
Komponentenorientierter Entwurf von PIMs und CIMs mit der Kobra-Methode

Colin Atkinson, **Matthias Gutheil**, Oliver Hummel
{atkinson, gutheil, hummel}@informatik.uni-mannheim.de

Lehrstuhl für Softwaretechnik
Universität Mannheim
<http://swt.informatik.uni-mannheim.de>



UNIVERSITY OF
MANNHEIM

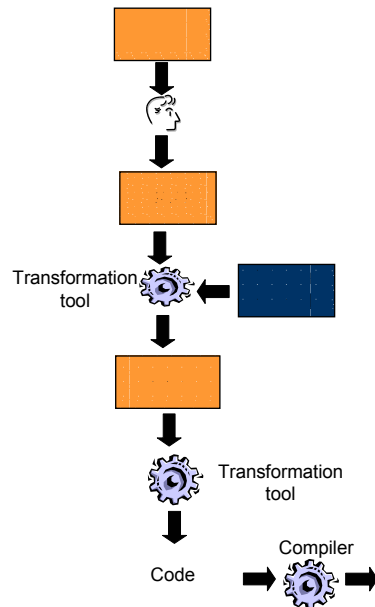
Überblick

- Modellgetriebene Softwareentwicklung
- Komponentenorientierte Softwareentwicklung
- MDA mit Kobra
- Zusammenfassung
- Literatur und Fragen



MDA: CIMs, PIMs und PSMs

- Die MDA unterscheidet zwischen
 - Computation Independent Models
 - beschreibt die Anforderungen an das System und seine Umgebung
 - keine Beschreibung der Struktur oder interner Abläufe des Systems
 - Platform Independent Models
 - beschreibt die Funktionalität des Systems ohne die notwendigen Details einer bestimmten Plattform
 - beschreibt die Teile des Systems, welche sich auch für verschiedene Plattformen nicht ändern
 - Platform Specific Models
 - kombiniert die plattformunabhängige Sicht mit plattformspezifischen Details zur Beschreibung des Systems



Komponentenorientierte Softwareentwicklung

■ Charakteristika

“ a component represents a modular, deployable, and replaceable **part of a system** that encapsulates implementation and exposes a set of interfaces ”

UML 1.5 Specification

“ a reusable software component is a logically cohesive, **loosely coupled module** that denotes a single abstraction ”

Booch 87

■ Vorteile

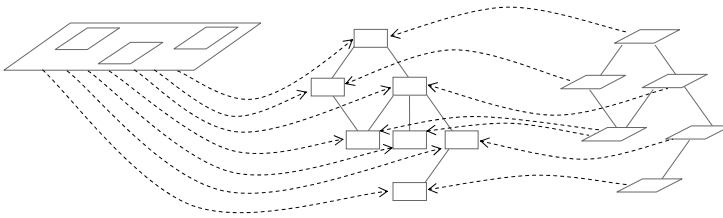
- erhöhte Zuverlässigkeit
 - durch Verwendung von getesteten Komponenten
- reduziertes Prozessrisiko
 - bessere Vorhersage der Entwicklungskosten
- standardkonform
 - Verwendung von Standards in wiederverwendbaren Komponenten
- kürzere Entwicklungszeit
 - durch Vermeidung von Eigenentwicklung



Komponentenorientierte SE vs. MDA

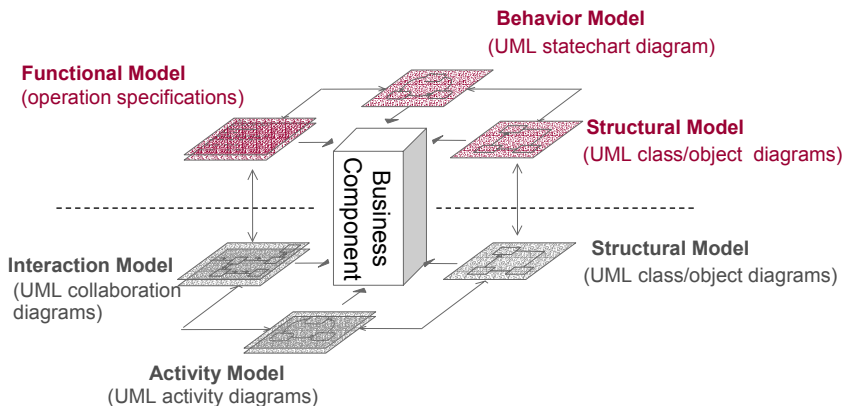
- Heutzutage versteht man unter einer Komponente eine JavaBean, eine CORBA Applikation oder ein .NET Objekt
- Komponenten sind primär ein Implementierungskonzept
- Beschreibung von Komponenten hauptsächlich auf PSM-Ebene
- So gut wie keine komponentenorientierte Betrachtung auf PIM-Ebene während modellgetriebener Entwicklung.

traditionell *Run-time Hierarchie* *komponentenorientiert*



Modellierung von Komponenten mit Kobra

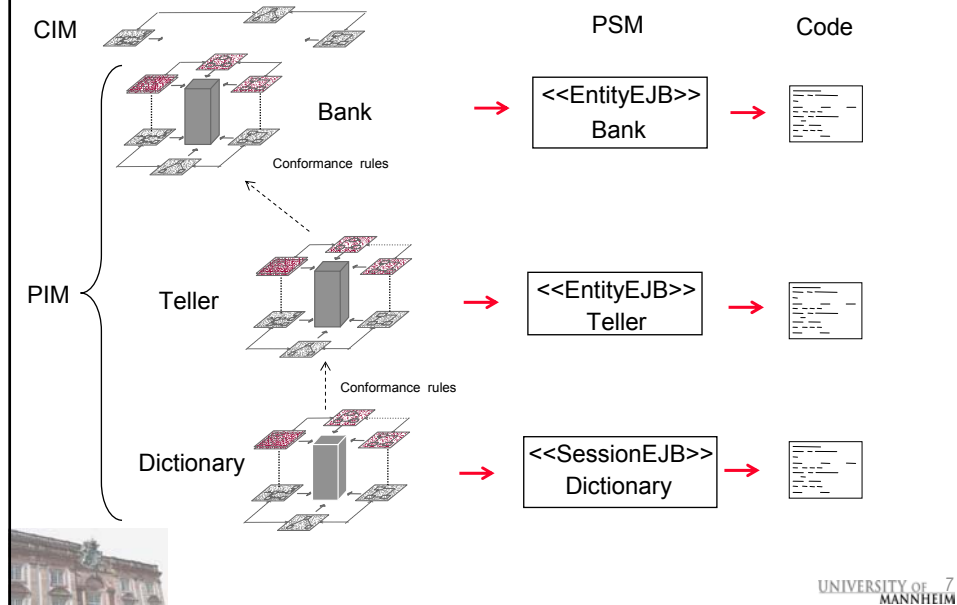
Specification



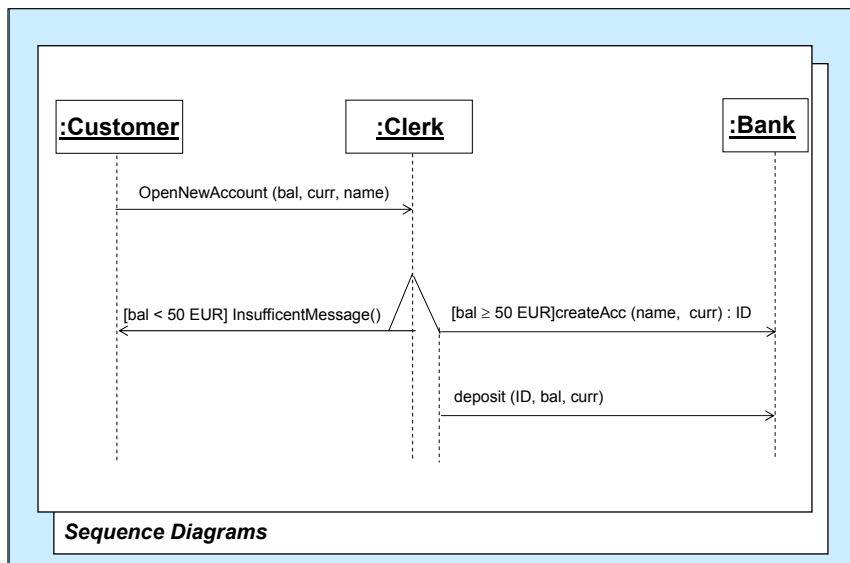
Realization



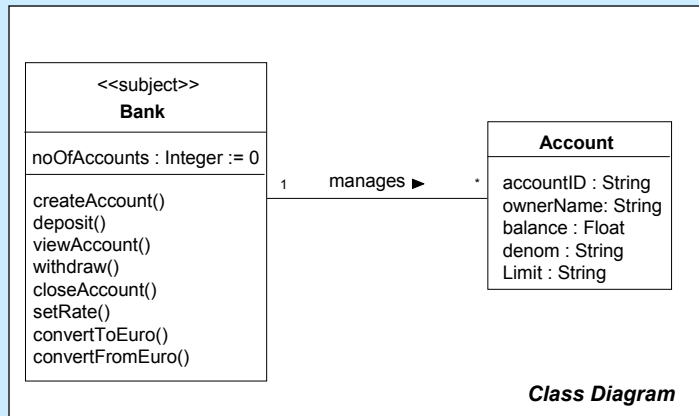
CIMs, PIMs und PSMs in Kobra



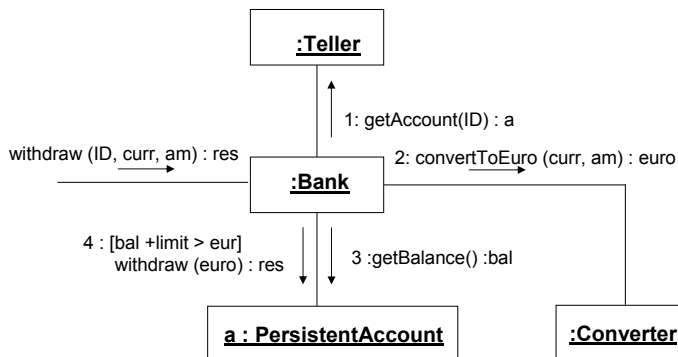
Context Realization (Bank)



Component Specification (Bank)



Component Realization (Bank)



Zusammenfassung Kobra und MDA

- Bisher
 - Hauptsächlich Komponentenorientierung nur auf PSM-Ebene
- MDA mit Kobra
 - Komponentenorientierung auch auf CIM- und PIM-Ebene
 - Vorteile:
 - Ermöglicht die Betrachtung und Wiederverwendung von BusinessKomponenten in der Analyse und Designebene
 - Effizientere Erstellung und Verifizierung von PIMs
 - Nutzen der Synergie sowohl komponentenorientierter als auch modellgetriebener Entwicklung auf jeder Ebene
 - Besonders geeignet zur Modellierung der Composition von Webservices und Beschreibung von SOA.



Literatur und Fragen

- Kobra Buch
 - 464 Seiten, November 2001
 - Component Series, Addison-Wesley
 - ISBN: 0-201-73791-4
- <http://swt.informatik.uni-mannheim.de/kobra>



- **Vielen Dank für Ihre Aufmerksamkeit!**

- Fragen?

