KTH

CID
Centre for
User Oriented IT Design

Conceptual Modeling in UML

A super-short introduction

by Ambjörn Naeve

amb@nada.kth.se

http://kmr.nada.kth.se

Concept Formation

Aim: Concept formation helps us
to disregard what is inessential
by creating idealised structures
that focus on what is essential.

Example: Point, Line, Plane, ….. in geometry.

Efficiency: Efficient concepts
disregard as much as possible
so that it is noticed as little as possible.

"The power of thinking
is knowing what NOT to think about."

Conceptual Modeling

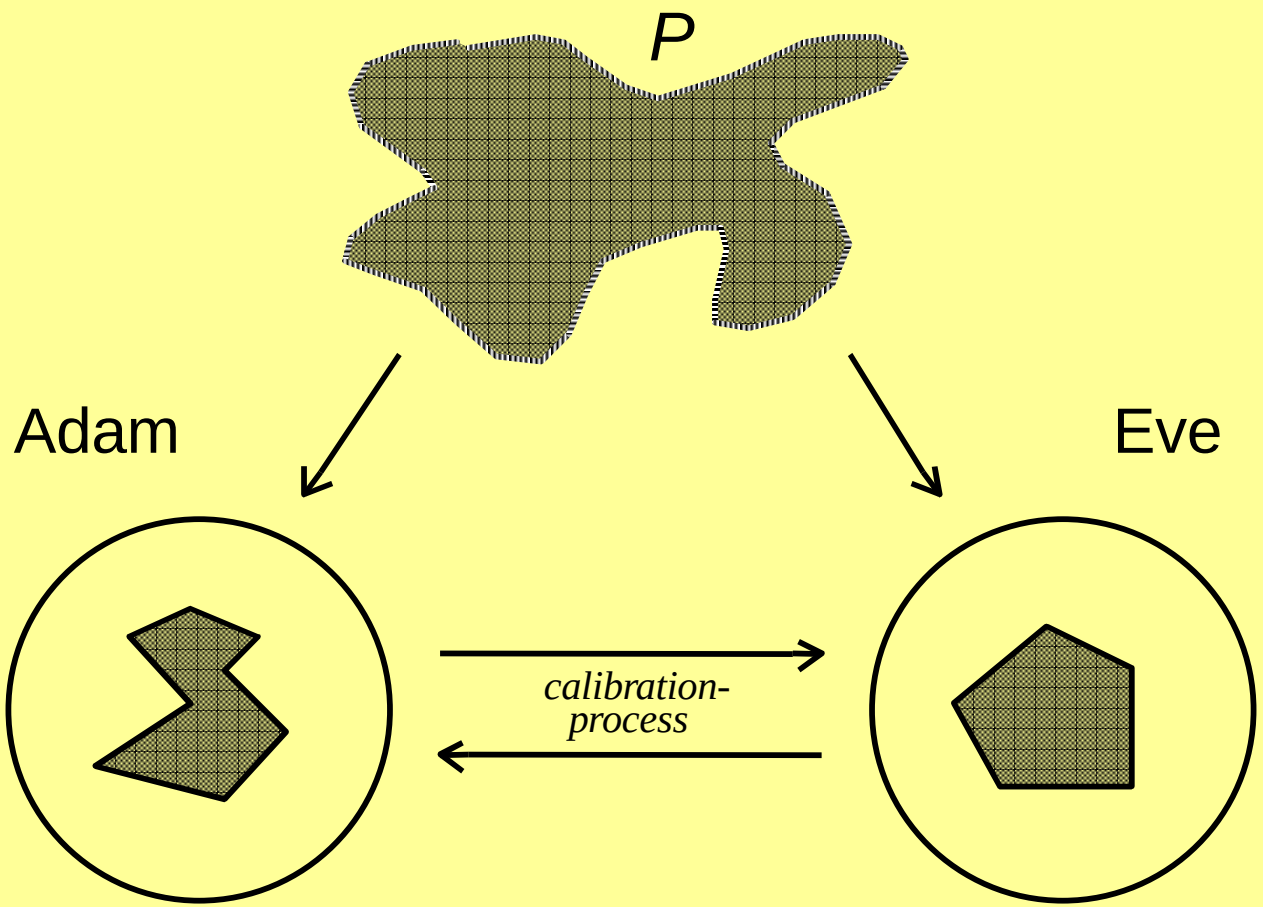
Def: A description of the most important concepts (and their relations) within a specific problem domain is called a conceptual model of the domain.

In order to create an efficient conceptual model of a domain you must be able to reduce its complexity.

Four important tools in complexity reduction are:

- Abstraction ⟷ disregard
- Hierarchy ⟷ resolution
- Encapsulation ⟷ interface
- Modularity ⟷ partition

Two conceptual models of a domain



Adam's image of *P*

Eve's image of *P*

The concept concept

Def: A concept is a representation of something
that we have experienced or that we can imagine,
and which we can apply to the objects
that we are aware of.

Def: The set of objects that belong to a concept
is called the extension of the concept
(or its examples or its instances).

Def: To identify a concept
by observing similarities and differences
within a group of objects
is called to classifiy the objects.

The concept concept (cont.)

Def: The definition of a concept
describes its intention,
i.e. what qualities (= properties and behaviour)
it wants to express and delimit
with respect to its surroundings.

Def: We say that a concept can be applied
to a specific entity (= object = instance)
if that entity fullfills the intention of the concept,
i.e. the conditions of its definition.

Properties of the concept concept

- A concept must always be defined
  by making use of other concepts.
- A concept can be denoted
  by one or several names (= symbols).
- A concept is always idealised,
  because it contains simplifications
  that focus on some aspects and disregard others.
- The definition of a concept always depends on
  the context within which it will be used.
- The aim is always
  to disregard the inessential
  and focus on the essential.

To symbolise a concept

Def: Two symbols are called synonyms if they denote the same concept.

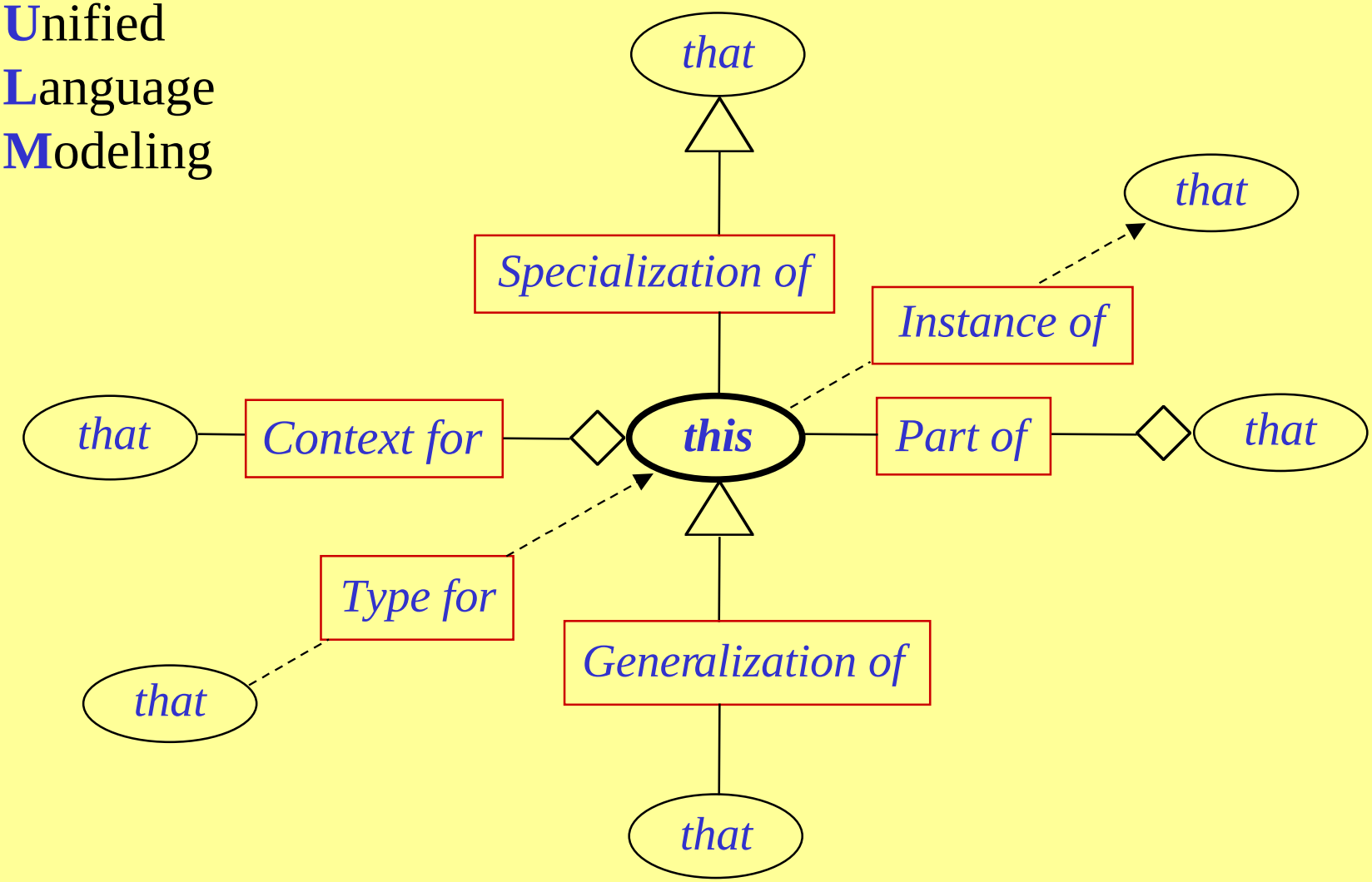
Example: Customer and client can denote the same concept in a model of a business system.

Def: Two concepts are called homonyms if they can be denoted by the same symbol.

Example: In mathematics, negation och subtraction are both denoted by the symbol minus (-).

The hierarchical directions from *this* to *that*

Unified
Language
Modeling



UML - a global modeling language

- UML (= Unified Modeling Language) is a language for specifying, visualising and documenting conceptual models within many different domains.

- UML was developed during 1993 -1997 within the object-oriented software industry as an attempt to unify the 250 different modeling languages that were in use by the mid 1990s.

- UML represents a collection of practically tested modeling techniques that have proven to be effective in the description of large and complex systems.

UML: a visual language for concept relations

- UML provides a visual language
  where you can draw the concepts
  and their relations in different types of diagrams.

- The aim is to display how you think about
  a specific problem domain.

- Words pass - images remain!

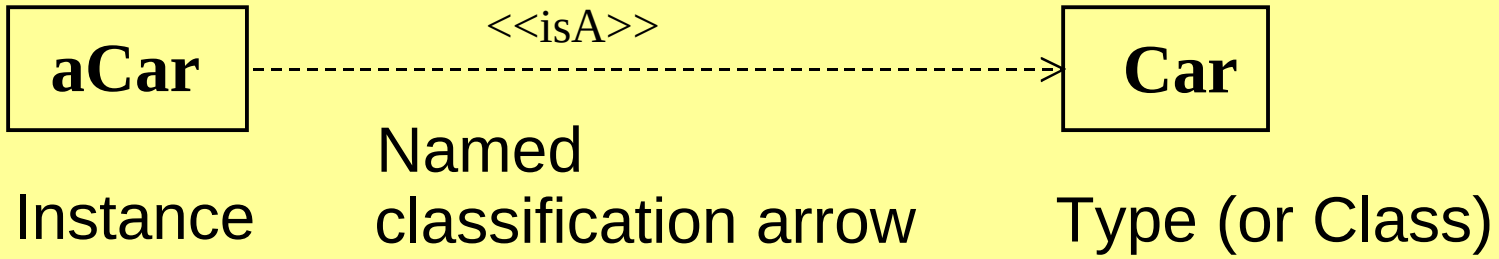
UML: a visual language for concept relations

- Diagrams create an overview.
- You get a visible background
  against which you can discuss
  and where it is clear how you have been thinking so far.
- This
  - facilitates further development of the conceptual model.
  - increases the possibilities to “calibrate the model”
    and reach consensus with respect to
    what aspects that are important.

Type and Class - two synonyms for Concept

Def: The concept, whose extension
consists of a set of instances
and whose intention
describes their common structure
is referred to (within computer science)
as the type or class of these instances

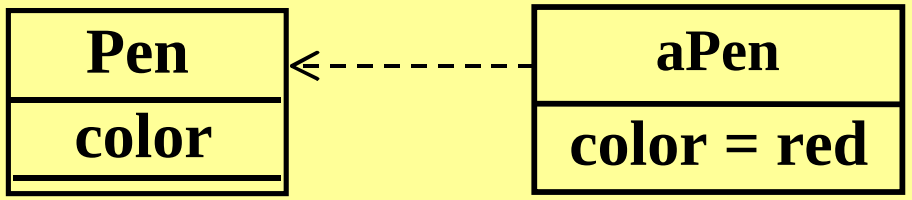
Classification of cars expressed in UML:



Attributes and operations of a concept

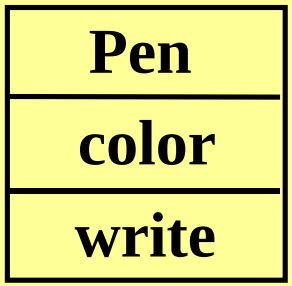
Def: The static properties that belong to a concept are called its attributes.

UML-example:



Def: The dynamic behaviour that belongs to a concept is described by its operations.

UML-example:

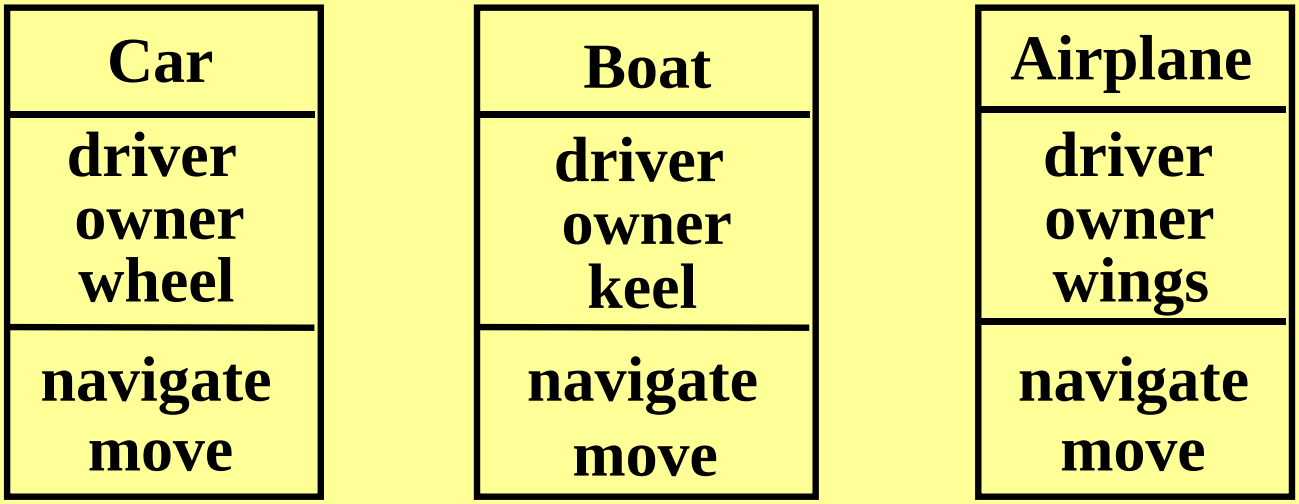


aPen.write( )

Gen/Spec - a kind of concept relation

Different concepts can have common properties and operations.

Example:

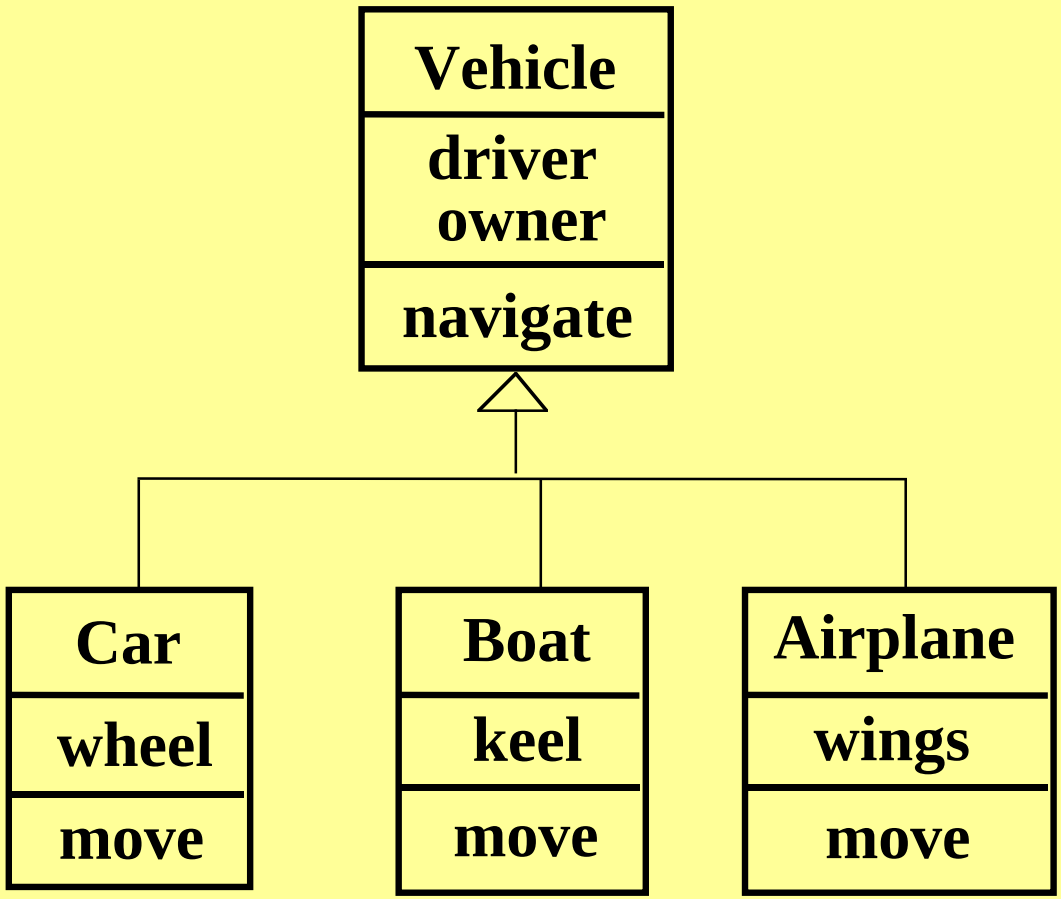


These concepts can be regarded as specialisations of a generalised concept.

Gen/Spec (cont.)

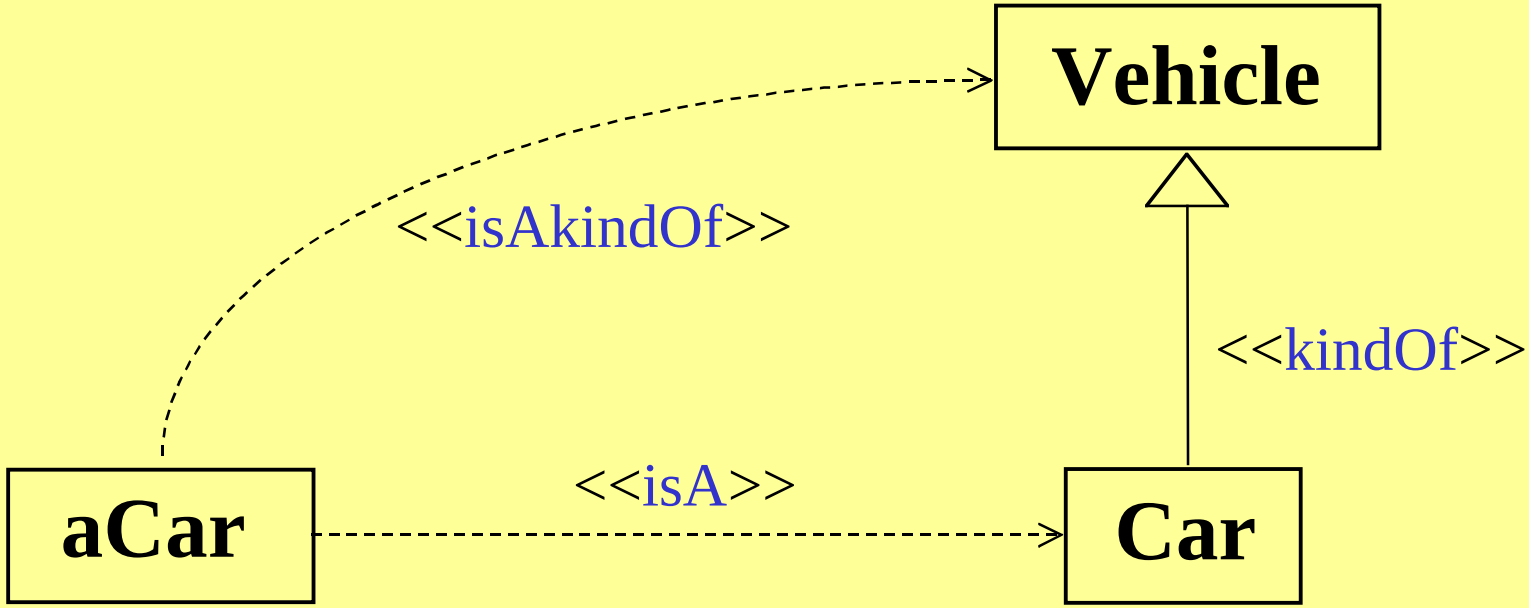
Generalised concept:

Specialised concepts:

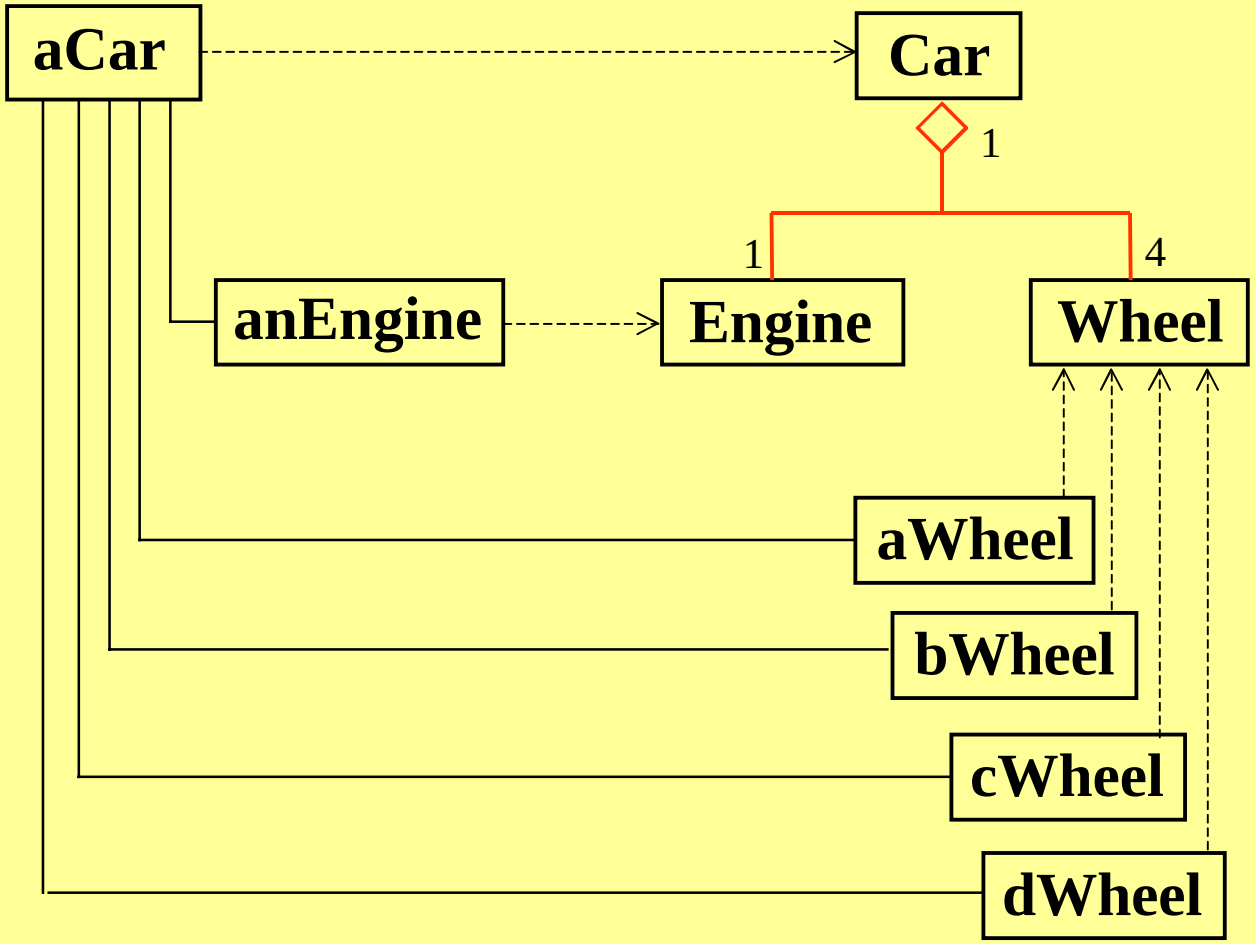


Gen/Spec (cont.)

aCar isAkindOf **Vehicle**

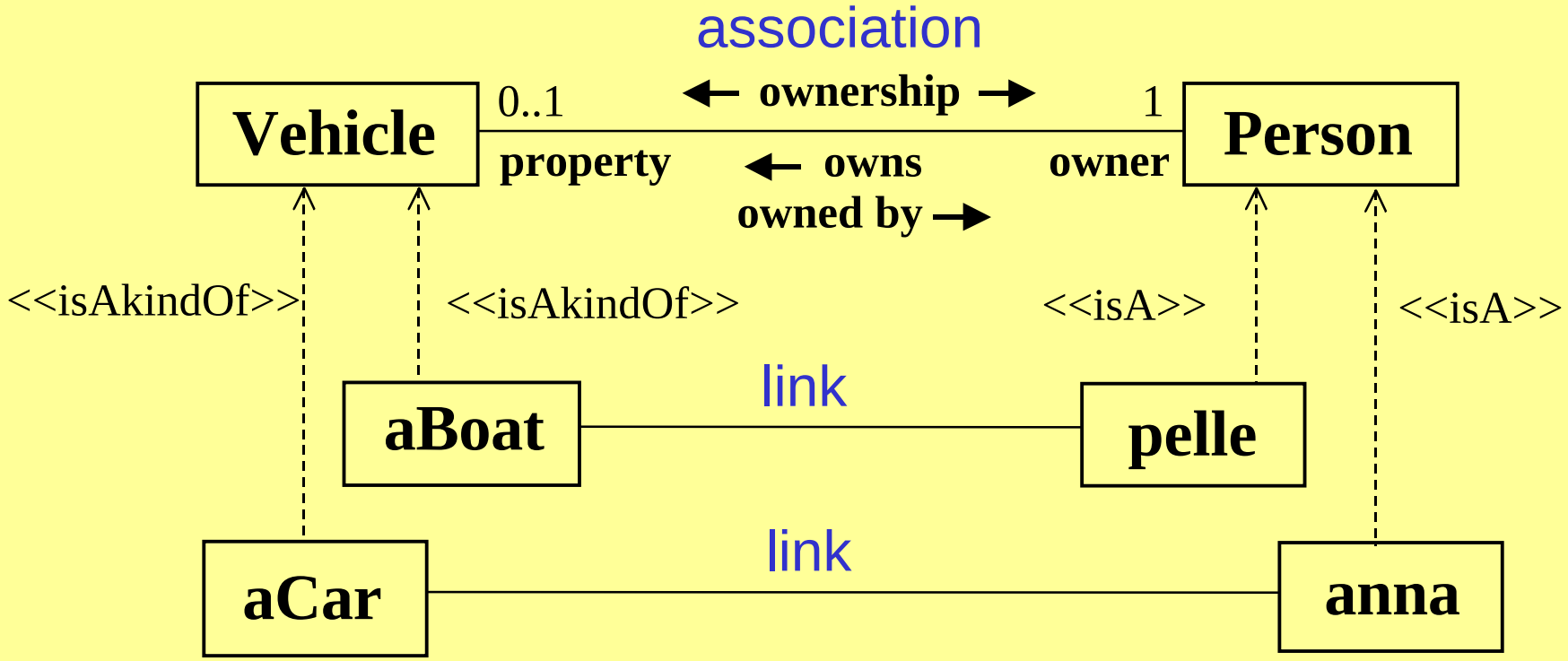


Aggregation - a type of concept relation



An aggregation is a kind of assocation that expresses a whole-part relation between the corresponding instances.

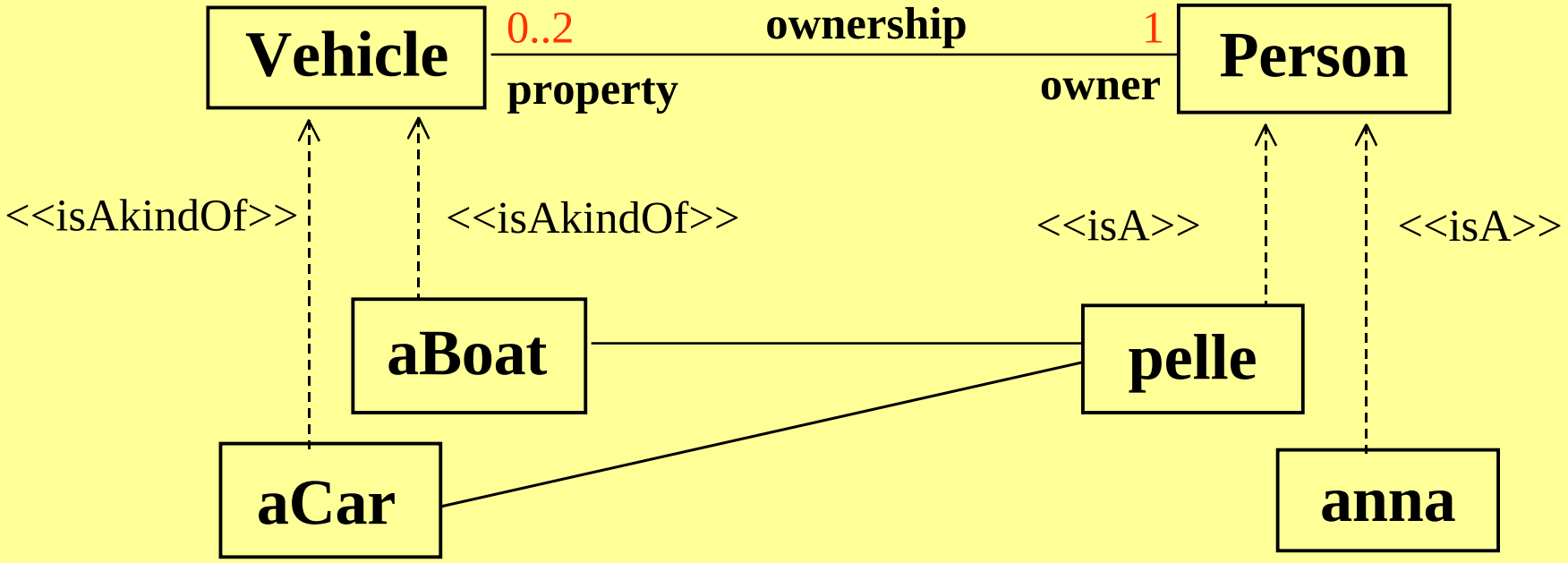
Association - a type of concept relation



A link is an instance of an association.

An association between two concepts describes limitations in the link strukture between instances of the corresponding type.

Link limitations are modeled by multiplicity



Meaning: Every instance of type **Vehicle** is linked to exactly 1 instance of type **Person**.

Every instance of type **Person** is linked to 0, 1, or 2 instances of type **Vehicle**.

Unified Language Modeling