

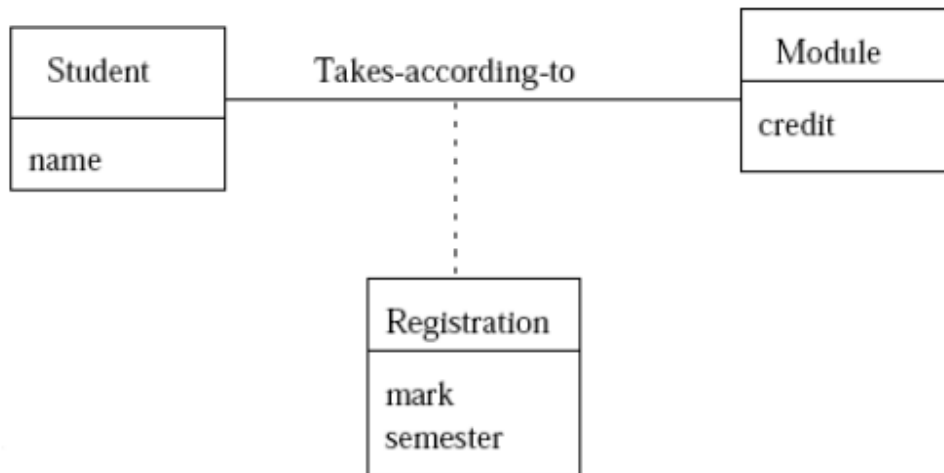
Im Unified Process werden alle Anforderungen in UseCases dargestellt; aus diesen heraus das Klassen, Sequenz und Kollaborationsdiagramm;

Es ist zu unterscheiden ob es wenige, dafür komplexe Use Cases gibt, oder ob es viele kleine Use Cases geben soll;

Nach der Ersten phase mit groben Ideen gibt es mehrer Verfeinerungsstufen; dabei kann man immer detailliert werden, ev. auch externe aktoren (z.B.: die Daten, Validierungen, ... von aussen liefern)

Die Noun Methode liefert Klassennamen;

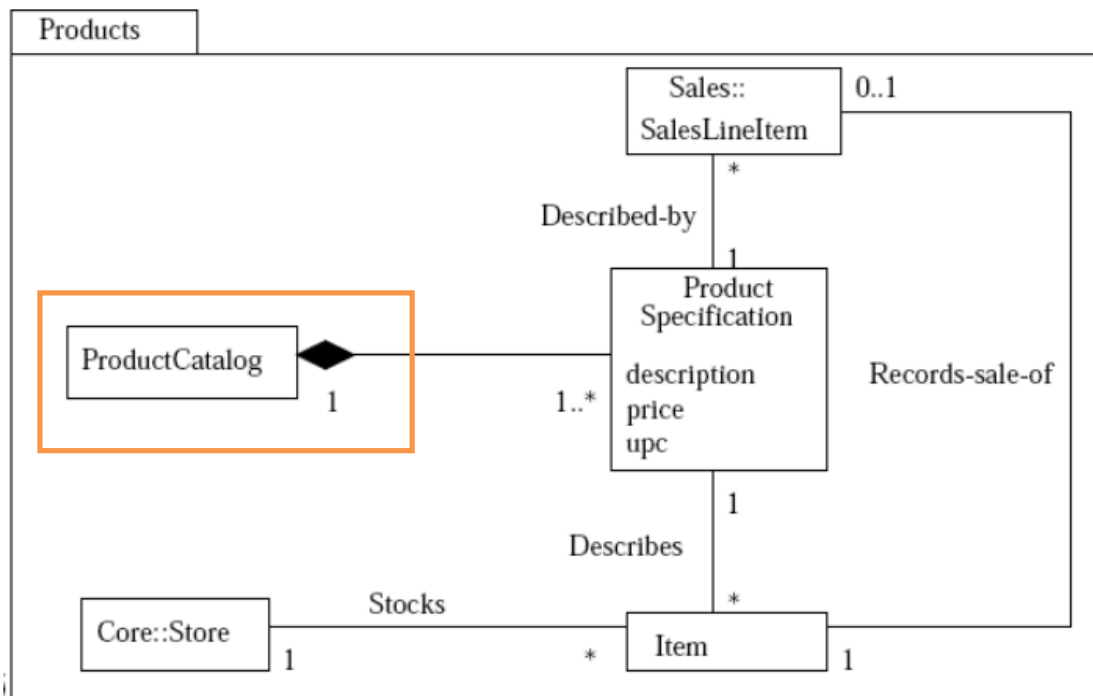
Assoziationsklasse: d.h. es besteht eine Assoziation zwischen zwei Klassen, diese hat jedoch selbst auch attribute.



Die Note passt sowohl zum Studenten als auch zum Modul das er besucht; ein Student wird jedoch pro Modul eine Note bekommen; daher gibt man der Assoziation die Note mit;

Ein Qualifier wird verwendet um ein Objekt eindeutig anzusprechen (Indexfunktion); Im Gespräch mit dem Kunden ist es ev. sinnvoll Produktnummern, ... etc zu verwenden, nicht irgendwelche UUIDs oder DBKeys.

Gängig ist die Verwendung bei 1 ... * Relationen; durch Verwenden des Qualifiers wird diese Relation fast immer eine 1 zu 1 Relation.



Hier waere der UPC (unified product counter) fuer die Klasse ProductCatalog als Qualifier sinnvoll.

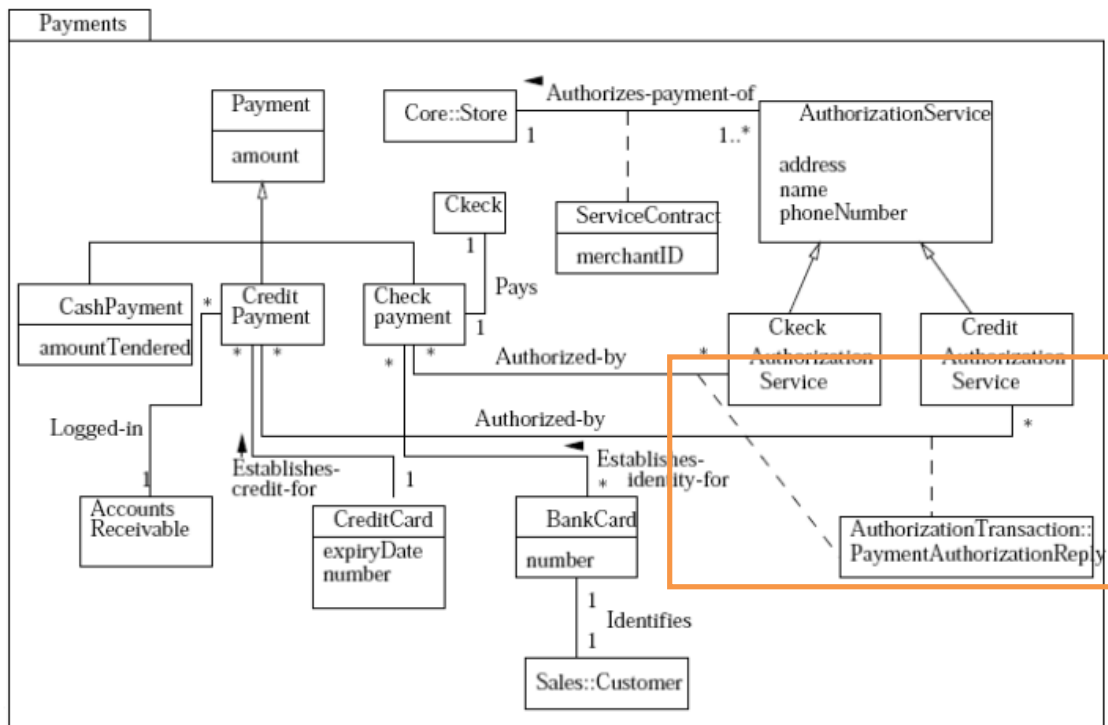
Packages werden verwendet um Software zu gliedern;

Innerhalb eines Packages kann man auf eine Klasse eines anderen packages zugreifen, dazu den Identifier [Packagename>::[Klassenname] verwenden; Innerhalb des zugreifenden Packages darf/kann man die Klasse nicht aendern, aber durchaus assoziationen setzen;

Packages koennen durchaus von anderen abhaengen

Packages haben nichts mit Komponenten zu tun; sie sind rein zur Strukturierung des codes gedacht.

Oft werden in fruehen Phasen bereits packages verwendet; eventuell entstehen aus ihnen heraus spaeter die Komponenten;



Hier ist eine gute Anwendung fuer eine Assoziationsklasse; die Daten die der Assoziation mitgegeben werden muessen werden in der Klasse zusammengefasst; z.B.: zur Realisierung eines Uebertragungsprotokolls.

State Chart diagramme sind keine Sequenz oder Kollaborationsdiagramme;
 Das State Chart diagramm beschreibt immer das gesamte System aus Sicht von aussen;
 Die Ablaeufe von hier werden in einer z.B.: Managerklasse implementiert, wie eben ein UseCase Controller