

# Kurs 1793 “Software Engineering I”

## Hauptklausur am 05.08.2000

---

### Aufgabe 1 (25 Punkte)

#### Gründung eines E-Commerce-Unternehmens — ER-Analyse

Als Systemanalyst einer Softwarefirma besuchen Sie einen jungen Unternehmer, der Ihnen seine neue Geschäftsidee schildert und Sie mit der Entwicklung eines Softwaresystems beauftragen möchte:

„Ich möchte, dass die Besucher meiner Website kostenlos über das Internet Musik hören und diese Musik auch auf CD von mir kaufen können. Dazu soll es Künstlern möglich sein, ihre Musikaufnahmen auf meine Website hochzuladen und zu CDs zusammenzustellen. Die Künstler sollen am halben Verkaufserlös der CDs beteiligt werden und (als zusätzlichen Anreiz) für jeden (für die Hörer kostenlosen) Anhörvorgang auch eine (werbefinanzierte) Geldsumme erhalten. Dafür müssen natürlich die Anhörvorgänge genau protokolliert werden, am besten erlaubt man den Künstlern sogar Einsicht in diese Statistik. Einmal pro Quartal soll das (nach der Anzahl der Anhörvorgänge und CD-Bestellungen berechnete) Einkommen jedes Künstlers auf ein von ihm benanntes Konto überwiesen werden...“

An dieser Stelle unterbrechen Sie den eifrigen Unternehmer und beginnen eine systematische Analyse der Datensicht des Problems durch gezielte Fragen. Während des Gesprächs machen Sie sich folgende Notizen:

- Es gibt zwei Arten von Personen: Von den Hörern ist nur die E-Mailadresse bekannt, von den Künstlern auch Name und Anschrift.
- Ein Künstler kann eine oder mehrere Webseiten erstellen und auf jede Webseite eine oder mehrere Musikaufnahmen hochladen. Es ist nicht vorgesehen, dass von mehreren Seiten aus auf dieselbe Musikaufnahme verwiesen wird.
- Der Künstler kann eine oder mehrere Aufnahmen zu einer CD zusammenstellen. Nicht jede Aufnahme muss zu einer CD gehören, sie kann jedoch Teil von verschiedenen CDs sein.
- Ein Hörer kann Musikaufnahmen anhören, der Anhörvorgang wird mit Datum und Uhrzeit festgehalten. Ein Hörer kann das gleiche Stück auch mehrmals anhören, das sind dann aber verschiedene Anhörvorgänge.
- Ein Hörer kann auch eine oder mehrere CDs eines Künstlers bestellen. Jede Bestellung bezieht sich aber nur auf eine CD.
- Jeder Künstler benennt ein Konto, auf das er quartalsweise eine aus Anhörvorgängen und CD-Bestellungen resultierenden Vergütung überwiesen bekommt. Die Anhörvorgänge und CD-Bestellungen bleiben auch nach der Auszahlung weiterhin gespeichert.

## Kurs 1793 “Software Engineering I”

Hauptklausur am 05.08.2000

- 
- Der Künstler kann den Preis seiner CDs jederzeit individuell verändern. Bei einer Bestellung gilt der zu diesem Zeitpunkt aktuelle Preis der CD.

### *Aufgabenstellung:*

Geben Sie ausgehend von den obigen Notizen ein möglichst einfaches ER-Modell an, das aber ausreicht, um die folgenden Anfragen zu beantworten:

1. Welche Kontonummer hat Künstler K?
2. Welche URLs (WWW-Adressen) haben die Webseiten des Künstlers K?
3. Sind die Musikaufnahmen M1 und M2 auf der gleichen Webseite?
4. Welche Musikaufnahmen hat Künstler K auf CDs zusammengestellt?
5. Wie viele Minuten Musik sind auf der CD X?
6. Was ist der aktuelle Preis der CD X?
7. Wieviel Anhörvorgänge gab es im Juni 2000 für die Musikaufnahme M des Künstlers K?
8. Welche Hörer haben bisher die CD X des Künstlers K bestellt?
9. Wieviel muss Hörer H im Juni 2000 für bestellte CDs bezahlen?
10. Welche Künstler erzielten im Juni 2000 mehr Einnahmen durch Anhörvorgänge als durch CD-Bestellungen?

Geben Sie pro Entität durchschnittlich 2 Attribute an.

## Kurs 1793 “Software Engineering I”

### Hauptklausur am 05.08.2000

### Aufgabe 2 (15 + 6 + 4 = 25 Punkte)

#### Abiturplanungssystem — ER-Transformation in einen Entwurf

Aus der Musterlösung zur Aufgabe 1 der Probeklausur kennen Sie folgendes ER-Diagramm eines Abiturplanungsystems (AbiPS):

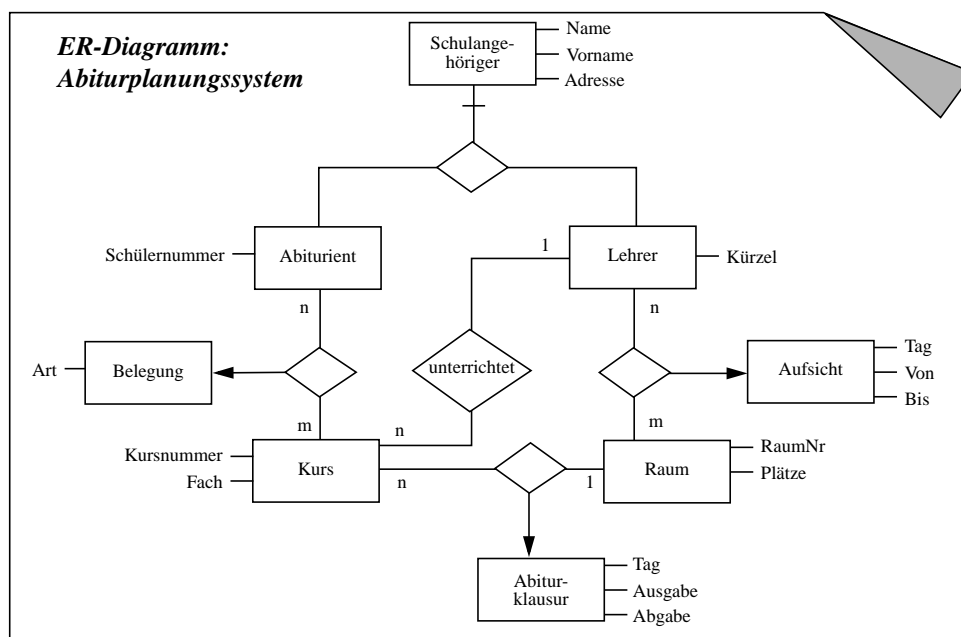


Abb. 2.1 ER-Diagramm des AbiPS

- Transformieren Sie dieses ER-Diagramm nach dem im Kurstext (Kap. 15) vorgestellten Schema in eine modulare Grobarchitektur, indem Sie die entstehenden Teilsysteme und Moduln graphisch skizzieren. Die in Teilsystemen enthaltenen Moduln brauchen nicht dargestellt zu werden. Erläuterungen und besonders übersichtliches Layout sind nicht notwendig.
- Man kann die Entität „Schulangehöriger“ aus dem ER-Modell entfernen, wenn man ihre Attribute („Name“, „Vorname“ und „Adresse“) bei den Subentitäten „Abiturient“ und „Lehrer“ ergänzt. Skizzieren Sie den modularen Entwurf, der sich durch Transformation des in dieser Weise geänderten ER-Modells ergibt.
- Nennen Sie für die beiden Varianten aus a) und b) jeweils einen Vorteil und geben Sie an, für welchen Entwurf Sie sich entscheiden würden.



# Kurs 1793 "Software Engineering I"

## Hauptklausur am 05.08.2000

**MODULSPEZIFIKATION ADT** Berechtigungsschablone;

### SCHNITTSTELLENSPEZIFIKATION

#### IMPORTE

**IMPORTIERE** TKontoNr, TKundenID;

#### EXPORTE

##### DATENTYPEN

TBerechtigung;

##### OPERATIONEN

Create : TBerechtigung,

Dispose (**INOUT** ZB: TBerechtigung),

IsFull (**IN** ZB: TBerechtigung): BOOLEAN,

IsEmpty (**IN** ZB: TBerechtigung): BOOLEAN,

Einfügen (**INOUT** ZB : TBerechtigung; **IN** KtoNr : TKontoNr; KID : TKundenID),

Entfernen (**INOUT** ZB : TBerechtigung; **IN** KtoNr : TKontoNr; KID : TKundenID),

IstBerechtig (**IN** ZB : TBerechtigung; KtoNr : TKontoNr; KID : TKundenID): BOOLEAN,

ErsterBerechtigter (**INOUT** ZB : TBerechtigung;

**IN** KtoNr : TKontoNr; **OUT** KID : TKundenID) : BOOLEAN,

NächsterBerechtigter (**INOUT** ZB : TBerechtigung;

**IN** KtoNr : TKontoNr; **INOUT** KID : TKundenID) : BOOLEAN,

ErstesVerfügbaresKonto (**INOUT** ZB : TBerechtigung;

**IN** KID : TKundenID; **OUT** KtoNr : TKontoNr) : BOOLEAN,

NächstesVerfügbaresKonto (**INOUT** ZB : TBerechtigung;

**IN** KID : TKundenID; **INOUT** KtoNr : TKontoNr) : BOOLEAN;

#### SEMANTIK

(\* Create, Dispose, IsFull, IsEmpty mit üblicher Semantik \*)

(\* Sei ZB eine Relation zwischen Berechtigten und Konten \*)

Einfügen: Vorbed.:  $\neg$  IsFull (ZB)

Nimmt (KtoNr, KID) in die Relation ZB auf, wenn nicht bereits enthalten

Nachbed.:  $\text{InRelation}(\text{ZB}, \text{KID}, \text{KtoNr}) \wedge \neg \text{IsEmpty}(\text{ZB});$

Entfernen: Vorbed.: keine

Löscht (KtoNr, KID) in der Relation ZB, falls vorhanden

Nachbed.:  $\neg \text{InRelation}(\text{ZB}, \text{KID}, \text{KtoNr});$

IstBerechtig: Vorbed.: keine

Prüft, ob (KtoNr, KID) in der Relation ZB vorhanden ist.

Bezeichne *exists* den booleschen Rückgabeparameter, dann gilt:

Nachbed.:  $\text{exists} \Leftrightarrow \text{InRelation}(\text{ZB}, \text{KID}, \text{KtoNr});$

ErstesVerfügbaresKonto:

Vorbed.: keine

Liefert zum Kunden mit der KID das erste verfügbare Konto, falls dieses existiert.

Bezeichne *exists* den booleschen Rückgabeparameter, dann gilt:

Nachbed.:  $(\text{exists} \Rightarrow (\text{RückgabeADO KtoNr existiert}) \wedge \text{InRelation}(\text{ZB}, \text{KID}, \text{KtoNr}))$   
 $\wedge (\neg \text{exists} \Rightarrow \text{RückgabeADO KtoNr existiert nicht});$

NächstesVerfügbaresKonto:

Vorbed.:  $\text{InRelation}(\text{ZB}, \text{KID}, \text{KtoNr})$

Ermöglicht zusammen mit der Operation ErstesVerfügbaresKonto die Rückgabe aller verfügbaren Konten. Liefert die Nummer des Kontos, das in der internen Liste auf das Konto mit der übergebenen KtoNr folgt.

Bezeichne *exists* den booleschen Rückgabeparameter, dann gilt:

Nachbed.:  $(\text{exists} \Rightarrow (\text{RückgabeADO KtoNr existiert}) \wedge \text{InRelation}(\text{ZB}, \text{KID}, \text{KtoNr}))$   
 $\wedge (\neg \text{exists} \Rightarrow \text{RückgabeADO KtoNr existiert nicht});$

(\*\*\*\*\*)

**Kurs 1793 “Software Engineering I”**  
Hauptklausur am 05.08.2000

```
RUMPFESPEZIFIKATION  
IMPORTE  
  IMPORTIERE GenRelSchablone;  
INSTANTIIERUNG  
  GENERIERE Berechtigungsschablone AUS GenRelSchablone MIT  
  DATENTYPEN  
    TUrbbild => TKundenID,  
    TBild => TKontoNr;  
UMBENENNUNGEN  
  TGenRelation [TKundenID, TKontoNr] => TBerechtigung,  
  InRelation => IstBerechtigt,  
  ErstesBild => ErsterBerechtigter,  
  NächstesBild => NächsterBerechtigter,  
  ErstesUrbbild => ErstesVerfügbaresKonto,  
  NächstesUrbbild => NächstesVerfügbaresKonto;  
SPEZIFIKATIONSENDE MODUL Berechtigungsschablone.
```

*Abb. 3.2 Das Teilsystem Zeichnungsberechtigung*

a) Vervollständigen Sie die Modulspezifikation des ADO-Moduls **Berechtigung** auf der folgenden Seite an den mit (\* ? \*) bezeichneten Stellen. Zur Verringerung der Schreibarbeit dürfen Sie bei den Exporten die Operationen **ErsterBerechtigter** und **NächsterBerechtigter** weglassen und brauchen nur die zwei angegebenen Prozessspezifikationen auszuführen.

# Kurs 1793 “Software Engineering I”

## Hauptklausur am 05.08.2000

**MODULSPEZIFIKATION ADO** Berechtigung;

**SCHNITTSTELLENSPEZIFIKATION**

**IMPORTE**

(\* ? \*)

**EXPORTE**

**DATENTYPEN**

(\* ? \*)

**OPERATIONEN**

(\* ? \*)

**SEMANTIK**

Die Semantik der Operationen kann von den entsprechenden  
Operationen des ADT-Modul Berechtigungsschablone übertragen werden.

(\*\*\*\*\*)

**RUMPFESPEZIFIKATION**

**IMPORTE**

(\* ? \*)

(\* ? \*)

*Prozessspezifikation* Create

(\* ? \*)

**END** Create;

*Prozessspezifikation* NächstesVerfügbaresKonto

(\* ? \*)

**END** NächstesVerfügbaresKonto;

... (\* weitere Prozessspezifikationen, die in dieser Aufgabe nicht verlangt sind \*)

**SPEZIFIKATIONSENDE MODUL** Berechtigung.

Abb. 3.3 Der ADO-Modul Berechtigung

## Kurs 1793 “Software Engineering I”

### Hauptklausur am 05.08.2000

b) Der Funktionsmodul **Berechtigungsverwaltung** soll für statistische Zwecke eine Operation **AnzahlKundenOhneBerechtigung** bekommen, die die Anzahl der Kunden liefert, die für kein einziges Konto zeichnungsberechtigt sind. Die Operation **AnzahlKundenOhneBerechtigung** soll sich auf dem ADO-Modul **Berechtigung** und dem Teilsystem **Kunden** abstützen, dessen Spezifikation Sie auszugsweise der Abb. 3.4 entnehmen können.

Ergänzen Sie bitte in der Modulspezifikation des Moduls **Berechtigungsverwaltung** (Abb. 3.5) die Prozessspezifikation **AnzahlKundenOhneBerechtigung** und die hierfür notwendigen Rumpfimporte.

*Hinweis:* Denken Sie daran, nicht mehr benötigte Objekte wieder zu löschen!

#### TEILSYSTEMSPEZIFIKATION TS Kunden;

##### SCHNITTSTELLENSPEZIFIKATION

###### IMPORTE

...

###### EXPORTE

###### DATENTYPEN

TKunde, TKundenID;

###### OPERATIONEN

CreateKunde (...): TKunde,

DisposeKunde (**INOUT** Kunde : TKunde),

... (\* -- weitere ADT-Basisoperationen sowie Lese- und Schreiboperationen  
-- auf den Kundendaten \*)

DisposeKundenID (**INOUT** KundenID : TKundenID),

DuplicateKundenID (**IN** KundenID) : TKundenID,

... (\* -- weitere Operationen auf KundenIDs \*)

Suche (**IN** KundenID : TKundenID; **OUT** Kunde : TKunde) : BOOLEAN,

ErsterKunde (**OUT** KundenID : TKundenID) : BOOLEAN,

NächsterKunde (**INOUT** KundenID : TKundenID) : BOOLEAN,

###### SEMANTIK

Die Basisoperationen auf Kunden und KundenIDs haben die gewöhnliche Semantik.

Suche:

Vorbed.: keine

Liefert den Kunden mit der gegebenen KundenID.

Bezeichne Kunde<sub>neu</sub> den INOUT-Parameter Kunde nach dem Aufruf und  
exists den booleschen Rückgabeparameter, dann gilt:

Nachbed.: (exists ⇔ Kunde<sub>neu</sub> existiert)

ErsterKunde:

Vorbed.: keine

Liefert die ID des ersten Kunden, falls einer existiert.

Bezeichne exists den booleschen Rückgabeparameter, dann gilt:

Nachbed.: (exists ⇔ KundenID<sub>neu</sub> existiert)

NächsterKunde:

Vorbed.: Kunde existiert

Ermöglicht zusammen mit der Operation ErsterKunde die Rückgabe aller  
KundenIDs. Liefert die ID des Kunden, der in der internen Liste auf die übergebene KundenID folgt.

Bezeichne KundenID<sub>neu</sub> den INOUT-Parameter KundenID nach dem Aufruf und  
exists den booleschen Rückgabeparameter, dann gilt:

Nachbed.: (exists ⇔ KundenID<sub>neu</sub> existiert)

Abb. 3.4 Auszug aus der Spezifikation des Teilsystems Kunden



# Kurs 1793 “Software Engineering I”

## Hauptklausur am 05.08.2000

**MODULSPEZIFIKATION FM** Berechtigungsverwaltung;  
**SCHNITTSTELLENSPEZIFIKATION**

**EXPORTE**

**OPERATIONEN**

AnzahlKundenOhneBerechtigung : CARDINAL;

...

**SEMANTIK**

(\* -- siehe Aufgabenstellung \*)

(\*\*\*\*\*)

**RUMPFESPEZIFIKATION**

**IMPORTE**

(\* ? \*)

*Prozeßspezifikation* AnzahlKundenOhneBerechtigung

(\* ? \*)

**END** AnzahlKundenOhneBerechtigung;

... (\* -- weitere Operationen, die nicht Bestandteil der Aufgabenstellung sind \*)

Abb. 3.5 Der FM-Modul Berechtigungsverwaltung

**Aufgabe 4** (5 + 5 + 7 + 4 + 4 = 25 Punkte)**Nullstellen — Funktionaler Test**

Die Nullstellen der quadratischen Gleichung  $x^2 + px + q = 0$  lassen sich bekanntlich nach folgender Formel bestimmen:

$$x_{1,2} = -\frac{p}{2} \pm \sqrt{\left(\frac{p}{2}\right)^2 - q}$$

Die Prozedur `NullstellenBestimmen` berechnet die Nullstellen dieser Gleichung in Abhängigkeit von  $p$  und  $q$ . Sie hat folgenden Prozedurkopf:

```
PROCEDURE NullstellenBerechnen(inP: REAL;
                              inQ: REAL;
                              outAnzahl: CARDINAL;
                              outErsteNullstelle: REAL;
                              outZweiteNullstelle: REAL);
(* berechnet die Nullstellen der quadratischen Gleichung x^2+px+q=0
in Abhängigkeit von p und q.
outAnzahl enthält die Anzahl der Nullstellen.
outErsteNullstelle enthält die erste Nullstelle, falls es mind. eine gibt,
sonst 0.
outZweiteNullstelle enthält die zweite Nullstelle, falls es zwei gibt,
sonst 0. *)
```

**Aufgabenstellung:**

Führen Sie einen funktionalen Test aus, indem Sie die folgenden Teilaufgaben bearbeiten:

- Geben Sie geeignete Äquivalenzklassen für den funktionalen Test an (drei bis fünf Äquivalenzklassen reichen).
- Geben Sie für jede Äquivalenzklasse ein Testdatum an (bestehend aus Eingabe und erwarteter Ausgabe).
- Stellen Sie sich eine einfache Implementierung der Prozedur `NullstellenBerechnen` vor und zeichnen Sie deren Kontrollflussgraph.
- Geben Sie die Pfade an, die bei den unter b) ermittelten Testdaten durchlaufen werden. Ist Anweisungs- und Zweigüberdeckung vorhanden?
- Nachdem die Prozedur `NullstellenBerechnen` schon längere Zeit im industriellen Einsatz ist, erreicht Sie die Nachricht, dass die Prozedur - obwohl sie gut getestet war - bei manchen Eingaben falsch arbeitet (z.B.  $\text{inP}=20000.0$ ,  $\text{inQ}=99999999.0$ ). Wie konnte das trotz der Tests passieren?