

**Klausur zur Vorlesung**  
**„Software Engineering“**

**2.3.2005**

**Prof. Dr. Ralf Möller**

Name, Vorname
Matrikelnummer
Hauptfach:
Ist meine Matrikelnummer auf der Liste (Ja/Nein)?
Habe ich einen Vorbehalt ausgefüllt (Ja/Nein)?
Unterschrift:

Bitte beachten Sie:

1. Überprüfen Sie, ob Ihre Matrikelnummer in der ausgehändigten Liste erscheint. Wenn nicht, so verfahren Sie wie folgt:
  - a. Füllen Sie vor Beginn der Klausur einen Vorbehalt aus. Diesen händigen Sie der Klausuraufsicht aus. Sie dürfen die Klausur sonst nicht antreten.
  - b. Füllen Sie außerdem den Formzettel zur Nachmeldung aus, der Ihnen ebenfalls von der Klausuraufsicht ausgehändigt wird. Lassen Sie diesen Formzettel von „Ihrem“ Prüfungsamt unterzeichnen und abstempeln. Bringen Sie den Formzettel dann in das STS-Sekretariat (Harburger Schloßstr. 20, 2. Stock, Frau Hantschmann).
2. Legen Sie Ihren Studentenausweis und Personalausweis auf den Tisch zwecks Einsichtnahme durch die Klausuraufsicht.
3. Sie haben 90 Minuten Zeit. Es dürfen keine zusätzlichen Quellen verwendet werden.
4. In der Klausur ist ausreichend Platz für Ihre Antworten vorgesehen.
5. Die Minutenangabe soll Ihnen helfen, den Bearbeitungsaufwand abzuschätzen.
6. Sollten Sie zusätzliches Papier benötigen, so melden Sie sich bei der Klausuraufsicht. Jeder zusätzliche Bogen muss Name und Matrikelnummer aufführen.
7. Sollten Sie den Prüfungsraum verlassen müssen, so melden Sie sich bei der Klausuraufsicht. Bleiben Sie dann solange ruhig sitzen, bis die Klausuraufsicht an Ihren Tisch tritt und Ihnen die Erlaubnis zum Verlassen des Raumes erteilt. Es darf nur eine Person zur Zeit abwesend sein.

**Mit dem Öffnen der Klausur bis zur Aufforderung warten! Viel Erfolg!**

Name:

ID:

**Product Line Software Engineering** (Pts: 2) ⊕ 3 Min.

	<i>wahr</i>	<i>falsch</i>
Ein Framework (z.B. für GUI Anwendungen) ist eine traditionelle wiederverwendbare Einheit. Product Line Assets sind dazu gedacht, die Wiederverwendung in einer Familie von verwandten Produkten zu ermöglichen	<input type="checkbox"/>	<input type="checkbox"/>
Man kann ein Asset in einer Produktlinie entwickeln ohne die Anforderungen der Geschäftsdomäne zu berücksichtigen. Gedanken über die Integration des Assets mit anderen kann man sich erst machen nachdem alle Assets als ausführbare Komponenten vorhanden sind.	<input type="checkbox"/>	<input type="checkbox"/>
Wenn ein Unternehmen Legacy Systeme hat, braucht man nur ein minor Reengineering um ein Product-Line basiertes Entwicklungsmodell einzusetzen	<input type="checkbox"/>	<input type="checkbox"/>
Reverse Engineering ist ein integraler Bestandteil von Product Line Software Engineering	<input type="checkbox"/>	<input type="checkbox"/>

**Reverse Engineering** (Pts: 4) ⊕ 4 Min.

Wäre es sinnvoll, sowohl Extreme Programming (XP) als auch Reverse Engineering in einem Projekt einzusetzen, indem nach jedem Release Cycle des XP Teams das andere Team die Softwarearchitektur des Releases ableitet (als UML-basiertes Entwurfmodell), so dass dann eine architekturbasierte, forward-engineering Implementierung stattfinden kann? Warum?

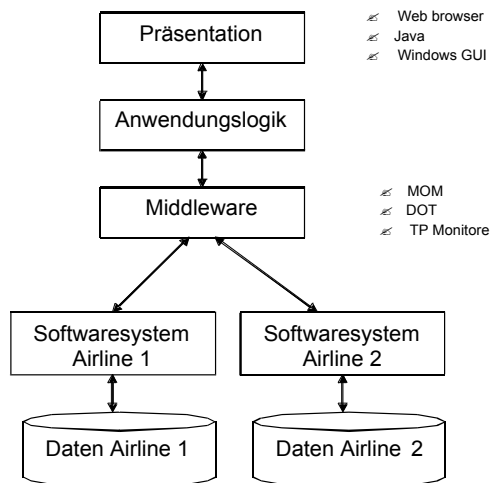
**Capability Maturity Model (CMM)** (Pts: 6) ⊕ 6 Min.

Das CMM des SEI ordnet Softwareunternehmen nach den Projektmanagementeigenschaften, die ihre Entwicklungsprozesse bestimmen. Die Stufen des Rankings sind: *Initial*, *Repeatable*, *Defined*, *Managed*, und *Optimising*. Erläutern Sie die Merkmale der *Defined*, *Managed*, und *Optimising* Kategorien.

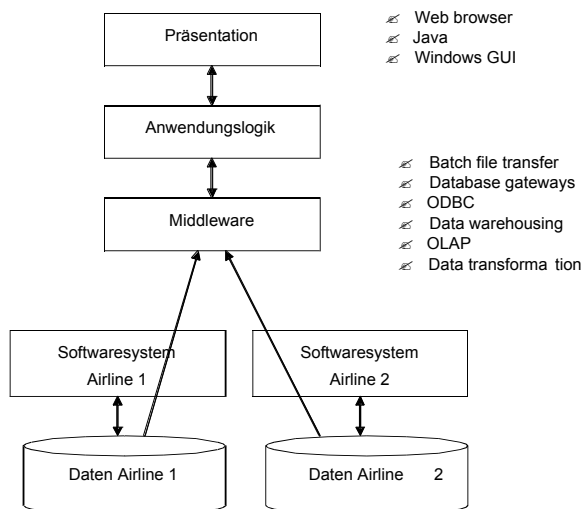
**Enterprise Application Integration (EAI) (Pts: 12) ⊕ 12 Min.**

Durch eine Vereinbarung verpflichten sich zwei Fluglinien, Tickets für gemeinsam bediente Routen gegenseitig als eigene anzuerkennen, um Kosten zu sparen und die Kundenzufriedenheit zu verbessern. Um Kapazitäten zu planen, muss jede Fluglinie über die von der anderen Fluglinie verkaufte Tickets für diese Routen (sowie über Stornierungen) benachrichtigt werden, sobald diese geschehen.

Erläutern Sie wie die Implementierung unter Berücksichtigung der drei Integrationsmodelle von EAI aussehen sollte. Als Hinweis sind die Vorlage der Integrationsmodelle unten zu sehen.



**Abbildung 1: Funktionales-Integrationsmodell**

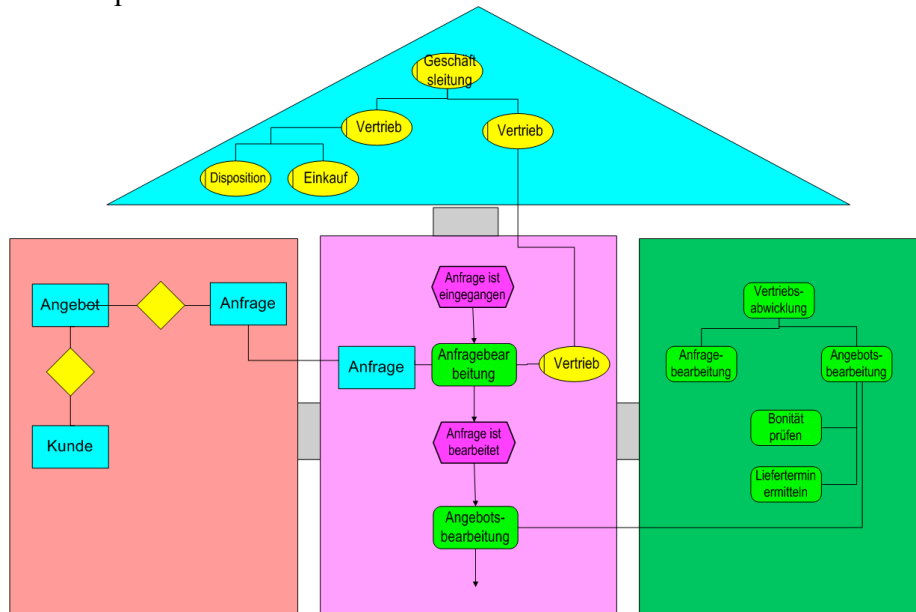


**Abbildung 2: Daten-Integrationsmodell**



**ARIS** (Pts: 8) ⌚ 8 Min.

Das folgende Diagramm (Abbildung 4) beinhaltet Beispiele verschiedener Visuellenotationen, die beim Modellieren mit ARIS benutzt werden. Bitte beschreiben Sie die visuellen Elemente, die in zwei dieser Sprachen benutzt sind.



**Abbildung 4**



Name:

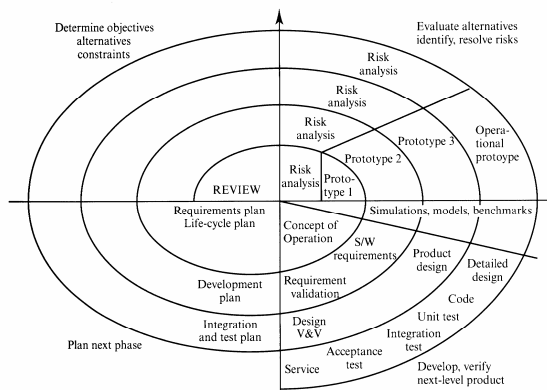
ID:

d) Erläutern Sie die folgende Konzepte: Software Testing, Software Verification, Software Validation.

e) Sie haben die Aufgabe bekommen die hausintern entwickelte CRM Webanwendung mit einer neuen standard ERP Lösung (z.B. SAP) zu integrieren. Die CRM Anwendung wird hauptsächlich für die Abwicklung der telefonischen Kundenabfragen benutzt. Welche Testmethode würden Sie benutzen um sicherzustellen, ob Ihre Integration während des normalen Betriebes die erwartete Leistung bringt? Erläutern Sie die Schwierigkeiten die man in diesem Bereich hat.

**Planung** (Pts: 4) ⌚ 6 Min.

Bitte beschreiben Sie die verschiedenen Phasen des Spiral Model.



Name:

ID:

**Aufwandsschätzung** (Pts: 6) ⊕ 7 Min.

a)

	<i>wahr</i>	<i>falsch</i>
Mehr Mitarbeiter erhöhen den Kommunikationsaufwand im Entwicklungsteam	<input type="checkbox"/>	<input type="checkbox"/>
Software-Entwicklung mit weitgehender Übernahme von existierender Software benötigen ungefähr 1/4 der Zeit von Neuentwicklungen.	<input type="checkbox"/>	<input type="checkbox"/>
Bei der Multiplikatormethode zu Aufwandsschätzung ist es eine umfangreiche, empirische Datensammlung und -auswertung erforderlich, um die zu berücksichtigenden Faktoren unternehmensspezifisch zu bewerten	<input type="checkbox"/>	<input type="checkbox"/>

b) Erläutern Sie die Kernidee der Aufwandsschätzung nach der Funktion-Point Methode.