Adding attributes to classes

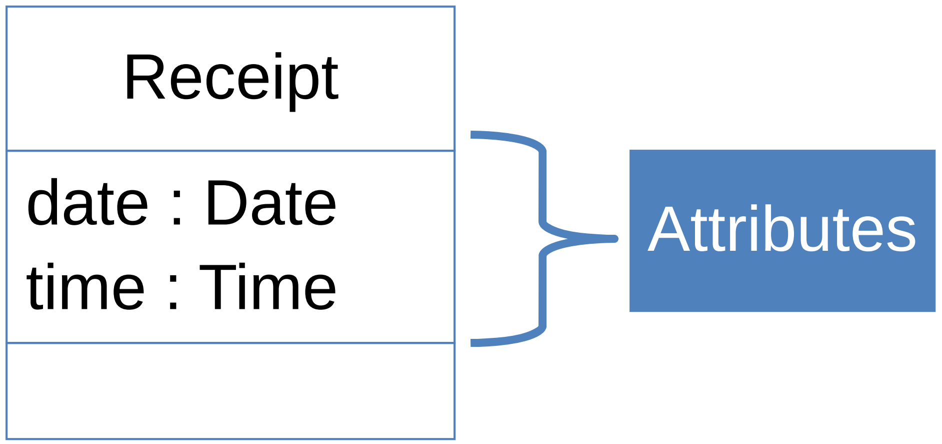
COMP 3831

Larman: Chapter 9

Attributes

- An attribute is a logical data value of an object
- Include attributes for which the requirements (example: use cases) suggest or imply a need to remember information
  - Example: a receipt (which represents information of a sale) includes date and time which management needs to know about for a variety of reasons.
- Data value held by object
- Collectively store the state of the object

UML attribute Notation



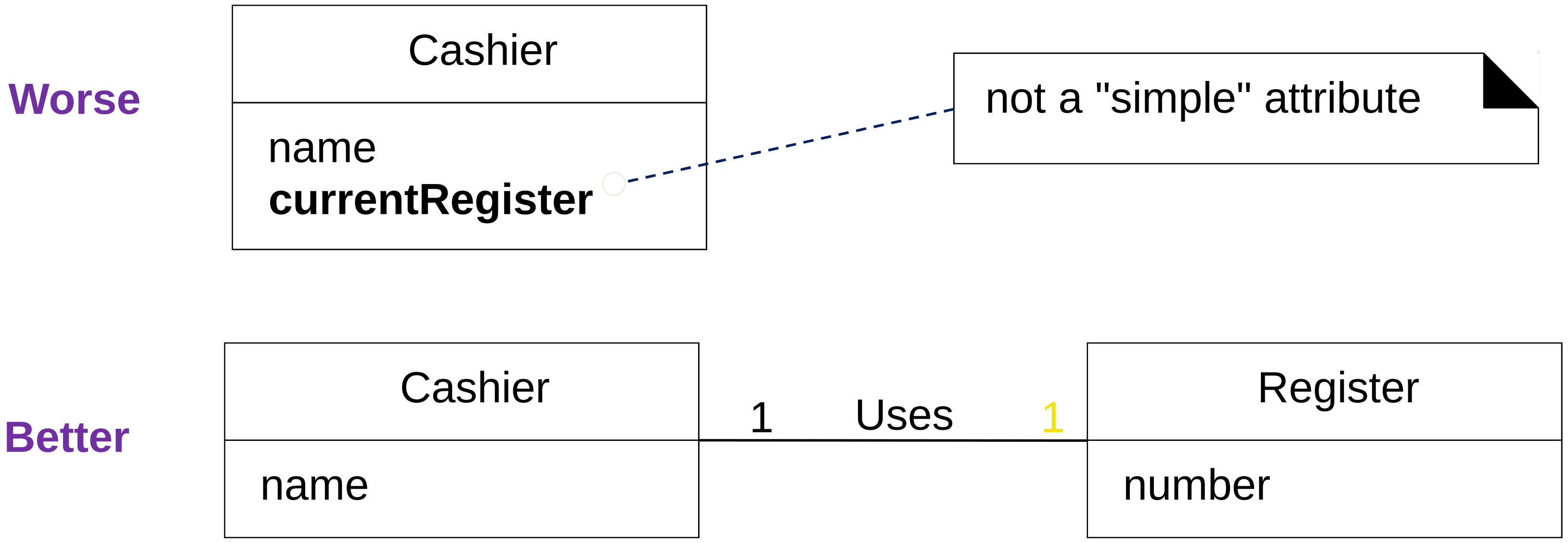
- Attributes are shown in the second compartment of the class box
- Their type may be optionally shown

Attribute types in conceptual model

- The type of an attribute should NOT normally be a complex domain concept such as a *Sale* or *Airport*.
- Attributes should be preferably be simple attributes or data types.
  - Example:
    - Boolean, Date, Number, String (Text), Time, etc…
  - Other common types include:
    - Address, Color, PhoneNumber, SocialSecurityNumber, UniversalProductCode (UPC), PostalCode, etc.…

Attribute verses Association (conceptual)

- Avoid representing complex domain concepts as attributes – use associations instead
  - The most useful way to express that a cashier uses a Register is with an association, not with an attribute.

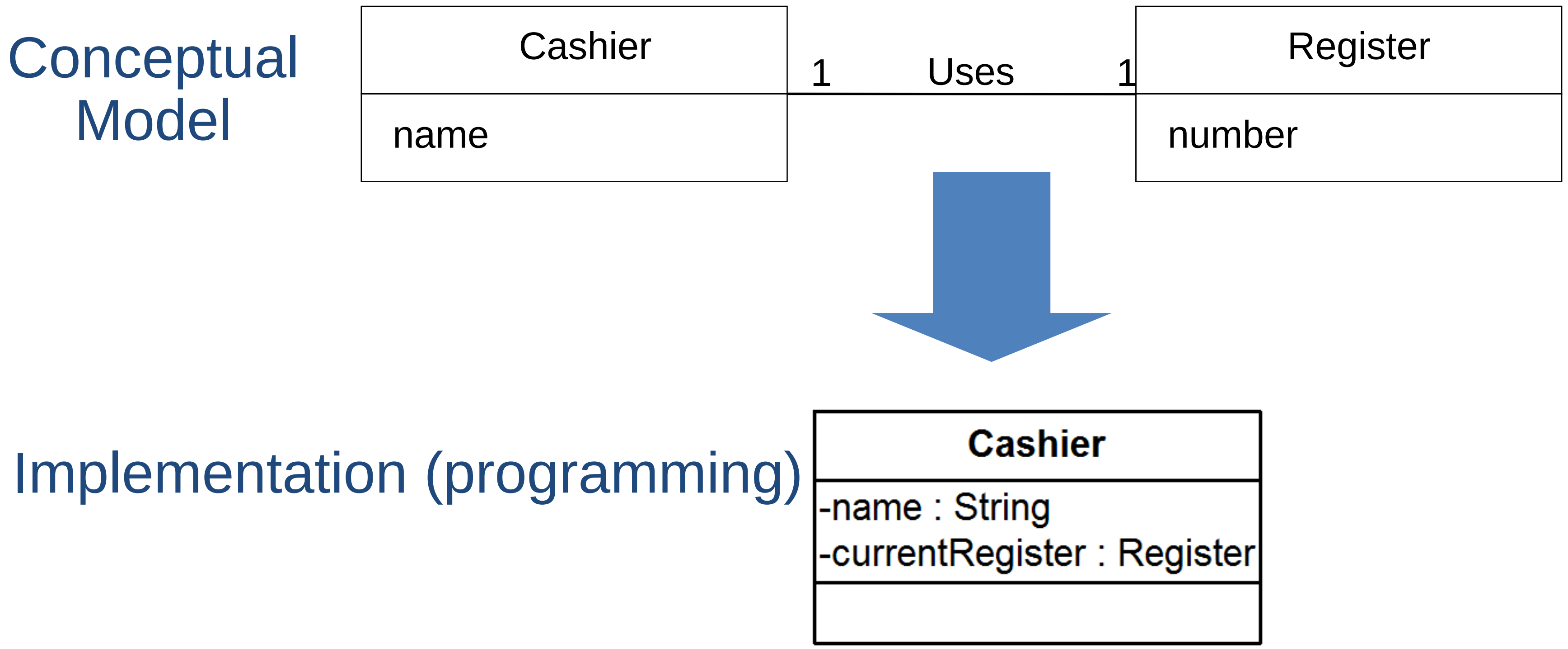


Conceptual Modeling verses Programming

- In the modeling domain:
  - an attribute cannot be an instance of another object
  - situation is modeled as a relationship between classes or objects ("has a" relationship)
- In the programming domain data members are often objects.
  - Decision should be deferred during domain modeling

What about attributes in code?

- During the design and implementation work, associations between objects will often be implemented as attributes that reference other complex software objects



Non-primitive data type classes

Represent what may initially be considered a primitive data type as a class under the following situations:

Situation	Example
It is composed of separate sections	Phone number, name of person
There are operations usually associated with it, such as parsing or validation.	Social security number
It has other attributes	Promotional price could have a start date and end date
It is a quantity with a unit	Payment amount has a unit of currency
It is an abstraction of one or more types with some of these qualities.	UPC (Universal Product Code) or EAN (European Article Number). These numeric coding schemes have subparts identifying the manufacturer, product, country, and a check-sum for validation.

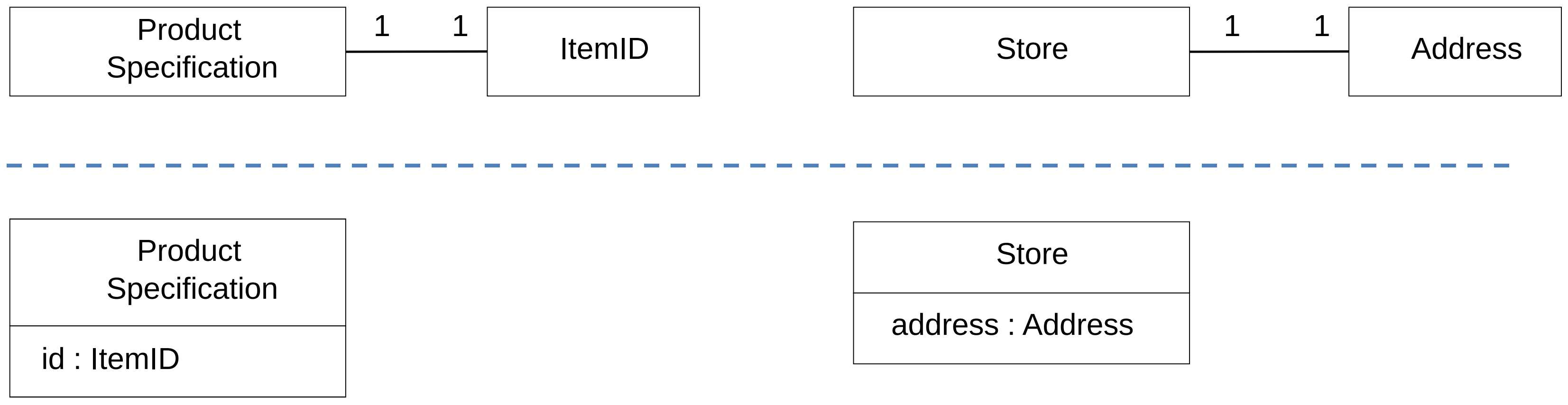
Data type classes in POS domain

- Based on previous slide, following are good candidates for non-primitive classes:
  - Item identifier
  - Price and amount
  - Address

- Whether or not these are shown as separate conceptual classes really depends on what the analyst wishes to emphasize in a domain model.
  - may be shown either as an attribute or a conceptual class
  - No correct answer – all depends on how the domain model is being used as a communication tool

Data type classes in POS domain

Domain Model

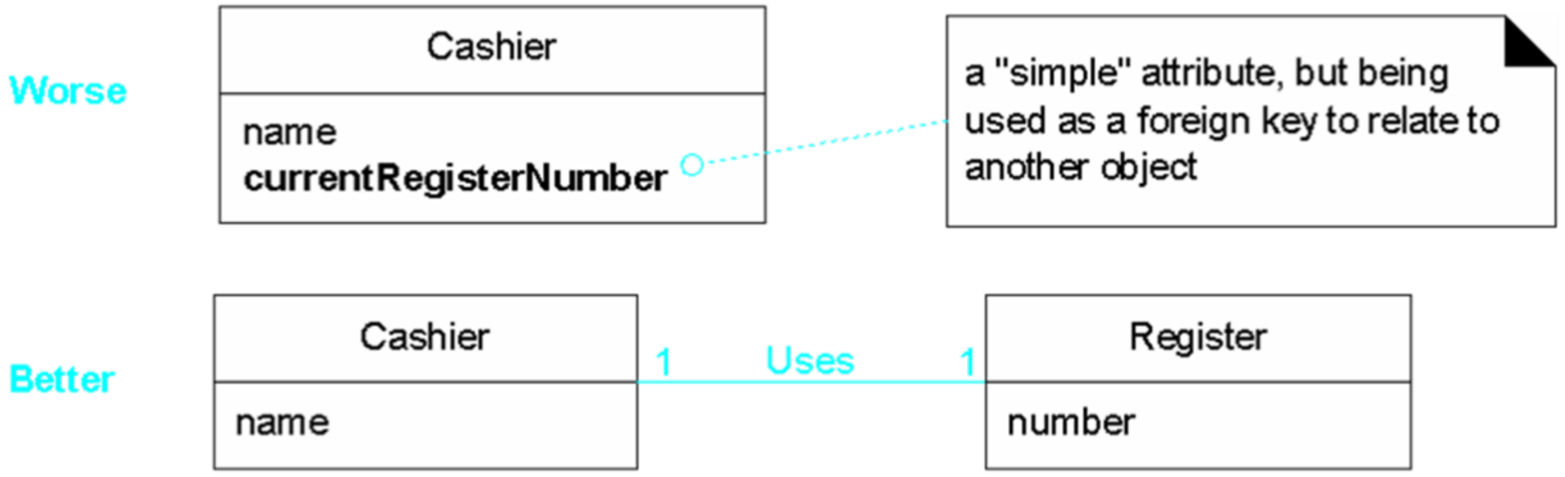


Design Model

Design Creep

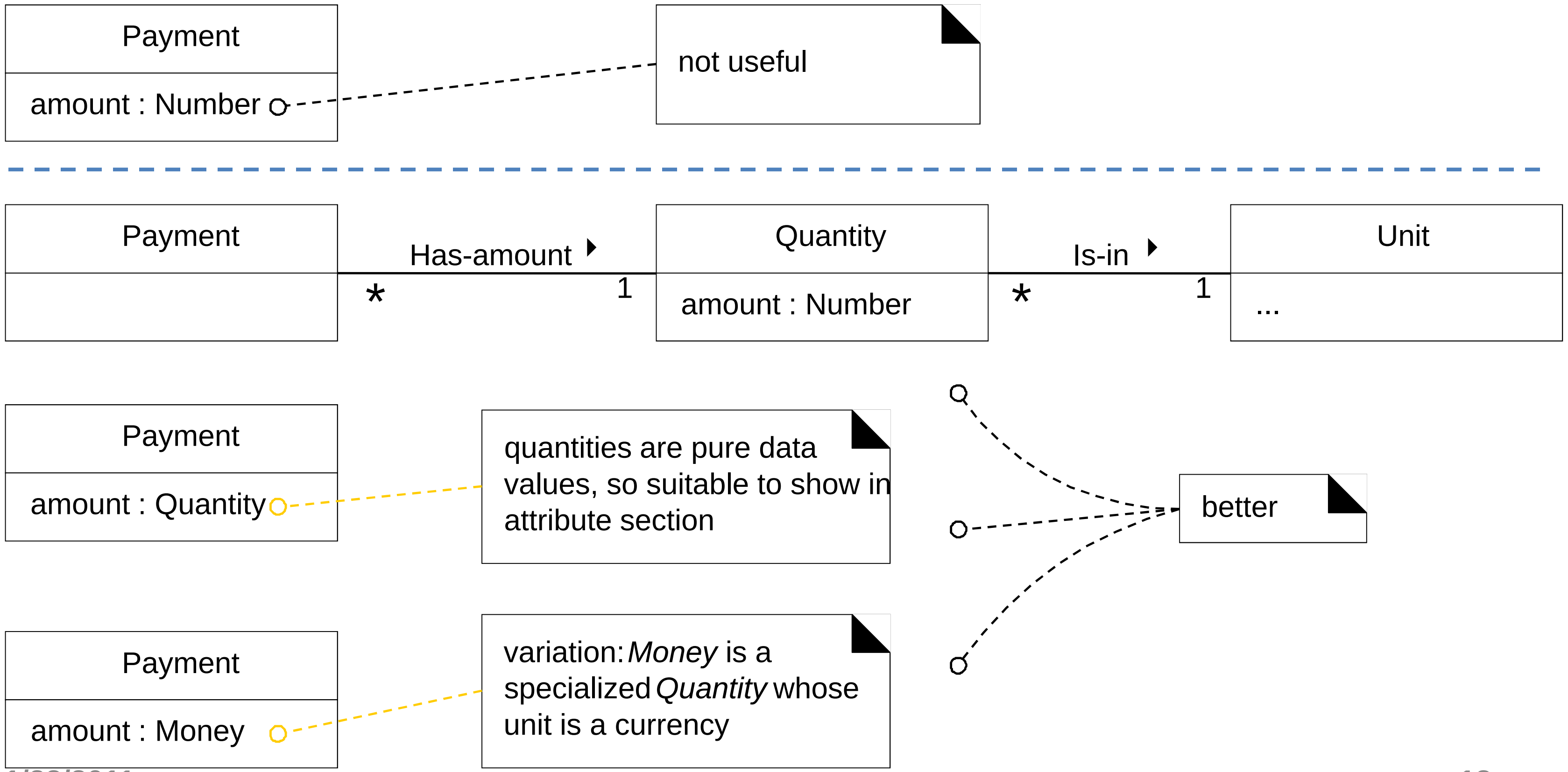
- Attributes should not be used to relate conceptual classes as in the design model
  - Do *not* add a kind of *foreign key attribute* in order to associate two types, as is typical in relational database design.

– In the example below, the *currentRegisterNumber* in *Cashier* class undesirable because its purpose is to relate the *Cashier* and *Register* objects. Better use association.


Worse
Cashier
name
currentRegisterNumber
a "simple" attribute, but being used as a foreign key to relate to another object
Better
Cashier
name
1
Uses
1
Register
number

Modeling attribute Quantities & Unit

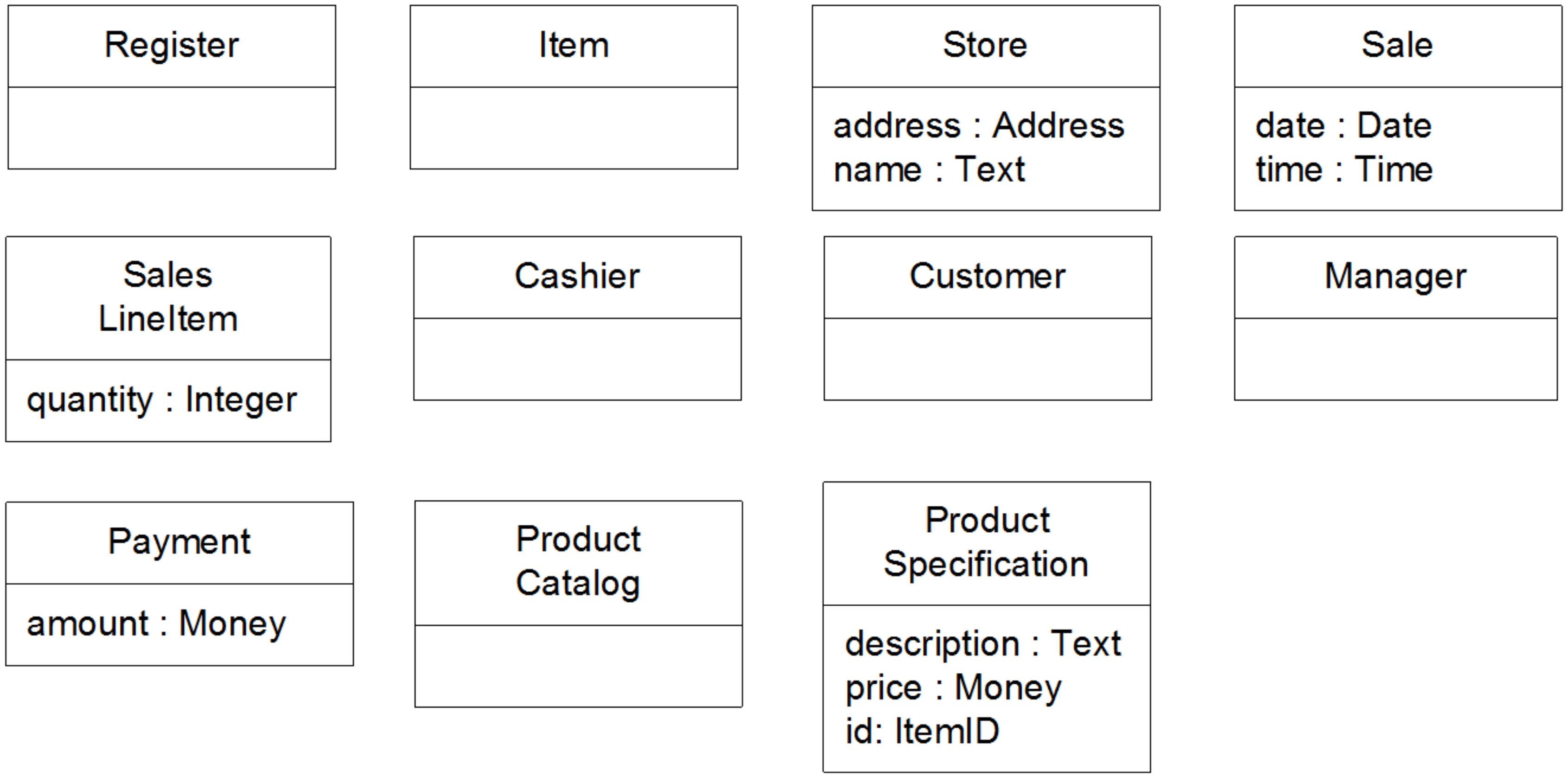
- Most numeric quantities should not be represented as plain numbers.
  - Example: Speed required knowledge of unit (I.E. Kilometers/Hr or Meters/Sec or Miles/Hr …)



Payment
amount : Number
not useful
Payment
Has-amount
*
1
Quantity
amount : Number
Is-in
*
1
Unit
...
Payment
amount : Quantity
quantities are pure data values, so suitable to show in attribute section
better
Payment
amount : Money
variation: *Money* is a specialized *Quantity* whose unit is a currency

Case Study: POS Model

- Attributes for the POS Domain Model



Questions and Conclusions