

# ***Software-Engineering***

Sebastian Iwanowski  
FH Wedel

Klausurabgrenzung WS 2006 / 2007

# Klausurabgrenzung Software-Engineering

- 2**      **Grundbegriffe und Prinzipien**
  - Systeme und Modelle, Prinzipien der Abstraktion, modulare und hierarchische Zerlegung, Perspektivenbildung
- 3**      **Softwareplanung**
  - Lastenheft und Pflichtenheft: Zweck, wesentliche Merkmale, Unterschiede
- 4**      **Systemanalyse**
  - Einteilung nach Sichten,
- 4.1**      **Prozessorientierte Modellierungsmethoden:**
  - Funktionsbaum, Datenflussdiagramm, Entscheidungstabelle/-baum, Abgrenzung zu Kontrollflussdiagrammen.
  - EPKs: Modellierung einfacher Beispiele aus dem täglichen Leben
  - Petri-Netze: 3 unterschiedliche Typen, Erkennen der Typen an Beispielen, Modellierung einfacher Beispiele aus dem täglichen Leben
- 4.2**      **Datenorientierte Modellierungsmethoden:**
  - Entity-Relationship-Modellierung und Objektorientierte Modellierung: Wesentliche Begriffe und Prinzipien, Gemeinsamkeiten und Unterschiede, Interpretierung und Modellierung einfacher Beispiele in einer der beiden Modellierungstechniken, Konversionen zwischen diesen Techniken

# Klausurabgrenzung Software-Engineering

## 4 Systemanalyse

### 4.3 UML:

Unterschiede zwischen Systemanalyse und Softwareentwurf, Klassendiagramme, Use-Case-Diagramme, Sequenzdiagramme, Zustandsübergangsdigramme, Aktivitätsdiagramme (UML2), Interpretation und Modellierung kleiner Beispiele

### 4.4 Aris:

Aris-Haus mit den wesentlichen Sichten, EPKs (mit Erweiterung), Zusammenspiel von EPKs mit UML-Bausteinen in Aris  
Eignung von UML und Aris für die verschiedenen Entwicklungsphasen von Software

## 5 Softwareentwurf

Abgrenzung zur Systemanalyse, Entwurfsmethoden top-down und bottom-up, Leitlinien für die Modularisierung

# Klausurabgrenzung Software-Engineering

## 6 Projektmanagement

Grobe Zeitplanung:

Wasserfallmodell, Prototyping, Spiralmodell: Definitionen und die wesentlichen Vor- und Nachteile

Feine Zeitplanung:

RUP, XP: Wesentliche Prinzipien, Unterschiede und Gemeinsamkeiten, Vorteile der Chefprogrammiererorganisation zur streng hierarchischen Organisation

## 7 Aufwandsabschätzung

Zusammenhang von Kosten und Zeit bei Software, verschiedene Messgrößen  
Basismethoden: Gewichtungsmethode, parametrische Gleichungen, Multiplikatormethode, Analogiemethode, Relationsmethode, Kennzahlenverfahren, Prozentsatzverfahren (mit Einordnung in übergeordnete Ziele)

Allgemeine Schätzverfahren: Function-Point-Methode (mit Kenntnis des allgemeinen Verfahrens, Anwendungsmöglichkeiten, Vorteilen und Nachteilen),  
**keine Beispiele dazu nötig**

# Klausurabgrenzung Software-Engineering

## 8 Qualitätsmanagement

Qualität im Wettbewerb mit anderen Entwicklungskriterien, Klassifizierung der verschiedenen Qualitätskriterien für Software: Zusammenhang zwischen diesen, statische und dynamische Qualitätssicherungsverfahren

Dynamische Verfahren im Detail:

Blackbox und Whitebox-Tests, Grundsätzlicher Unterschied

Blackbox-Tests: Bestimmung der Testfälle, Strategien für die Testdatenauswahl, kleine Beispiele

Whitebox-Tests im Detail:

Kontrollflussbezogene Verfahren: Überdeckungskriterien, Zusammenhang zwischen diesen, kleine Beispiele