



University of Zurich
Department of Informatics



Feature-Oriented Domain Requirements Analysis

Requirements Engineering II - Assignment 5

Jaroslav Habr
Thomas Maurer
Lining Liu

Content

- **Definition**
- **Basic Approach FODA**
- **Feature Tree Models**
- **Advantages and Limitations**
- **Conclusion**

Definition(1)

■ Domain Engineering

- A process of designing a whole family of software systems.
- Model the whole domain all systems belong to.

■ Domain Analysis

- First Step of domain engineering
- Domain identification and definition, scope, stakeholders
- Domain model
 - common and variable requirements for a family of similar systems.
 - functionality: reused across family members
 - dependencies

Definition(2)

- **FODA (Feature-Oriented Domain Analysis)**
 - FODA is a domain analysis method based upon identifying the prominent or distinctive features of a class of a system.
 - Feature models
 - - capture the commonalities and variabilities of a system in a Domain
 - - Feature combinations, condition and why
 - Feature
 - Characteristic of a system that is visible to the end-user
 - Common features
 - - Aggregation and generalization
 - Variable features
 - - decomposed, specialized

FODA Basic Approach

■ FODA Process

- Context Analyse: boundyries of a domain
- Domain Modelling: Produce a Domain Model
 - - Domain knowledge and data requirements
 - - Feature analysis
 - - Operational analysis
- Architecture Modeling: Provides a software solution

Feature Tree Models (1)

■ Overview

- Domain Analysis: Model mandatory and variant requirements
- Origin: Original Feature Tree (FODA)
- OFT: Concept (root node), features, decomposition, constraints

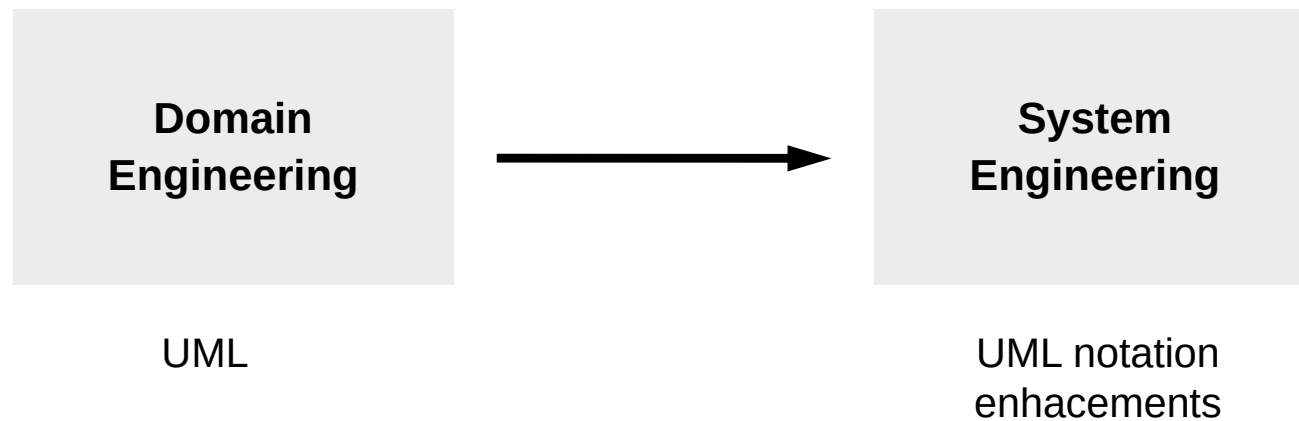
■ Feature Diagrams

- Different languages: FORM, FeatuRSEB, van Gurp, Riebisch, PLUSS
- Nodes, edges, structural dependencies
- Different notations and semantics (“features”)
- Types: Tree, DAG (directed acyclic graph)

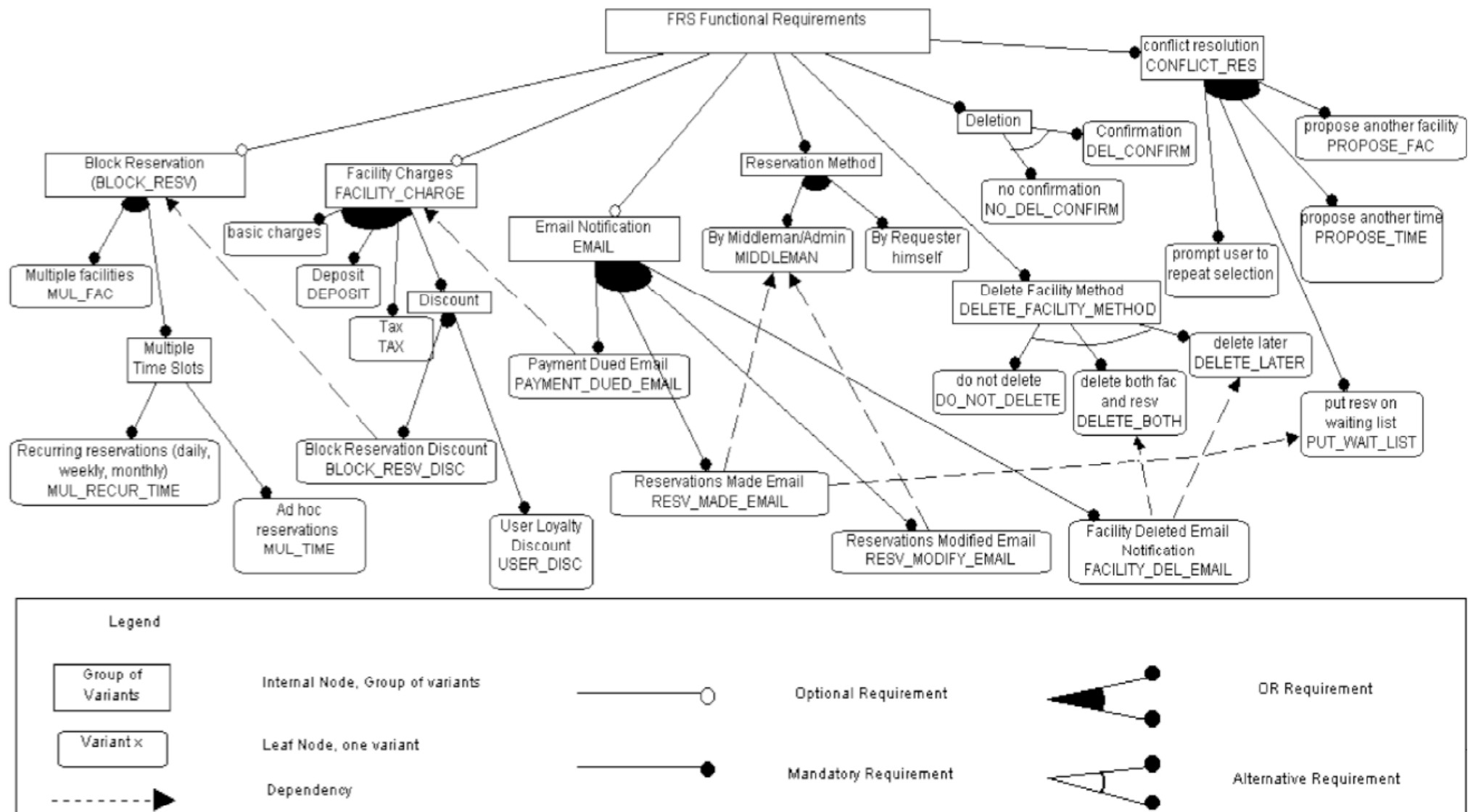
Feature Tree Models (2)

■ Jarzabek Approach

- **Domain Engineering: UML domain defaults (understanding)**
- **System Engineering: Modeling of Variants (customization)**



Feature Tree Models (3)



Advantages & Limitations

■ Advantages

- Clear for small models (overview variants)
- More or less easy to understand
- Variant types: optional, mandatory
- Decomposition: OR, XOR, AND
- Variant dependency (constraints) presentable

■ Limitations

- Scalability: Models get complicated with increasing number of variants
- Ambiguity: Purly defined semantics
- Portability: Merging models modeled with different languages
- Limited expressiveness

**Thank you for your
attention!**
