

# Automated Model Grouping

Michael Becker  
University of Leipzig

## Forecast

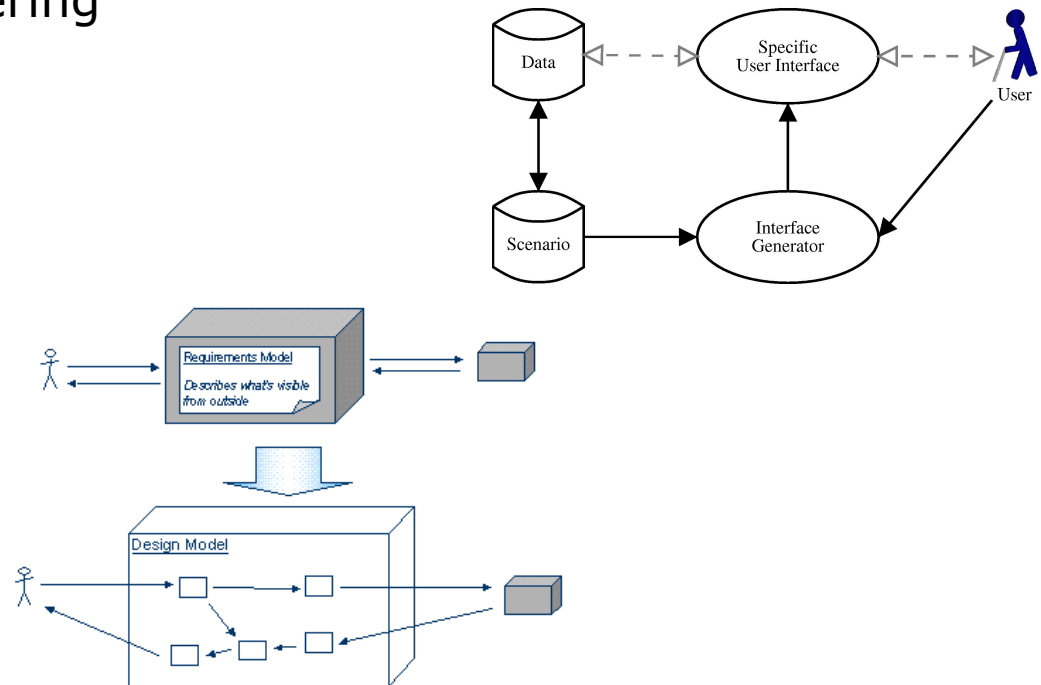
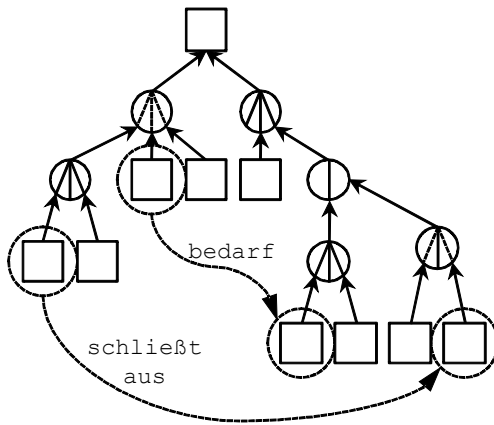
- Problem
  - Handling large amount of existing models
  - Not limited to software engineering
    - Service Engineering, Workflow Management, etc.
- Approach
  - Group models according to their similarity
    - What is similarity?
    - How can models be grouped automatically?
- State
  - Very beginning

## Outline


- Actual Situation and Problem Description
  - Models today
- Research Targets and Research Method
  - Planned steps in the PhD
- Preliminary Results
  - Application on an existing model collection
- Next Steps

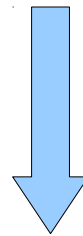
## Actual Situation

- Models are used everywhere
  - Description of software, services, work flows, etc.
- Models base on different notations
  - UML, EPC, BPMN, Service Configuration Graph
- Models are becoming more and more important
  - MDE, Service Engineering



## Problem Description

- Growing number of models adds to the difficulty of
    - Organising models
    - Searching models
  - Existing models are not reused
    - Developers will not reuse models if it is more time-consuming to find, understand, and integrate existing models than to create new models.
- 



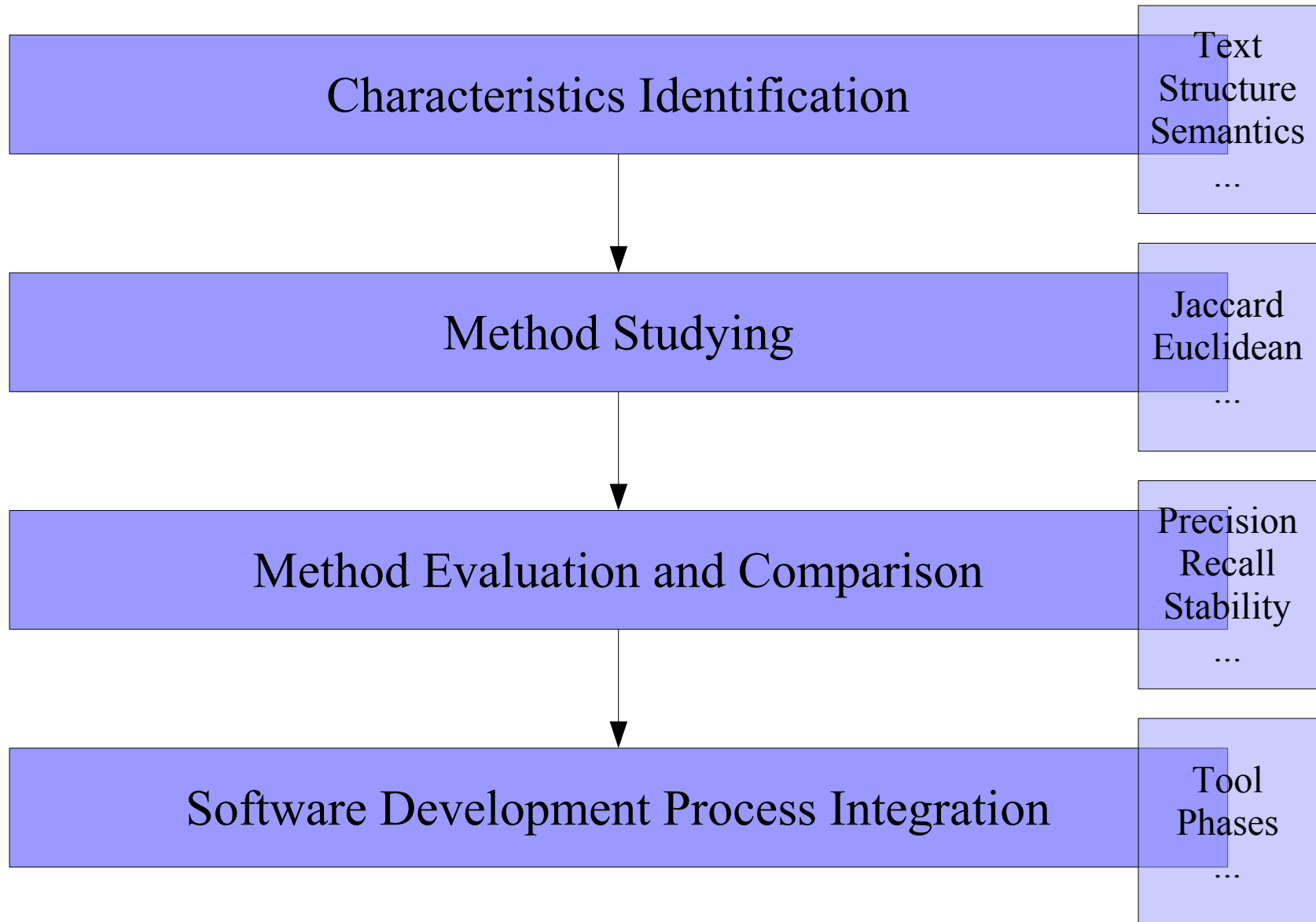
Existing knowledge is not used  
Engineering costs increase

## Reuse Models

- Reuse Requirements
  - Structuring: explication of implicit knowledge
  - Documenting: better understandability
- Naïve Approach
  - Manually keyword and describe models, group them in categories
    - Labor-intensive, time-consuming, error-prone
    - Higher number of models adds to the problems
    - At a certain number manual indexing is not possible
- Proposed Solution
  - Automated Model Grouping

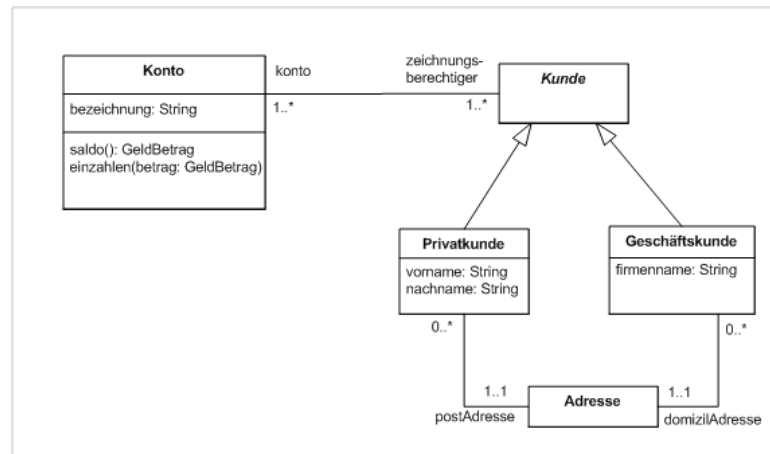
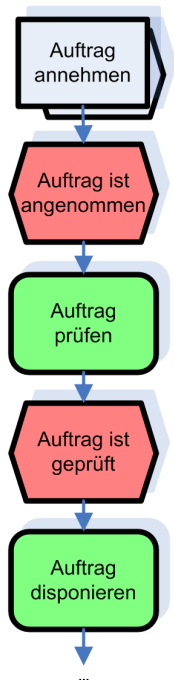
## Research Targets

- Structure model collections automatically
  - Model Grouping based on similarity
  - Improves Model Browsing, Model Retrieval
- SDP integration
  - Seamless integration of model retrieval and grouping into the software development process
  - Resulting in a tool

**Research Method**

## Step 1: Identify model characteristics

- Identify characteristics to distinct models from each other
- Automatic grouping requires automatic extraction
- Concentration on labels
  - easy to extract, generic enough
  - enables text, structure, and semantics extraction

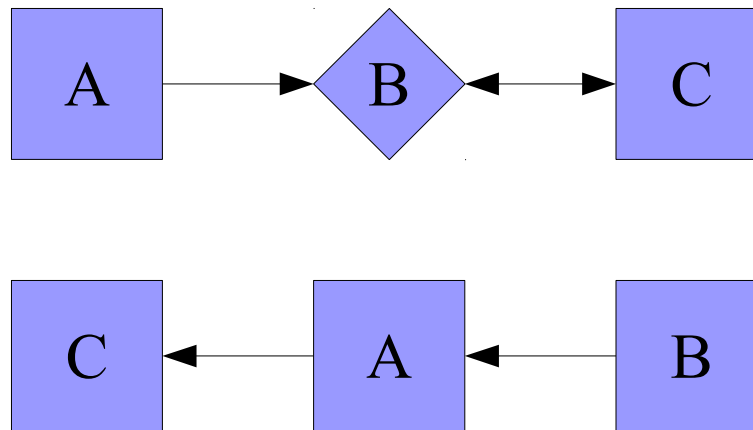


### Characteristics

- Labels
- Colours
- Size (width, height)
- Amount of elements
- Semantics

## Labels: Text

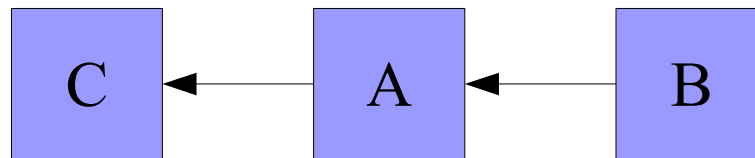
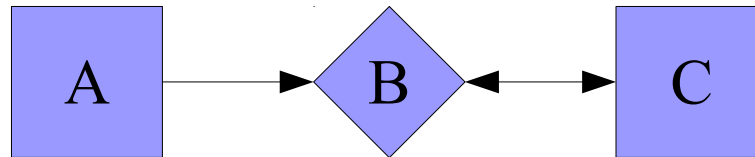
- Group models according to text of persistent representation
- Extract text from labels (naïve approach)
- Extract text from labels (NLP approach)



- A = Delegate the work to person X
- B = Assign the work to person X
- C = Put it on the work bench
  - Synonyms, Homonyms

## Labels: Structure, Semantics

- Distinguish different types of elements
  - Representation of types in different notations
- Take connection of elements in account



- Semantics: Meaning of words
  - Concepts: A=auto, B=car, C=duck
  - Latent Semantic Indexing, etc.

- Group models according to operational semantics
  - Unified Semantic Representation
  - Semantics of non-executable models
  - May be only possible using homogeneous collections

## Step 2 and 3: Analyse and evaluate methods

- Grouping of models with existing methods
  - Fuzzy or strict?
- Required steps
  - Formatting models
  - Selecting a similarity measure
  - Calculating a similarity matrix
  - Calculating groups
  
- Application on different types of model collections
  - 1<sup>st</sup> step: group models of same type
  - 2<sup>nd</sup> step: group models of different types
- Baseline: Manual grouping
  - SAP Reference Model

## Step 4: SDP Integration

- Seamless integration of model reuse
  - Search for models in early phases
  - Inspiration from existing models
- Tool development
- Usage of model repositories
- Model Merging

## Preliminary Results

- Applied on SAP R/3 Reference Model
  - 1593 models in English and German
  - Average amount of words per model: 36
    - Calculating similarity between documents with few words
- Jaccard Similarity, Euclidean Distance
  - Jaccard groups very similar to manual grouping but has low similarity values
  - Euclidean Distance has one document that is grouped with all others, needs further examination



Thanks for your attention

Feedback

## Backup

# Jaccard Similarity

$$J(A, B) = \frac{A \cap B}{A \cup B}$$

*A = cat eats mouse*

*B = dog hunts cat*

$$J(A, B) = \frac{|\{cat\}|}{|\{cat, eats, mouse, dog, hunts\}|} = \frac{1}{5}$$

## Euclidean Distance

$$d(x, y) = \sqrt{(x_1 - y_1)^2 + \dots + (x_n - y_n)^2}$$

$$\vec{X} = (w_1 = \text{cat}, w_2 = \text{eats}, w_3 = \text{mouse})$$

$$\vec{Y} = (w_4 = \text{dog}, w_5 = \text{hunts}, w_1 = \text{cat})$$

$$d(x, y) = \sqrt{(|w_{1_x}| - |w_{1_y}|)^2 + (|w_{2_x}| - |w_{2_y}|)^2 + (|w_{3_x}| - |w_{3_y}|)^2 + (|w_{4_x}| - |w_{4_y}|)^2 + (|w_{5_x}| - |w_{5_y}|)^2}$$

$$\sqrt{(1-1)^2 + (1-0)^2 + (1-0)^2 + (0-1)^2 + (0-1)^2}$$

$$\sqrt{0+1+1+1+1} = 2$$