

Block B (Software-Architektur): Methodischer Entwurf 09.11.2004 1

Vorlesung „Methoden des Software Engineering“

Block B „Software Architektur“  
**Methodischer Entwurf**

Martin Wirsing

Einheit B.1, 09.11.2004

Methoden des Software Engineering (c) 2004, Koch, Störle, Wirsing, LMU München

Block B (Software-Architektur): Methodischer Entwurf 09.11.2004 2

**Gliederung B.1**

- **Einleitung in den Block B „Software-Architektur“**
  - Definition, Gegenstand, Motivation
  - Gliederung und Einbettung in die Vorlesung
- **Teil 1: Methodischer Entwurf**
  - Notationen & Konzepte
  - Fallbeispiel
  - Hilfsmittel & Literatur
- **Ausblick auf Teil 2: Systemarchitektur**

Methoden des Software Engineering (c) 2004, Koch, Störle, Wirsing, LMU München

Block B (Software-Architektur): Methodischer Entwurf 09.11.2004 3

**Software-Architektur – Gegenstand**

Die klassische Trias der Architektur seit Vitruv.

Methoden des Software Engineering (c) 2004, Koch, Störle, Wirsing, LMU München

Block B (Software-Architektur): Methodischer Entwurf 09.11.2004 4

**Software-Architektur – Definitionen**

„Architecture is the set of design decisions [...] that keeps its implementors and maintainers from exercising needless creativity.“  
(D'Souza & Wills, Catalysis)

„Architecture is a framework for change.“  
(Tom De Marco)

„Architecture is defined [...] as the fundamental organization of a system, embodied in its components, their relationships to each other and the environment, and the principles governing its design and evolution.“  
(IEEE Architecture Working Group, P1471)

Siehe z.B. [www.sei.smu.edu/architecture/definitions.html](http://www.sei.smu.edu/architecture/definitions.html) für weitere Definitionen.

Methoden des Software Engineering (c) 2004, Koch, Störle, Wirsing, LMU München

Block B (Software-Architektur): Methodischer Entwurf 09.11.2004 5

**Die Rolle des Software-Architekten**

Methoden des Software Engineering (c) 2004, Koch, Störle, Wirsing, LMU München

Block B (Software-Architektur): Methodischer Entwurf 09.11.2004 6

**Gliederung Block B**

Methoden des Software Engineering (c) 2004, Koch, Störle, Wirsing, LMU München

Block B (Software-Architektur): Methodischer Entwurf 09.11.2004 7

## Gliederung B.1

- Vorstellung
- Einleitung in den Block B „Software-Architektur“
  - Definition, Gegenstand, Motivation
  - Gliederung und Einbettung in die Vorlesung
- **Teil 1: Methodischer Entwurf**
  - Notationen & Konzepte
  - Fallbeispiel
  - Hilfsmittel & Literatur
- **Ausblick auf Teil 2: Systemarchitektur**

Kontexteinbettung  
Geschäftsvorfälle  
Facharchitektur  
Nutzfälle  
Schnittstellen

Methoden des Software Engineering (c) 2004, Koch, Störle, Wirsing, LMU München

Block B (Software-Architektur): Methodischer Entwurf 09.11.2004 8

## Methodik Dokumentenbasierte Entwicklung

- Ein System entsteht und entwickelt sich als eine Menge von Dokumenten: Konzepte, Präsentationen, Testfälle, Code, ...
- Drei Dimensionen der Klassifizierung
  - Historie („Version“, „Phase“)
  - Perspektive („Sicht“)
  - Granularität („Ebene“)
- Verschiedene überlappende Perspektiven
  - Überlappungen sind Quellen von Inkonsistenzen, aber auch Möglichkeit der Konsistenzprüfung („Sollbruchstellen“).
  - Überlappungen bieten außerdem Möglichkeiten, von einer Sicht eine andere methodisch abzuleiten („opportunistisches Vorgehen“).

Methoden des Software Engineering (c) 2004, Koch, Störle, Wirsing, LMU München

Block B (Software-Architektur): Methodischer Entwurf 09.11.2004 9

## Rückblick betrachtete Aspekte und Darstellungsmittel

<b>Kontext</b>	Fachliches Klassenmodell
<b>Einbettung</b>	Zusammenhänge
<b>Quantifizierung</b>	Objektlebenszyklus
	Klassendetails
	<b>Struktur</b>
	Facharchitektur
	Subsysteme & Schnittstellen
	(fachliche) Interaktionen
<b>GeVo</b>	
hierarchischer Text	
GeVo-Diagramm	
GeVo-Tabelle	
Aktivitätsdiagramm	

Methoden des Software Engineering (c) 2004, Koch, Störle, Wirsing, LMU München

Block B (Software-Architektur): Methodischer Entwurf 09.11.2004 10

## Ebene 0 Kontexteinbettung

Ebene: 0 Phase: Architektur-Entwurf Autor: HS Version: 0.1 Sicht: Kontext

Methoden des Software Engineering (c) 2004, Koch, Störle, Wirsing, LMU München

Block B (Software-Architektur): Methodischer Entwurf 09.11.2004 11

## Ebene 0 Kontexteinbettung

Ebene: 0 Phase: Architektur-Entwurf Autor: HS Version: 0.1 Sicht: Kontext

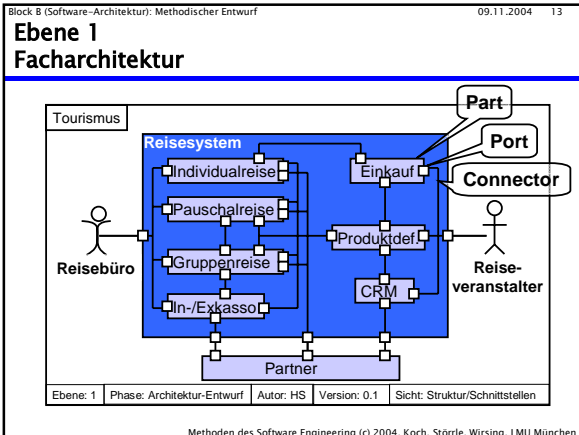
Methoden des Software Engineering (c) 2004, Koch, Störle, Wirsing, LMU München

Block B (Software-Architektur): Methodischer Entwurf 09.11.2004 12

## Ebene 1 Facharchitektur

Ebene: 1 Phase: Architektur-Entwurf Autor: HS Version: 0.1 Sicht: Facharchitektur

Methoden des Software Engineering (c) 2004, Koch, Störle, Wirsing, LMU München

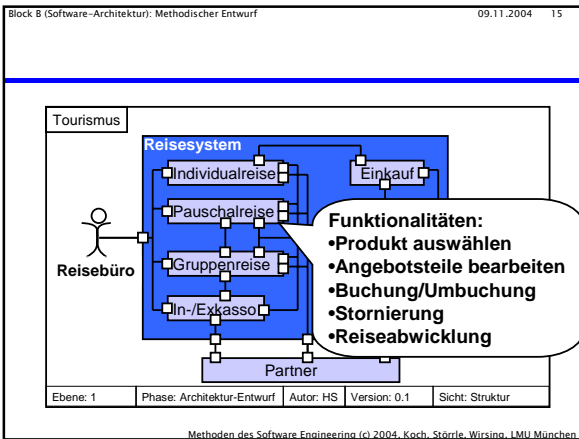


Block B (Software-Architektur): Methodischer Entwurf 09.11.2004 14

### Spezifikation von Ports mit Protokollrollen

- **Im Allgemeinfall:**
  - je eine Liste für eingehende und ausgehende Signale plus ein gemeinsamer Zustandsautomat.
- **Im Modell:**
  - je ein (UML-)Interface pro Richtung (können optional einen Zustandsautomaten haben).
- **Für Benutzerschnittstellen:**
  - Window-Event-Diagramm, Storyboards, GUI-Prototypen.
- **Statt UML-Interfaces sind auch XML-DTDs, Corba-IDLs, oder DB-Schemata (z.B. als DDL-Ausdrücke oder ER-Modelle) nützlich.**

Metoden des Software Engineering (c) 2004, Koch, Störle, Wirsing, LMU München

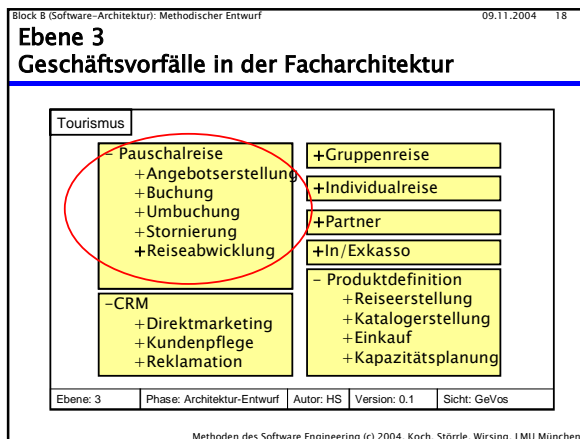
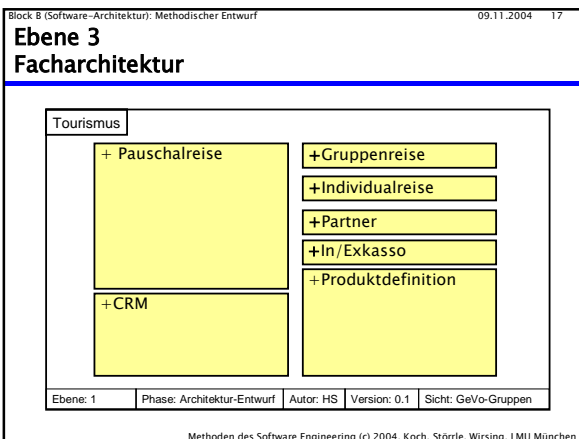


Block B (Software-Architektur): Methodischer Entwurf 09.11.2004 16

### Rückblick betrachtete Aspekte und Darstellungsmittel

<b>Kontext</b>	<b>Fachliches Klassenmodell</b>
<b>Einbettung</b>	Zusammenhänge
<b>Quantifizierung</b>	Objektlebenszyklus
	Klassendetails
	<b>Struktur</b>
<b>GeVo</b>	<b>Facharchitektur</b>
hierarchischer Text	<b>Subsysteme &amp; Schnittstellen</b>
GeVo-Diagramm	(fachliche) Interaktionen
GeVo-Tabelle	
Aktivitätsdiagramm	

Metoden des Software Engineering (c) 2004, Koch, Störle, Wirsing, LMU München



Block B (Software-Architektur): Methodischer Entwurf 09.11.2004 19

### Ebene 3 Geschäftsvorfälle in der Facharchitektur

Tourismus

- Pauschalreise
- +Angebotserstellung
- +Buchung
- +Umbuchung
- +Stornierung
- +Reiseabwicklung

...

Ebene: 3	Phase: Architektur-Entwurf	Autor: HS	Version: 0.1	Sicht: GeVos
----------	----------------------------	-----------	--------------	--------------

Methoden des Software Engineering (c) 2004, Koch, Störle, Wirsing, LMU München

Block B (Software-Architektur): Methodischer Entwurf 09.11.2004 20

### Ebene 3...4 Geschäftsvorfälle/Nutzfälle in der Facharchitektur

Tourismus

- Pauschalreise
- Angebotserstellung
  - +Reise auswählen
  - +Verfügbarkeit prüfen
  - +Angebot reservieren
- +Buchung
- +Umbuchung
- +Stornierung
- +Reiseabwicklung

...

Ebene: 3	Phase: Architektur-Entwurf	Autor: HS	Version: 0.1	Sicht: GeVos
----------	----------------------------	-----------	--------------	--------------

Methoden des Software Engineering (c) 2004, Koch, Störle, Wirsing, LMU München

Block B (Software-Architektur): Methodischer Entwurf 09.11.2004 21

### Ebene 3...4 Geschäftsvorfälle/Nutzfälle in der Facharchitektur

Tourismus

- Pauschalreise
- Angebotserstellung
  - Reise auswählen
    - +Termine wählen
    - +Hotelkategorie wählen
    - +Flug wählen
    - +Zusatzangebote wählen
  - +Verfügbarkeit prüfen
  - +Angebot reservieren
- +Buchung
- +Umbuchung
- ...

...

Ebene: 3	Phase: Architektur-Entwurf	Autor: HS	Version: 0.1	Sicht: GeVos
----------	----------------------------	-----------	--------------	--------------

Methoden des Software Engineering (c) 2004, Koch, Störle, Wirsing, LMU München

Block B (Software-Architektur): Methodischer Entwurf 09.11.2004 22

### Ebene 3...4 Geschäftsvorfälle/Nutzfälle in der Facharchitektur

Tourismus

- Pauschalreise
- Angebot
  - Reise auswählen
    - +Termine wählen
    - +Hotelkategorie wählen
    - +Saisonkalender anzeigen
    - +Anfangstermin angeben
    - +Endtermin angeben
    - +Termingenauigkeit einstellen
  - +Flug wählen
  - +Zusatzangebote wählen
- ...

...

Ebene: 3	Phase: Architektur-Entwurf	Autor: HS	Version: 0.1	Sicht: GeVo-Verfeinerung
----------	----------------------------	-----------	--------------	--------------------------

Methoden des Software Engineering (c) 2004, Koch, Störle, Wirsing, LMU München

Block B (Software-Architektur): Methodischer Entwurf 09.11.2004 23

### Unterschied Geschäftsvorfall/Nutzfall

Geschäftsvorfall	Nutzfal
<ul style="list-style-type: none"> <li>• überschreitet Systemgrenzen</li> <li>• benutzt/enthält Nutzfälle</li> <li>• möglicherweise nur teilweise automatisch</li> <li>• kann (leicht) unterbrochen werden</li> <li>• wird erbracht von einer Organisation, umfasst mehrere Systeme &amp; Personen</li> <li>• von messbarem Wert oder Kosten, für einen Akteur, von außen sichtbar</li> <li>• white box view</li> </ul>	<ul style="list-style-type: none"> <li>• definiert Systemgrenze</li> <li>• kann in GeVos vorkommen</li> <li>• automatisch: umfasst keine manuellen Zwischenschritte</li> <li>• läuft atomar in einem Zeitintervall ab</li> <li>• wird erbracht von einem System</li> <li>• black box view</li> </ul>

Methoden des Software Engineering (c) 2004, Koch, Störle, Wirsing, LMU München

Block B (Software-Architektur): Methodischer Entwurf 09.11.2004 24

### Ebene 3...4 Geschäftsvorfälle/Nutzfälle in der Facharchitektur

Tourismus

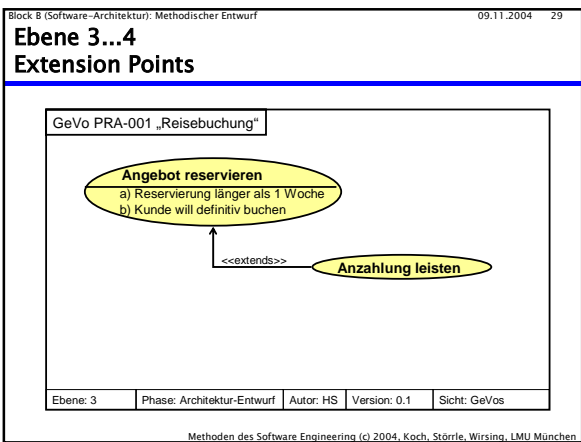
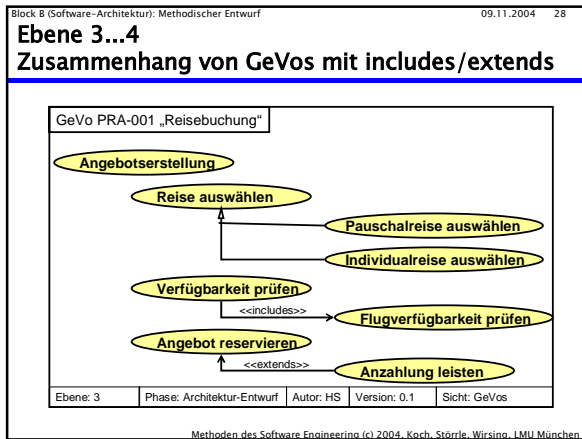
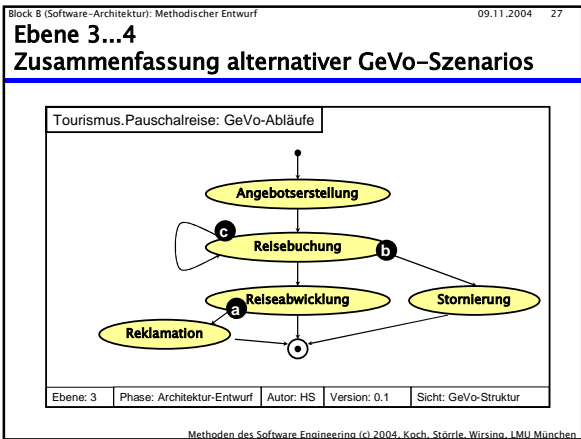
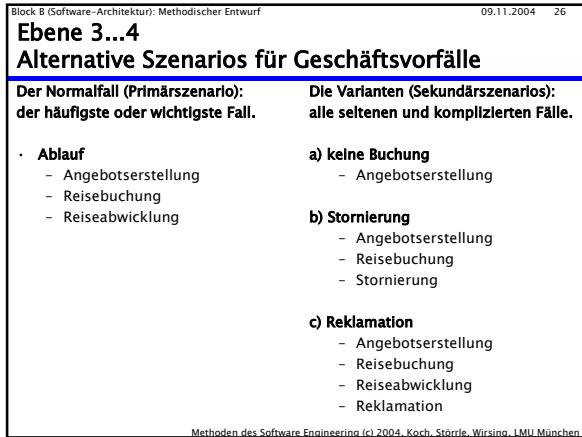
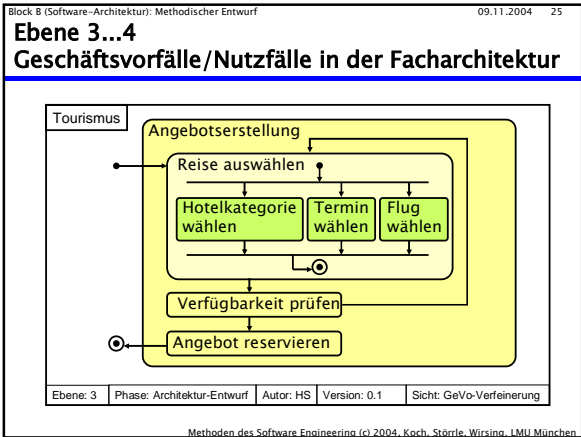
Angebotserstellung

- Reise auswählen
- Verfügbarkeit prüfen
- Angebot reservieren

...

Ebene: 3	Phase: Architektur-Entwurf	Autor: HS	Version: 0.1	Sicht: GeVo-Verfeinerung
----------	----------------------------	-----------	--------------	--------------------------

Methoden des Software Engineering (c) 2004, Koch, Störle, Wirsing, LMU München



Block B (Software-Architektur): Methodischer Entwurf 09.11.2004 30

### Ebene 3...4 GeVo-Details in Tabellenschema

GeVo PRA-001 „Reisebuchung“	
Kurzbeschreibung <b>Ein Sachbearbeiter im Reisebüro oder Callcenter macht dem Kunden ein Angebot. Der Kunde bucht die Reise und leistet eine Anzahlung.</b>	
Aktivierer <b>Kunde akzeptiert Angebot</b>	Beteiligte Akteure <b>1. Kunde, 2. Sachbearbeiter</b>
Vorbedingung <b>Angebot ist verfügbar</b>	Häufigkeit <b>ca. 10<sup>4</sup> pro Tag und Vertriebskanal</b>
Primärszenario <b>1. Angebotserstellung 2. Reisebuchung 3. Reiseabwicklung</b>	Varianten <b>a) 4. Reklamation b) 3. Storno c) 3. Umbuchung 4. Reiseabwicklung</b>
Nachbedingung <b>Kapazitätsplanung angepasst</b>	Referenzen =
Ergebnis <b>Kunde hat Reservierung,</b>	Anmerkungen, offene Punkte =
Ebene: 3	Phase: Architektur-Entwurf
Autor: HS	Version: 0.1
Sicht: GeVos-Details	

Methoden des Software Engineering (c) 2004, Koch, Störle, Wirsing, LMU München

Block B (Software-Architektur): Methodischer Entwurf 09.11.2004 31

## Zusammenfassung Geschäftsvorfälle

- Definition**
  - Jeder GeVo besteht aus vielen Einzelschritten, die maschinell/ manuell von verschiedenen Systemen/Aktoren erbracht werden.
- Darstellung**
  - Die Abläufe und ihre Verfeinerung werden zunächst durch hierarchischen Text, später dann durch Aktivitätsdiagramme festgehalten.
  - Der Zusammenhang der GeVos wird durch GeVo-Diagramme dargestellt, die Details einzelner GeVos werden in einem vorgegebenen Schema tabellarisch angegeben.
- Herstellung**
  - Die Menge der GeVos und ihre Details sind entweder bekannt (Vorwissen, Vorgängersystem, Kunden-Dokumente), oder werden z.B. durch Rollenspiele und Interviews erhoben.

Methoden des Software Engineering (c) 2004, Koch, Störle, Wirsing, LMU München

Block B (Software-Architektur): Methodischer Entwurf 09.11.2004 32

## Ableitung von Schnittstellen Ausgangspunkt Facharchitektur

Diagram showing the derivation of interfaces from a domain architecture for 'Tourismus'. The diagram is organized into a grid with functional areas and specific services:

- Top Left:** - Pauschalreise, +Angebotserstellung, +Buchung, +Umbuchung, +Stornierung, +Reiseabwicklung
- Top Right:** +Gruppenreise, +Individualreise, +Partner, +In-/Exkasso
- Bottom Left:** -CRM, +Direktmarketing, +Kundenpflege, +Reklamation
- Bottom Right:** - Produktdefinition, +Reiseerstellung, +Katalogerstellung, +Einkauf, +Kapazitätsplanung

Metadata: Ebene: 3 | Phase: Architektur-Entwurf | Autor: HS | Version: 0.1 | Sicht: GeVos

Methoden des Software Engineering (c) 2004, Koch, Störle, Wirsing, LMU München

Block B (Software-Architektur): Methodischer Entwurf 09.11.2004 33

## Ableitung von Schnittstellen Ausgangspunkt Facharchitektur

Diagram showing the derivation of interfaces from a domain architecture for 'Tourismus'. It shows a central 'Reisesystem' box with various services and actors like 'Reisebüro' and 'Reiseveranstalter'.

Metadata: Ebene: 1 | Phase: Architektur-Entwurf | Autor: HS | Version: 0.1 | Sicht: Struktur/Schnittstellen

Methoden des Software Engineering (c) 2004, Koch, Störle, Wirsing, LMU München

Block B (Software-Architektur): Methodischer Entwurf 09.11.2004 34

## Ableitung von Schnittstellen Ausgangspunkt Facharchitektur

Diagram showing the derivation of interfaces from a domain architecture for 'Tourismus'. It highlights specific services like 'Pauschalreise' and 'Produktdef.' with red boxes.

Metadata: Ebene: 1 | Phase: Architektur-Entwurf | Autor: HS | Version: 0.1 | Sicht: Struktur/Schnittstellen

Methoden des Software Engineering (c) 2004, Koch, Störle, Wirsing, LMU München

Block B (Software-Architektur): Methodischer Entwurf 09.11.2004 35

## Ableitung von Schnittstellen von Facharchitektur nach Nutzfalkarte

Diagram showing the derivation of interfaces from a domain architecture for 'Tourismus'. It maps specific use cases like 'Reise buchen' and 'Preis ermitteln' to system components.

Metadata: Ebene: 1 | Phase: Architektur-Entwurf | Autor: HS | Version: 0.1 | Sicht: Struktur/Interaktion

Methoden des Software Engineering (c) 2004, Koch, Störle, Wirsing, LMU München

Block B (Software-Architektur): Methodischer Entwurf 09.11.2004 36

## Alternative: Aktivitätsdiagramme statt Nutzfalkarten

Diagram showing an alternative approach to deriving interfaces from a domain architecture for 'Tourismus'. It uses activity diagrams to show the flow of data and control between system components.

System-Schnittstellen  
Daten

Methoden des Software Engineering (c) 2004, Koch, Störle, Wirsing, LMU München

Block 8 (Software-Architektur): Methodischer Entwurf 09.11.2004 37

## Rückblick betrachtete Aspekte und Darstellungsmittel

<b>Kontext</b> Einbettung Quantifizierung	<b>Fachliches Klassenmodell</b> Zusammenhänge Objektlebenszyklus Klassendetails
<b>GeVo</b> hierarchischer Text GeVo-Diagramm GeVo-Tabelle Aktivitätsdiagramm	<b>Struktur</b> Facharchitektur Subsysteme & Schnittstellen (fachliche) Interaktionen

Methoden des Software Engineering (c) 2004, Koch, Störle, Wirsing, LMU München

Block 8 (Software-Architektur): Methodischer Entwurf 09.11.2004 38

## Klassenübersicht „Angebot/Buchung/Reise“

```

classDiagram
    class Produkt
    class Reise
    class Partner
    class Tarif
    class Produktteil
    class Reisteil

    Produkt "1" -- "*" Produktteil
    Reise "*" -- "1" Reisteil
    Partner "*" -- "*" Reisteil
    Tarif "1" -- "*" Produktteil
    Tarif "1" -- "*" Reisteil
  
```

Reise: Reisender (+), Partner (+)

Produktteil: Flug, Unterbringung, Verpflegung, Transfer, Zubringer, Mietwagen, Extras

Reisteil: Flug, Unterbringung, Verpflegung, Transfer, Zubringer, Mietwagen, Extras

Ebene: 3 | Phase: Fachliche Analyse | Autor: HS | Version: 0.1 | Sicht: Klassen-Details

Methoden des Software Engineering (c) 2004, Koch, Störle, Wirsing, LMU München

Block 8 (Software-Architektur): Methodischer Entwurf 09.11.2004 39

## Objektlebenslauf „Angebot/Buchung/Reise“

```

sequenceDiagram
    participant Angebot
    participant Buchung
    participant Reise

    Angebot -->> Angebot: reservieren
    Angebot -->> Angebot: bearbeiten, löschen
    Angebot -->> Buchung: buchen
    Buchung -->> Buchung: umbuchen
    Buchung -->> Reise: antreten
    Reise -->> Reise: verlängern, hochstufen, Extras_buchen
    Reise -->> Reise: beenden
    Reise -->> Buchung: abrechnen
    Buchung -->> Buchung: stornieren
  
```

Ebene: 3 | Phase: Fachliche Analyse | Autor: HS | Version: 0.1 | Sicht: Objekt-Lebenslauf

Methoden des Software Engineering (c) 2004, Koch, Störle, Wirsing, LMU München

Block 8 (Software-Architektur): Methodischer Entwurf 09.11.2004 40

## Klassendetails „Angebot/Buchung/Reise“

Angebot	Buchung	Reise
+ von: Datum + bis: Datum + ab: Ort + preis: Euro + anmerkungen: String	+ buchungsdatum: Datum + sachbearbeiter: Partner + agentur: Partner + anzahlung: Euro + restforderung: Euro + tickets_verschickt: bool	+ ist_abgerechnet: bool
+ buchen(): Buchung	+ tickets_ausstellen(): void	+ hochstufen(): Reise + verlängern(): Reise + beenden(): Reise + reklamieren(): Reise + abrechnen(): void
+ berechnen(): Euro + ist_verfügbar(): bool + reservieren(frist: Datum): void	+ stornieren(): Angebot + antreten(): Reise + umbuchen(): Buchung	

Ebene: 3 | Phase: Fachliche Analyse | Autor: HS | Version: 0.1 | Sicht: Klassen-Details

Methoden des Software Engineering (c) 2004, Koch, Störle, Wirsing, LMU München

Block 8 (Software-Architektur): Methodischer Entwurf 09.11.2004 41

## Rückblick betrachtete Aspekte und Darstellungsmittel

<b>Kontext</b> Einbettung Quantifizierung	<b>Fachliches Klassenmodell</b> Zusammenhänge Objektlebenszyklus Klassendetails
<b>GeVo</b> hierarchischer Text GeVo-Diagramm GeVo-Tabelle Aktivitätsdiagramm	<b>Struktur</b> Facharchitektur Subsysteme & Schnittstellen (fachliche) Interaktionen

Methoden des Software Engineering (c) 2004, Koch, Störle, Wirsing, LMU München

Block 8 (Software-Architektur): Methodischer Entwurf 09.11.2004 42

## Hilfsmittel 1: Repository Problem

- Ein **Repository** ist eine Datenbank für semantische komplexe Daten.
- CASE-Tools haben **zwangsläufig ein Repository eingebaut**. Dies kann
  - als Datenbank realisiert sein (z.B. Rational Rose, Aonix StP),
  - als XML-Datei gespeichert werden (z.B. Magic Draw, ArguUML),
  - oder im Code versteckt sein (Together).
- Oft kann kein Repository benutzt werden. Mögliche Gründe:**
  - Es ist keines (auf der „richtigen“ Plattform) vorhanden.
  - Es darf nicht für alle Zwecke/von allen Personen genutzt werden.
  - Es ist aufwendig zu bedienen, und man soll „nur mal eben schnell“... („Overkill“).
  - Es ist nicht (gut) integriert mit den anderen Werkzeugen.

Methoden des Software Engineering (c) 2004, Koch, Störle, Wirsing, LMU München

Block B (Software-Architektur): Methodischer Entwurf 09.11.2004 43

## Hilfsmittel 1: Repository

### Lösung: Primitiv-Repository

- Vorteile**
  - gibt's auf jeder Plattform
  - keinerlei technischer Aufwand, minimale Kosten
  - keine Genehmigung erforderlich
- Nachteile**
  - Vergabe von Identifiern & Zuständen wird nicht automatisch erledigt, dadurch Mehraufwand, und mögliche Inkonsistenzen
  - nebenläufiges Arbeiten schwierig
- Anzuwenden bei**
  - kleinen Projekten / in der Frühphase
  - in technisch rückständigen Umgebungen – d.h. in 70% aller SW-Buden

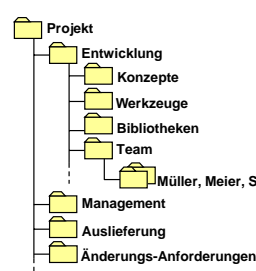
Methoden des Software Engineering (c) 2004, Koch, Störle, Wirsing, LMU München

Block B (Software-Architektur): Methodischer Entwurf 09.11.2004 44

## Hilfsmittel 1: Repository

### Lösung: Primitiv-Repository

#### Ablagestruktur



#### Dokumente/Modelle

Buchstabe: a, b, c  
Uhrzeit: hh-mm  
Datum: yyyy-mm-dd

<Name> <Versions>.doc

lfd. Nr.  
Typ  
Gruppe  
System

**Beispiele**  
•Bib-AUS\_GP\_007

Methoden des Software Engineering (c) 2004, Koch, Störle, Wirsing, LMU München

Block B (Software-Architektur): Methodischer Entwurf 09.11.2004 45

## Hilfsmittel 2: Glossar

- Ist von Anfang an zu führen, zu pflegen und zu verwenden!
- In Excel, Word o.ä. -> Sortierung der Einträge

Begriff	Synonyme	Erklärung(en)	Verantw.	Status
Leser	-	Natürliche Person, die mit Namen, Geburtsdatum und Adresse im Leser-verwaltungssystem erfasst ist.	Störle	ok

- Sollte zentral gepflegt werden
  - z.B. über einheitlich Email: glossar@Firma.com
  - 4-Augen-Prinzip (QS)
  - verfügbar im Intranet (-> Knowledge Management)

Methoden des Software Engineering (c) 2004, Koch, Störle, Wirsing, LMU München

Block B (Software-Architektur): Methodischer Entwurf 09.11.2004 46

## Hilfsmittel 2: Glossar

### Zweck & Gütekriterien

- Standardisierte Sprache
  - erhöht die Konsistenz, und
  - hilft Missverständnisse vermeiden.
- Technische Prosa darf nicht nur „langweilig“ sein, sie muß es!
- und bitte: kein Denglisch!  
„Um die Usage-Policy zu updaten muß zuerst das File vom User downgeloadet werden.“

Weitere Tips bei:  
Peter Rechenberg: Technisches Schreiben. Hanser, 2002

Methoden des Software Engineering (c) 2004, Koch, Störle, Wirsing, LMU München

Block B (Software-Architektur): Methodischer Entwurf 09.11.2004 47

## Literatur für B.1

- Hofmeister, Nord, Soni: Applied Software Architecture. Addison-Wesley, 2000
- Störle: Models of Software Architecture. Book-on-demand, 2001
- Sewell, Sewell: The Software Architect's Profession. Prentice Hall PTR, 2002
- Rechenberg: Technisches Schreiben. Hanser, 2002
- Mellor, Balcer: Executable UML. Addison-Wesley, 2002
- Marshall: Enterprise Modeling with UML. Addison-Wesley, 2000??

Methoden des Software Engineering (c) 2004, Koch, Störle, Wirsing, LMU München

Block B (Software-Architektur): Methodischer Entwurf 09.11.2004 48

## Gliederung B.1

- Vorstellung
- Einleitung in den Block B „Software-Architektur“
  - Definition, Gegenstand, Motivation
  - Gliederung und Einbettung in die Vorlesung
- Teil 1: Methodischer Entwurf
  - Notationen & Konzepte
  - Fallbeispiel
  - Hilfsmittel & Literatur
- Ausblick auf Teil 2: Systemarchitektur

Methoden des Software Engineering (c) 2004, Koch, Störle, Wirsing, LMU München

## Ausblick auf Block B Teil 2: Systemarchitektur

