it may be transformed into valid entries for other feature modeling tools such as FeatureIDE, pure :: variants, and ToolDay

THANKS!

First International Workshop on Requirements Engineering
on Software Product Line Engineering

Generating Feature Model from Creative Requirements using Model Driven Design

Fernando Wanderley
Universidade de Pernambuco
Recife - Brasil

Denis Silva da Silveira
Universidade Federal de Pernambuco
Recife - Brasil

João Araujo
Faculdade de Ciência e Tecnologia
Universidade Nova de Lisboa
Lisboa - Portugal

Maria Lencastre
Universidade de Pernambuco
Recife - Brasil