



Halle, 06. Dezember 2005

Software-Engineering (WS 2005/2006)

Übungsserie 7

Aufgabe 1 (UML, OCL)

Betrachten Sie das Beispiel aus Aufgabe 2 der 11. Übungsserie.

- Auf die Assoziation zwischen Konto und Kontovertrag kann verzichtet werden, da vom Konto über Kunde zu Kontovertrag navigiert werden kann. Dann muß aber sichergestellt werden, daß die Navigation von Konto zu Kontovertrag den gleichen Vertrag liefert, wie die Navigation über den Kunden. Drücken Sie diese Zusicherung(en) in UML aus.

- Abhängigkeiten zwischen Assoziationen können auch durch `{or}` ausgedrückt werden.

In unserem betrachteten Banksystem wird zwischen Privatkunde und Geschäftskunde unterschieden, damit kann auch ein Konto eine entsprechende Zugehörigkeit haben. Eigentümer eines Kontos sind ein oder mehrere Privatpersonen. Es sind aber auch ein oder mehrere Unternehmen als Eigentümer denkbar. Inhaber eines Kontos dürfen aber nicht gleichzeitig ein Privatkunde und ein Geschäftskunde sein. Beschreiben Sie das zugehörige Klassendiagramm mit UML.

- Jetzt gibt es in dem System nur noch Kunden, die als Eigentümer mindestens ein Konto besitzen. Ein Konto hat genau einen Eigentümer. Das Konto stellt das Vermögen des Kunden dar. Die Beziehung Kunde - Konto, die Kontobedingung, legt fest, um welche Art von Konto es sich handelt, welcher monatliche Betrag als Gutschrift erfolgen muß und welcher Verfügungsrahmen besteht.

Geben Sie das entsprechende Klassendiagramm an.

Aufgabe 2 (UML, OCL)

Beim Monopoly kann ein Spieler bei einer Auktion eine Straße nicht an sich verkaufen. Geben Sie dafür das Klassendiagramm mit einer Klasse an und die Bedingung in Form eines OCL-Ausdrucks.

Aufgabe 3 (OCL)

Betrachtet werde das abgebildete Klassendiagramm.

Beschreiben Sie natürlichsprachlich, welche Forderungen durch die folgenden OCL-Constraints ausgedrückt werden.

a. Kredit

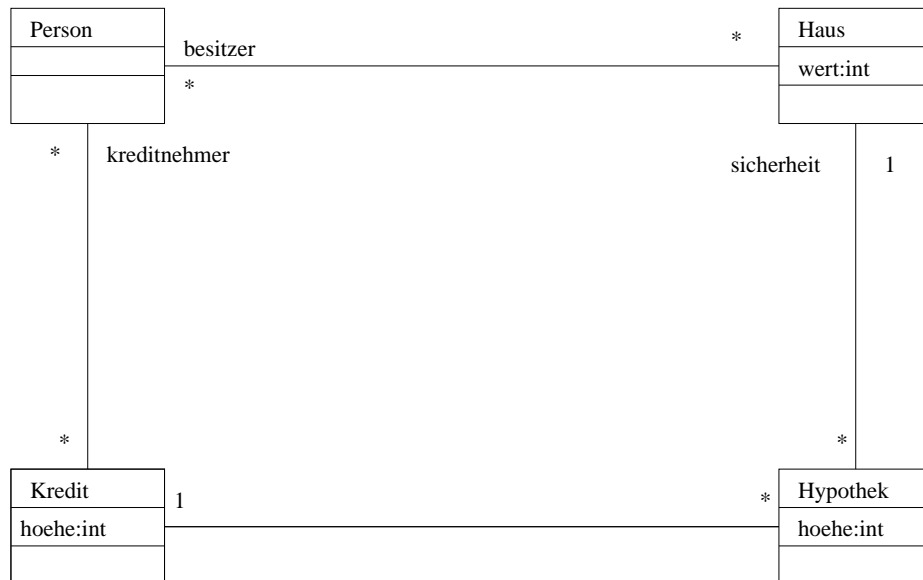
`self.kreditnehmer->includesAll(self.hypothek.sicherheit.besitzer)`

b. Haus

`self.wert >= self.hypothek->collect(hoehe)->sum`

c. Kredit

`self.hoehe = self.hypothek->collect(hoehe)->sum`



Besprechung ist am 28. Januar 2005.