Introduction to Modelling in ISO 15926

The Data Model

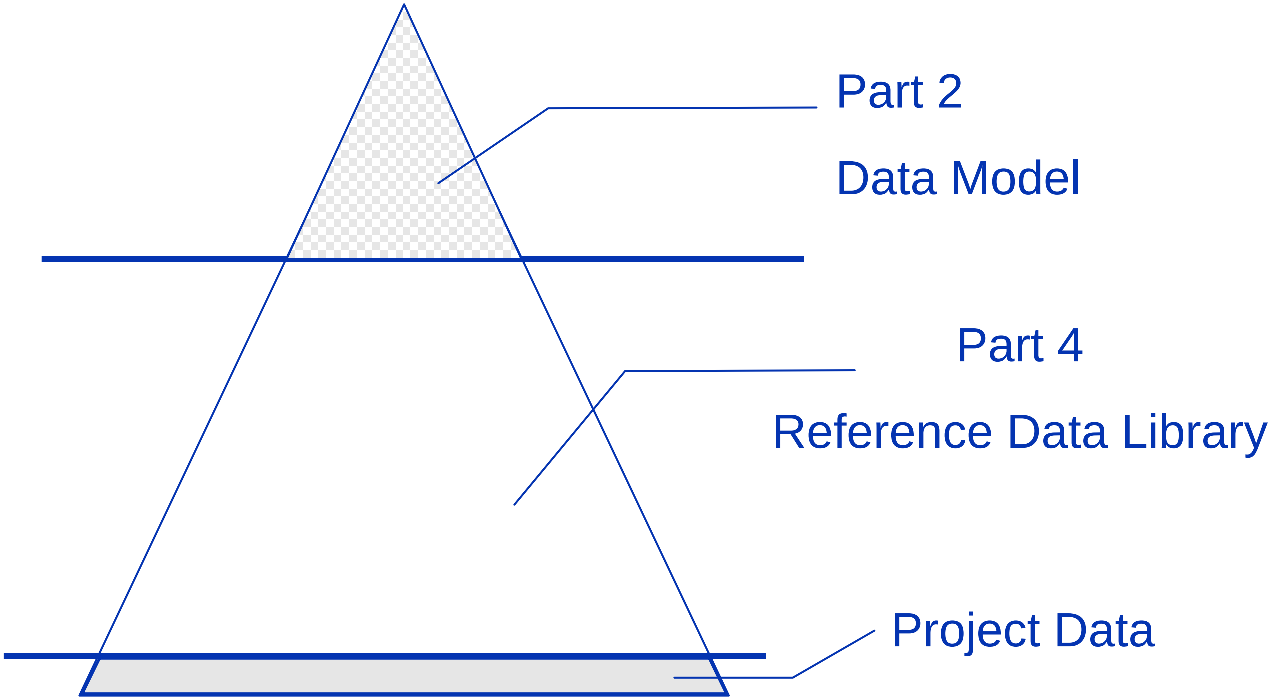
abelia POSC Caesar Association IOHN olf MANAGING RISK DNV

Tutorial Overview

- **The Data Model**
  - Fundamental Concepts
  - The EXPRESS Modeling Language
  - The Basic Classes of 15926
  - Types of Reference Data Classes

MANAGING RISK

Class Hierarchy



Part 2

Data Model

Part 4

Reference Data Library

Project Data

Source Material and Accessories

- Part 2 can be bought from ISO as a pdf file.
  - Navigate to http://iso.org and search for ISO 15926
- An overview of Part 2 can be found at http://ht.vestforsk.no/demo/iso15926/
- The EXPRESS code and the EXPRESS diagrams of Part 2 can be found at http://www.tc184-sc4.org/wg3ndocs/wg3n1328

© Det Norske Veritas AS. All rights reserved

Tutorial Overview

- The Data Model
  - **Fundamental Concepts**
    - Class
    - Relationship
    - Entity types, classes and objects
  - The EXPRESS Modeling Language
  - The Basic Classes of 15926
  - Types of Reference Data Classes

MANAGING RISK

Two fundamental concepts

- Class
  - **Pump** – The collection of all pumps
  - **Bicycle** – The collection of all bicycles
  - **Pebbles** – The collection of all pebbles

- Relationship
  - **HusbandOf** – The relationship between a man and his wife
  - **ConsistsOf** – The relationship between e.g. a pump and its parts

Fundamental Concepts: Class

- Class
  - A collection of objects.
  - The objects in such a collection are called *members* of the class
  - A class is an abstraction:
    - You can have a pile of pebbles and a class of the same pebbles. Dispersing the pebbles will distroy the pile, but the class remains the same.
  - A class is also an object, and can therefore …
    - be a member of a class
    - be related to other objects

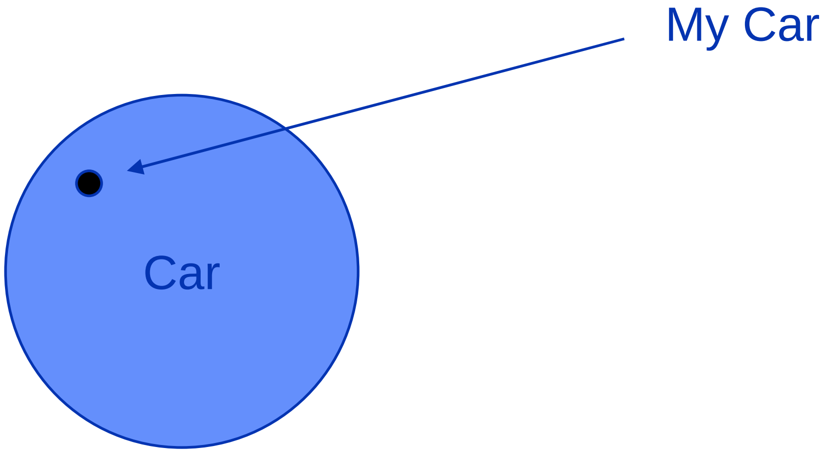
© Det Norske Veritas AS. All rights reserved

Classes and Classifying Concepts

- By a classifying concept we mean a concept which holds, or is true of, objects.
  - *”is a car”*
- The collection of things that the concept is true of forms a class.
  - *”the class of all cars”*
- The principle of extensionality for classes:
  - *”If two classes have the same members, they are the same class.”*
- Classifying concepts are <u>intentional</u>: Two concepts can be true of the same things.
  - *”is a creature with heart” and ”is a creature with kidney”*
- *Are 15926 classes Classes or classifying concepts?*
  - *For our purposes, it makes little difference*

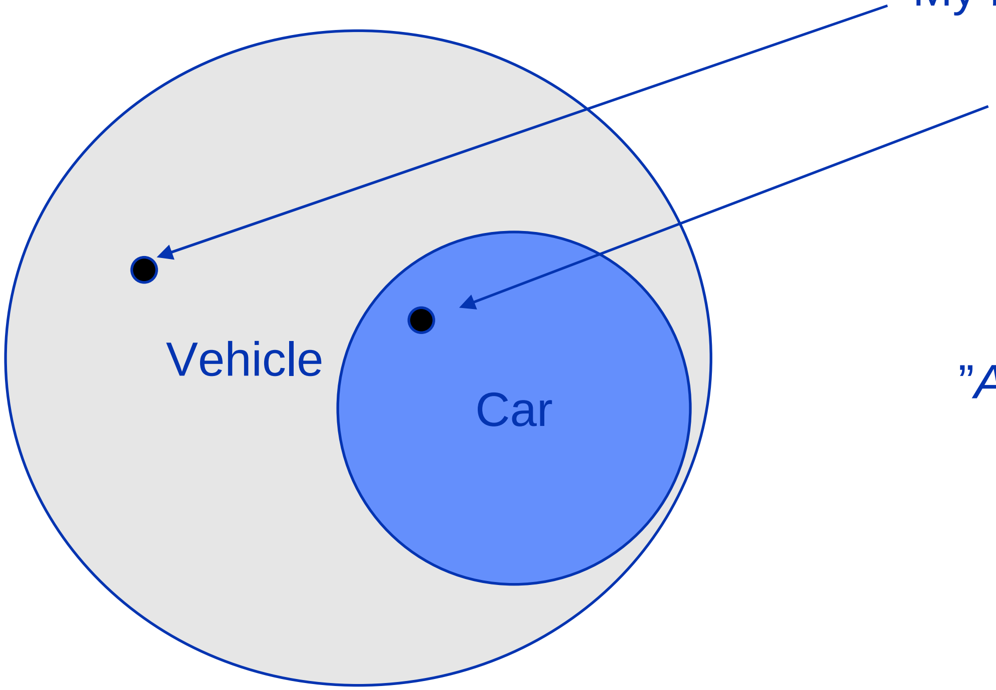
Two Important Notions

- Membership



- Inclusion

Inclusion

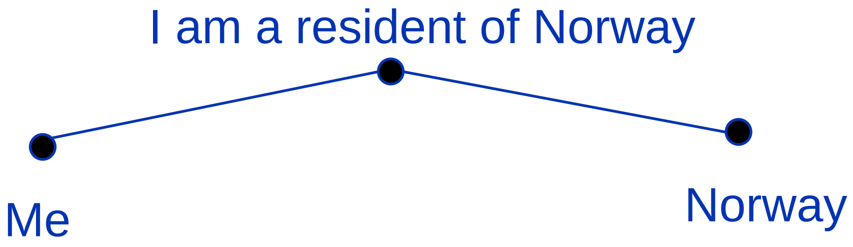


”All cars are vehicles”

© Det Norske Veritas AS. All rights reserved

Fundamental Concepts: Relationship

- A relationship is something that holds of or between two objects.

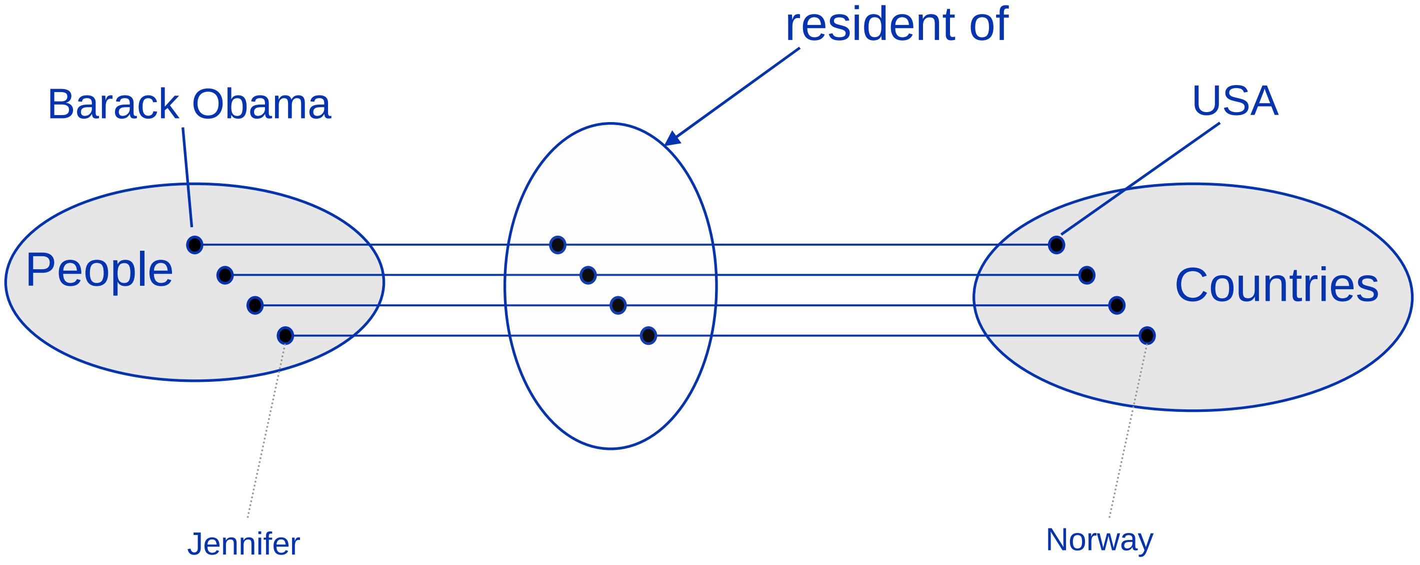


- Relationships are also objects, and can be used in other relationships and be members of classes.

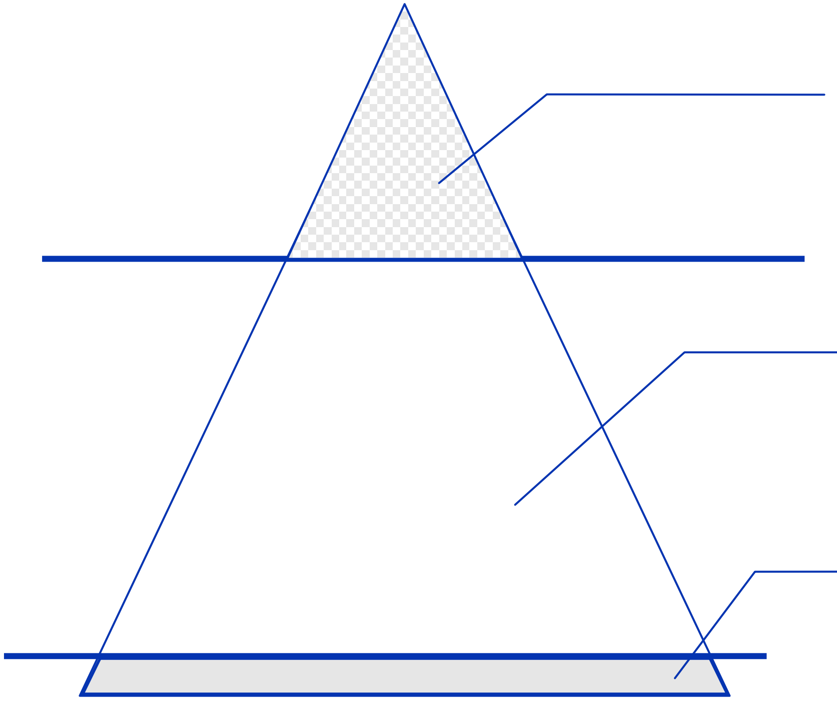
© Det Norske Veritas AS. All rights reserved

Fundamental Concepts: Relationship, cont.

- A concept with two entries corresponds to a Class of Relationship
  - *… is a resident of …*
  - A Relationship on the other hand, corresponds to a sentence: `Jennifer` is a resident of `Norway`
- Instead of connecting two *objects,* a class of relationship connects two *classses*



Part 2 vs. Reference Data



Data Model

Very general classes and relations

Reference Data Library (RDL)

Industry specific classes and relations

Project Data

Project or company specific data

© Det Norske Veritas AS. All rights reserved

Entity Types, classes and objects.

- There is a ’syntactical’ difference between classes of Part 2, and every other type of class in the ISO15926 standard:

- Classes in Part 2 are modelled as *entity types* in the language EXPRESS.
  - Entity types are only classes, they cannot be a member of any other class, and not be related to any entity.

- Every class *not* in Part 2 are *entities*, sometimes also called objects. This is because they are inevitably members of at least one *entity type*.
  - Entities *may* be classes, but they are *always* objects. This means they can be members of classes, and related to other objects.

© Det Norske Veritas AS. All rights reserved

Tutorial Overview

The Data Model

- Fundamental Concepts
- **The EXPRESS Modeling Language**
- The Basic Classes of 15926
- Types of Reference Data Classes

abelia POSC Caesar Association IOHN olf MANAGING RISK DNV

EXPRESS

- Standardized as part of the STEP standard (ISO 10303)
- Data modeling language
- The EXPRESS-G notation will be used to visualize EXPRESS models. (Such as 15926-2)
- We will limit our discussion to features of EXPRESS used in ISO15926-2

Entity type

- Specification of a type of data
- Consists of
  - Name
  - Attribute(s)
  - Subtype and Supertype declarations

```
ENTITY Person
  name   : STRING;
  mother : Woman;
END_ENTITY
```

Attributes

- Built-in types
  - STRING, INTEGER, BOOLEAN, etc
- Entity types:
  - An entity type defined elsewhere in the data model
- Aggregate types:
  - A list where all elements are of the same type.
- Optional attributes.

```
ENTITY Person
  name      : STRING;
  age       : INTEGER;
  mother    : OPTIONAL Woman;
END_ENTITY
```

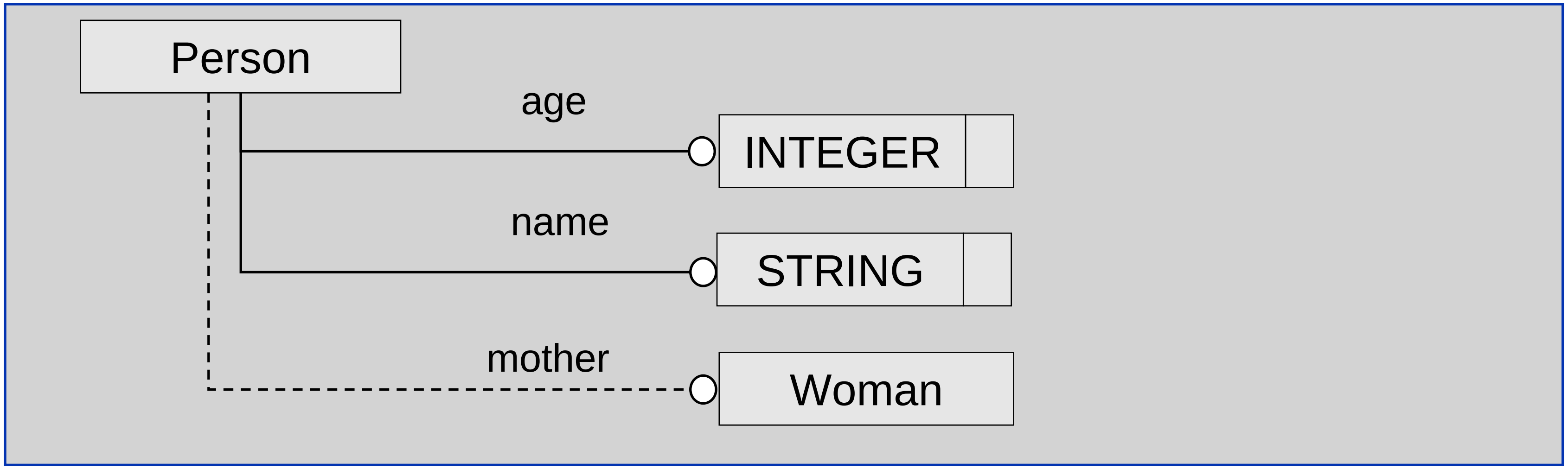
© Det Norske Veritas AS. All rights reserved

The EXPRESS-G notation

EXPRESS

```
ENTITY Person
  name      : STRING;
  age       : INTEGER;
  mother    : OPTIONAL Woman;
END_ENTITY
```

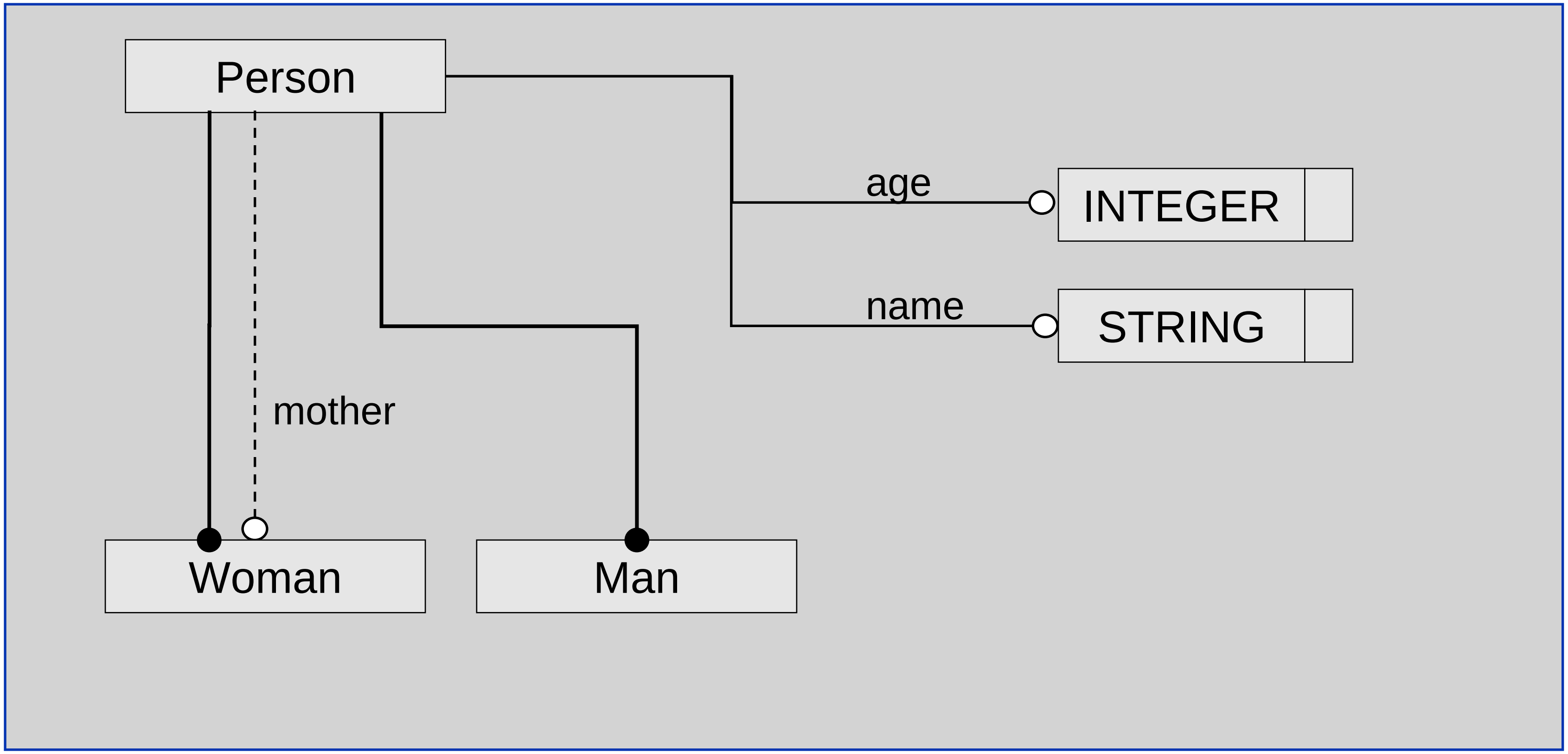
EXPRESS-G



© Det Norske Veritas AS. All rights reserved

Type Hierarchy

- Entity types in a data model are usually organized in a type hierarchy.
- A subtype inherits all the attributes of its supertype.



© Det Norske Veritas AS. All rights reserved

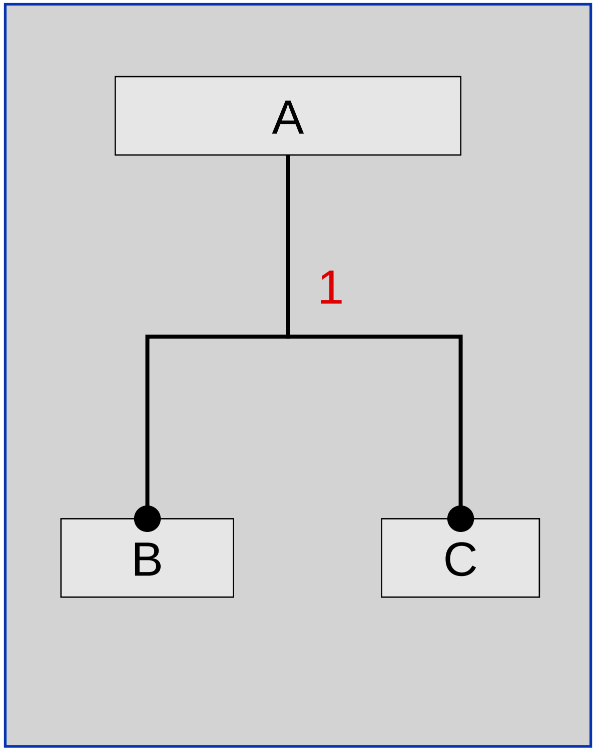
Hierarchy Restrictions

- ONEOF – disjoint entity types

```
ENTITY A
  SUPERTYPE OF (ONEOF (B, C));
END_ENTITY;

ENTITY B
  SUBTYPE OF (A)
END_ENTITY;

ENTITY C
  SUBTYPE OF (A)
END_ENTITY
```



- An **A** cannot be both a **B** and a **C**

© Det Norske Veritas AS. All rights reserved

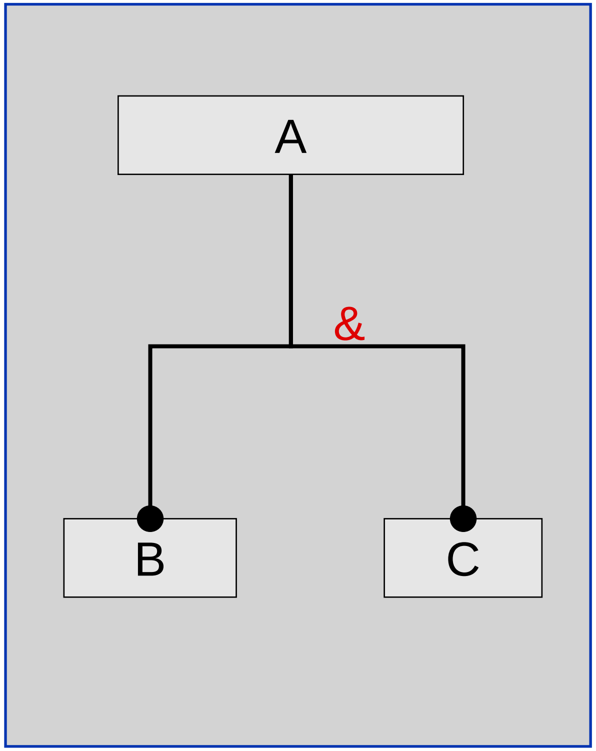
Hierarchy Restrictions

- ANDOR – possible overlaping entity types

```
ENTITY A
  SUPERTYPE OF (B ANDOR C));
END_ENTITY;

ENTITY B
  SUBTYPE OF (A)
END_ENTITY;

ENTITY C
  SUBTYPE OF (A)
END_ENTITY
```



- An **A** may be either a **B** or a **C**, both or neither.

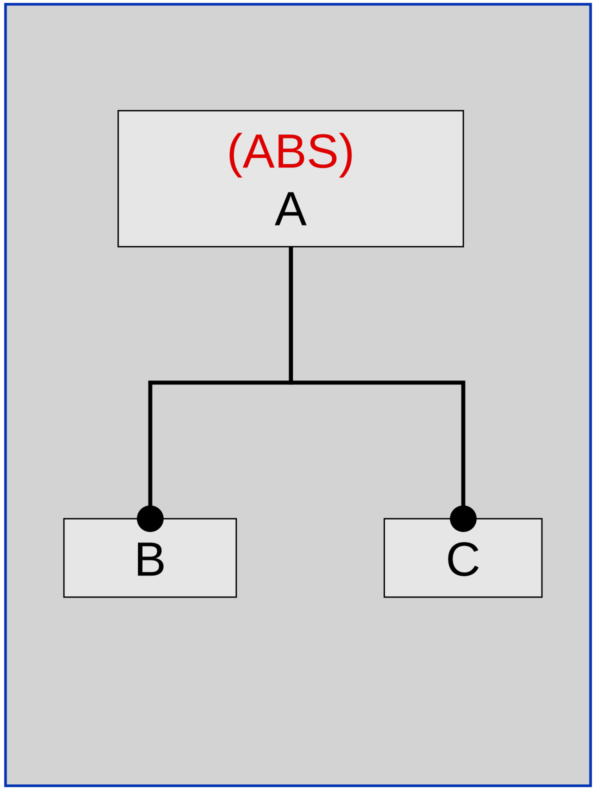
Hierarchy Restrictions

- ANDOR – possible overlaping entity types

```
ENTITY A
  ABSTRACT SUPERTYPE OF
                    (B ANDOR C);
END_ENTITY;

ENTITY B
  SUBTYPE OF (A)
END_ENTITY;

ENTITY B
  SUBTYPE OF (A)
END_ENTITY;
```



- An **A** must be one of **B** or **C**, or both.

© Det Norske Veritas AS. All rights reserved

Tutorial Overview

The Data Model

Fundamental Concepts

The EXPRESS Modeling Language

The Basic Classes of 15926

- Thing
- Possible Individual
- Abstract Object: Class
- Abstract Object: Relationship

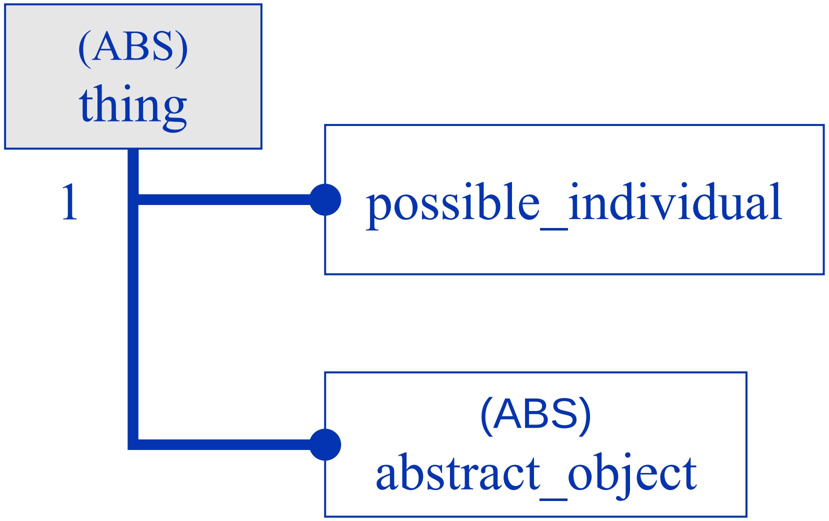
Types of Reference Data Classes

MANAGING RISK

© Det Norske Veritas AS. All rights reserved

Thing

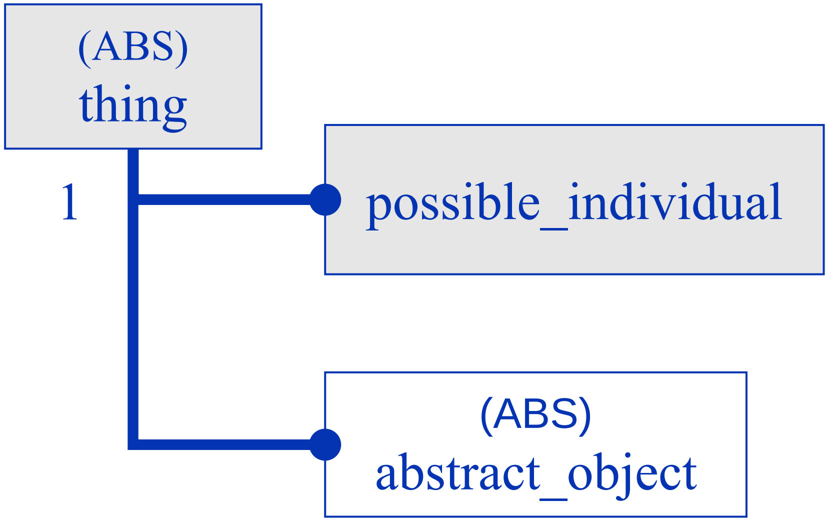
- The top class of 15926 is ***Thing***
- All 15926 objects are ***Thing***s
- All 15926 classes are subclasses of ***Thing***, exept ***Thing*** itself.



- Examples of things:
  - The French Revolution
  - Barack Obama
  - 152 kilometers
  - π (pi)
  - The class of all pumps

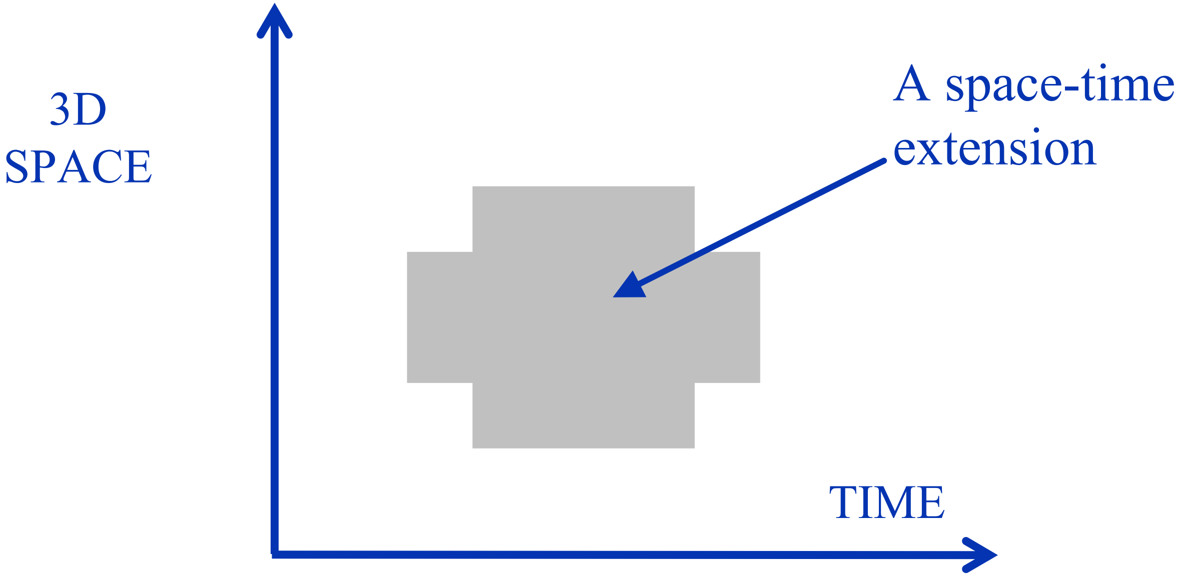
© Det Norske Veritas AS. All rights reserved

Possible Individual



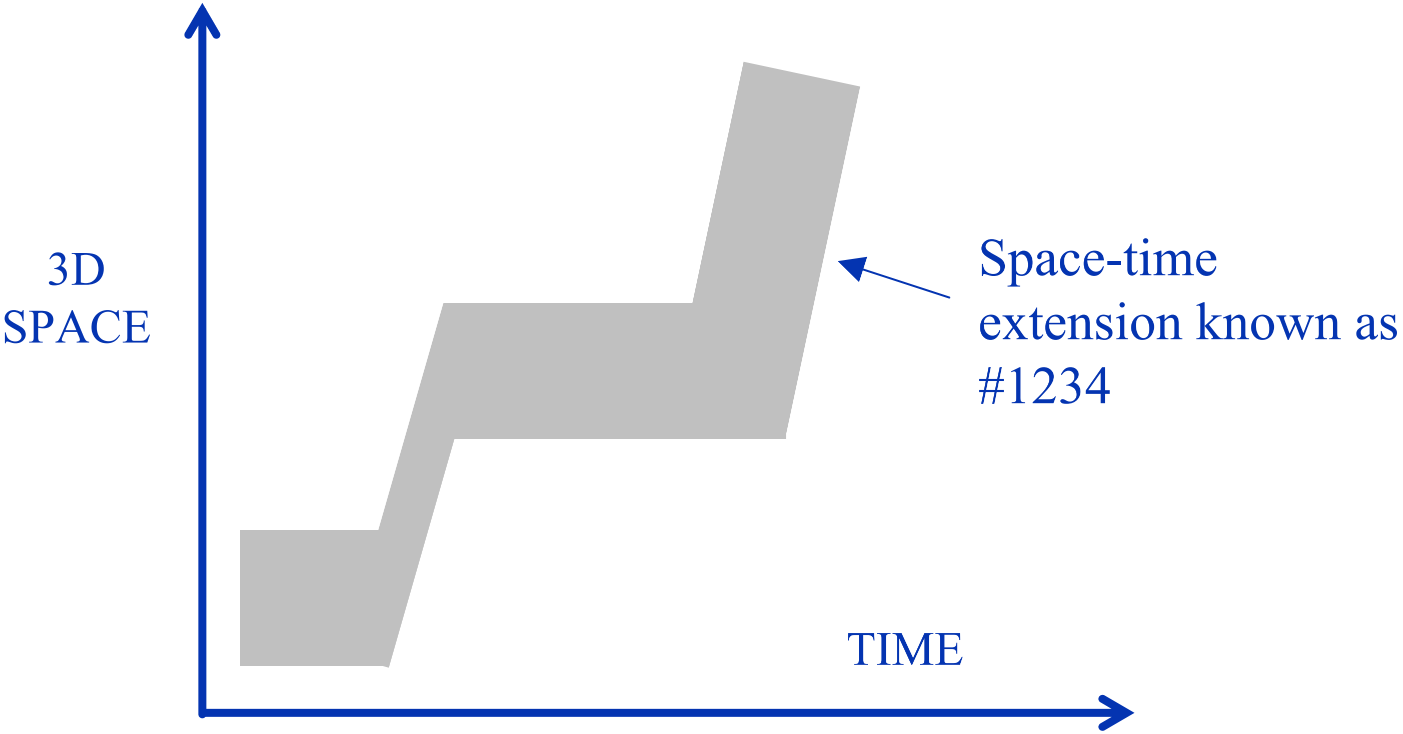
- All physical things
- All things bound by space or time
  - Includes past and future objects
- Considered as four dimensional space-time regions

Possible Individual in space time



A Possible Individual in space time

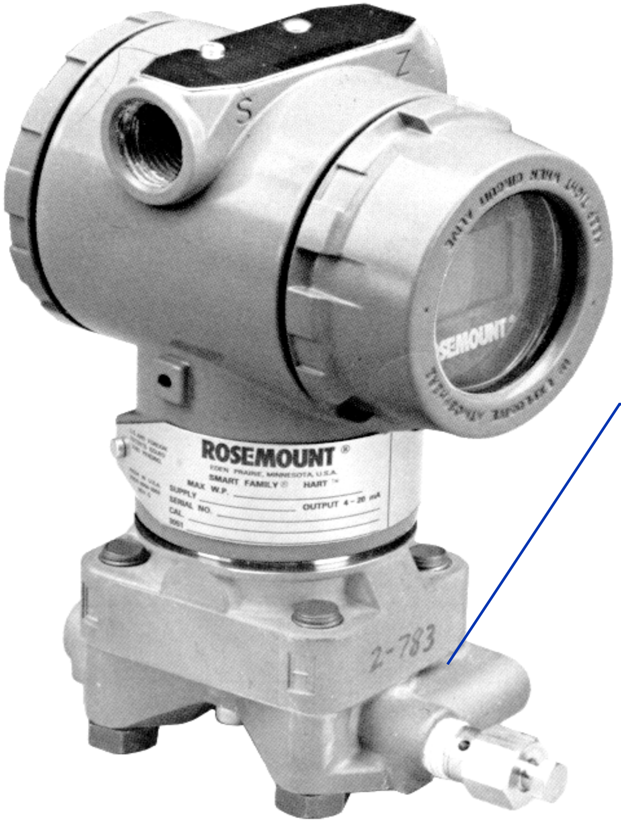
A unique space time extension



© Det Norske Veritas AS. All rights reserved

Unique space time extension

Gauge pressure transmitter



Same thing through time and space

Serial number

“2-783”

Scope of Possible Individual

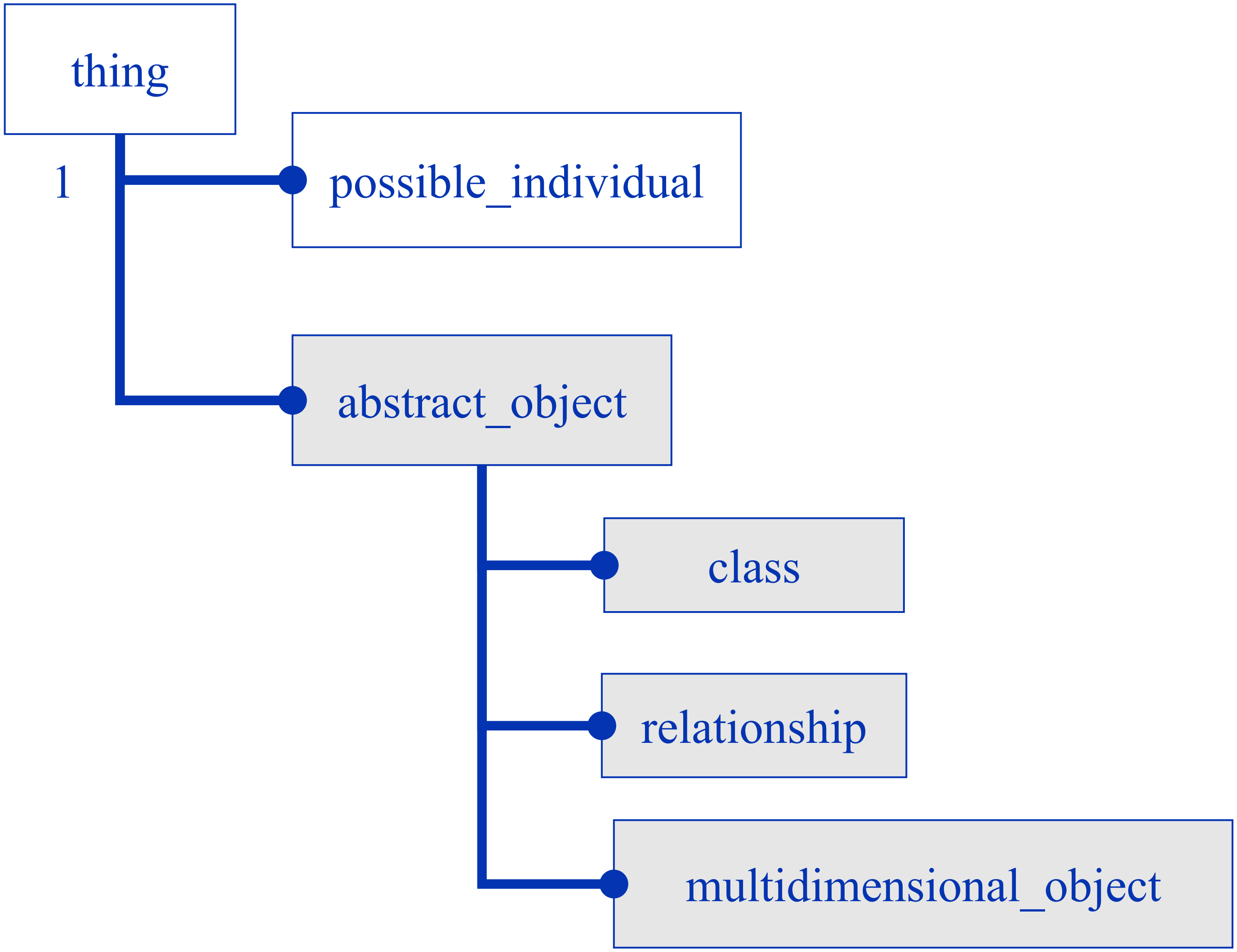
- Things that actually exist, or have existed
- Things that are fictional or conjectured and possibly exist in the past, present or future
- The entirety of all space time (Universe)
- Things that are either all space for any time (time periods)
- Things where the time dimension is vanishingly small (events)
- Things that have a specific position, but zero extent in one or more space dimensions, such as points, lines, and surfaces.
- Temporal parts (states) of other possible individuals

Abstract Objects

- Things that don’t have a space or time extension
  - Numbers
  - Classes
  - Relationships

© Det Norske Veritas AS. All rights reserved

Subtypes of Abstract Object


thing

1

possible_individual

abstract_object

class

relationship

multidimensional_object

© Det Norske Veritas AS. All rights reserved

Subtypes of Abstract Object: Class

- **Class** is the entity type of reference data classes.

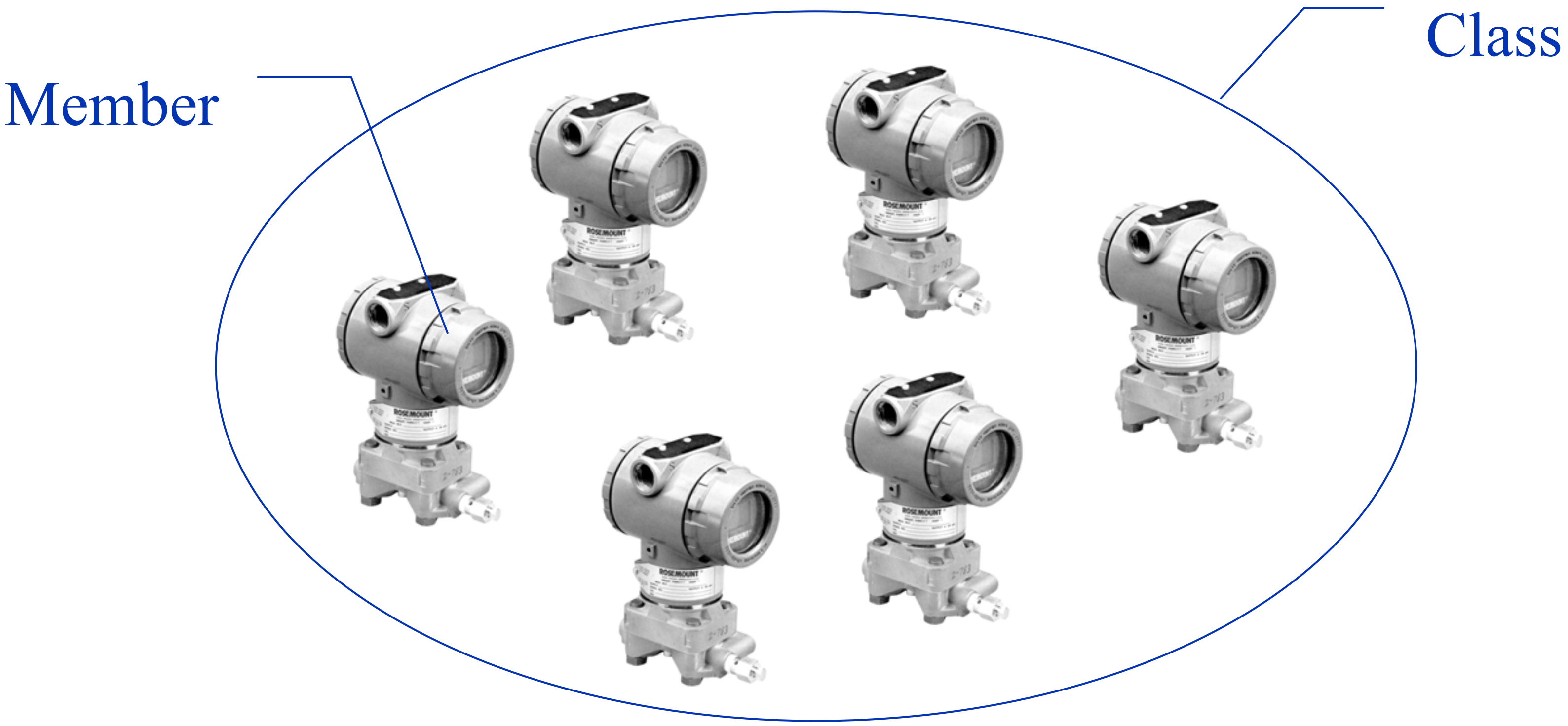
- A reference data class is an object with entity type **Class**.

© Det Norske Veritas AS. All rights reserved

Examples of Classes

- Apple
- Green
- VLCC – very large crude carriers
- Gauge pressure transmitter
- ASME B1.1 Bolt threads
- EN218 Gasoline
- SAE 20-40W Motor engine oil

Gauge Pressure Transmitter Class



© Det Norske Veritas AS. All rights reserved

Why is class important?

- Class enables us to describe common information once, as the class.
- In practice little information is recorded about individuals. Most design is about stating I want one of ”them” or some of ”that”. ”Them” or ”that” is a class, often a manufacture’s product.
- Specifications that enable compliance to be assessed are classes, a particular item can be judged against the criteria of membership of the class.
- Specification of classes based on criteria for membership is the basis for part piece assembly, mass production, and economies of scale.

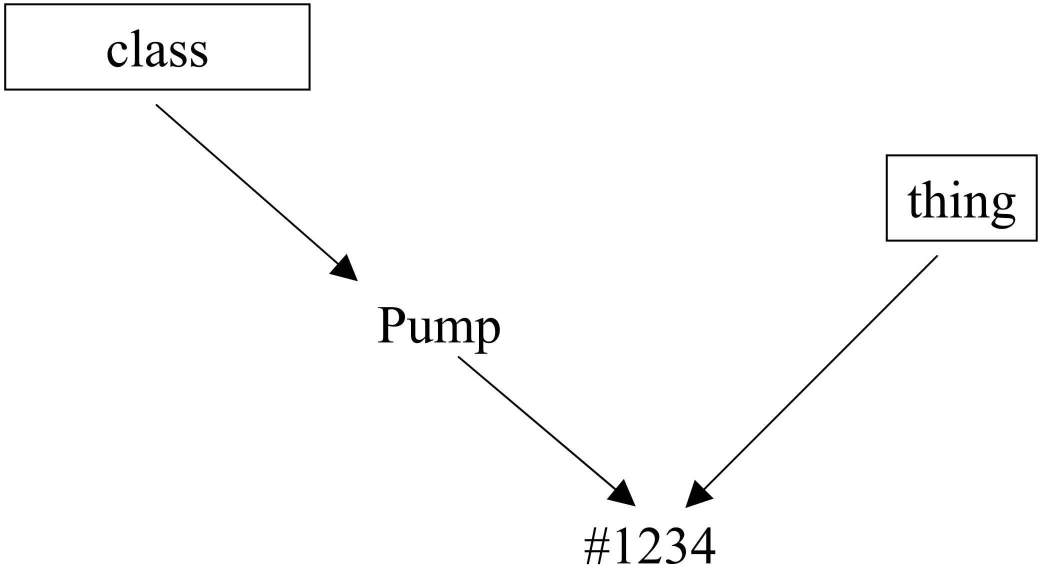
© Det Norske Veritas AS. All rights reserved

Diagram notation

abcde	an object
→	a membership relationship, the arrow head indicates the member of the class
$\boxed{\text{id}}$	a class that is a model entity type, where “id” is its identifier

© Det Norske Veritas AS. All rights reserved

Example Diagram



© Det Norske Veritas AS. All rights reserved

Exercise 1

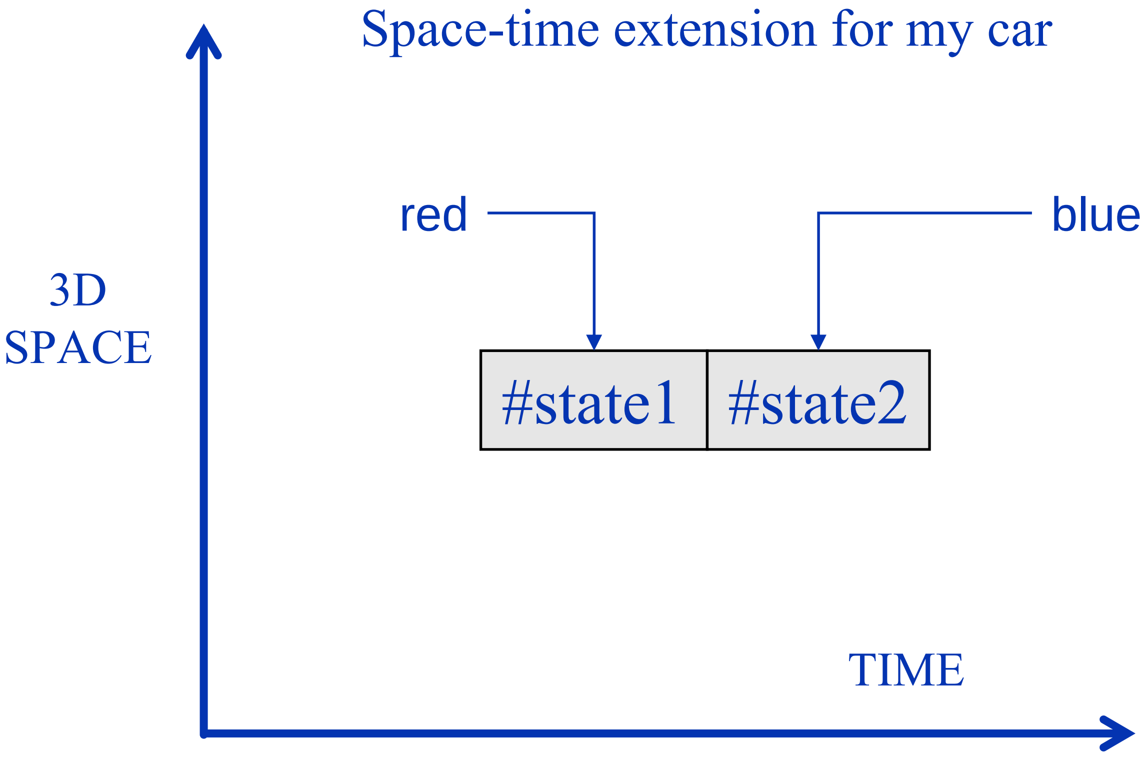
- Represent using a diagram
  - that 2-783 is a possible individual
  - that possible individual is an entity type
  - that 2-783 is a gauge pressure transmitter
  - that gauge pressure transmitter is a Class
  - that Class is an entity type

More about Class

- Classes are universal, independent of time
- Change is handled by classification of temporal parts of 4D objects
- For a car that is repainted from red to blue, the 4D part when it is red is a member of the red class, and the blue 4D part is a member of the blue class.

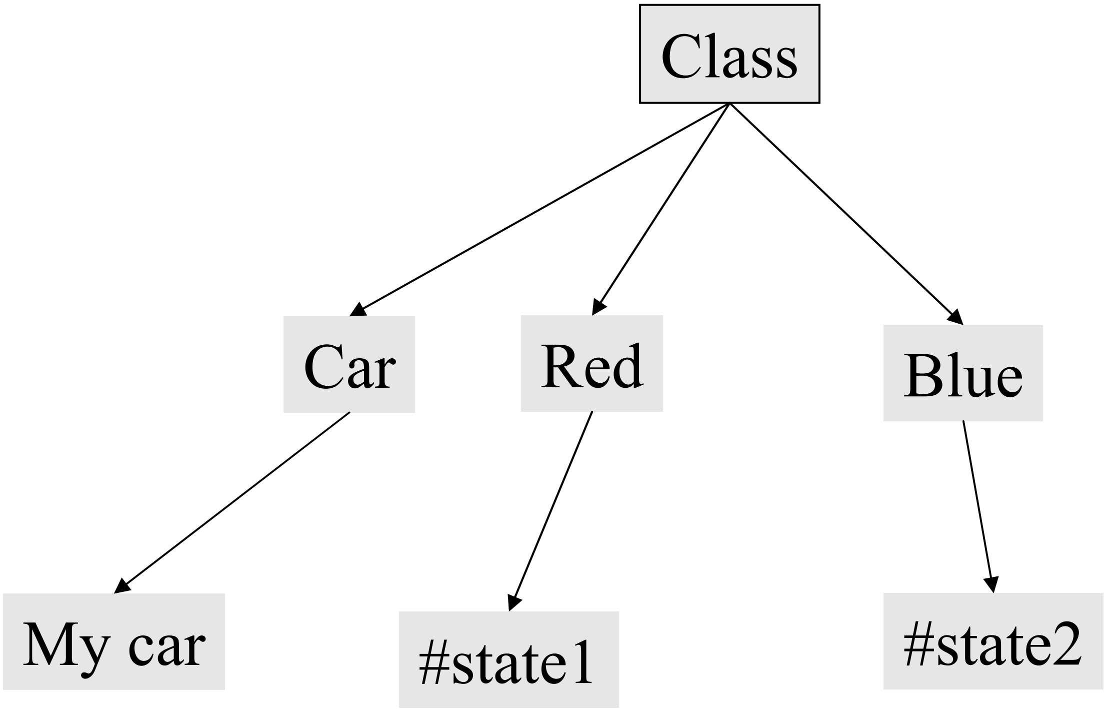
© Det Norske Veritas AS. All rights reserved

Class membership is independent of time



© Det Norske Veritas AS. All rights reserved

Membership is constant


Class

Car

Red

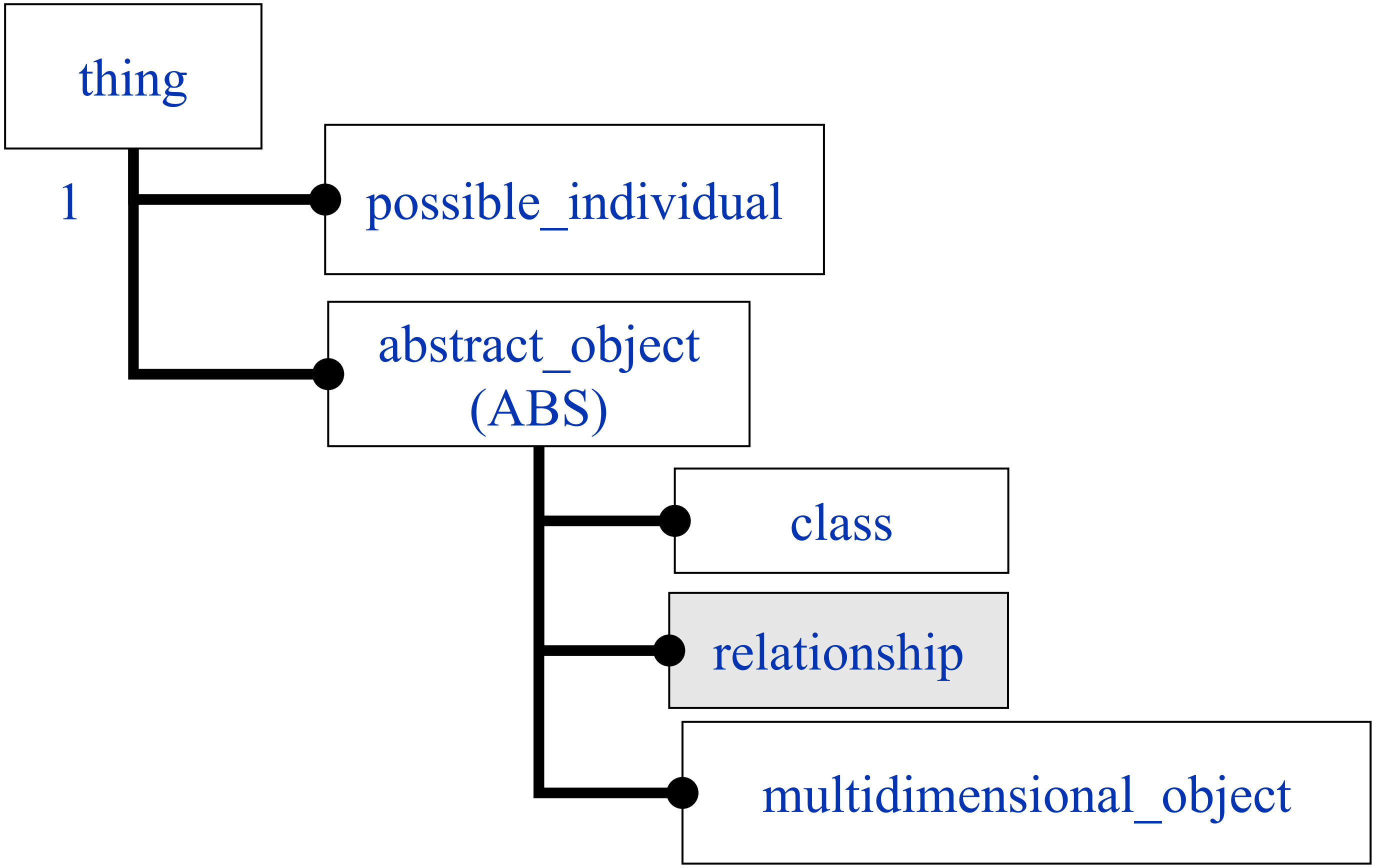
Blue

My car

#state1

#state2

Subclasses of Abstract Object: Relationship



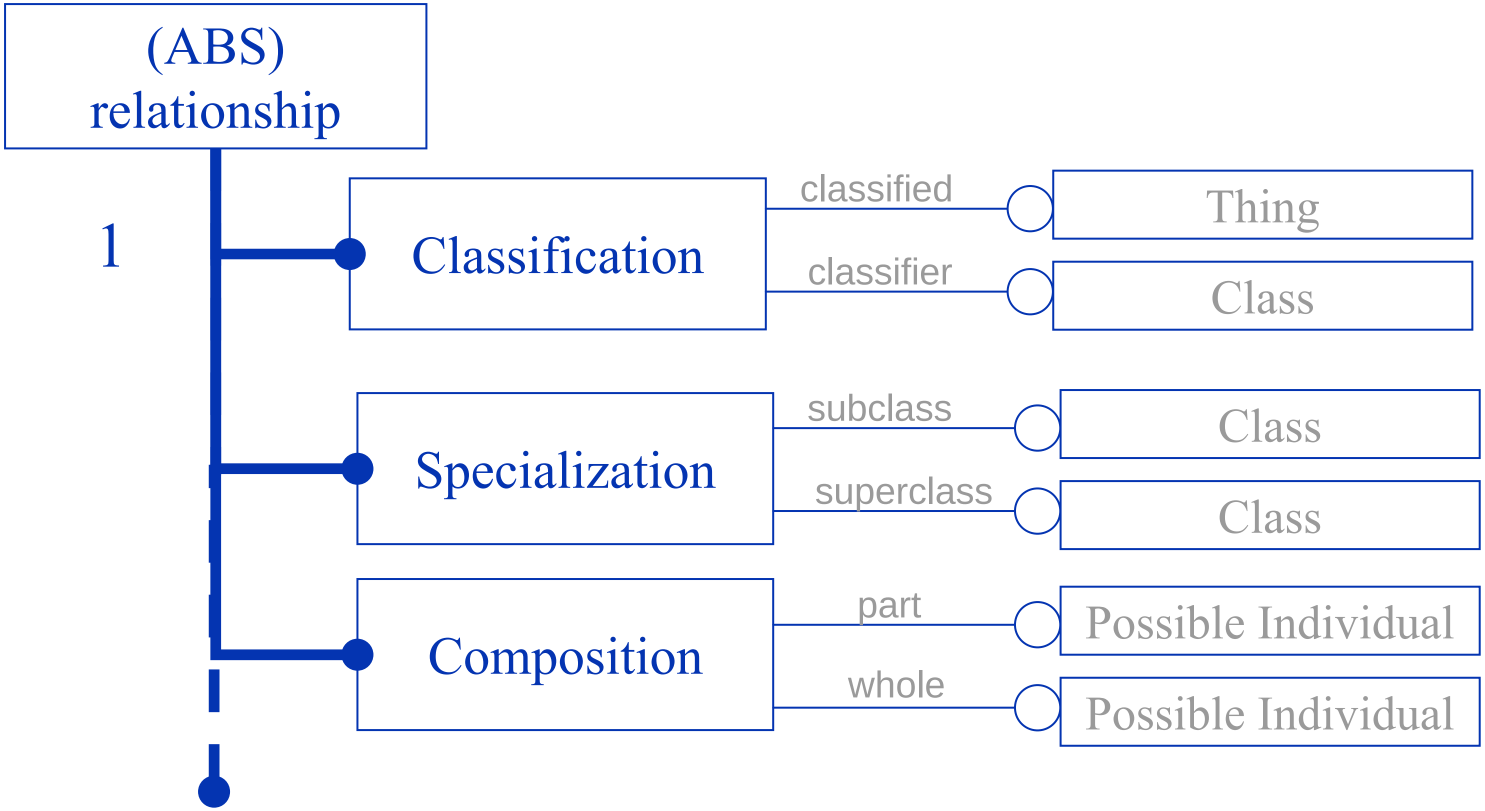
© Det Norske Veritas AS. All rights reserved

Subclasses of Abstract Object: Relationship

- A relationship is considered to be an abstract object.
- **Relationship**s are given names, as are all ***Thing***s. Moreover, they should always be classified by a ***ClassOfRelationship***.
- Only classes of binary relationship are supported. More complex objects can be supported using **MultidimensionalObject**.

Subclasses of Relationship

- A Relationship has the pair of objects related as attributes of the relationship.
- Relationship has several subtypes:



© Det Norske Veritas AS. All rights reserved

Classification and Specialization

- Classification ↔ membership
- Specialization ↔ inclusion

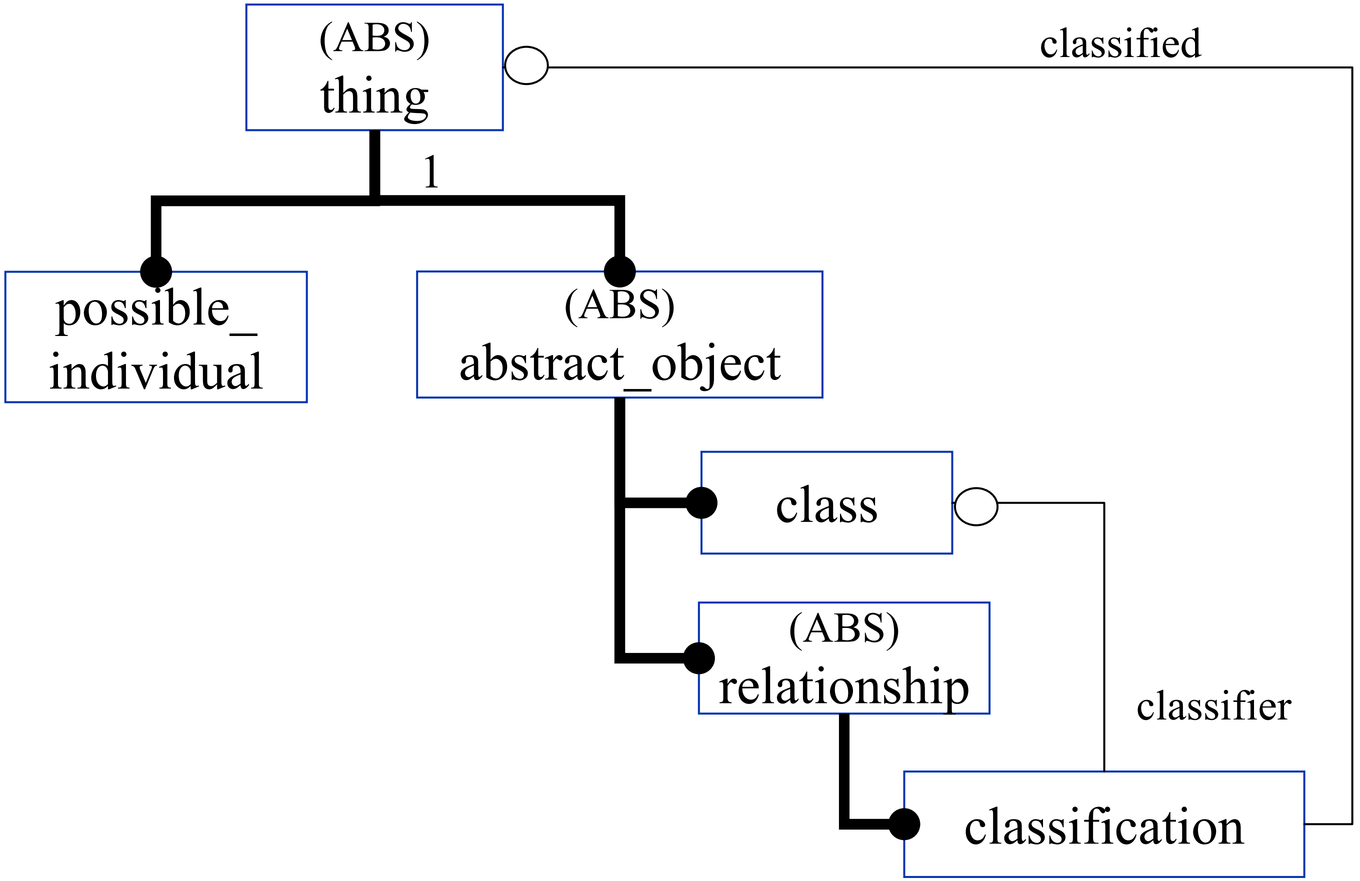
- Why create new names for familliar relations?
  - There is a distinction between the data model (Part2) and the RDL (Part 4)
  - Classes in Part 2 are entity types
  - Classes in Part 4 are instances of the Class entity type
  - Every object in the RDL must be a member of at least one entity type of Part 2

© Det Norske Veritas AS. All rights reserved

Classification and Specialization, cont.

- An RDL class such as Pump is an object with a certain entity type (Class of arranged Individual)
- Because it is a class, it may have members, or enter into inclusion relationships.
- How to indicate these relationships?
- Answer: You record the required relationship as a new piece of reference data. In other words, you create a Relationship object.
- The Classification entity type contains the membership relationships
- The Specialization entity type contains the inclusion relationships

Classification relationship



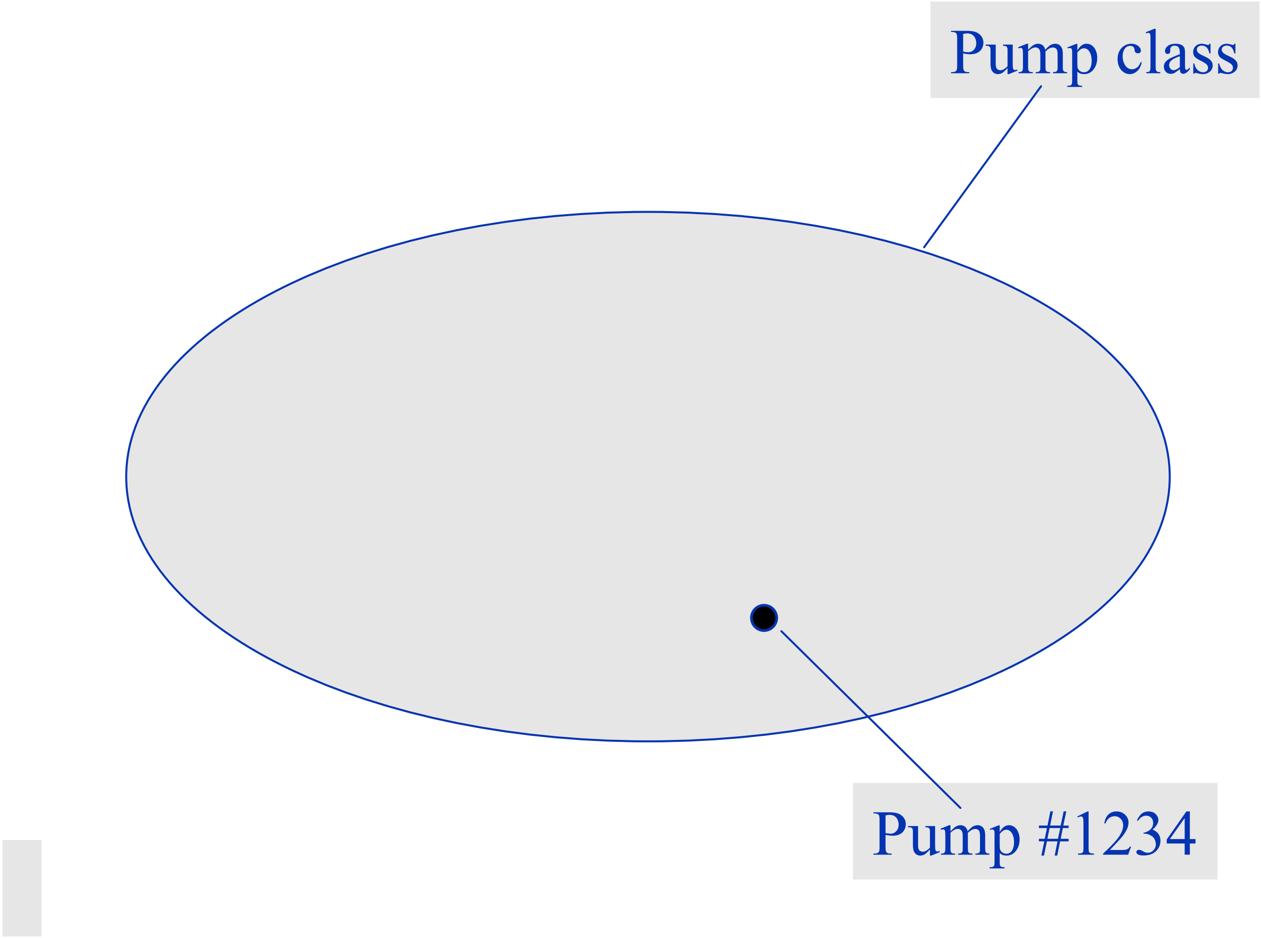
© Det Norske Veritas AS. All rights reserved

Classification Relationship

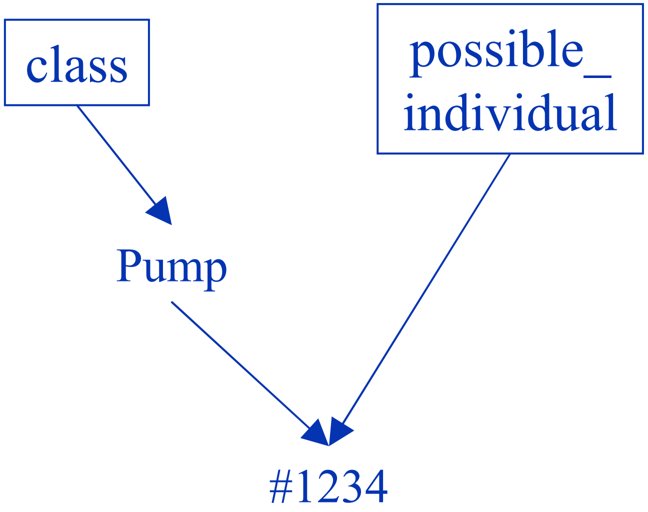
- A **Classification** is a **Relationship**
- The first attribute, called “classified”, is a **Thing**
- The second attribute, called “classifier” is a **Class**
- A **Classification** between a **Thing** and a **Class** says that the classified **Thing** is a member of the classifying **Class**

© Det Norske Veritas AS. All rights reserved

Classification Illustration



Example Diagram

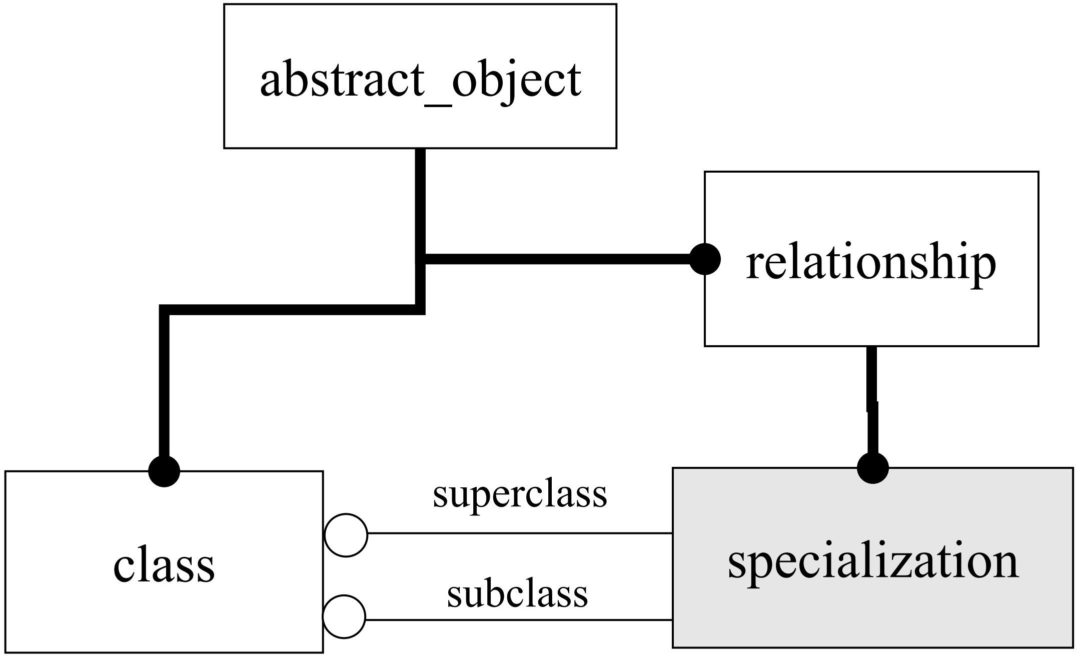


- #1234 is a possible_individual
- #1234 is a pump
- pump is a class
- class in an entity type
- possible_individual is an entity type

abcde	an object
xyz	a class that is a schema entity
→	classification relationship, pointing from the classifier to the classified

© Det Norske Veritas AS. All rights reserved

Specialization Relationship



abstract_object

relationship

class

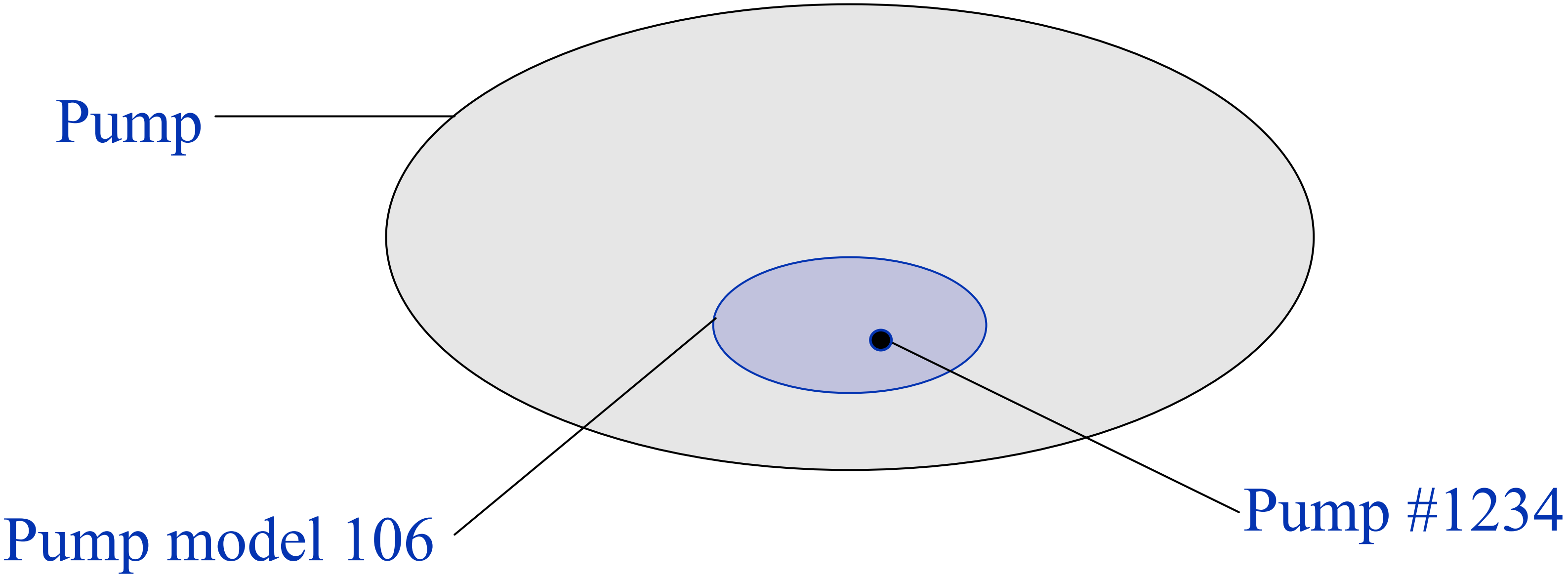
superclass

subclass

specialization

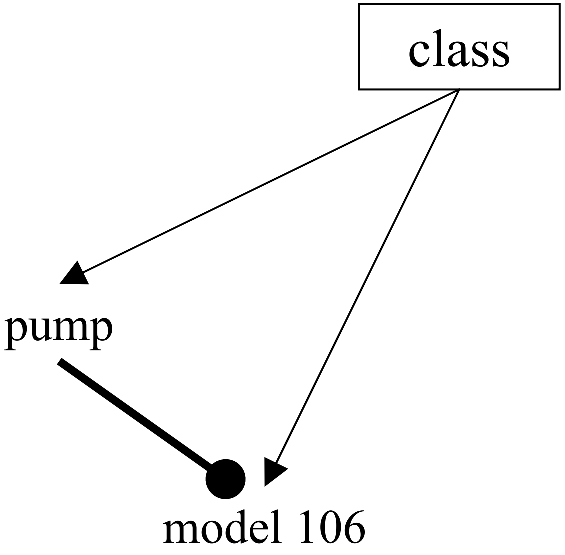
Specialization Relationship, cont.

- A specialization is a relationship between two classes that says that all members of the subclass are members of the superclass.
- Specialization is transitive. If A is a specialization of B and B is a specialization of C, then A is necessarily a specialization of C.



© Det Norske Veritas AS. All rights reserved

Specialization Example



All model 106 members are members of pump

—● a specialization relationship, the circle indicates the subclass

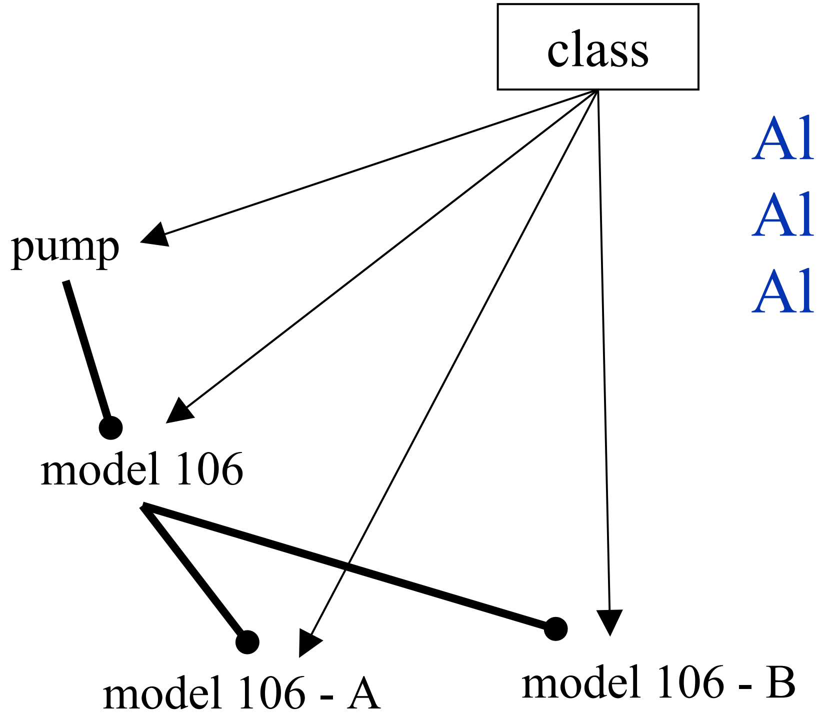
© Det Norske Veritas AS. All rights reserved

Exercise 2

- Rosemont manufacture a type of Gauge Pressure Transmitter know as 3051C
- Represent by means of a diagram
  - That 3051C is a class
  - That 3051Cs are Gauge Pressure Transmitters
  - Gauge Pressure Transmitter is a class
  - Class is an entity type

Transitivity

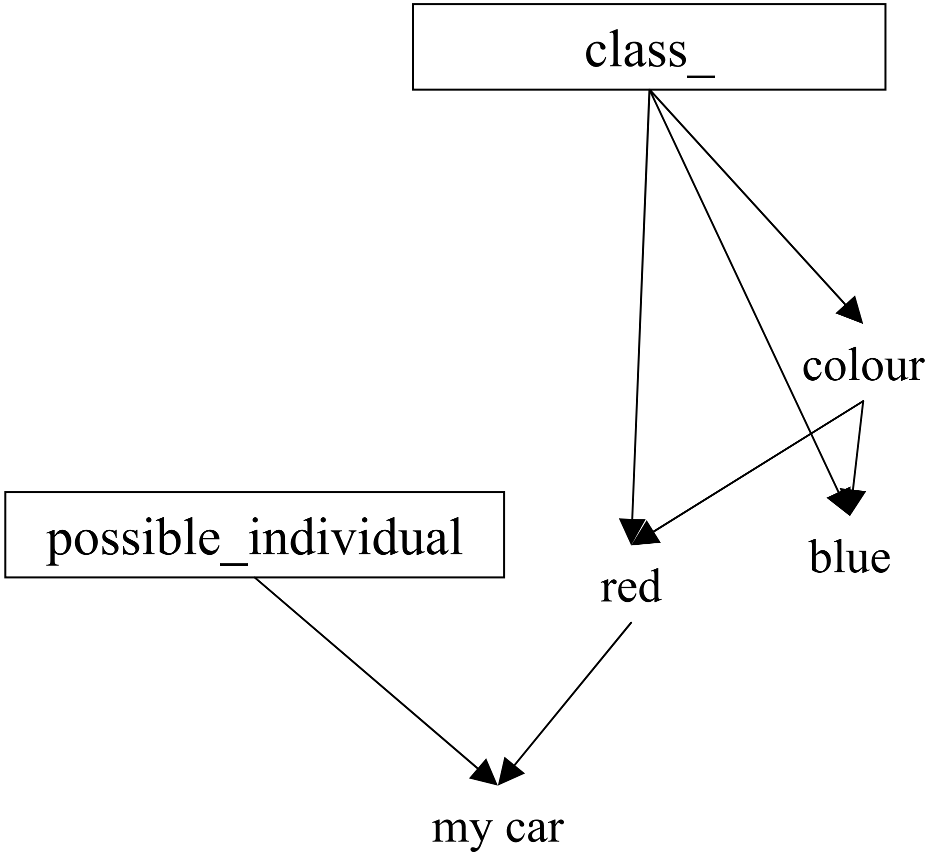
- Specialization is transitive



All model 106-As are model 106s
All model 106s are pumps
All model 106-As are pumps

Transitivity, cont.

- Classification is **not** transitive

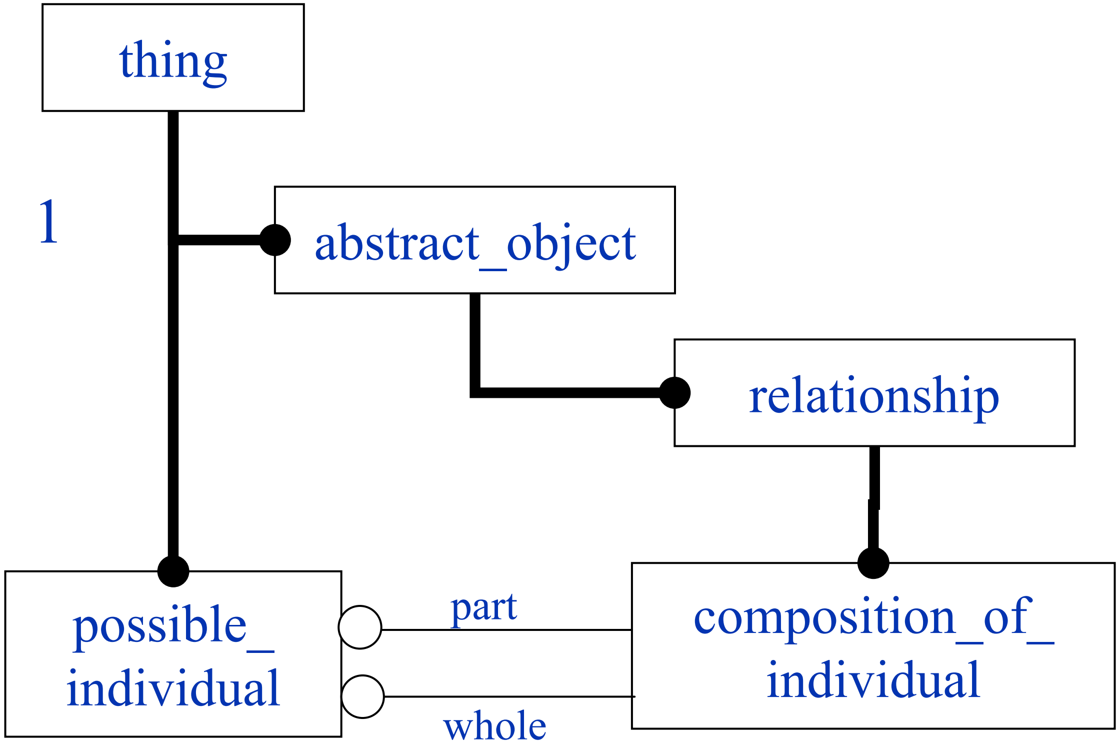


My car is red
Red is a colour
My car is not a colour
Red is class
Colour is a class
My car is not a class!

© Det Norske Veritas AS. All rights reserved

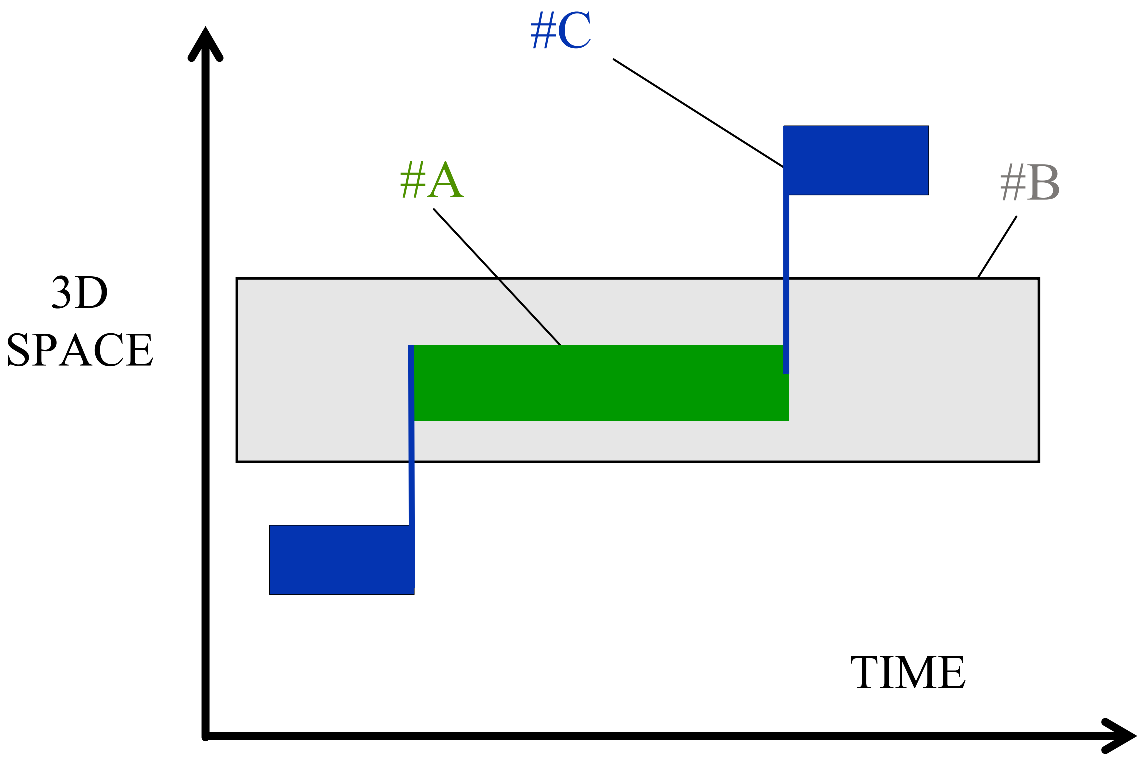
Composition Relationship

- Says that something is a part of something else



© Det Norske Veritas AS. All rights reserved

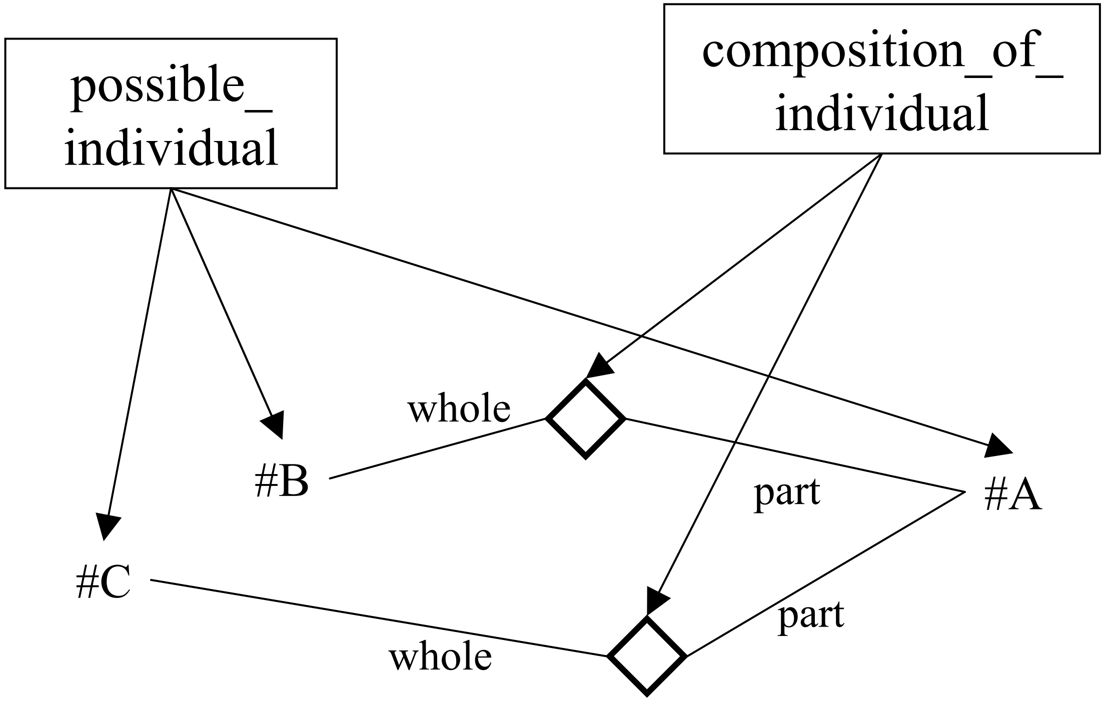
Composition in Four Dimensions

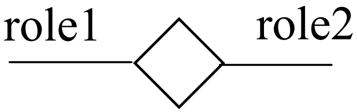


#A is a part of #B and of #C

© Det Norske Veritas AS. All rights reserved

Composition Example





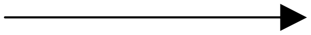
a relationship that is not a classification or specialization, role1 and role2 are the role names of the relationship

© Det Norske Veritas AS. All rights reserved

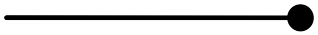
Diagrammatic Notation

abcde

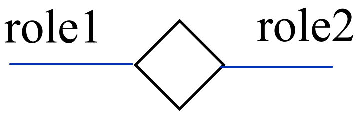
an object



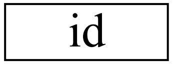
a classification relationship, the arrow head indicates the member of the class



a specialization relationship, the circle indicates the subclass



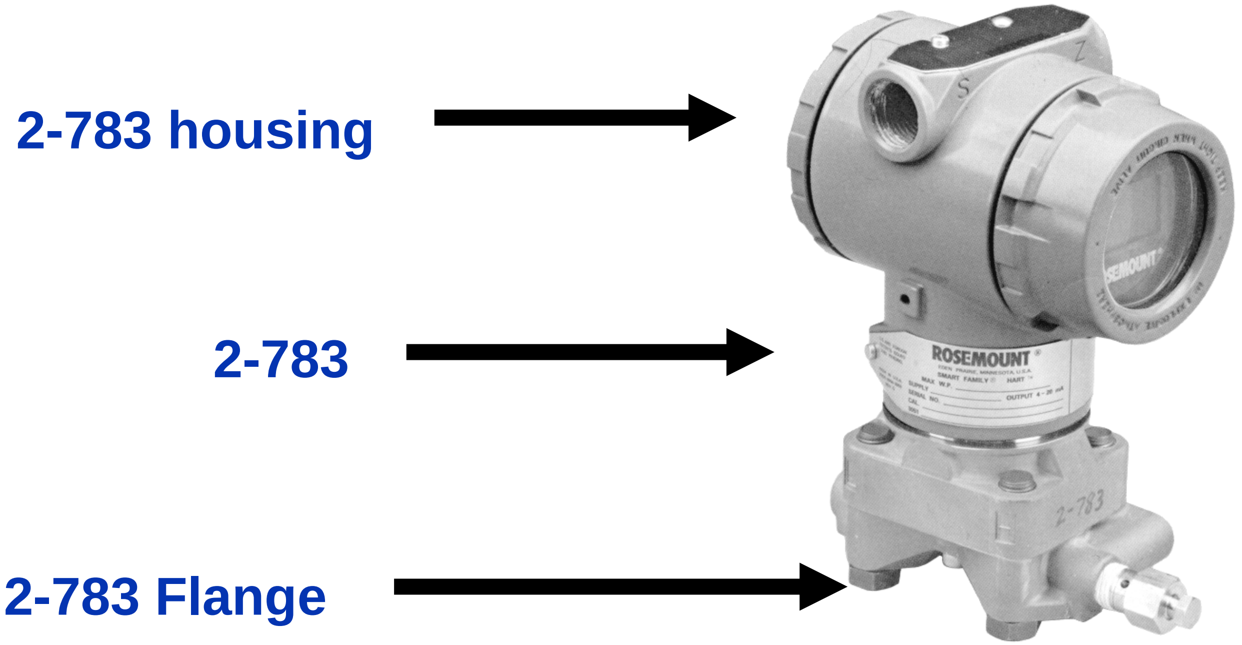
a relationship that is not a classification or a specialization, role1 and role2 are the role names of the relationship



a class that is a model entity type, where “id” is its identifier

Exercise 3

- Represent by means of a diagram the parts of 2-783.



2-783 housing

2-783

2-783 Flange

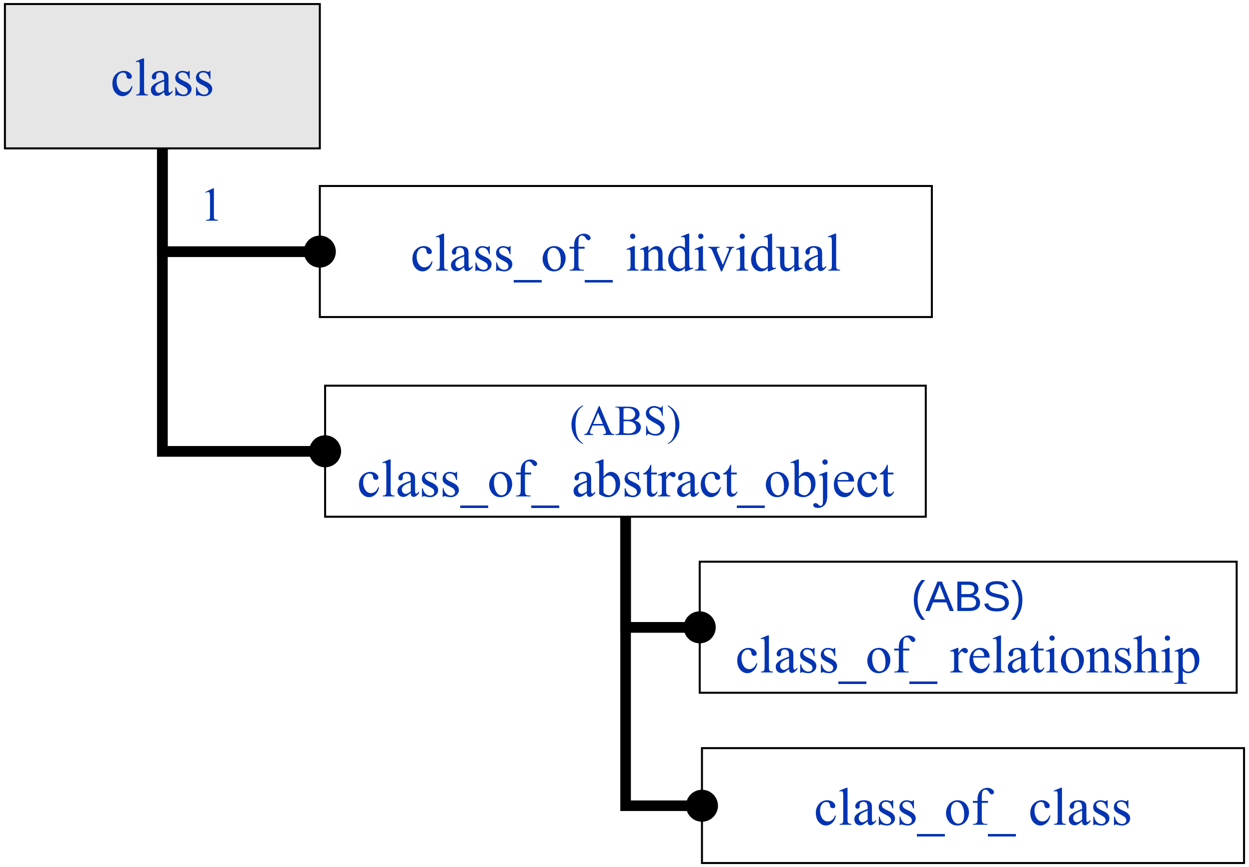
© Det Norske Veritas AS. All rights reserved

Tutorial Overview

The Data Model

- Fundamental Concepts
- The EXPRESS Modeling Language
- The Basic Classes of 15926
- **Types of Reference Data Classes**
  - Class of Class
  - Class of Relationship
  - Class of Individual
    - Class of Arranged Individual
    - Property
- Templates

Types of Class



class

1

class_of_ individual

(ABS)
class_of_ abstract_object

(ABS)
class_of_ relationship

class_of_ class

© Det Norske Veritas AS. All rights reserved

Class of Class Examples

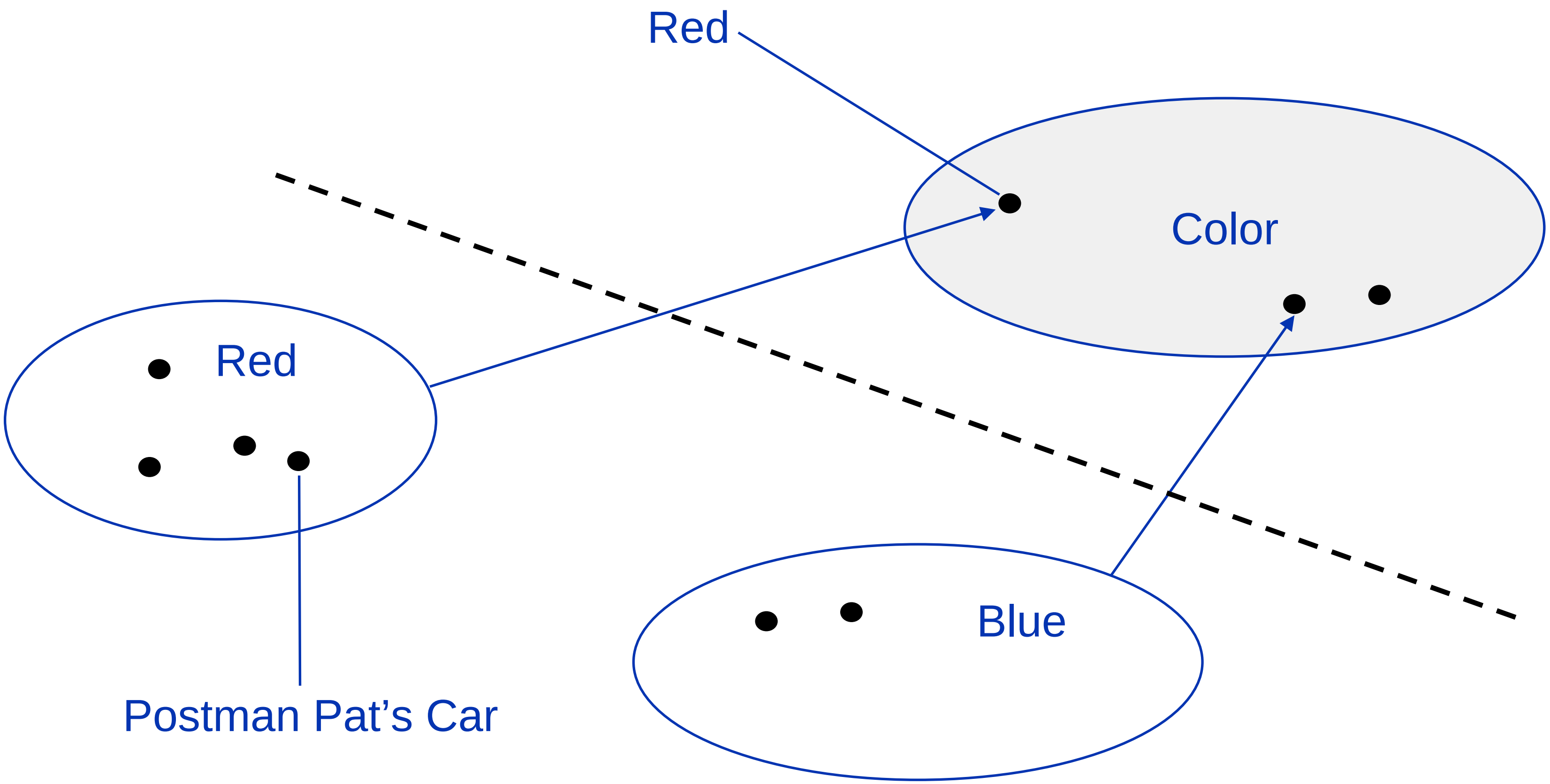
- **Red** is a class of individual, its members are possible individuals that are red, only possible individuals can be red.
- **Colour** is a class of class, the class **Red** is a member. The class **Blue** is another member of **Colour**. Individuals are not colours.

- **PolarBear** is a class of individual; Its members are individual polar bears.
- **EndangeredSpecies** is a class of class, where the class **PolarBear** is a member. It is the species that is endagered, not each and every polar bear.

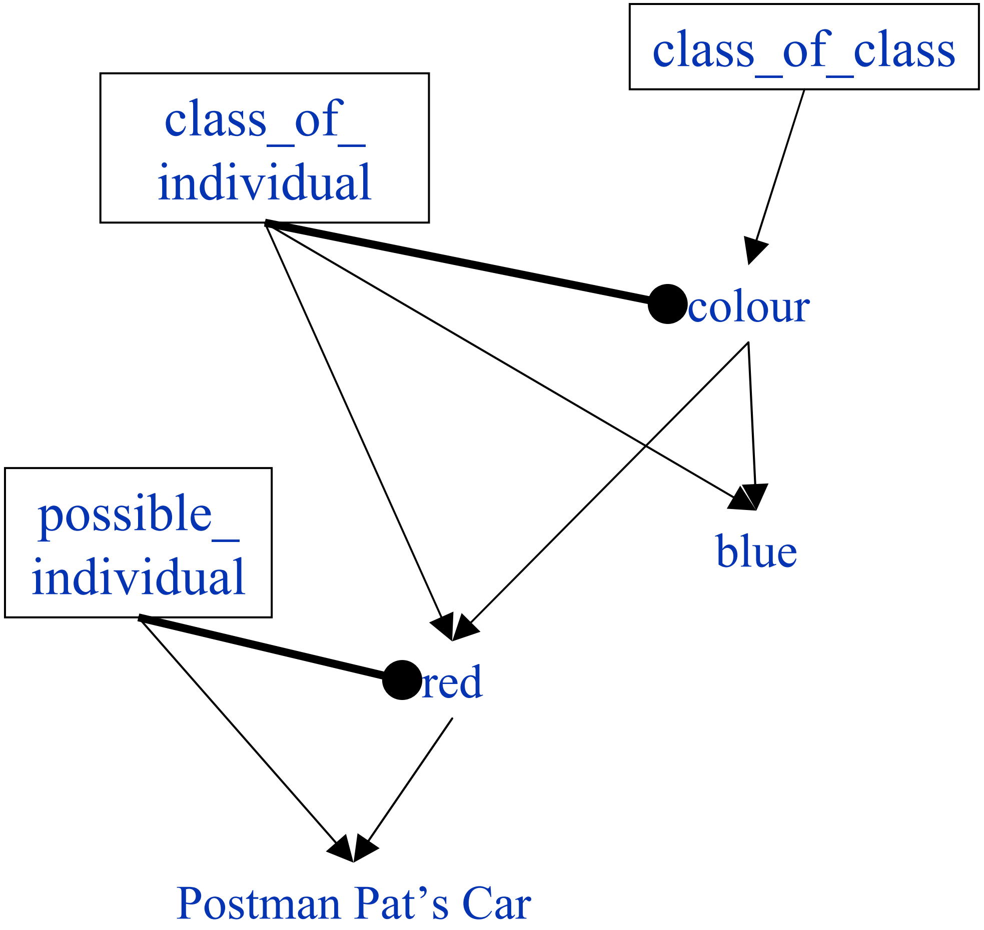
- **Honda** is a class of individual, its members are all Hondas.
- *Car is a class of individual also.* The members of **Car** are individual cars. **Honda** is a specialization of **Car**. Every Honda is a car
- **CarTypes** is a class of class, the class **Honda** is a member. Individual cars are not car types.

© Det Norske Veritas AS. All rights reserved

Class of Individual and Class of Class



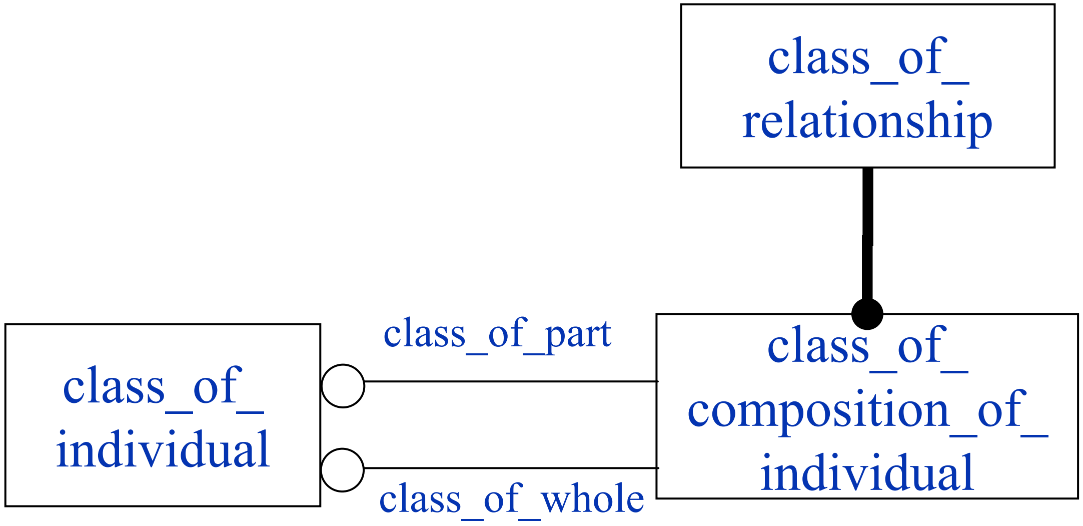
Class of Individual and Class of Class



© Det Norske Veritas AS. All rights reserved

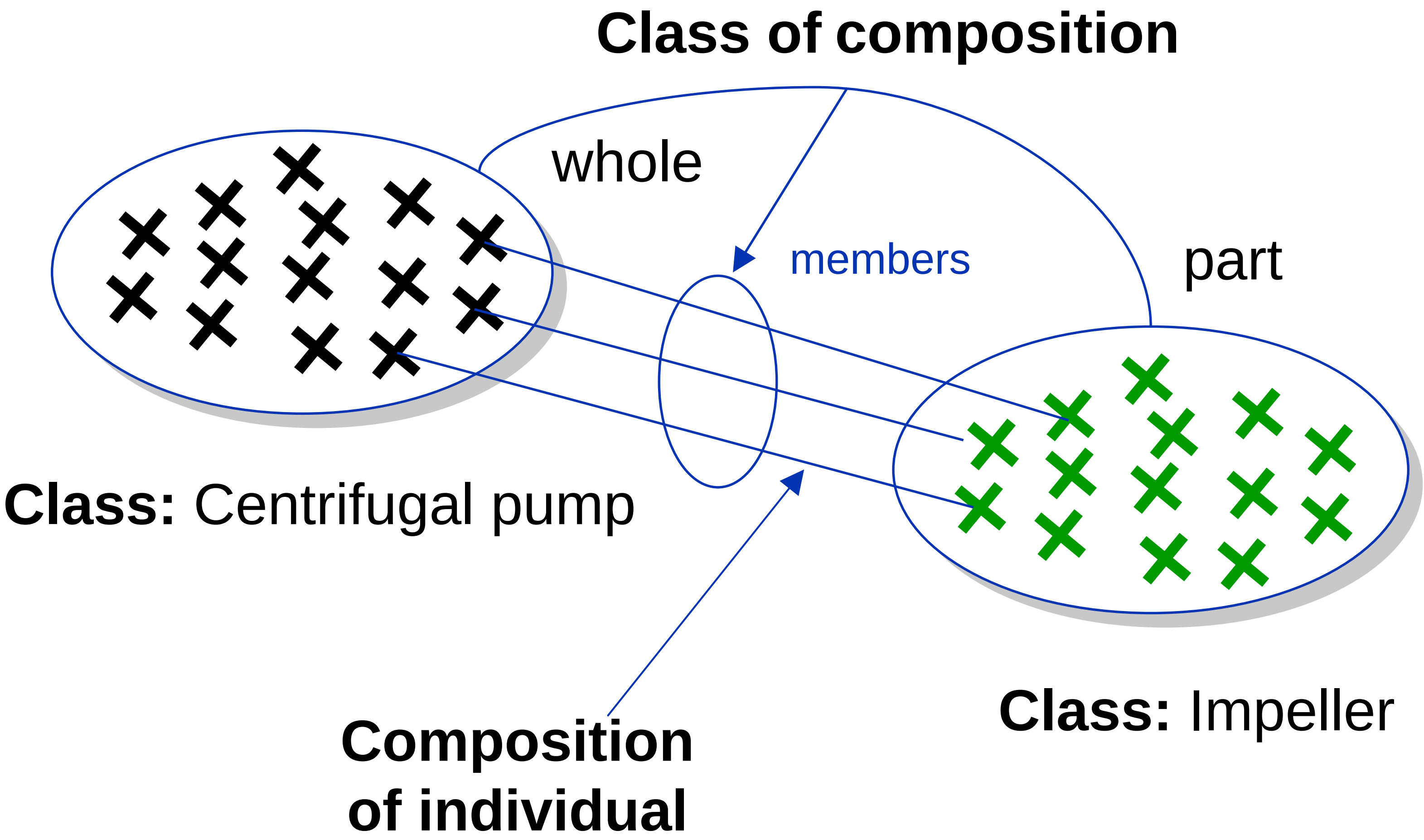
Class of Relationship

- Classes whose members are Relationships



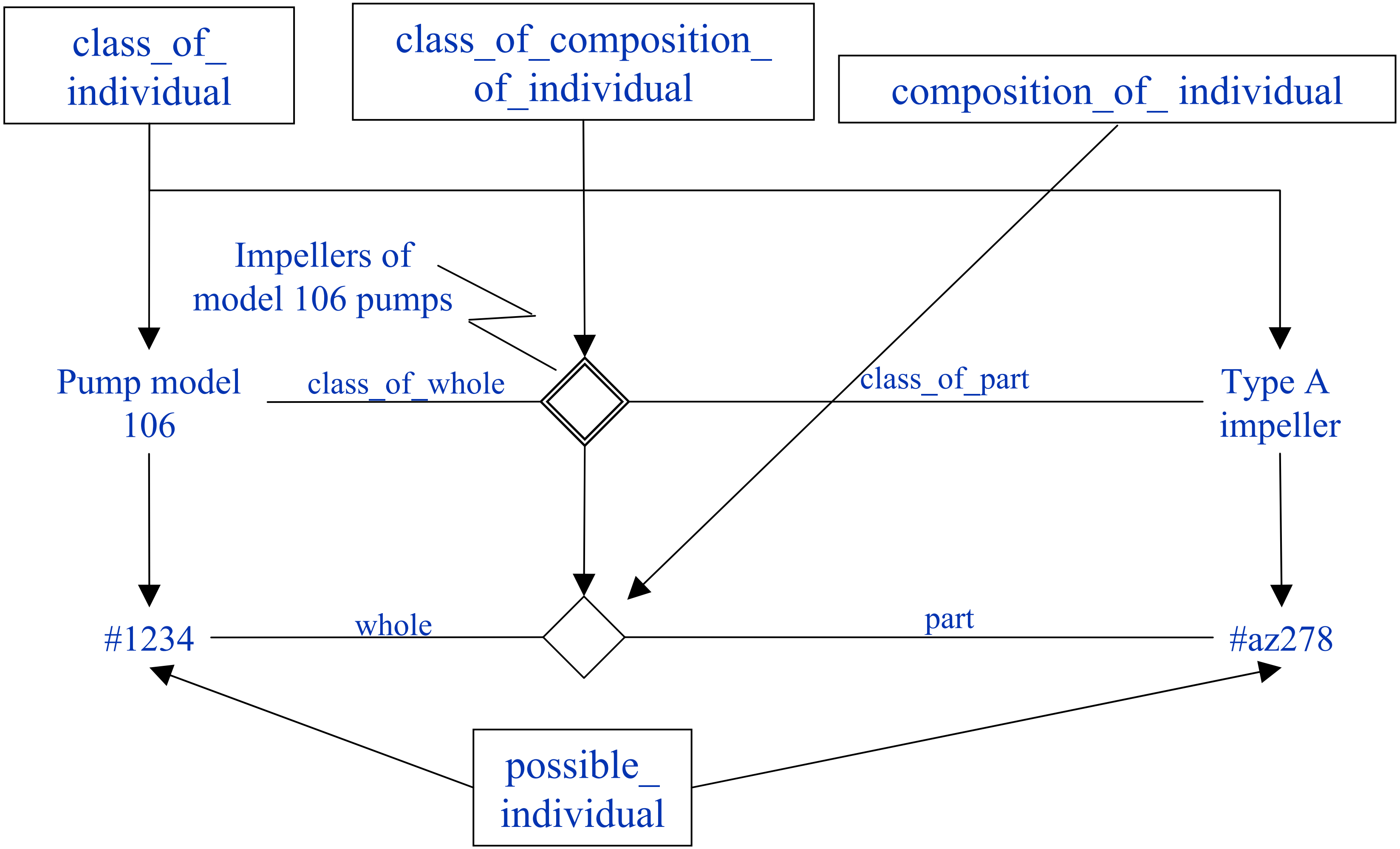
© Det Norske Veritas AS. All rights reserved

Class of Composition membership



© Det Norske Veritas AS. All rights reserved

Composition example: Pumps and Impellers



class_of_ individual

class_of_composition_ of_individual

composition_of_ individual

Impellers of model 106 pumps

Pump model 106

class_of_whole

class_of_part

Type A impeller

#1234

whole

part

#az278

possible_ individual

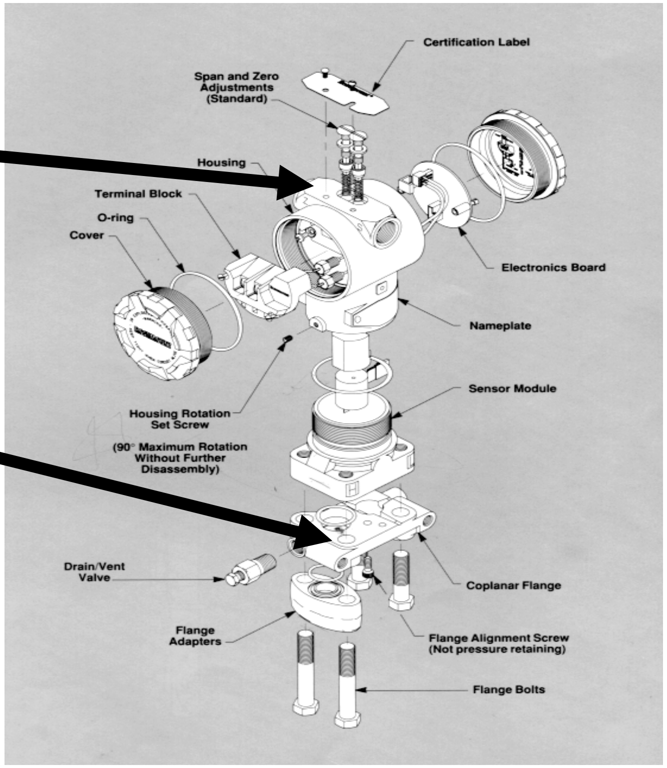
Exercise 5

- Represent by means of a diagram that all GPT-305Cs consist of a housing and a flange

GPT-3052C

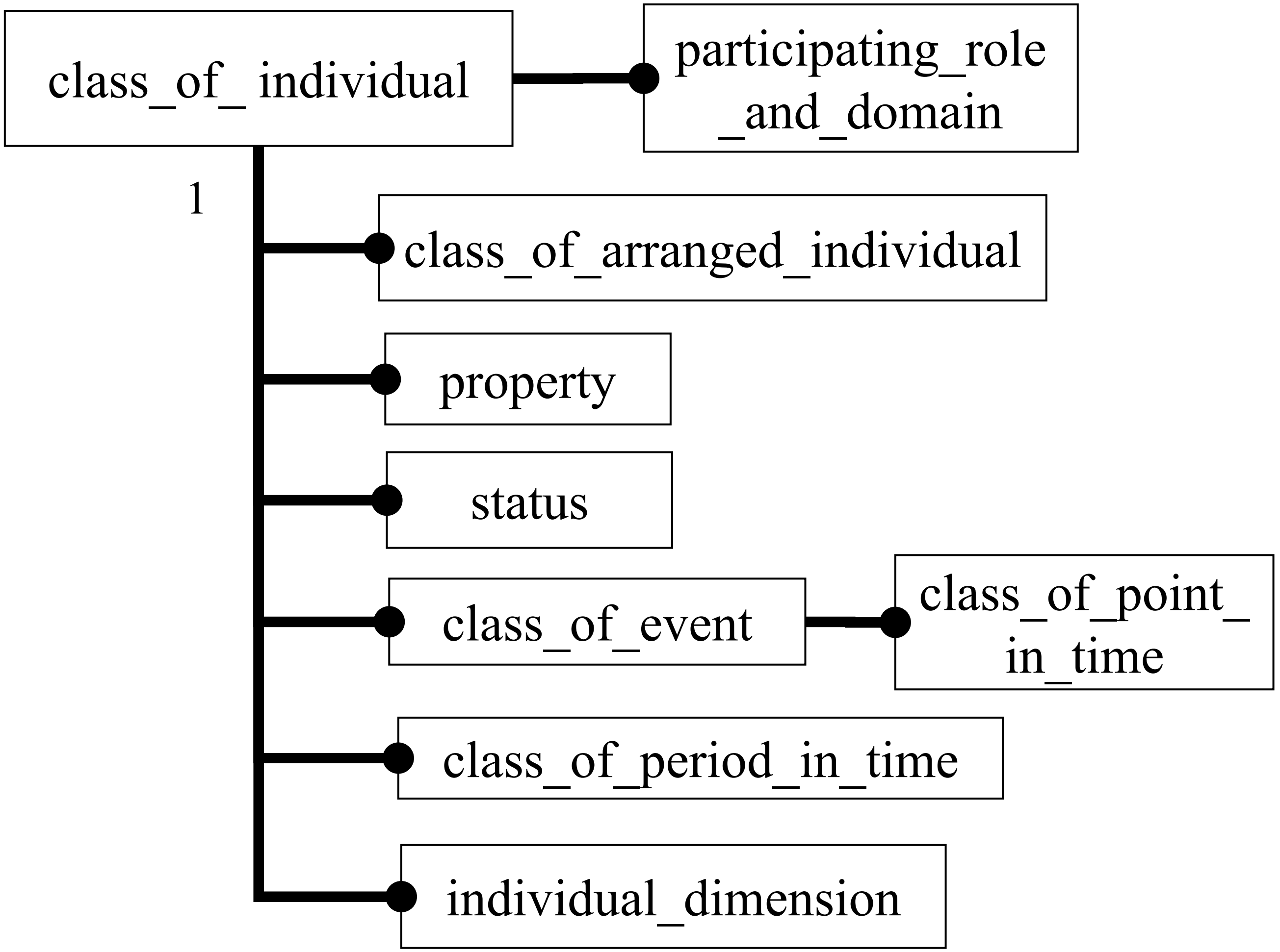
GPT-3052C housing

GPT-3052C flange


Certification Label
Span and Zero Adjustments (Standard)
Housing
Terminal Block
O-ring
Cover
Electronics Board
Nameplate
Sensor Module
Housing Rotation Set Screw
(90° Maximum Rotation Without Further Disassembly)
Drain/Vent Valve
Coplanar Flange
Flange Adapters
Flange Alignment Screw (Not pressure retaining)
Flange Bolts

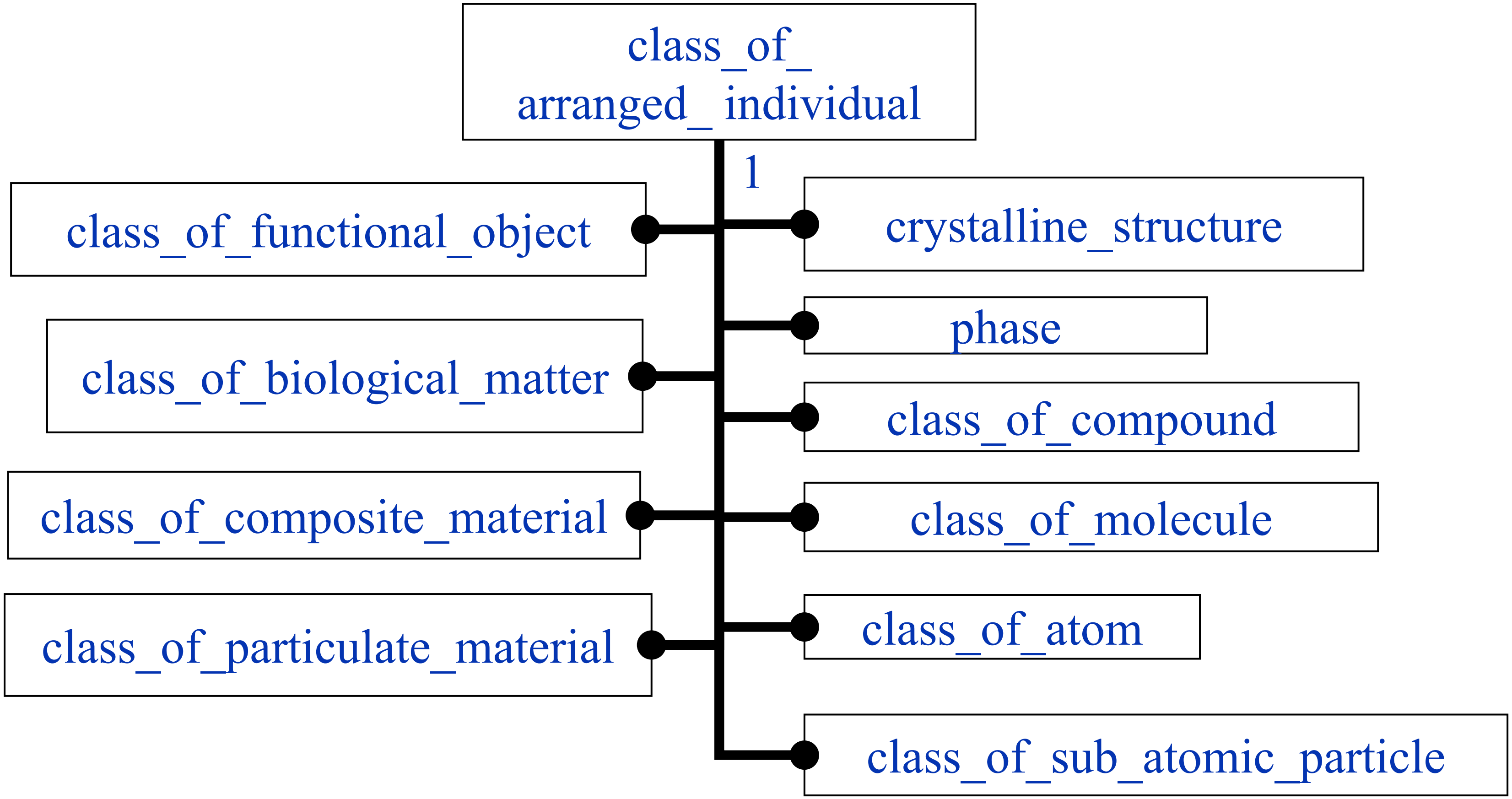
© Det Norske Veritas AS. All rights reserved

Classes of Individual



© Det Norske Veritas AS. All rights reserved

Class of Arranged Individual



class_of_arranged_ individual

1

class_of_functional_object

class_of_biological_matter

class_of_composite_material

class_of_particulate_material

crystalline_structure

phase

class_of_compound

class_of_molecule

class_of_atom

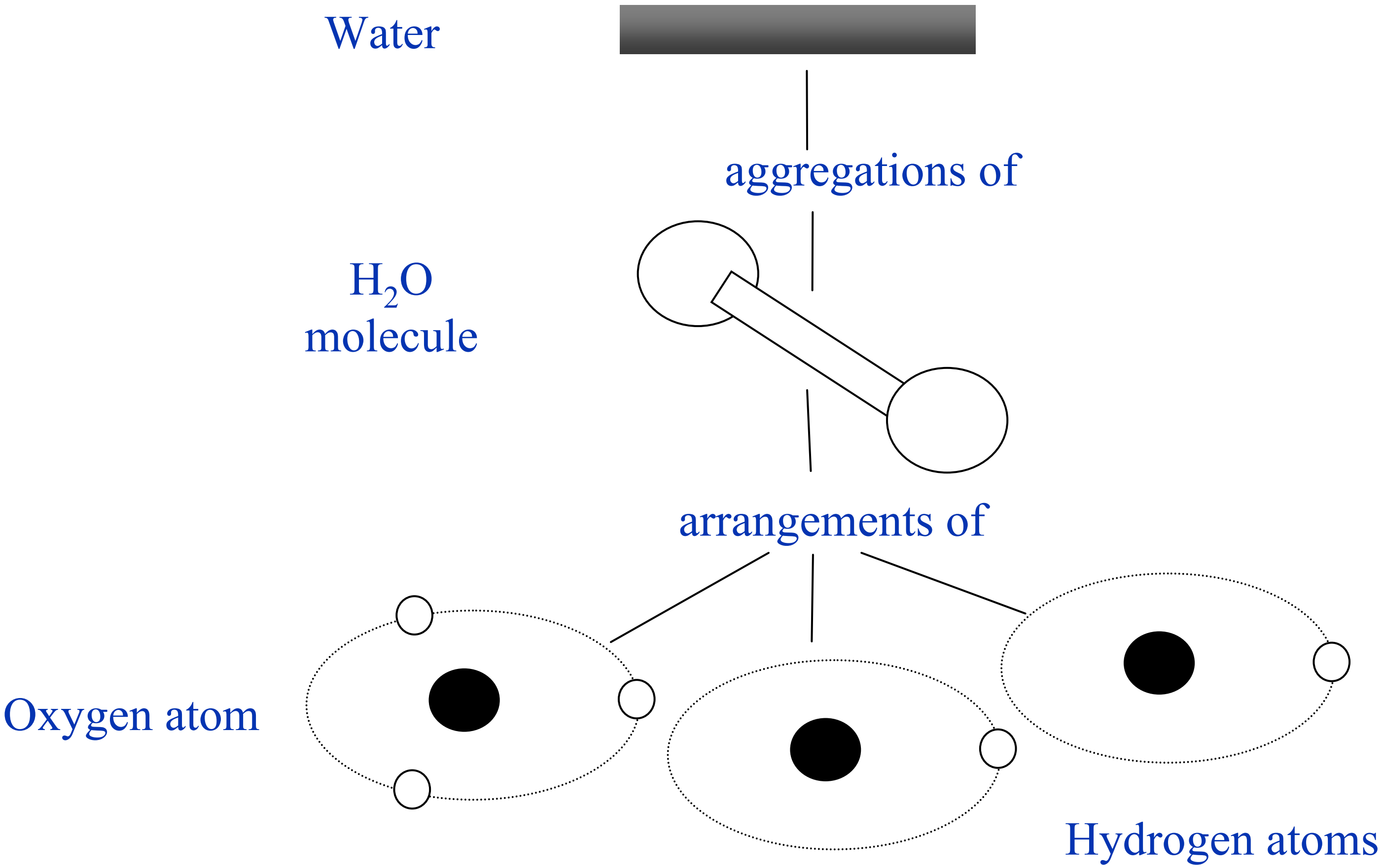
class_of_sub_atomic_particle

© Det Norske Veritas AS. All rights reserved

Increasing levels of aggregation

- The classes of arranged individual recognise increasing levels of arrangement, starting at the sub atomic level and finishing at the levels of biological material and functional objects.
- Each level consists of an aggregation of material of the level beneath

Aggregation illustration



Water

aggregations of

H_2O molecule

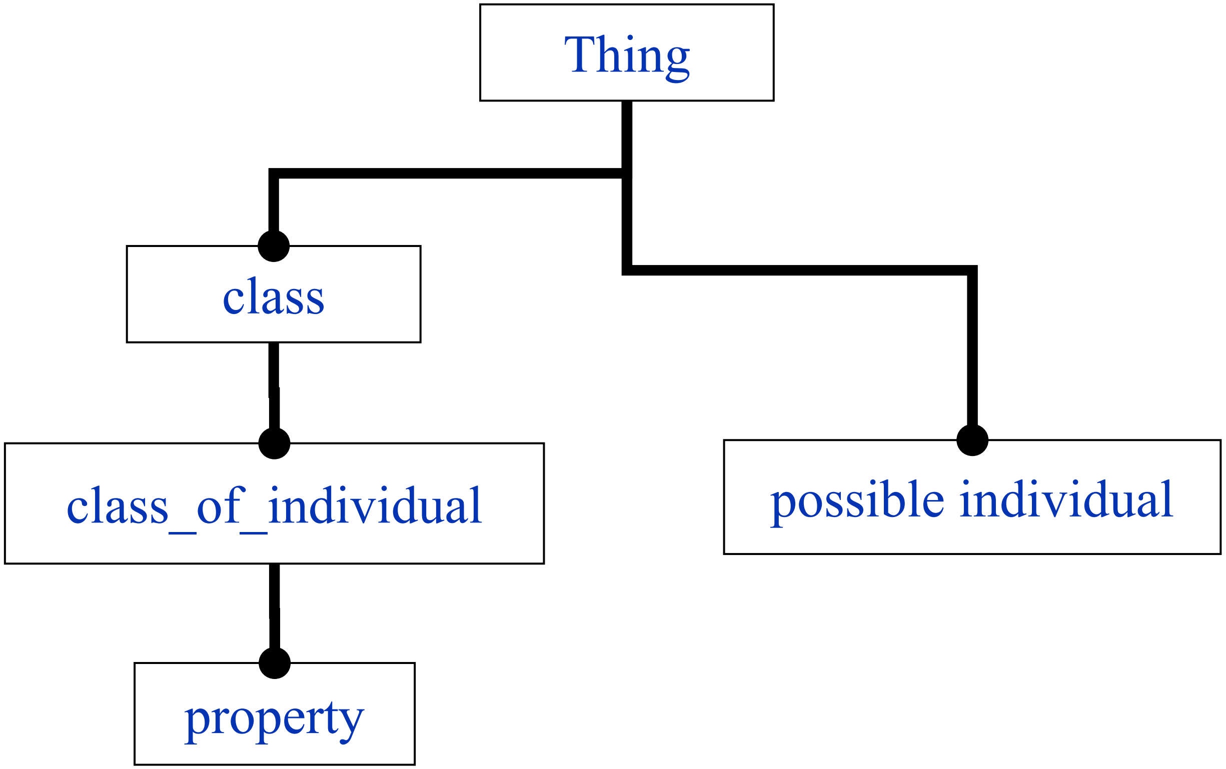
arrangements of

Oxygen atom

Hydrogen atoms

© Det Norske Veritas AS. All rights reserved

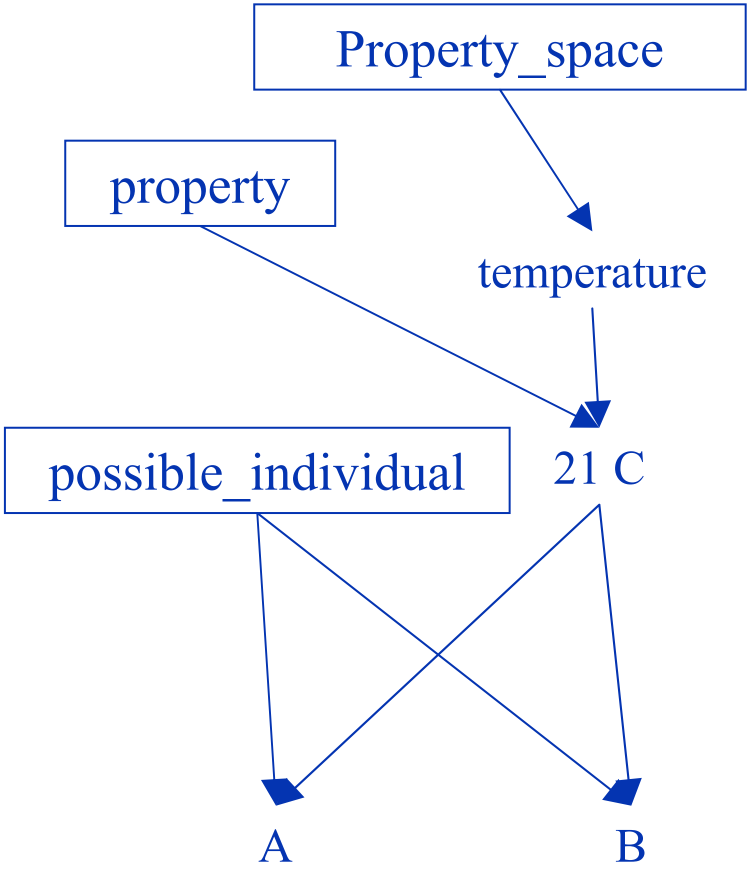
The Property Entity Type



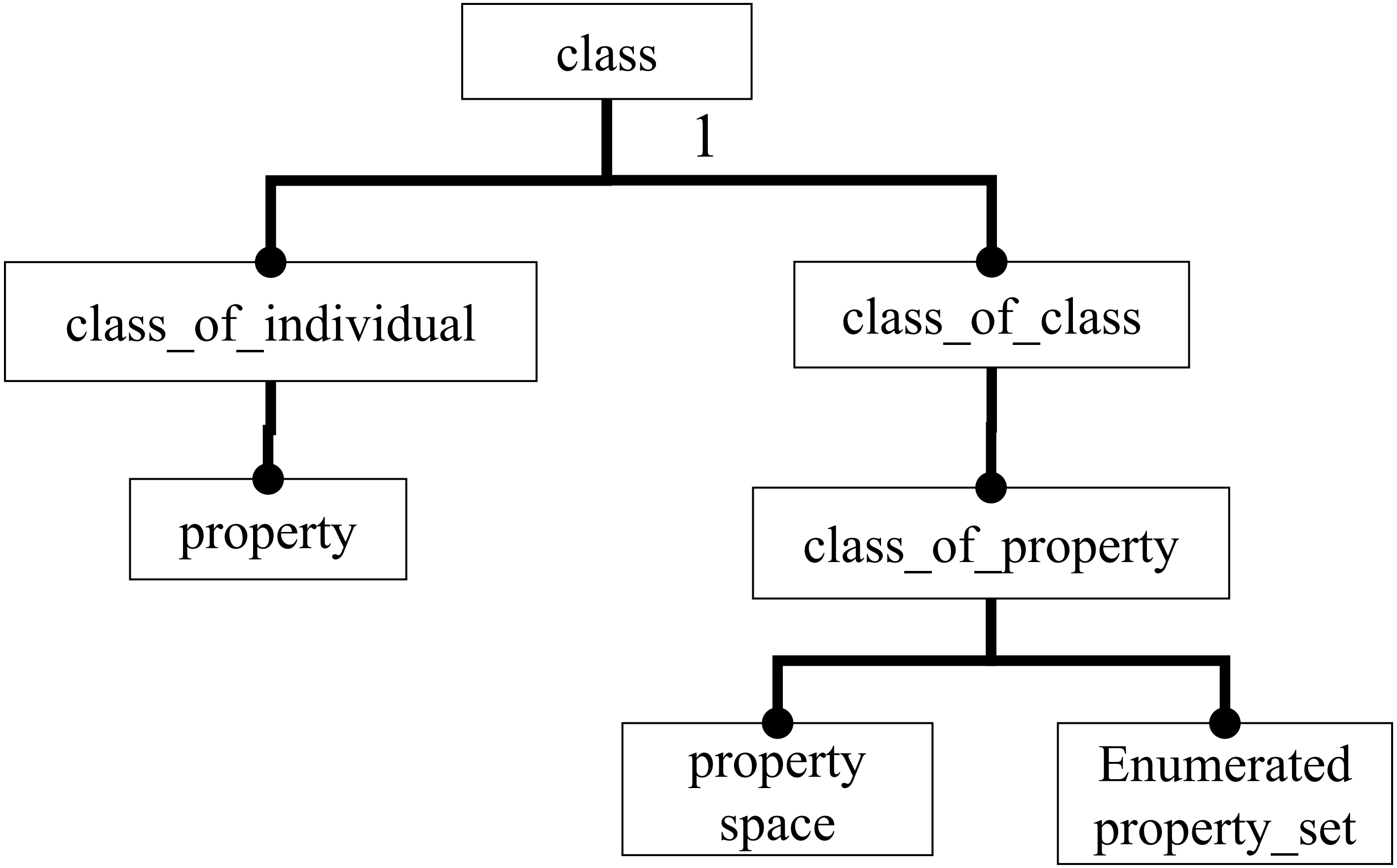
© Det Norske Veritas AS. All rights reserved

Property

- Property is a subtype of Class of Individual
- Examples of Properties
  - 400 Tonnes
  - 4ºF
- A Possible Individual is a member of a Property if and only if it ”has” this property
  - *”The `Eiffel Tower` is a member of the class ”`100000 tonnes`”*
- Every property is a member of a Class of Property
  - Weight
  - Temperature
  - Volume

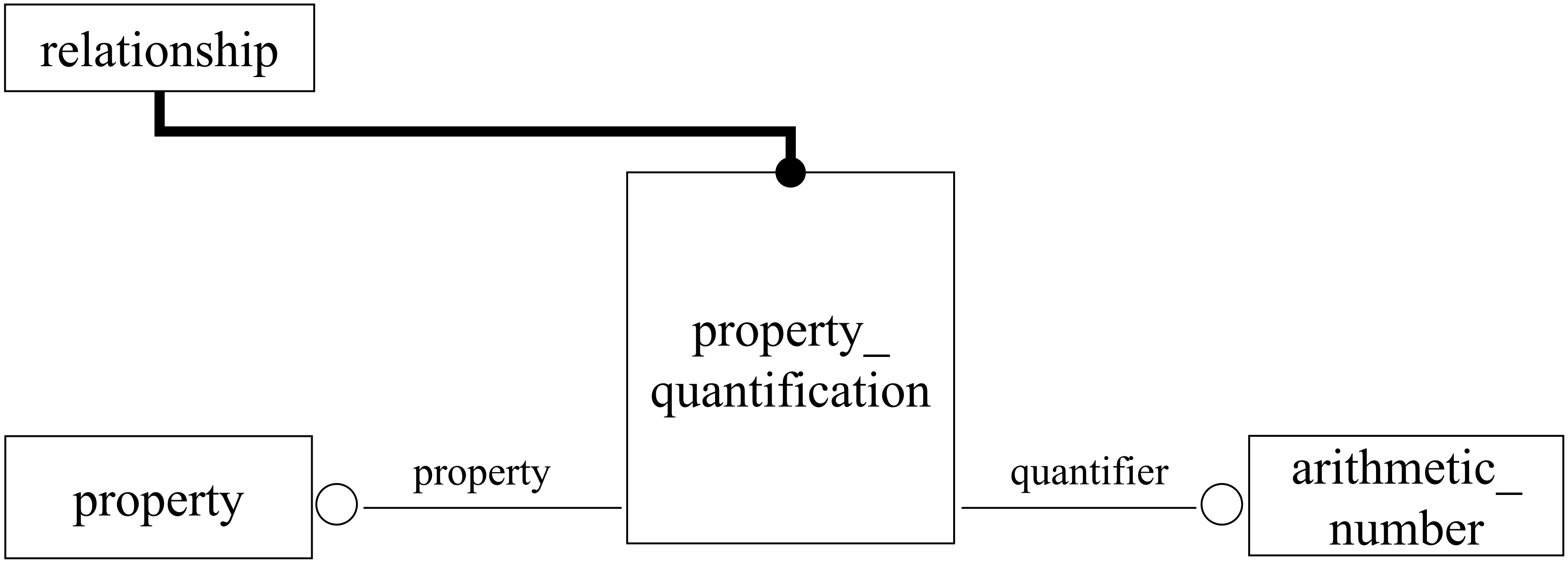


Property and Class of Property



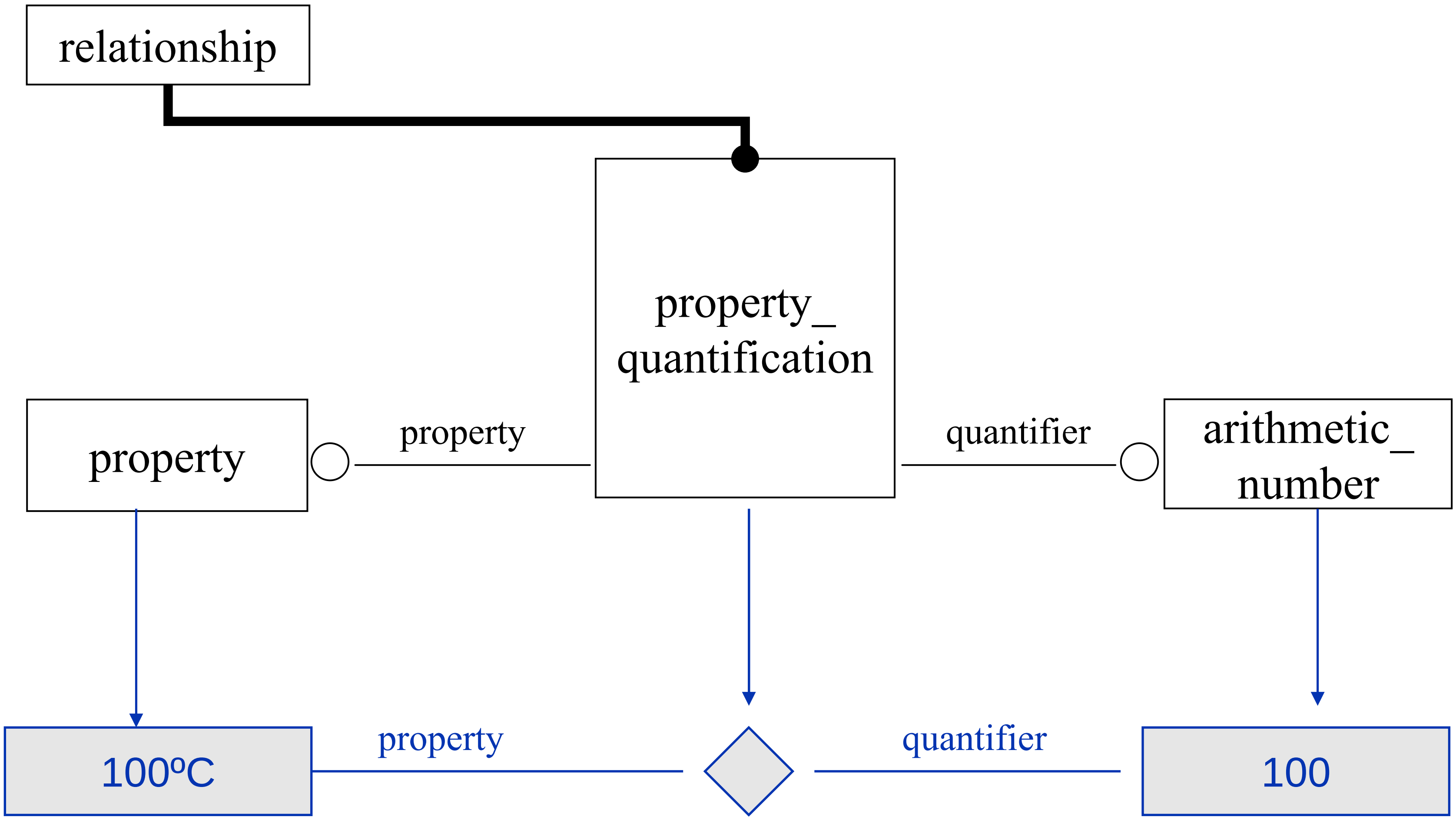
© Det Norske Veritas AS. All rights reserved

Property Quantification



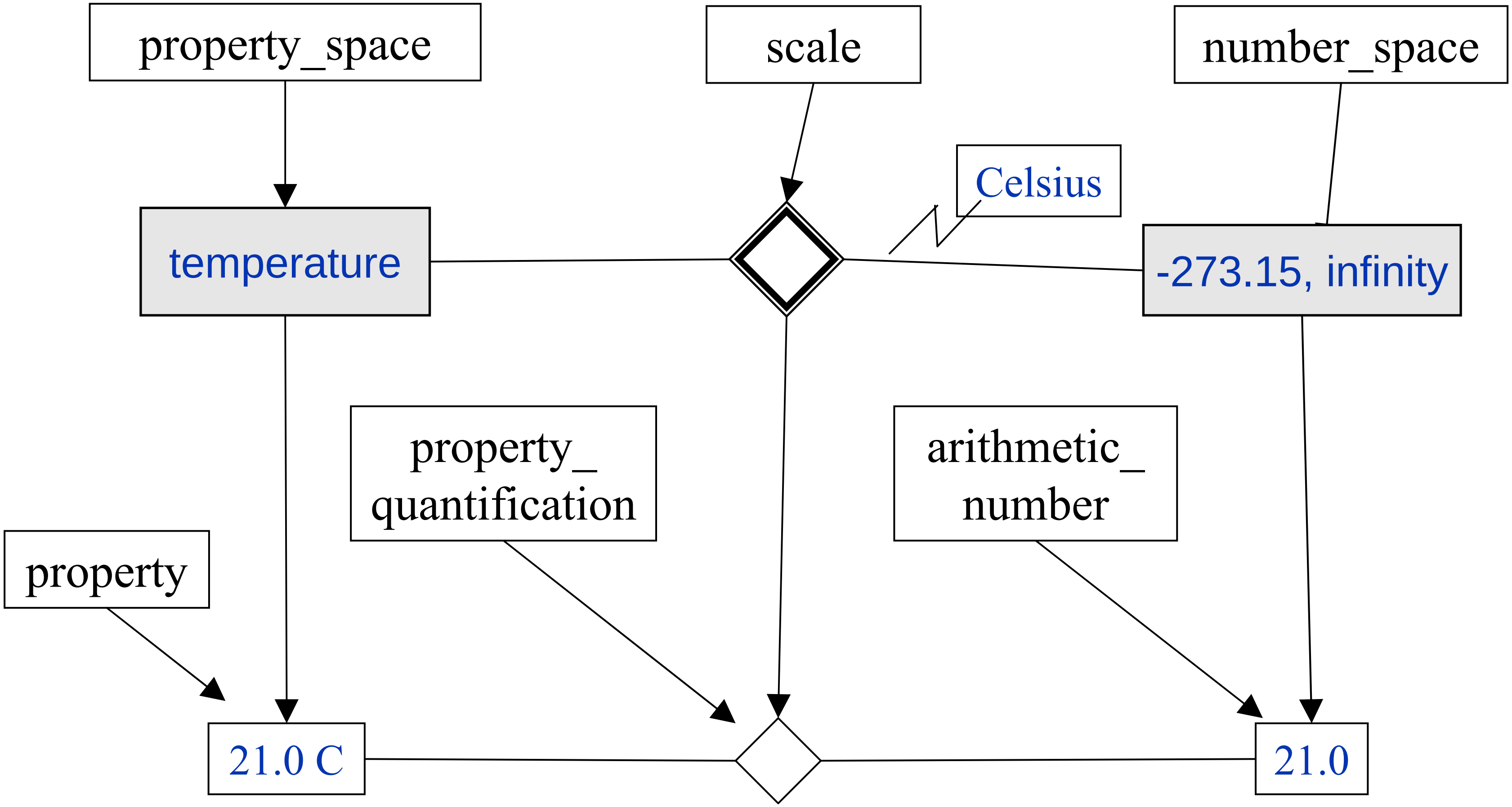
© Det Norske Veritas AS. All rights reserved

Property Quantification



© Det Norske Veritas AS. All rights reserved

Property Scales



Indirect property

- An **IndirectProperty** is derived from doing some tests or calculations to determine its value (as opposed to it being a current measurement).
- A relationship between a **Property** and a **PossibleIndividual**.
- The nature of the **IndirectProperty** is defined by its classification by a **ClassOfIndirectProperty**

© Det Norske Veritas AS. All rights reserved

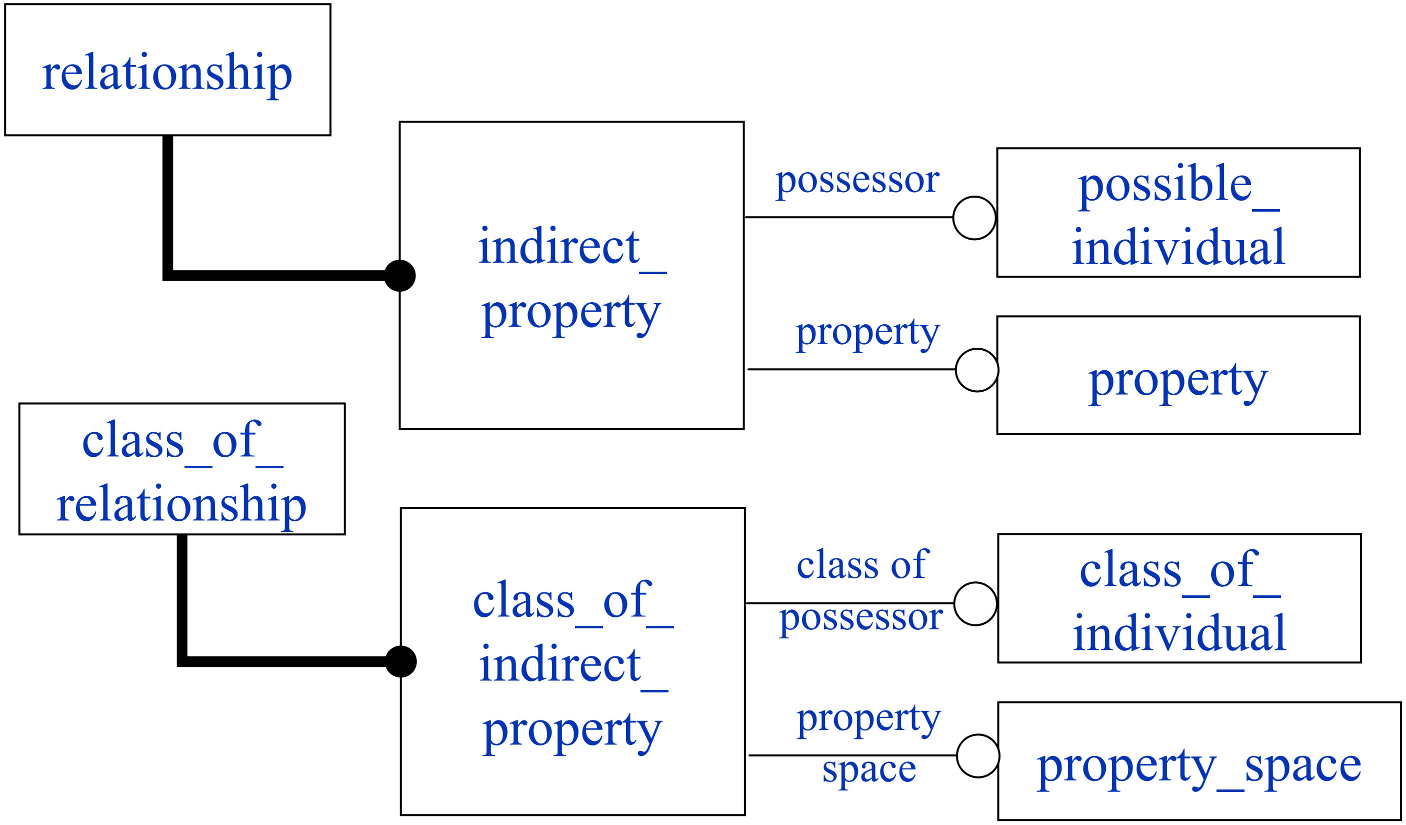
Indirect Property

- Example: The maximum allowable working temperature of an instrument is an indirect property, it is different from its current temperature.
- An indirect property is derived from doing some tests or calculations to determine its value (as opposed to it being a current measurement). This is what makes it indirect
- A relationship between a property and a possible individual.

  The nature of the indirect property is defined by its classification by a class of indirect property.

  A property is indirect when it does not directly apply to the possible individual it applies to, but is derived from some process.

Indirect property Model



© Det Norske Veritas AS. All rights reserved

Tutorial Overview

The Data Model

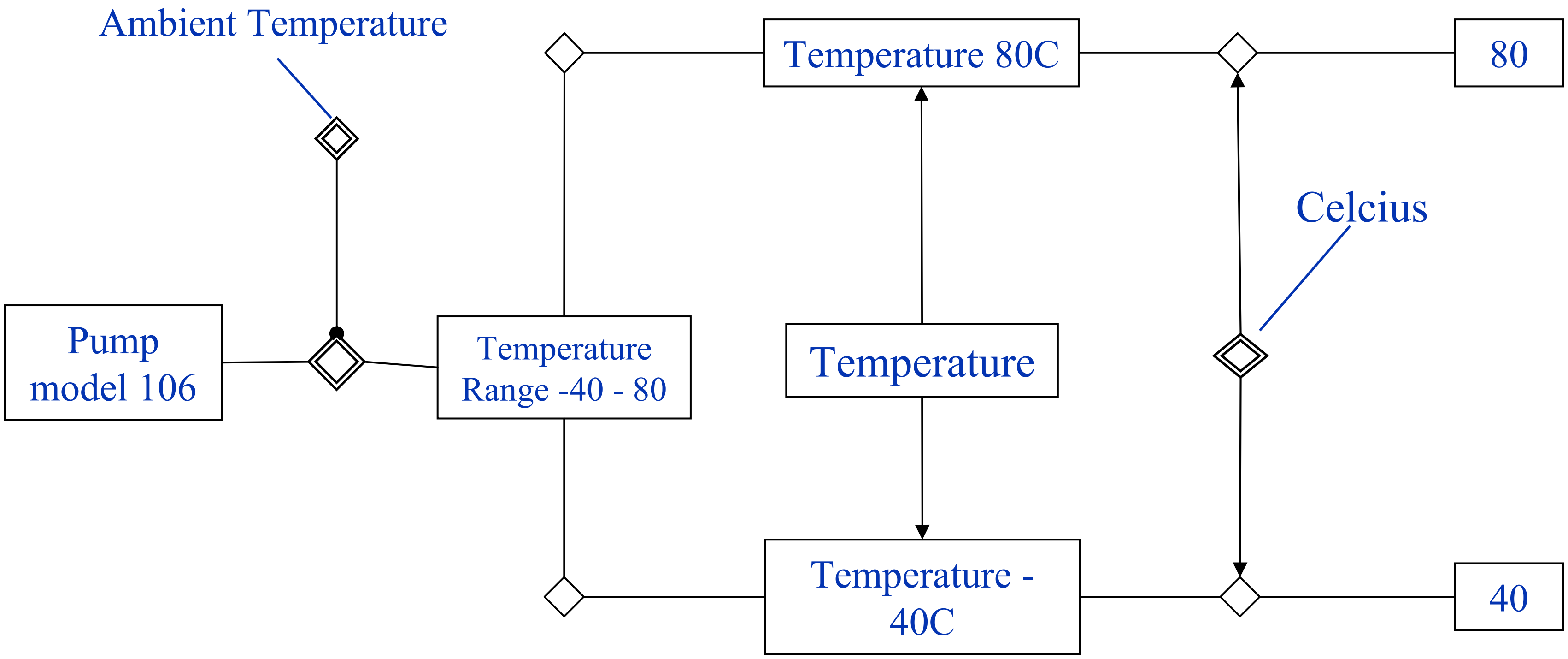
- Fundamental Concepts
- The EXPRESS Modeling Language
- The Basic Classes of 15926
- Types of Reference Data Classes
- **Templates**

MANAGING RISK

Templates

- Even simple statements become complex 15926 structures.
- ”3051CG should have an ambient temperature between -40 to 80 degrees Celsius”

Complex 15926 Structure



Ambient Temperature

Temperature 80C

80

Celcius

Pump model 106

Temperature Range -40 - 80

Temperature

Temperature - 40C

40

part

whole

CO Possessor

© Det Norske Veritas AS. All rights reserved

Template Signatures

- Template Signature

Roles	Class	Type of Restriction	Measurement scale	Upper Bound	Lower Bound
Entity Types	CO Individual	CO CO Relationship	Scale	Arithmetic Number	Arithmetic Number
Example	*3051CG Pressure Transmitter*	*Ambient Temperature*	*Celcius*	*80*	*-40*

End user challenge: Filling out the template may require explicit information that is only implicitly given by the end user’s documentation.

© Det Norske Veritas AS. All rights reserved