

Application Engineering

Eduardo Figueiredo

<http://www.dcc.ufmg.br/~figueiredo>

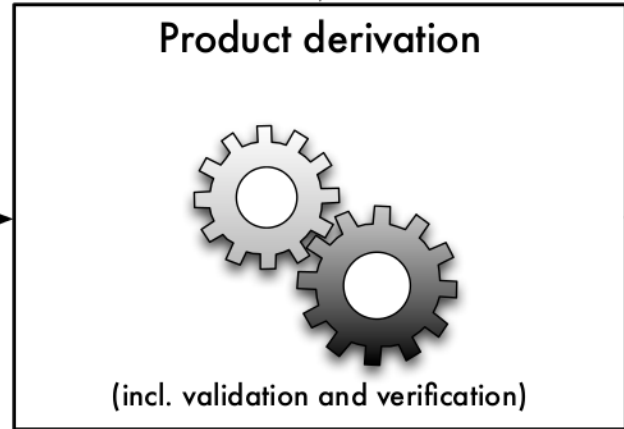
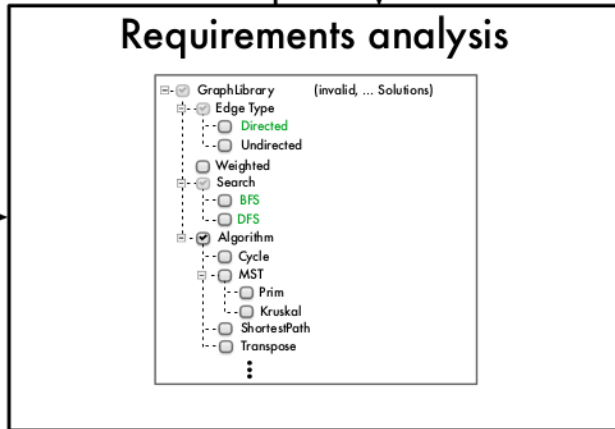
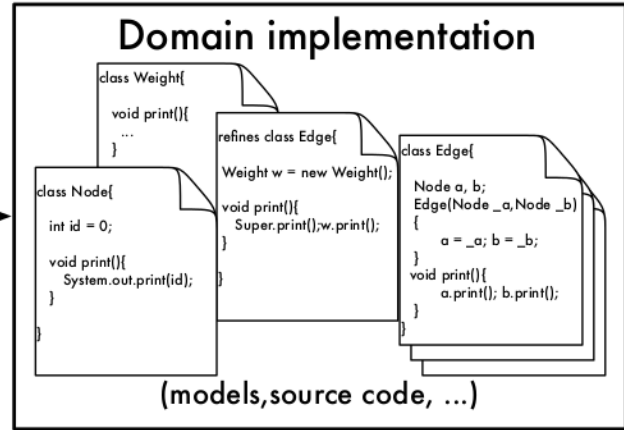
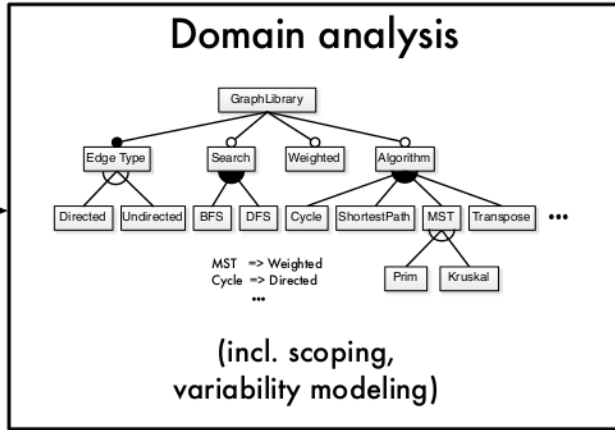
Engineering Process of SPL

Domain engineering

Application engineering

Problem Space

Solution Space



Mapping

Common implementation artifacts

Feature selection

Product

Domain knowledge

Customer needs

New requirements

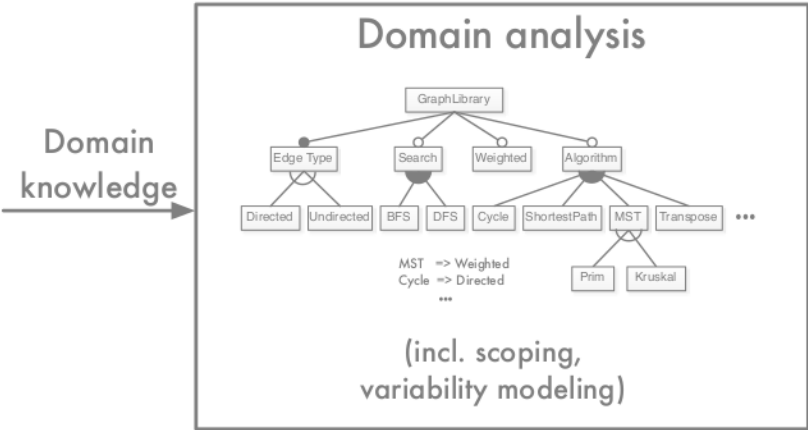
Features

Engineering Process of SPL

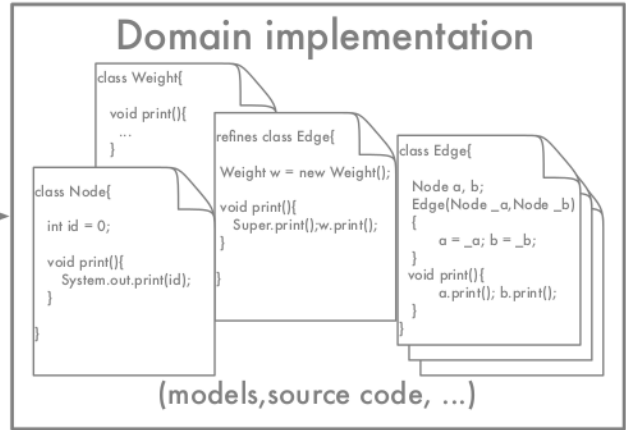
Domain engineering

Problem Space

Solution Space



Mapping

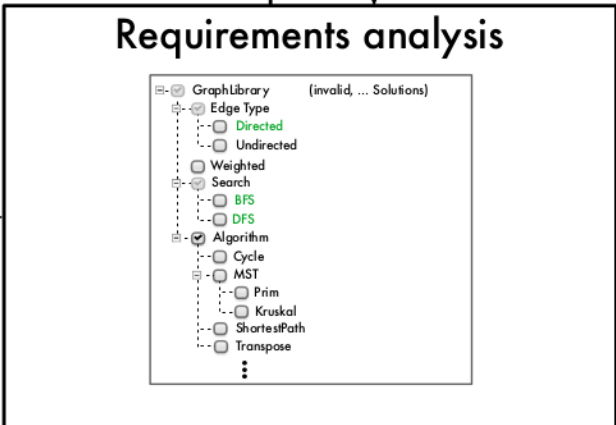


Application engineering

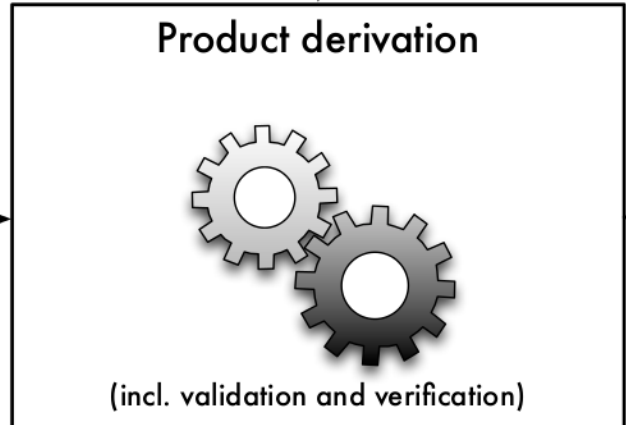
New requirements

Features

Common implementation artifacts



Feature selection



[Tasks in SPL Development]

- Four clusters of tasks in SPL development
 1. Domain Analysis
 2. **Requirements Analysis**
 3. Domain Implementation
 4. **Product Derivation**



Requirements Analysis

[Requirements Analysis]

- It is similar to requirements engineering in traditional software engineering
 - In SPL, we build knowledge from domain analysis
- Requirements analytics typically use well-known requirements techniques
 - Interviews, document analysis, etc.
 - But, they try to map specific customer requirements to those of domain analysis

[New Requirements]

- We have three options if a new requirement cannot be mapped to an existing feature
 1. We may decide that the requirements is out of scope (exclude)
 2. We may change the scope of the SPL to include the new requirement
 3. We may configure a product without this requirement and manually implement it

[Staged Configuration]

- Requirements analysis is mainly based on feature selection
- Feature selection is not always easy
- When multiple stakeholders are involved, features have to be selected in consecutive steps



Product Derivation

[Automatic Product Derivation]

- Depending on the implementation strategy, this task can be fully automated
 - That is, we select feature and call a composition engine to combine them
- Automating product derivation eliminates most costs of this task

[Manual Product Derivation]

- Fully automated product derivation may not be possible
- If manually composed, artifacts are reused and assembled from the domain implementation
 - Developers have to write “glue code” to connect the artifacts

[Product Validation]

- The last step of application engineering is the product validation
- That is, after product derivation, it has to be validated before delivery
 - This task includes running test cases

[Bibliography]

- S. Apel, D. Batory, C. Kastner, G. Saake. **Feature-Oriented Software Product Lines: Concepts and Implementation**. Springer; 2013.
 - Section 2.2 (2.2.2 and 2.2.4)