

Softwaretechnik

Model Driven Architecture

Anwendungen von Metamodellierung

Prof. Dr. Peter Thiemann, Stefan Wehr

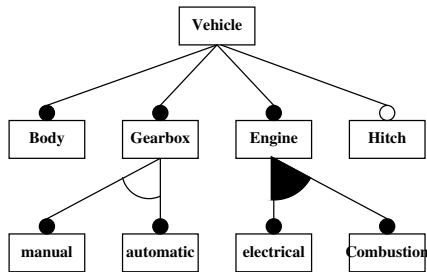
Universität Freiburg

18.07.2008

- Feature Modelle als Werkzeug zur Analyse einer Domäne
 - Hierarchische Anordnung der Features und ihrer Abhängigkeiten
 - Kategorisierung
- Visualisierung mit Hilfe von Feature Diagrammen
- Erfunden zur Analyse von Domänen im Zuge der Entwicklung von Softwaresystemen: *Kang, Cohen, Hess, Novak, Peterson. Feature-Oriented Domain Analysis (FODA) Feasibility Study. Technical report CMU/SEI-90-TR-21. 1990.*
- Auch verwendet im Bereich des generativen Programmierens (Czarnecki, Eisenäcker)

Featuremodellierung

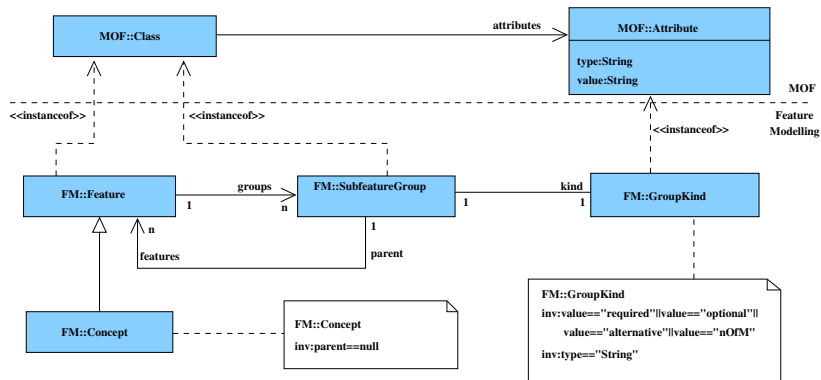
Beispiel



- Hierarchisch, aber **keine** “ist-ein” Beziehung (wie in Klassendiagrammen)
- Features können “required”, “optional”, “alternative”, oder “*n-of-m*” (Auswahl) sein

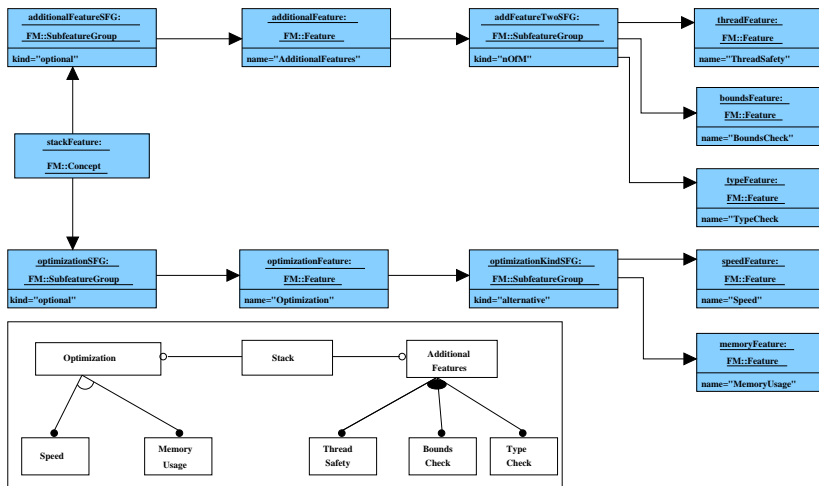
Featuremodellierung

MOF-basiertes Metamodell



Featuremodellierung

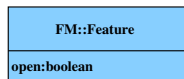
Beispielmodell als UML-Objektdiagramm



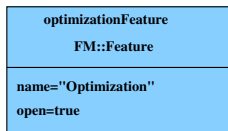
Featuremodellierung

Erweiterung des Metamodells

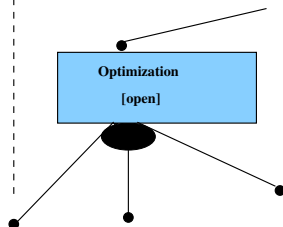
Metamodel



Object diagram



Feature diagram



Neue Eigenschaft \Rightarrow

- Neues Attribut im Metamodell
- Neuer Slot im Modell
- Erweiterung der konkreten Syntax

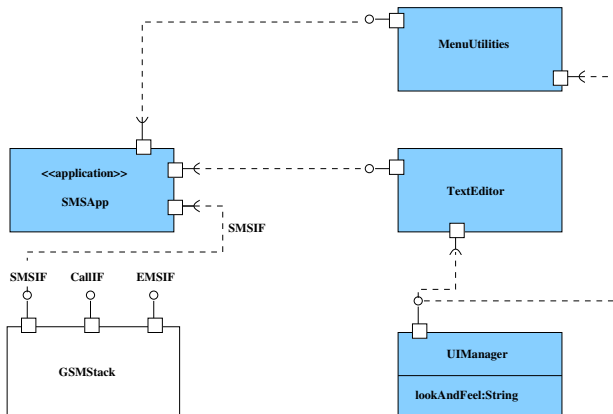
Anwendung von Metamodellierung

Komponenteninfrastruktur

- DSL für kleine, eingebettete Systeme
- Zentrale Abstraktion: Komponente
- Eine Komponente
 - *stellt Dienste bereit (mittels Schnittstellen),*
 - *benötigt Dienste (mittels Schnittstellen),*
 - *hat Konfigurationsparameter.*
- Eine Komponente die keine Dienste bereitstellt ist eine *Anwendung.*

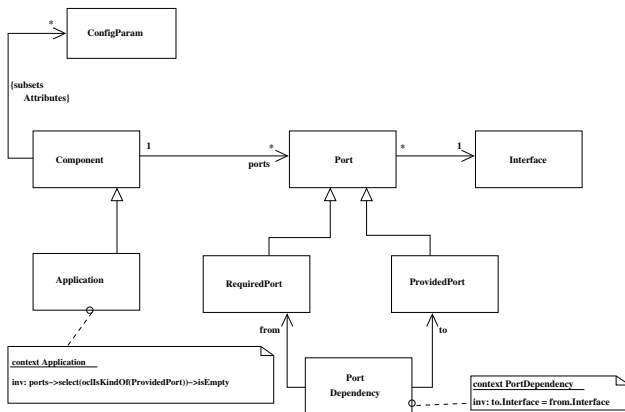
Komponenteninfrastruktur

Beispiel



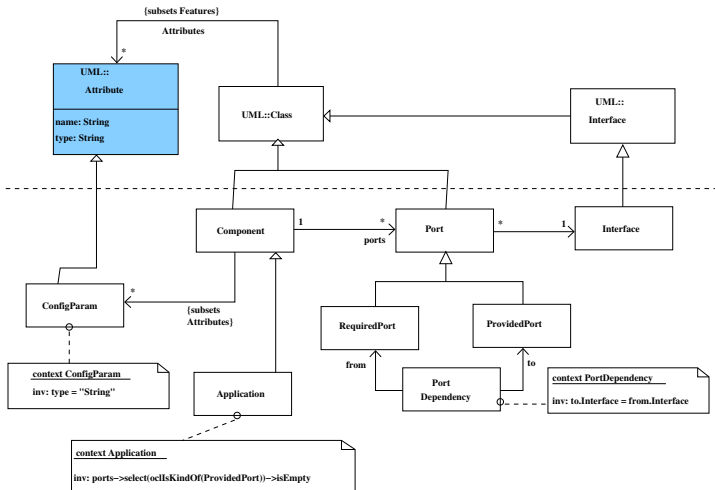
Komponenteninfrastruktur

Metamodell



Komponenteninfrastruktur

Metamodell als Erweiterung des UML Metamodells



Stolperfallen bei der Metamodellierung

Mögliche Probleme:

- Verwechslung mit UML Notation
- Vermischen von (Meta-) Ebenen

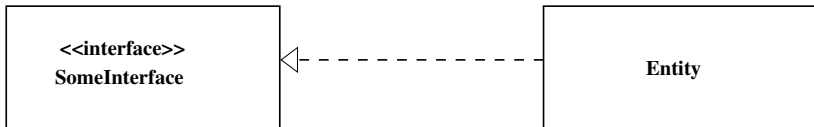
Hilfreich:

- Wie sieht die Entsprechung in einer Programmiersprache aus?

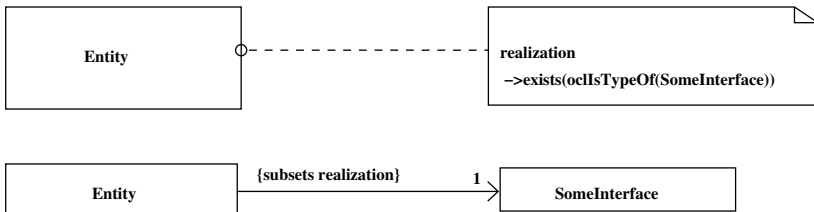
Beispiel: Interfaces

Jede Instanz der Metaklasse **Entity** soll das Interface **SomeInterface** implementieren

- **Falscher Ansatz**



- **Lösung aus Stahl & Völter Buch: benutze OCL oder Metaassoziation mit Teilmengeeinschränkung**



Beispiel: Interfaces / 2

Jede Instanz der Metaklasse **Entity** soll das Interface **SomeInterface** implementieren

- **Richtige Lösung:** benutze OCL

