Descriptive Ontology for Linguistic and Cognitive Engineering ($DOLCE$)

The view, the system, the use (and other scattered thoughts)

S. Borgo & all LOA-ists

Laboratory for Applied Ontology (LOA)
CNR-ISTC, Trento (IT)

Oslo, 21 May 2025

University of Oslo

Schedule for today

Part 1	9:00-9:50	generalities on AO and *DOLCE*
	10 min break	
Part 2	10:00-10:45	bits of formality in *DOLCE*
	15 min coffee break	
Part 3	11:00-12:00	modeling with *DOLCE*
	1.15 min lunck break	
Part 4	13:15-15:15	hands-on modeling
	15 min coffee break	
Part 5	15:30-17:00	wash-up the models

Legenda

AO = Applied Ontology (a theoretical and methodological approach)

AO includes two intertwined steps:

(i) ontological analysis

(ii) ontology construction

F^2O = Foundational and Formal Ontology
(any logical theory on a way to understand reality obtained via AO)

AO and F^2O

[LOA view]

General observations /1 - AO at LOA

The purpose of AO is to make explicit how to **(*) understand reality**.
[Defending a specific stand about what is reality (per se) and even how to make sense of this question, is a secondary problem for AO.]

This **open and permissive** attitude has an important motivation: conceptual freedom is central in science as well as in everyday life.

Nonetheless, AO imposes important constraints on views and their F^2O development:

- the view has to be conceptually coherent
- the view has to be homogeneous across the entities
- the view has to be logically sound (no contextual truths)
- the view has to be logically consistent (no contradiction)
- (ideally) the F^2O that models the view has priority in conveying meaning over all other communication means

(*) = decide to

General observations /2 - Things AO is not

In particular, note that AO:

- is NOT a branch of Applied philosophy;
- is NOT a branch of Computer science (CS);
- is NOT a branch of Semantic web (SW).

It is interesting to observe that after 30 years of research in AO,

- there is no established relationship nor strong overlap btw IAOA and the Society for Applied Philosophy or between these communities.
- there is no established relationship btw the Applied Ontology Journal and the Journal of Applied Philosophy.

The relationship btw AO and CS/SW is stronger for historical and teleological reasons, their complementarity in terms of interests and goals is clear.

General observations /3 - Motivations

How did we get here?

- Immediate problem:
data and knowlege integration in the 1980s

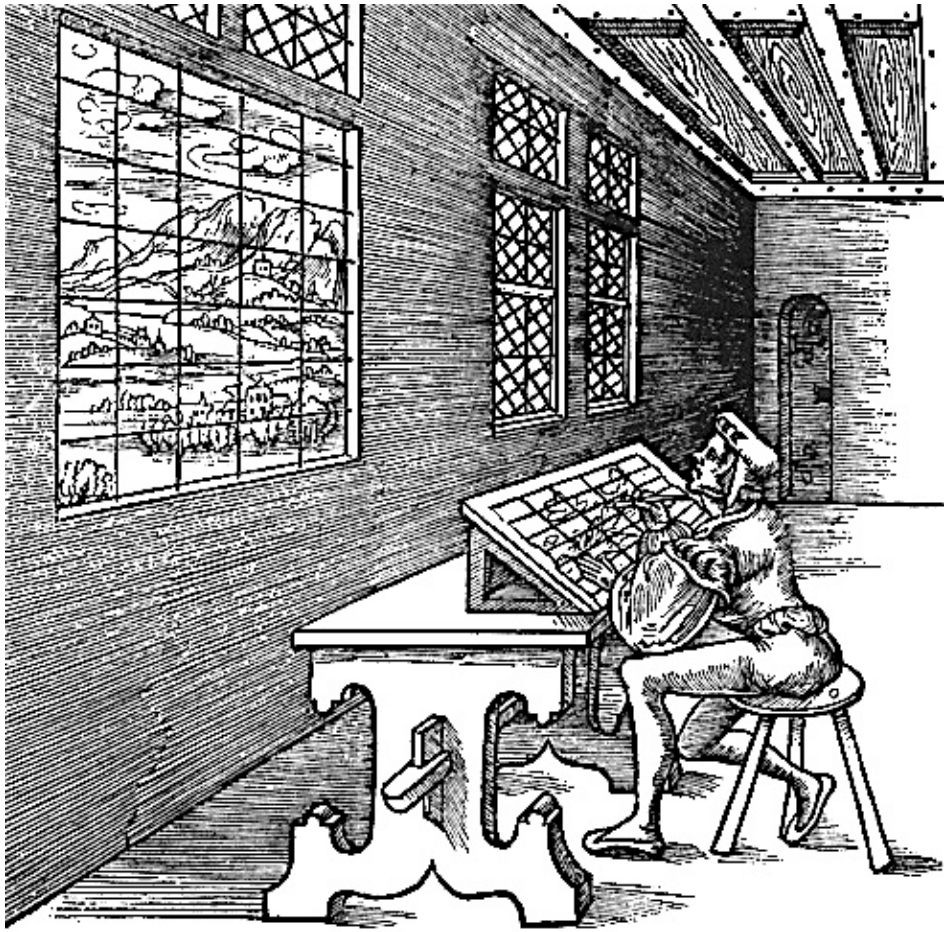
- Long term process:
formal approaches expanding from formal disciplines into cognitive and information sciences

To put it simply, AO reuses a successful pattern from (e.g.) mathematics and computer science, namely,
- state the assumptions you use,
- carefully clean your terminology,
- characterise the above in a formal language,
- and accept the consequences :-)

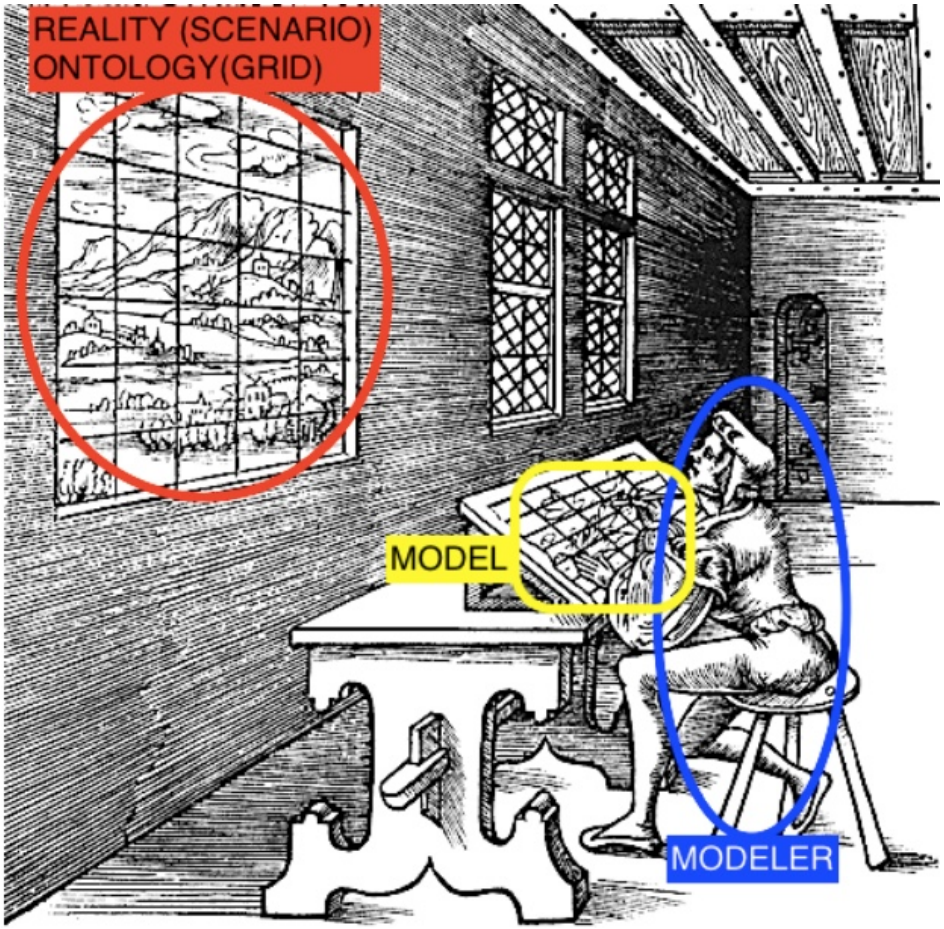
- in math one starts by declaring the properties that holds for operators and spaces one uses;
- in CS one (used to) starts by fixing a problem, a language and declaring the types of variables at disposal

General observations /4 - Just a tool?

The adoption of a F^2O is disruptive as F^2O is a **pervading tool**:



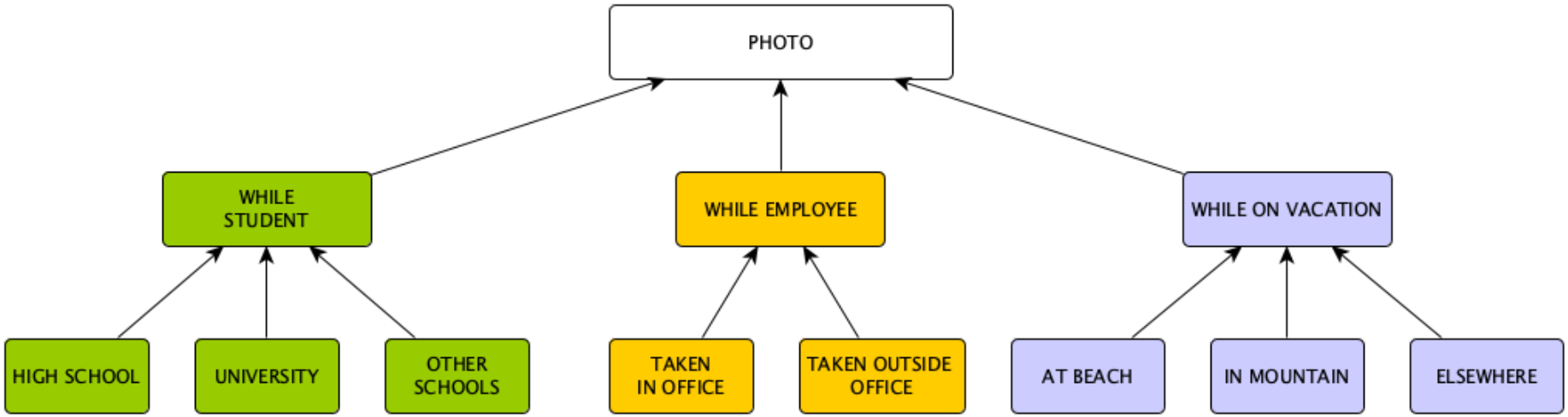
General observations /4.1 - Just a tool?



Use the ontology to clean your terminology /1

I want to organise some photos (it does not matter whether they are digital or printed photos).

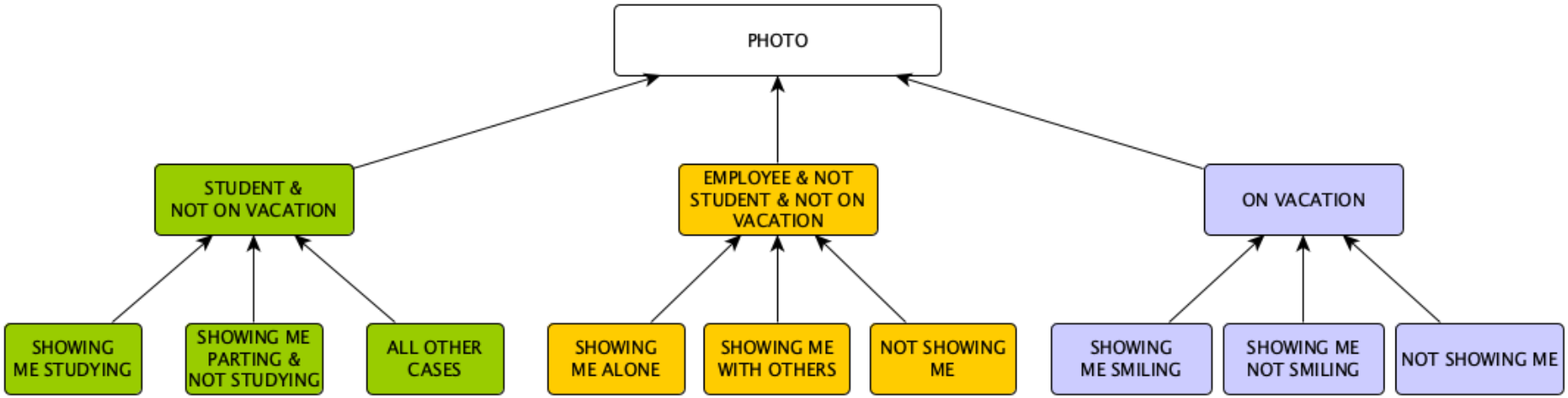
Here is the organization: there is a container "PHOTO" with three subsections, which are further refined.



Tell me why this organization is wrong.

Use the ontology to clean your terminology /2

Ok, let me try a different one.
Suppose that this time the photos are somehow only about me.

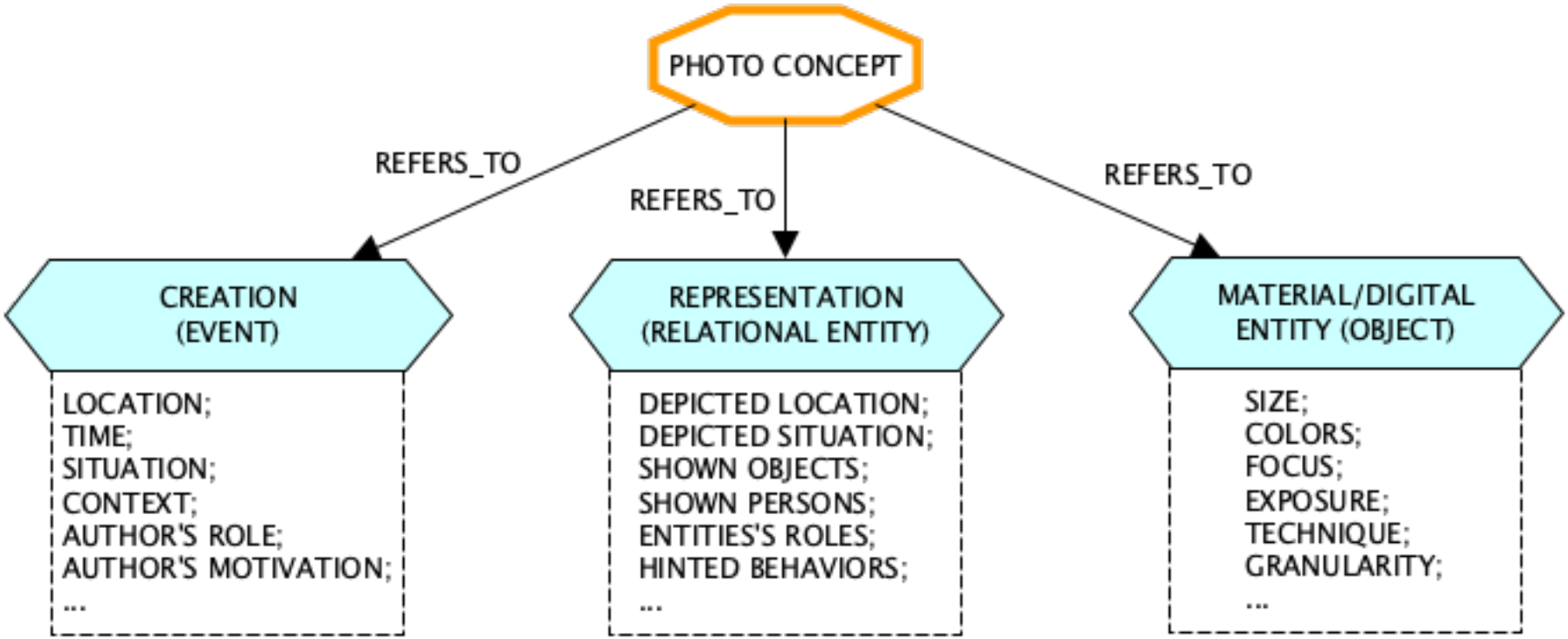


Is this classification better?
Why is this organization also wrong?

Use the ontology to clean your terminology /4

Let's turn to ontology.

Question: what do I refer to when I say "photo classification"?



DOLCE

DOLCE /1

DOLCE is a F^{2}O, its purpose is to formalise one of the views of reality that we use continuously.

The LOA group developed *DOLCE* with the attitude stated in slide 5: "conceptual freedom is central in science as well as in everyday life"

DOLCE embeds this view even in the ontology: it allows distinct modelling solutions.

It has been said (P. Hayes) that *DOLCE* is closer to an ontological framework than to a specific ontology.

Historically, it was the first F^{2}O released in first-order logic (FOL).

Soon after other F^{2}Os were axiomatised (and some are being developed today):
BFO, BORO, EMMO, GFO, GUM, TUPPER, UFO, YAMATO...

LOA has also invested in studying the alignment across F^{2}Os to ensure information sharing and interoperability across conceptualisations.

DOLCE /2 - “everything goes together”

- The physical layer:
  reality as an interconnected and evolving physical system;

DOLCE /2 - “everything goes together”

- The physical layer:
  reality as an interconnected and evolving physical system;
- The perspectival layer:
  reality as what determines an agent’s perception, understanding and acting (modulo the agent’s capabilities);

DOLCE /2 - "everything goes together"

- The physical layer:
  reality as an interconnected and evolving physical system;
- The perspectival layer:
  reality as what determines an agent's perception, understanding and acting (modulo the agent's capabilities);
- The social layer:
  reality as the combination of material, agentive and information entities that we call social systems.

It is already challenging to formalise a coherent and consistent theory for each of these layers.

DOLCE makes explicit a view of reality that (i) integrates the above layers, (ii) is fairly natural to human understanding, and (iii) is overall faithful to human practices.

Given the complexity of the goal, the development of *DOLCE* (and most F^{2}Os) relied on studies in philosophy, logic, linguistics, social studies as well as common sense.

The *DOLCE* ontology

- embraces a descriptive, non referentialist, metaphysics
  (No strong claim on the intrinsic nature of the world)
- models the mesoscopic and conceptual levels
  (following natural language, human cognition, social practices)
- is limited to the most general categories
- aims to be acceptable from different perspectives
- aims to be open to different perspectives
- adopts a structure that fosters interoperability

DOLCE /2.2 - *DOLCE*'s modeling toolkit

- There are different types of properties and they are formalised in different ways
- A notion of resemblance characterises (individual) qualities
- An individual quality can be evaluated in different kinds of spaces (discrete, continuous, topological, metrical...)
- Concepts are taken as individuals and have their own category

DOLCE /3 - The *DOLCE* history

In 2003, *DOLCE* was released by the European project "WonderWeb" as part of the Deliverable D18 (first version in 2002 as D17).
The axiomatisation was in first-order quantified modal logic QS5, including the Barcan and the converse Barcan formula (fixed domain).

DOLCE has been applied in many areas and applications
(by choice, LOA does not keep track of applications using *DOLCE*).
To answer applications' needs, a few "lite" versions in the Web Ontology Language (OWL) have been developed, among these: *DOLCE*-lite, *DOLCE*-ultralite, and *DOLCE*-zero.

In 2023, *DOLCE* has been included in standard ISO 21838 as part 3.
DOLCE is now written in Common Logic (CLIF) to comply with the ISO 21838 requirements.
As a byproduct of this process, LOA developed an official version of *DOLCE* in OWL: $DOLCE\,\mathrm{basic}_{OWL}$
LOA is also developing a richer one: $DOLCE\,\mathrm{naryRel}_{OWL}$.

DOLCE /4 - The *DOLCE* construction attitude

DOLCE, as a logical theory, has been enriched with several extensions to cover entities that were not considered top-level in 2003.
Technically, these are not parts of *DOLCE*.
They are modules that help to apply *DOLCE* coherently and consistently across domains.
E.g.: `roles, artefacts, functions, resources...`

How is LOA coping with openness in these extensions?
By choice, *DOLCE* extensions are built to be compatible with other ontologies as well. The goal remains to facilitate sharing and reuse.
To achieve this generality, the extensions list:
- the categories whose existence is needed, and
- the properties/relations on which the extension builds.

In these extensions, categories are minimally constrained (e.g., avoiding exploiting specific assumptions made by *DOLCE* or by other ontologies).

DOLCE /5 - The domain of *DOLCE*

DOLCE organises **particulars**, that is:
individuals that are directly or indirectly located in space or in time, or are abstract.

Properties and relations as universals are not discussed.
(Properties and relations are functional to define the ontology, are not included in the domain per se – *DOLCE* has its own ways to represent properties)

Regions and **spaces** are abstract entities.
(but see later how to contextualise them in time.)

Concepts (as well as mental and social entities) are temporal entities.

DOLCE /6 - The top categories of *DOLCE*

ENDURANT (ED)
Informally, the category of objects
Examples include: ANIMALS, MACHINES, ATOMS, GALAXIES, LAKES, WINDS, AMOUNTS OF MATTER, EDGES (of a material object), HOLES, THOUGHTS, ORGANISATIONS, GOODS and PRESIDENTS (role)

(INDIVIDUAL) QUALITY (Q)
Informally, the category of properties/apects of endurants or perdurants
Examples include: COLOR, WEIGHT (for material entities), AGE (for physical objects), DURATION, SPEED (for events), VALUE (for social objects).

PERDURANT (PD)
Informally, the category of events
Examples include: GIVING A LECTURE, BREAKING A GLASS, RUNNING TO A PARTY, PERFORMING, STOP FUNCTIONING, BEING WRONG (THE STATE OF), BEING STANDING, BREATHING and TRYING TO UNDERSTAND.

ABSTRACT (AB)
Informally, the category of abstract individuals
Examples include: NUMBERS, FACTS and SPACES (in the eternalist view).

NOTE: *DOLCE* does not enforce a specific list of qualities except for spatial location (for physical endurants) and temporal location (for events).

DOLCE /6.1 - Endurant vs. Perdurant

Endurants (objects):

- Need a time-indexed parthood relation
- Exist in time
- Can genuinely change in time
- May have non-essential parts
- Are fully given whenever they are present (wholly presence, no temporal parts)

Perdurants (events):

- Do not need a time-indexed parthood relation
- Happen in time
- Do not change in time (as a whole...)
- All parts are essential
- Only some proper parts might be present whenever they are present (partial presence, temporal parts)

DOLCE /7 - The core relations in *DOLCE*

- Constitution K
- Parthood (temporal, atemporal) P

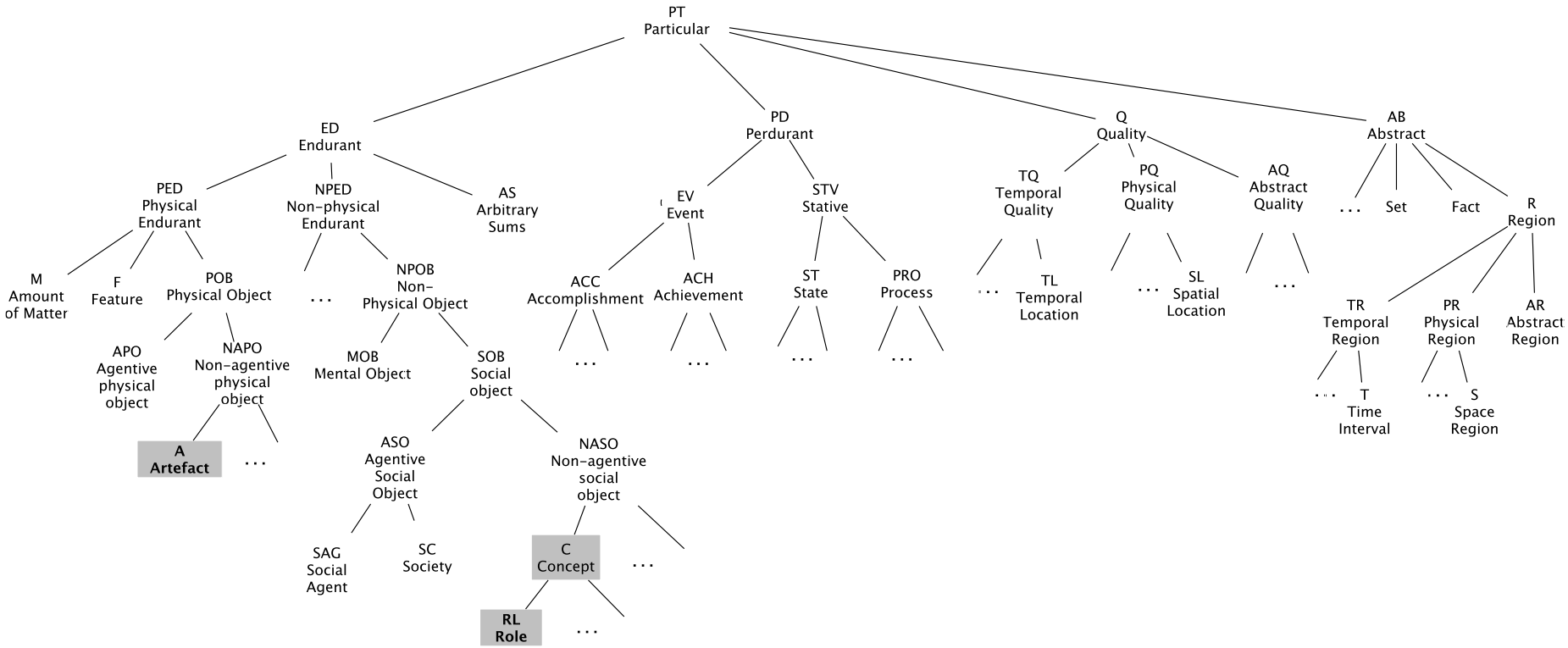
- Participation PC
- Being present at PRE

- Being quality of QT
- Being the quale of at QL

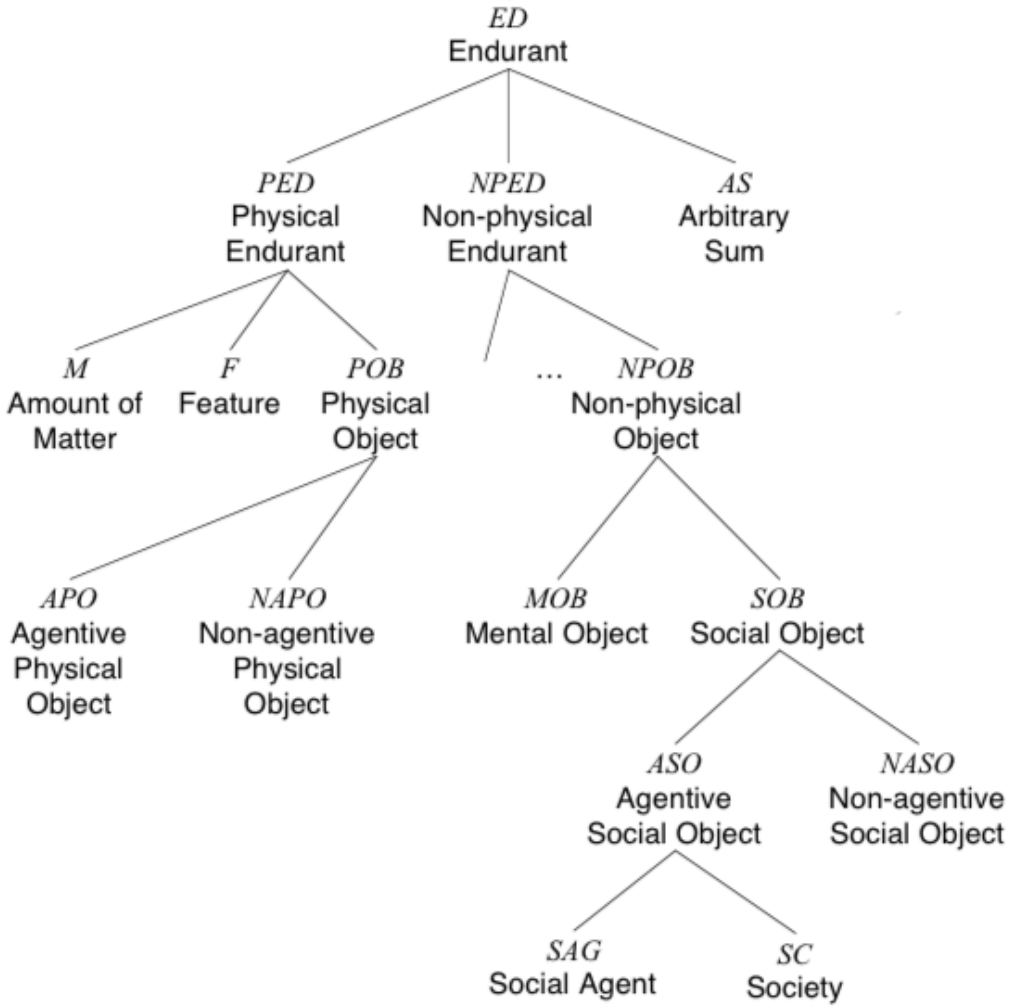
- Classification (being classified by at) CF
- Dependence (specific/generic, constant/partial, one-sided/mutual) D

other elements that structure the ontology are:
- subclass (is_a)
- membership (instance_of)
- OntoClean meta-properties

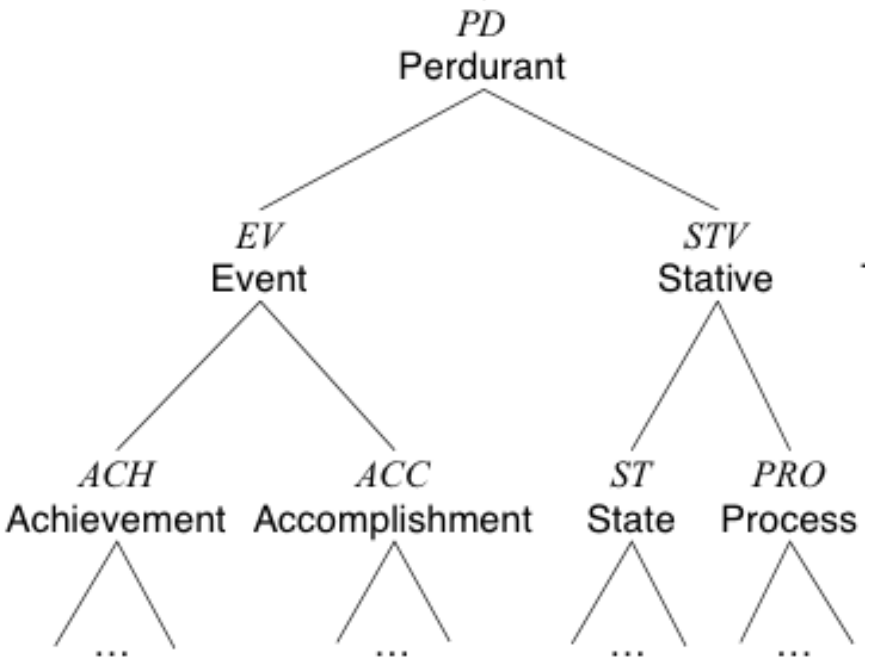
DOLCE /8 - The *DOLCE* taxonomy



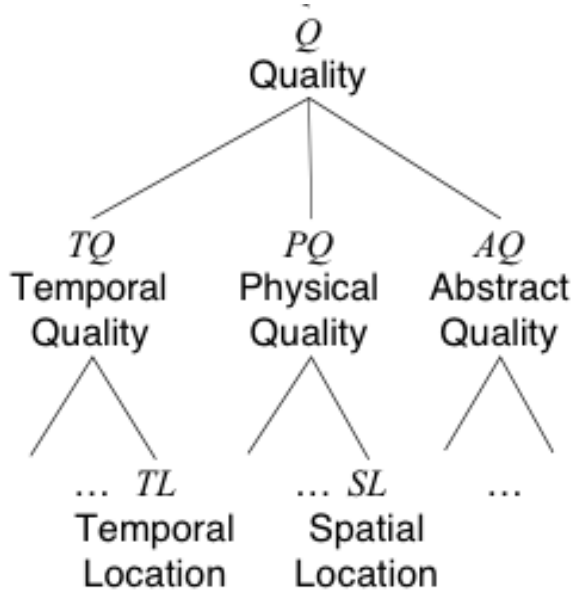
DOLCE /8.1 - The *DOLCE* taxonomy (ED)



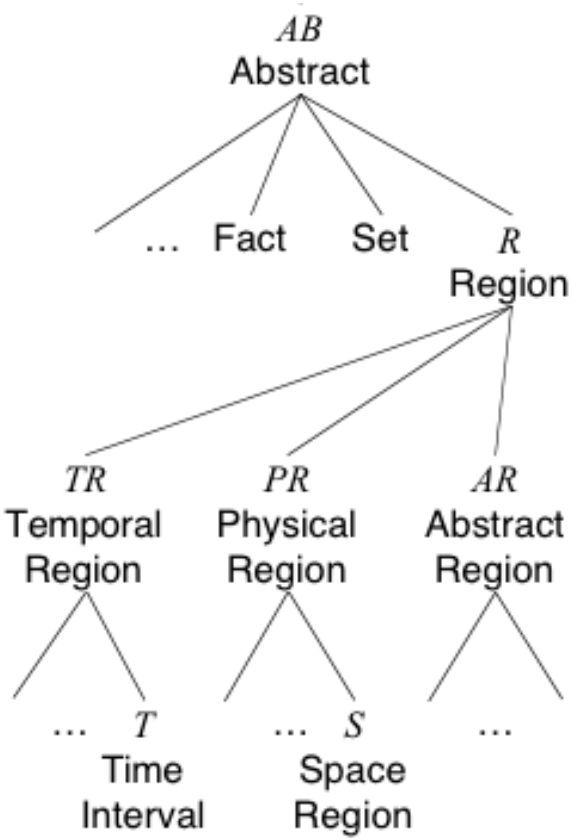
DOLCE /8.2 - The *DOLCE* taxonomy (PD)



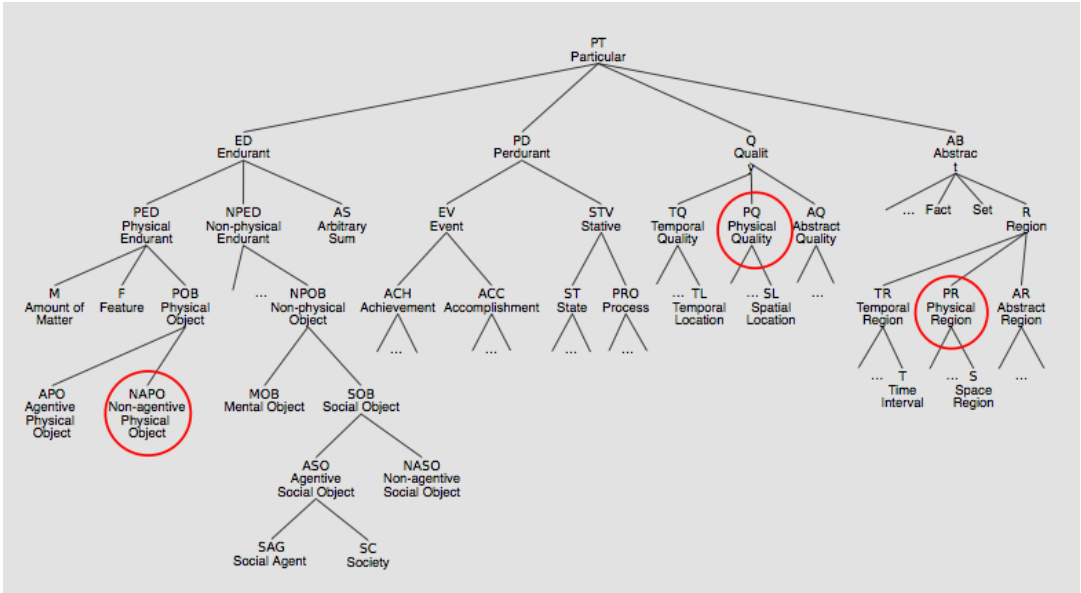
DOLCE /8.3 - The *DOLCE* taxonomy (Q)

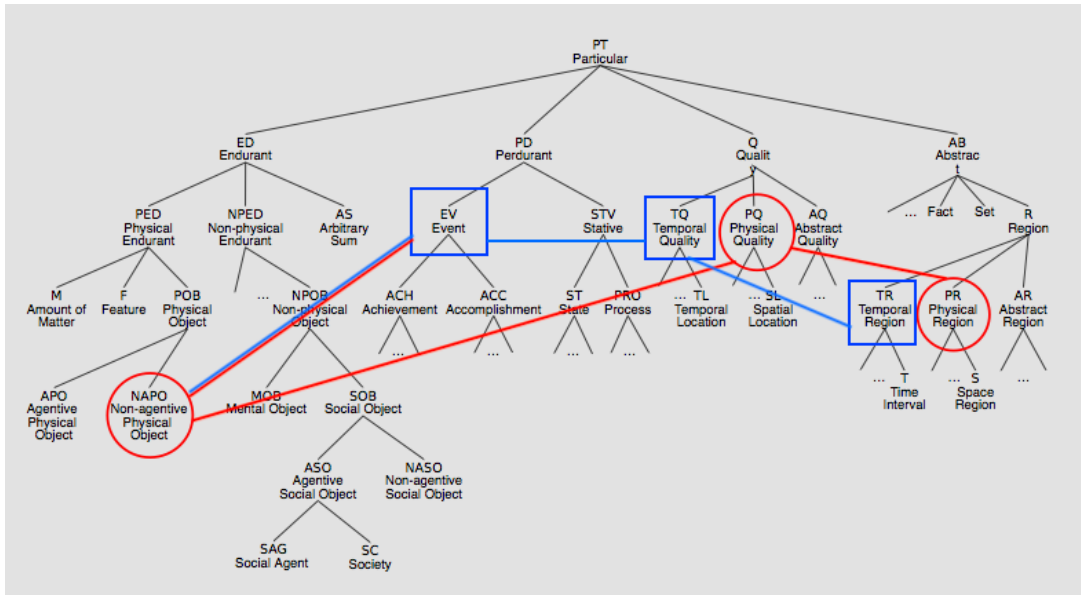


DOLCE /8.3 - The *DOLCE* taxonomy (AB)



DOLCE /9 - *DOLCE*'s 'everything goes together'





Bits of *DOLCE*'s axiomatisation

DOLCE - The core relations in *DOLCE*

- Constitution K
- Parthood (temporal, atemporal) P

- Participation PC
- Being present at PRE

- Being quality of QT
- Being the quale of at QL

- Classification (being classified by at) CF

DOLCE /1 - Constitution in DOLCE

The relationship $\mathsf{K}(x, y, t)$ stands for 'x constitutes y at time t'.

a1 $\mathsf{K}(x, y, t) \rightarrow ((\mathsf{ED}(x) \vee \mathsf{PD}(x)) \wedge (\mathsf{ED}(y) \vee \mathsf{PD}(y)) \wedge \mathtt{T}(t))$
(*Constitution typing*, cf. Ad20)

a2 $\mathsf{K}(x, y, t) \rightarrow (\mathsf{PED}(x) \leftrightarrow \mathsf{PED}(y))$ (cf. Ad21)

* $\mathsf{K}(x, y, t) \rightarrow (\mathsf{NPED}(x) \leftrightarrow \mathsf{NPED}(y))$ (cf. Ad22)

* $\mathsf{K}(x, y, t) \rightarrow (\mathsf{PD}(x) \leftrightarrow \mathsf{PD}(y))$ (cf. Ad23)

a3 $\mathsf{K}(x, y, t) \rightarrow \neg\mathsf{K}(y, x, t)$ (cf. Ad24)

* $\mathsf{K}(x, y, t) \wedge \mathsf{K}(y, z, t) \rightarrow \mathsf{K}(x, z, t)$ (cf. Ad25)

* $\mathsf{K}(x, y, t) \rightarrow \mathsf{PRE}(x, t) \wedge \mathsf{PRE}(y, t)$ (cf. Ad26)

* $\mathsf{K}(x, y, t) \rightarrow \forall t'(\mathsf{P}(t', t) \rightarrow \mathsf{K}(x, y, t'))$ (cf. Ad27)

DOLCE /2 - Parthood in DOLCE

Atemporal parthood: $\mathsf{P}(x, y)$ for 'x is part of y'
Temporal parthood: $\mathsf{P}(x, y, t)$ for 'x is part of y at time t'

Both satisfy the principles of Extensional Mereology, except for the antisymmetry axiom which holds for atemporal parthood only.

a4 $\mathsf{P}(x, y, t) \rightarrow \mathtt{ED}(x) \wedge \mathtt{ED}(y) \wedge \mathtt{T}(t)$ (*Temporary part typing*, cf. Ad10)

a5 $\mathsf{P}(x, y, t) \rightarrow \mathsf{PRE}(x, t) \wedge \mathsf{PRE}(y, t)$ (cf. Ad17)

d1 $\mathsf{PP}(x, y, t) \stackrel{def}{=} \mathsf{P}(x, y, t) \wedge \neg \mathsf{P}(y, x, t)$
(*Temporary proper part*, cf. Dd20)

d2 $\mathsf{O}(x, y, t) \stackrel{def}{=} \exists z(\mathsf{P}(z, x, t) \wedge \mathsf{P}(z, y, t))$ (*Temporary Overlap*, cf. Dd21)

d3 $x +_{te} y \stackrel{def}{=} \iota z \forall w, t(\mathsf{O}(w, z, t) \leftrightarrow (\mathsf{O}(w, x, t) \vee \mathsf{O}(w, y, t)))$
(*Temporary binary sum*, cf. Dd26)

(See D18 for atemporal parthood)

DOLCE - The core relations in *DOLCE*

- Constitution K
- Parthood (temporal, atemporal) P

- Participation PC
- Being present at PRE

- Being quality of QT
- Being the quale of at QL

- Classification (being classified by at) CF

DOLCE /3 - Participation in DOLCE

The participation (PC) relation connects endurants, perdurants, and times.
Here we write $\mathsf{PC}(x, y, t)$ for 'x participates in y at time t'.

a6 $\mathsf{PC}(x, y, t) \rightarrow \mathtt{ED}(x) \wedge \mathtt{PD}(y) \wedge \mathtt{T}(t)$ (*Participation typing*, cf. Ad33)

a7 $\mathtt{PD}(x) \wedge \mathsf{PRE}(x, t) \rightarrow \exists y(\mathsf{PC}(y, x, t))$ (cf. Ad34)

a8 $\mathtt{ED}(x) \rightarrow \exists y, t(\mathsf{PC}(x, y, t))$ (cf. Ad35)

a9 $\mathsf{PC}(x, y, t) \rightarrow \mathsf{PRE}(x, t) \wedge \mathsf{PRE}(y, t)$ (cf. Ad36)

a10 $\mathsf{PC}_{\mathsf{C}}(x, y) \stackrel{def}{=} \exists t(\mathsf{PRE}(y, t)) \wedge \forall t(\mathsf{PRE}(y, t) \rightarrow \mathsf{PC}(x, y, t))$
(*Const. Participation*, cf. Dd63)

DOLCE /4 - Being present in DOLCE

The primitive predicate 'being present at' (PRE) identifies at which times an entity exists.

d4 $\mathrm{PRE}(x,t) \stackrel{def}{=} \exists t'(\mathrm{ql}_T(t',x) \wedge \mathrm{P}(t,t'))$ (*being present at*)

t1 $\mathrm{PRE}(x,t) \rightarrow \mathtt{T}(t)$

t2 $\mathrm{PRE}(x,t) \wedge P(t',t) \rightarrow \mathrm{PRE}(x,t')$ (dissectivity)

a11 $\mathrm{PRE}(x,t') \wedge \mathrm{PRE}(x,t'') \wedge \mathrm{SUM}(t,t',t'') \rightarrow \mathrm{PRE}(x,t)$ (additivity)

No commitment to a specific notion of time is taken.

DOLCE - The core relations in *DOLCE*

- Constitution K
- Parthood (temporal, atemporal) P

- Participation PC
- Being present at PRE

- Being quality of QT
- Being the quale of at QL

- Classification (being classified by at) CF

DOLCE /5 - Qualities in DOLCE

$\mathsf{qt}(x, y)$ stands for 'x is a quality of y',
$\mathsf{SBL}_X(Q, \phi)$ stands for 'ϕ is a leaf category in the quality hierarchy".

d5 $\mathsf{qt}(\phi, x, y) \stackrel{def}{=} \mathsf{qt}(x, y) \wedge \phi(x) \wedge \mathsf{SBL}_X(Q, \phi)$

(*Quality of type* ϕ, cf. Dd29)

In *DOLCE* the temporal quale (relation ql) is the position occupied by an individual quality within a quality space (TL is the temporal location category):

d6 $\mathsf{ql}_{T,PD}(t, x) \stackrel{def}{=} \mathsf{PD}(x) \wedge \exists z(\mathsf{qt}(TL, z, x) \wedge \mathsf{ql}(t, z))$

(*Temporal quale of perdurants*, cf. Dd30)

d7 $\mathsf{ql}_{T,ED}(t, x) \stackrel{def}{=} \mathsf{ED}(x) \wedge t = \sigma t'(\exists y(\mathsf{PC}(x, y, t'))$

(*Temporal quale of endurants*, cf. Dd31)

d8 $\mathsf{ql}_T(t, x) \stackrel{def}{=} \mathsf{ql}_{T,ED}(t, x) \vee \mathsf{ql}_{T,PD}(t, x) \vee \mathsf{ql}_{T,Q}(t, x)$

(*Temporal Quale*, cf. Dd35)

Qualities are physical, temporal, or abstract.
A quality inheres in one and only one entity.

a12 $\mathsf{PQ}(x) \rightarrow \exists!y(\mathsf{qt}(x,y) \wedge \mathsf{PED}(x))$ (*Physical quality*, cf. Ad47)

a13 $\mathsf{TQ}(x) \rightarrow \exists!y(\mathsf{qt}(x,y) \wedge \mathsf{PD}(x))$ (*Temporal quality*, cf. Ad46)

a14 $\mathsf{AQ}(x) \rightarrow \exists!y(\mathsf{qt}(x,y) \wedge \mathsf{NPED}(x))$ (*Abstract quality*, cf. Ad48)

DOLCE - The core relations in *DOLCE*

- Constitution K
- Parthood (temporal, atemporal) P

- Participation PC
- Being present at PRE

- Being quality of QT
- Being the quale of at QL

- Classification (being classified by at) CF

DOLCE /6 - Classification in DOLCE

The relationship $\mathrm{CF}(x, y, t)$ stands for 'at the time t, x is classified by the concept y', from (Masolo et al., 2004) [=M4].
Relation CF applies to concepts (category C) among which are roles (category RL).

a15 $\mathrm{CF}(x, y, t) \rightarrow \mathrm{ED}(x) \wedge \mathrm{C}(y) \wedge \mathbb{T}(t)$ (cf. A11 in M4)

a16 $\mathrm{CF}(x, y, t) \rightarrow \mathrm{PRE}(x, t)$ (cf. A12 in M4)

a17 $\mathrm{CF}(x, y, t) \rightarrow \neg\mathrm{CF}(y, x, t)$ (cf. A14 in M4)

a18 $\mathrm{CF}(x, y, t) \wedge \mathrm{CF}(y, z, t) \rightarrow \neg\mathrm{CF}(x, z, t)$ (cf. A15 in M4)

d9 $\mathrm{AR}(x) \stackrel{def}{=} \forall y, t(\mathrm{CF}(x, y, t) \rightarrow \exists t'(\mathrm{PRE}(x, t') \wedge \neg\mathrm{CF}(x, y, t'))$
(cf. D1 in M4)

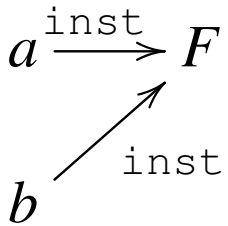
DOLCE /7 - Quality/property pattern in DOLCE

We may mean several things when we say *"it has a property"* and *"they have the same property."*

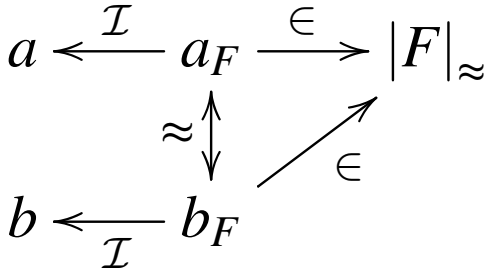
In one sense, in FOL property corresponds to a class or a predicate.

For modeling flexibility, *DOLCE* provides different representations for "individuals a and b share the property F":

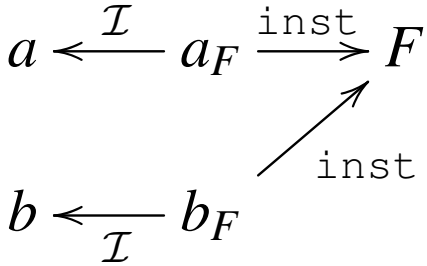
Universalism



Trope theory

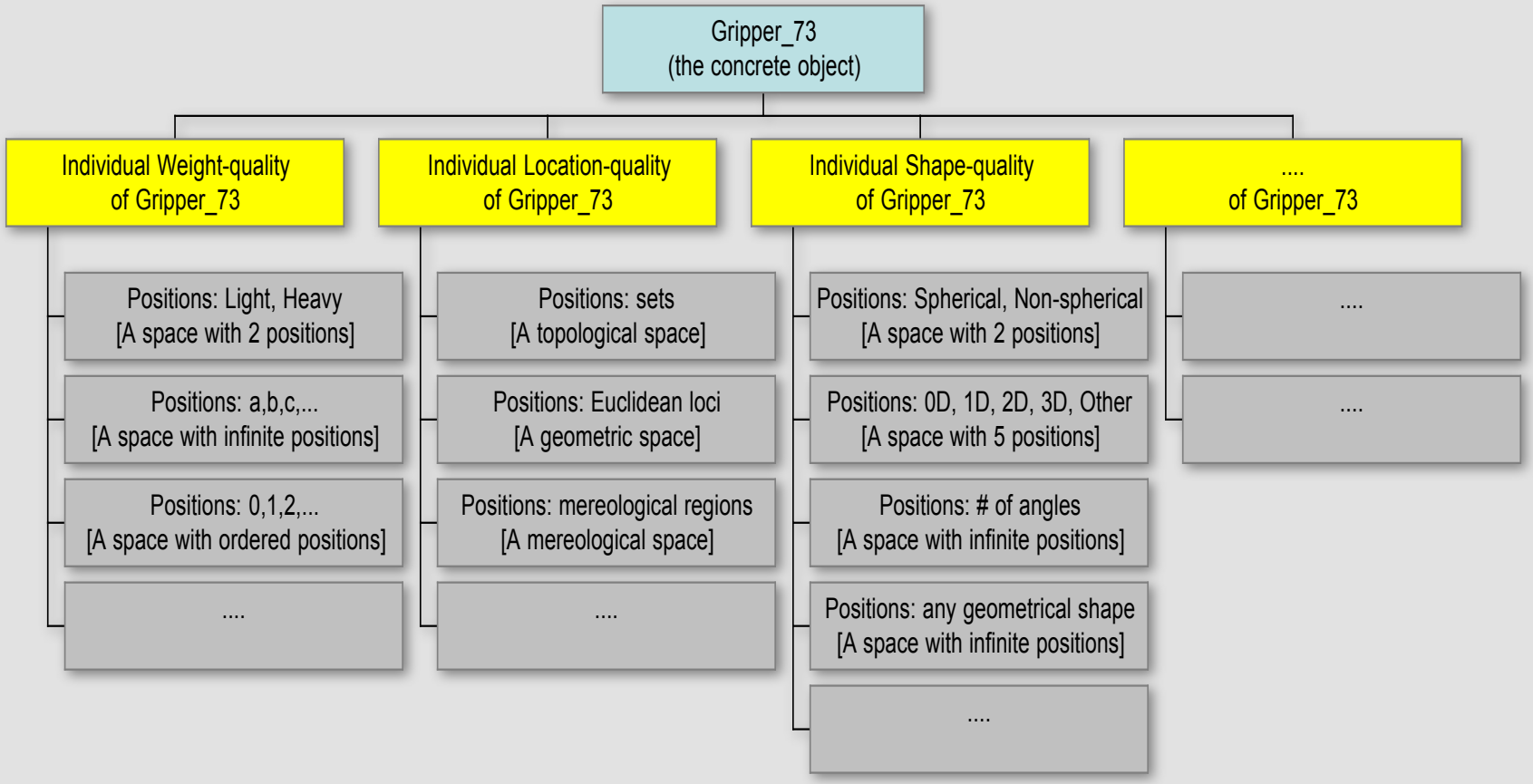


Universals+Tropes

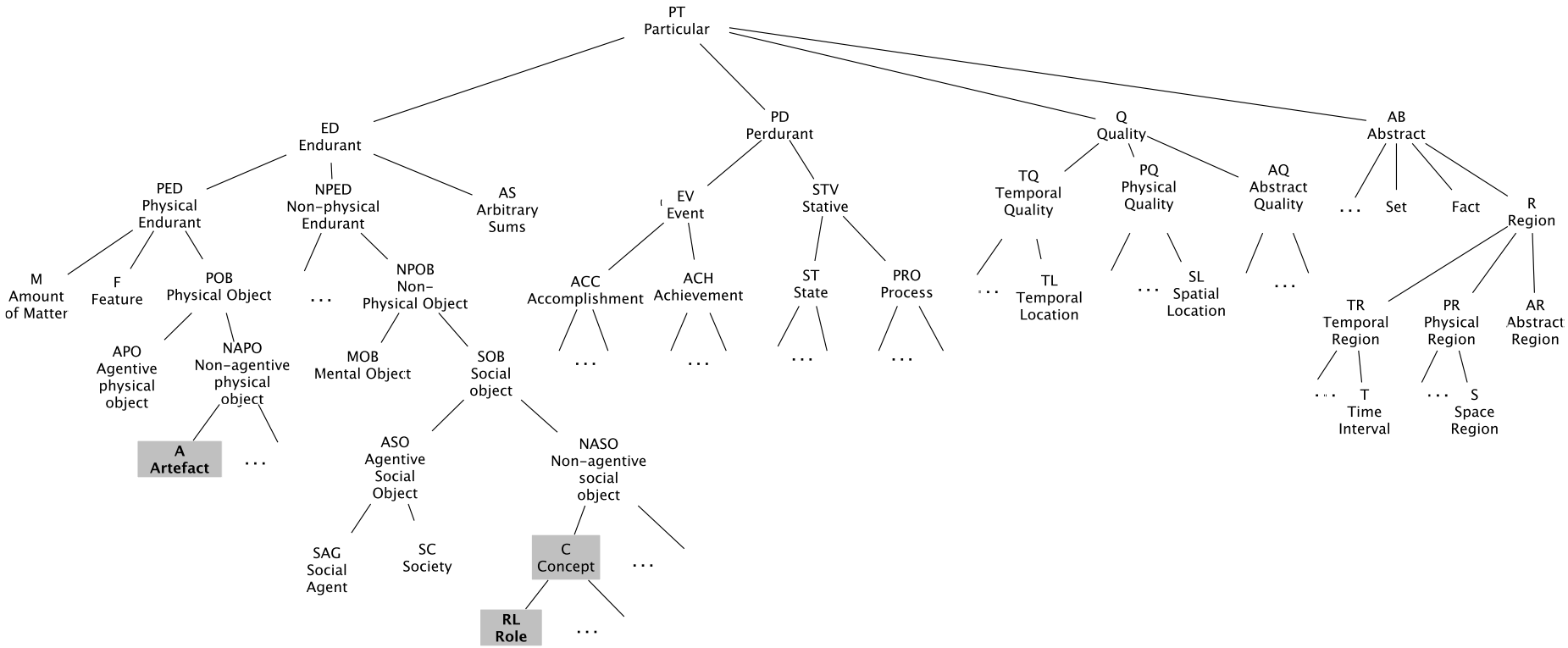


DOLCE /8 - Quality and quale (example)

An object, its qualities (yellow), and their quality-spaces (gray)



DOLCE /8 - The *DOLCE* taxonomy



Everything goes together

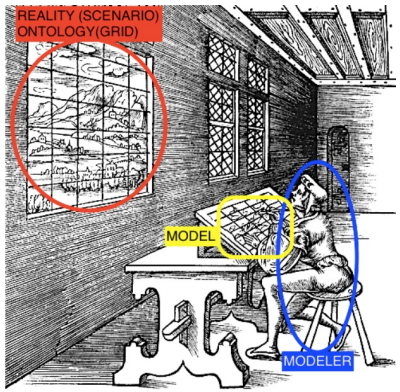
DOLCE /1 - Building a model

We are given this diorama and asked to use it to specialise/populate *DOLCE*

What should we do?

DOLCE /2 - Building a model: starting questions

We start from general questions suggested by (the structure and vision of) *DOLCE*:



- what endurants are there?
- to which categories do they belong?
- what qualities do they have?
- what relationships hold?

[the diorama is static, we do not investigate perdurants nor abstract (if not indirectly)]

PLAYER VIEW

- list of endurants
  house, person, ball, tree, horse, fence, window, door, roof, tree's pick, tree's trunk, person's head, person's legs, lawn . . .
- list of individual qualities and their spaces
  color [red, white, green . . .], shape [cone, sphere, prism . . .] volume, location
- list of relationships
  being part of, being connected to, being next to, being above/below, being in front/behind, being supported by, being friend of, being playing, being the owner of . . .

PRODUCER VIEW

- list of endurants
  Block 1, Block 2 . . .
  Assembled block 1, Assembled block 2,
- list of qualities/capabilities
  color, shape, volume, size, location, material, turnable, assemblable above/below/sidewise . . .
- list of relationships
  being touching, being attached, being above/below, being in front/behind . . .

DOLCE /5 - Building a SINGLE model

The vision behind the AO requires that

everything goes together

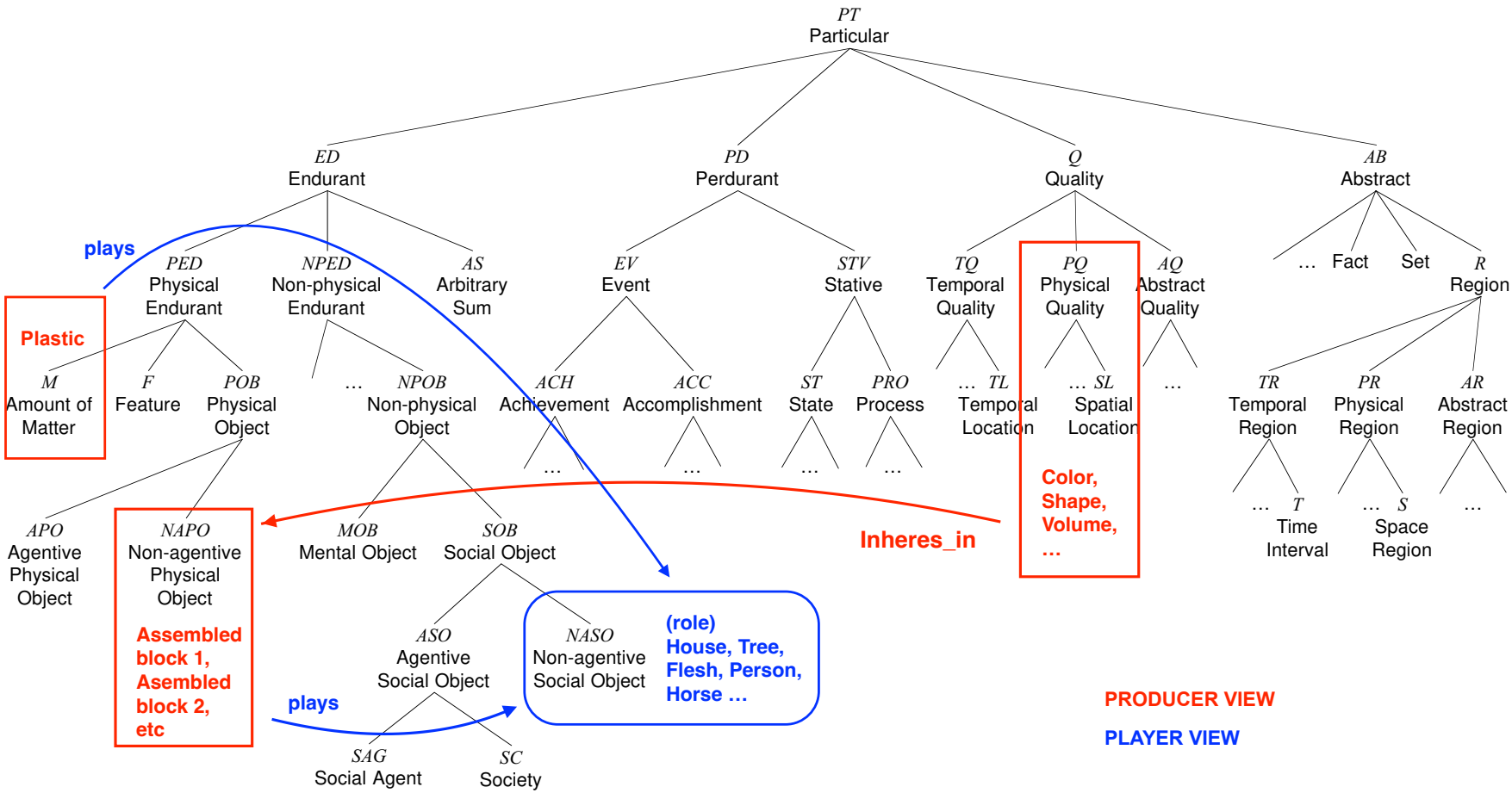
Informally, we know/believe this is possible.

Is it possible in *DOLCE*? If so, how?

For the logically oriented ontologists in the room:

Under which conditions is it possible to integrate different views of the 'same' reality?

DOLCE /6 - The unified model



Engineering functions

Engineering - general perspective

- structure, behaviour, function, and purpose
- structure, behaviour, and function
- structure and behaviour
- structure and function
- behaviour and function
- behaviour
- function

Motivation
we need to understand functions to develop information systems that support functional representation and reasoning!

Semantic incoherence of engineering models of artefacts

1. Behaviour is clearly defined as how something acts in response to its environment, while structure is the organization or arrangement of the constituents of an object. The differentiation between function and behaviour is achieved by considering results versus how the results are achieved. Thus, structure is what is, **behaviour how does**, function what does and purpose why does or what for. [. . .] The interaction between an object's existence and the natural environment causes the object to act in a certain manner, i.e. its behaviour. [. . .] Function is the result of the behaviour, i.e. as its product or effect, so that function is closely related to behaviour, the latter being the mechanism by which results are achieved. [Rosenman and Gero, 1998]
2. Structure is what the device is, and function is what the device is for, but **behaviour is what the device does**. [Kleer, 1984]
3. In general, function is what a design is going to do, while **behaviour is how a design will do it**.Thus, we present the following understanding of function: The function of a design system is its purpose and intention in some context, and is often expressed as functional requirements and restrained by functional constraints. [Zhang et al., 2001]

Semantic incoherence of engineering models of artefacts

1. Behaviour is clearly defined as how something acts in response to its environment, while structure is the organization or arrangement of the constituents of an object. The differentiation between function and behaviour is achieved by considering results versus how the results are achieved. Thus, structure is what is, behaviour how does, **function what does** and purpose why does or what for. [. . .] The interaction between an object's existence and the natural environment causes the object to act in a certain manner, i.e. its behaviour. [. . .] **Function is the result of the behaviour**, i.e. as its product or effect, so that function is closely related to behaviour, the latter being the mechanism by which results are achieved. [Rosenman and Gero, 1998]
2. Structure is what the device is, and **function is what the device is for**, but behaviour is what the device does. [Kleer, 1984]
3. In general, **function is what a design is going to do**, while behaviour is how a design will do it.Thus, we present the following understanding of function: The function of a design system is its purpose and intention in some context, and is often expressed as functional requirements and restrained by functional constraints. [Zhang et al., 2001]

Semantic incoherence of engineering models of artefacts

1. Behaviour is clearly defined as how something acts in response to its environment, while structure is the organization or arrangement of the constituents of an object. The differentiation between function and behaviour is achieved by considering results versus how the results are achieved. Thus, structure is what is, behaviour how does, function what does and purpose why does or what for. [. . .] The interaction between an object's existence and the natural environment causes the object to act in a certain manner, i.e. its behaviour. [. . .] Function is the result of the behaviour, i.e. as its product or effect, so that function is closely related to behaviour, the latter being the mechanism by which results are achieved. [Rosenman and Gero, 1998]
2. **Structure is what the device is**, and function is what the device is for, but behaviour is what the device does. [Kleer, 1984]
3. In general, function is what a design is going to do, while behaviour is how a design will do it.Thus, we present the following understanding of function: The function of a design system is its purpose and intention in some context, and is often expressed as functional requirements and restrained by functional constraints. [Zhang et al., 2001]

Semantic incoherence of engineering models of artefacts

1. Behaviour is clearly defined as how something acts in response to its environment, while structure is the organization or arrangement of the constituents of an object. The differentiation between function and behaviour is achieved by considering results versus how the results are achieved. Thus, structure is what is, behaviour how does, function what does and **purpose** **why does or what for**. [. . .] The interaction between an object's existence and the natural environment causes the object to act in a certain manner, i.e. its behaviour. [. . .] Function is the result of the behaviour, i.e. as its product or effect, so that function is closely related to behaviour, the latter being the mechanism by which results are achieved. [Rosenman and Gero, 1998]
2. Structure is what the device is, and function is what the device is for, but behaviour is what the device does. [Kleer, 1984]
3. In general, function is what a design is going to do, while behaviour is how a design will do it.Thus, we present the following understanding of function: **The function of a design system is its purpose and intention in some context**, and is often expressed as functional requirements and restrained by functional constraints. [Zhang et al., 2001]

Semantic incoherence of engineering models of artefacts

1. Behaviour is clearly defined as how something acts in response to its environment, while structure is the organization or arrangement of the constituents of an object. The differentiation between function and behaviour is achieved by considering results versus how the results are achieved. Thus, structure is what is, behaviour how does, function what does and purpose why does or what for. [. . .] The interaction between an object's existence and the natural environment causes the object to act in a certain manner, i.e. its behaviour. [. . .] Function is the result of the behaviour, i.e. as its product or effect, so that function is closely related to behaviour, the latter being the mechanism by which results are achieved. [Rosenman and Gero, 1998]
2. Structure is what the device is, and function is what the device is for, but behaviour is what the device does. [Kleer, 1984]
3. In general, function is what a design is going to do, while behaviour is how a design will do it.Thus, we present the following understanding of function: The function of a design system is its purpose and intention in some context, and is often expressed as functional requirements and restrained by functional constraints. [Zhang et al., 2001]
4. behaviour = how does

Semantic incoherence of engineering models of artefacts

1. Behaviour is clearly defined as how something acts in response to its environment, while structure is the organization or arrangement of the constituents of an object. The differentiation between function and behaviour is achieved by considering results versus how the results are achieved. Thus, structure is what is, behaviour how does, function what does and purpose why does or what for. [. . .] The interaction between an object's existence and the natural environment causes the object to act in a certain manner, i.e. its behaviour. [. . .] Function is the result of the behaviour, i.e. as its product or effect, so that function is closely related to behaviour, the latter being the mechanism by which results are achieved. [Rosenman and Gero, 1998]
2. Structure is what the device is, and function is what the device is for, but behaviour is what the device does. [Kleer, 1984]
3. In general, function is what a design is going to do, while behaviour is how a design will do it.Thus, we present the following understanding of function: The function of a design system is its purpose and intention in some context, and is often expressed as functional requirements and restrained by functional constraints. [Zhang et al., 2001]
4. behaviour = what does

Semantic incoherence of engineering models of artefacts

1. Behaviour is clearly defined as how something acts in response to its environment, while structure is the organization or arrangement of the constituents of an object. The differentiation between function and behaviour is achieved by considering results versus how the results are achieved. Thus, structure is what is, behaviour how does, function what does and purpose why does or what for. [. . .] The interaction between an object's existence and the natural environment causes the object to act in a certain manner, i.e. its behaviour. [. . .] Function is the result of the behaviour, i.e. as its product or effect, so that function is closely related to behaviour, the latter being the mechanism by which results are achieved. [Rosenman and Gero, 1998]
2. Structure is what the device is, and function is what the device is for, but behaviour is what the device does. [Kleer, 1984]
3. In general, function is what a design is going to do, while behaviour is how a design will do it.Thus, we present the following understanding of function: The function of a design system is its purpose and intention in some context, and is often expressed as functional requirements and restrained by functional constraints. [Zhang et al., 2001]

5. function = what does

Semantic incoherence of engineering models of artefacts

1. Behaviour is clearly defined as how something acts in response to its environment, while structure is the organization or arrangement of the constituents of an object. The differentiation between function and behaviour is achieved by considering results versus how the results are achieved. Thus, structure is what is, behaviour how does, function what does and purpose why does or what for. [. . .] The interaction between an object's existence and the natural environment causes the object to act in a certain manner, i.e. its behaviour. [. . .] Function is the result of the behaviour, i.e. as its product or effect, so that function is closely related to behaviour, the latter being the mechanism by which results are achieved. [Rosenman and Gero, 1998]
2. Structure is what the device is, and function is what the device is for, but behaviour is what the device does. [Kleer, 1984]
3. In general, function is what a design is going to do, while behaviour is how a design will do it.Thus, we present the following understanding of function: The function of a design system is its purpose and intention in some context, and is often expressed as functional requirements and restrained by functional constraints. [Zhang et al., 2001]

5. function = what for

Semantic incoherence of engineering models of artefacts II

1. [. . .] a function is defined as an instantiation of a part of a behaviour or of a complete behaviour.[Alberts, 1994]
2. Function is an abstraction of behaviour. Therefore, we propose to model function as a qualitative relationship between the input and the output ports of a device.[Kumar and Upadhyaya, 1998]
3. **The function of a design system is its purpose and intention in some context**, and is often expressed as functional requirements and restrained by functional constraints. **Function characterizes the abstracts of behaviour**. [Zhang et al., 2001]

Semantic incoherence of engineering models of artefacts II

1. [. . .] a function is defined as an **instantiation** of a part of a behaviour or of a complete behaviour.[Alberts, 1994]
2. Function is an **abstraction** of behaviour. Therefore, we propose to model function as a qualitative relationship between the input and the output ports of a device.[Kumar and Upadhyaya, 1998]
3. **The function of a design system is its purpose and intention in some context**, and is often expressed as functional requirements and restrained by functional constraints. **Function characterizes the abstracts of behaviour**. [Zhang et al., 2001]

Semantic incoherence of engineering models of artefacts II

1. [. . .] a function is defined as an instantiation of a part of a behaviour or of a complete behaviour.[Alberts, 1994]
2. Function is an abstraction of behaviour. Therefore, we propose to model function as a qualitative relationship between the input and the output ports of a device.[Kumar and Upadhyaya, 1998]
3. **The function of a design system is its purpose and intention in some context**, and is often expressed as functional requirements and restrained by functional constraints. **Function characterizes the abstracts of behaviour**. [Zhang et al., 2001]
4. function abstracts behaviour?

Semantic incoherence of engineering models of artefacts II

1. [. . .] a function is defined as an instantiation of a part of a behaviour or of a complete behaviour.[Alberts, 1994]
2. Function is an abstraction of behaviour. Therefore, we propose to model function as a qualitative relationship between the input and the output ports of a device.[Kumar and Upadhyaya, 1998]
3. **The function of a design system is its purpose and intention in some context**, and is often expressed as functional requirements and restrained by functional constraints. **Function characterizes the abstracts of behaviour**. [Zhang et al., 2001]
4. behaviour abstracts function?

Are these incoherences real?

The recent situation in engineering industry requires effective sharing of product knowledge among engineers for engineering activities such as designing and manufacturing during the product life-cycle. It is important to share not only product data but also designer's intention so-called design rationale of the product. One of the reasons of this difficulty is lack of semantic constraint for functional knowledge. Without guideline or restriction, **functional representation tends to be ad hoc**, specific to the target product, and hence not reusable. Although much research has been conducted on the representation of functionality in Artificial Intelligence, engineering design and Value Engineering, **there is no common definition of the concept of function itself** and semantic constraints are not enough for deriving effective guidelines. [Kitamura et al., 2005]

but beamers beam, planes fly, and . . . - how come?

Why do we need ontologies in engineering - let the engineers speak

- [Chandrasekaran et al., 1999]
  1. clarification of knowledge bases' vocabularies
  2. support for knowledge sharing
- [Mizoguchi, 2003]
  1. common vocabulary
  2. data structure
  3. presupposition mining
  4. increase in semantic interoperability
  5. theory of content
  6. **explication of design rationale**

Engineering functions - core theories

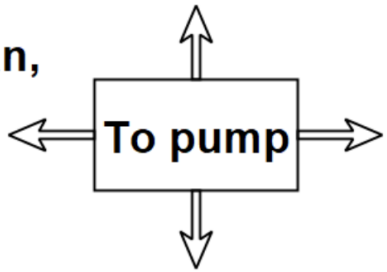
Three notions of function in engineering literature

1. **Functions as behaviors**
2. **Functions as operations on flows**
3. **Functions as roles**

Engineering functions: the source of confusion

WHAT HAPPENS?
To pump as a perdurant-type:
Ontological function, behavior
to transform electrical energy
into fluid motion

HOW DOES IT HAPPEN?
To pump as an implementation plan,
means(/way-of-achievements
/methods):
Is it done by means of a
gear pump or a cylinder pump?



WHY DOES IT HAPPEN?
To pump as a system role,
systemic function:
which is the role of
the pump in this system?

WHAT CAN HAPPEN?
To pump as a quality:
the pumping capability

Artefacts

Three definitions of artefact

Definition (Ontological Artefact)

A (technical) artefact $\mathfrak{a}$ is a physical object which a (group of) agent creates by two, possibly concurrent, intentional acts: the selection of a material entity, as the only constituent of $\mathfrak{a}$, and the attribution to $\mathfrak{a}$ of a (technical) quality or capacity.

Definition (Engineering Artefact)

A technical artefact $\mathfrak{a}$ is a physical object created by an intentionally performed production process. The process is intentionally performed by one or more agents with the goal of producing the object $\mathfrak{a}$ which is expected to realize intended behavior in some given generic technical situation.

Definition (Technological Artefact)

A technical artefact $\mathfrak{a}$ is a physical entity created by the carrying out by a (group of) agent of a make plan for a physical entity with a physical description id.

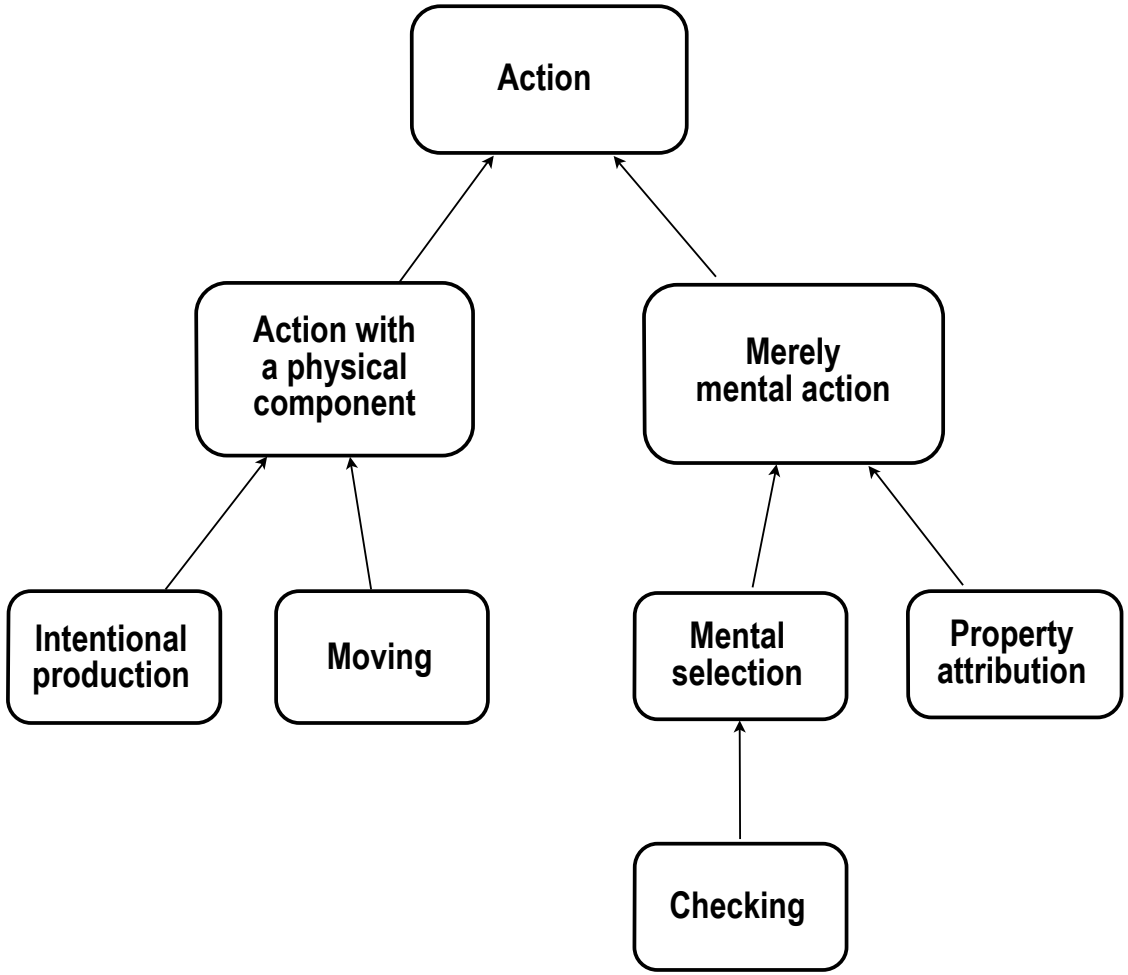


Figure: Types of actions for an artefact theory

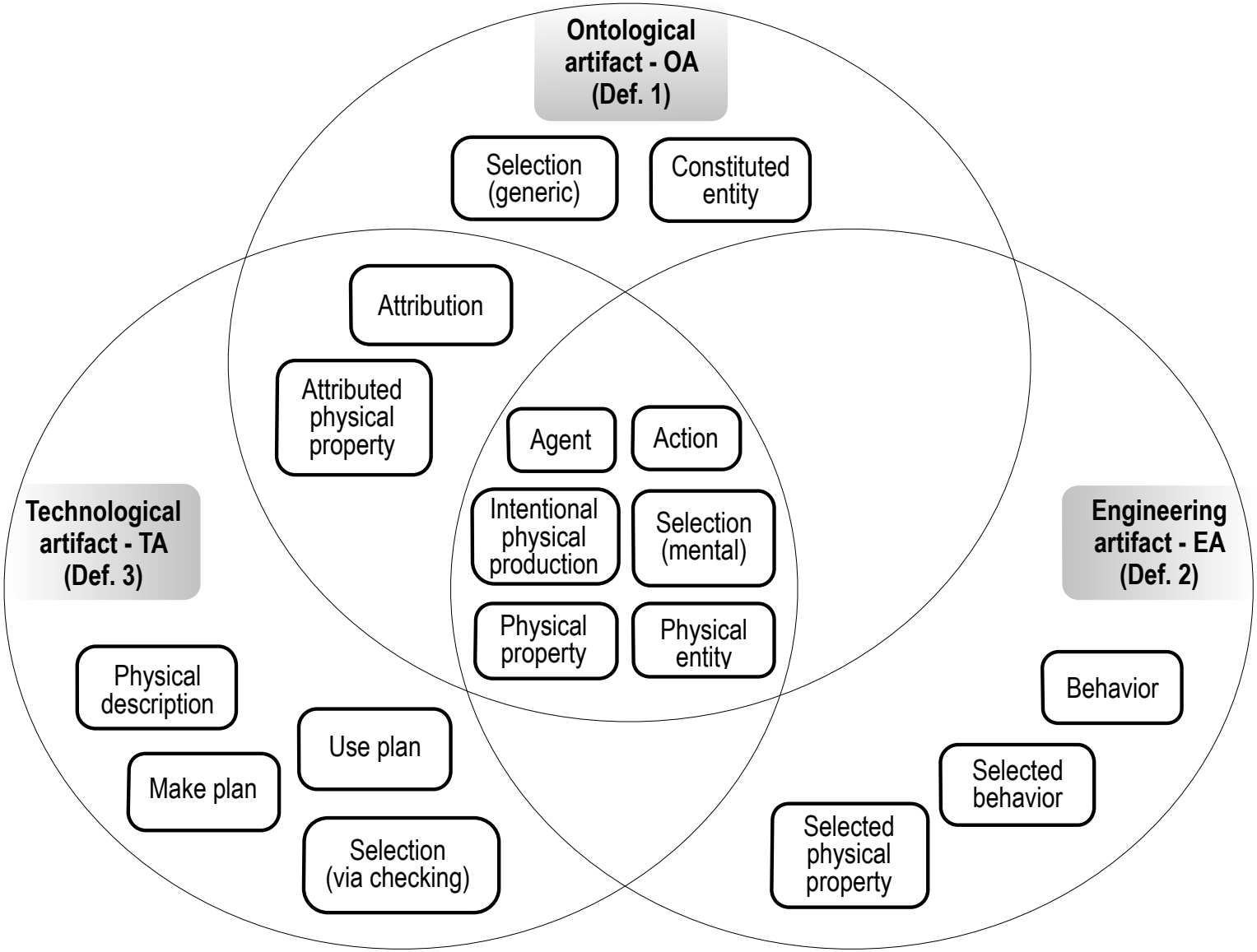


Figure: Ontological concepts for artefact definition

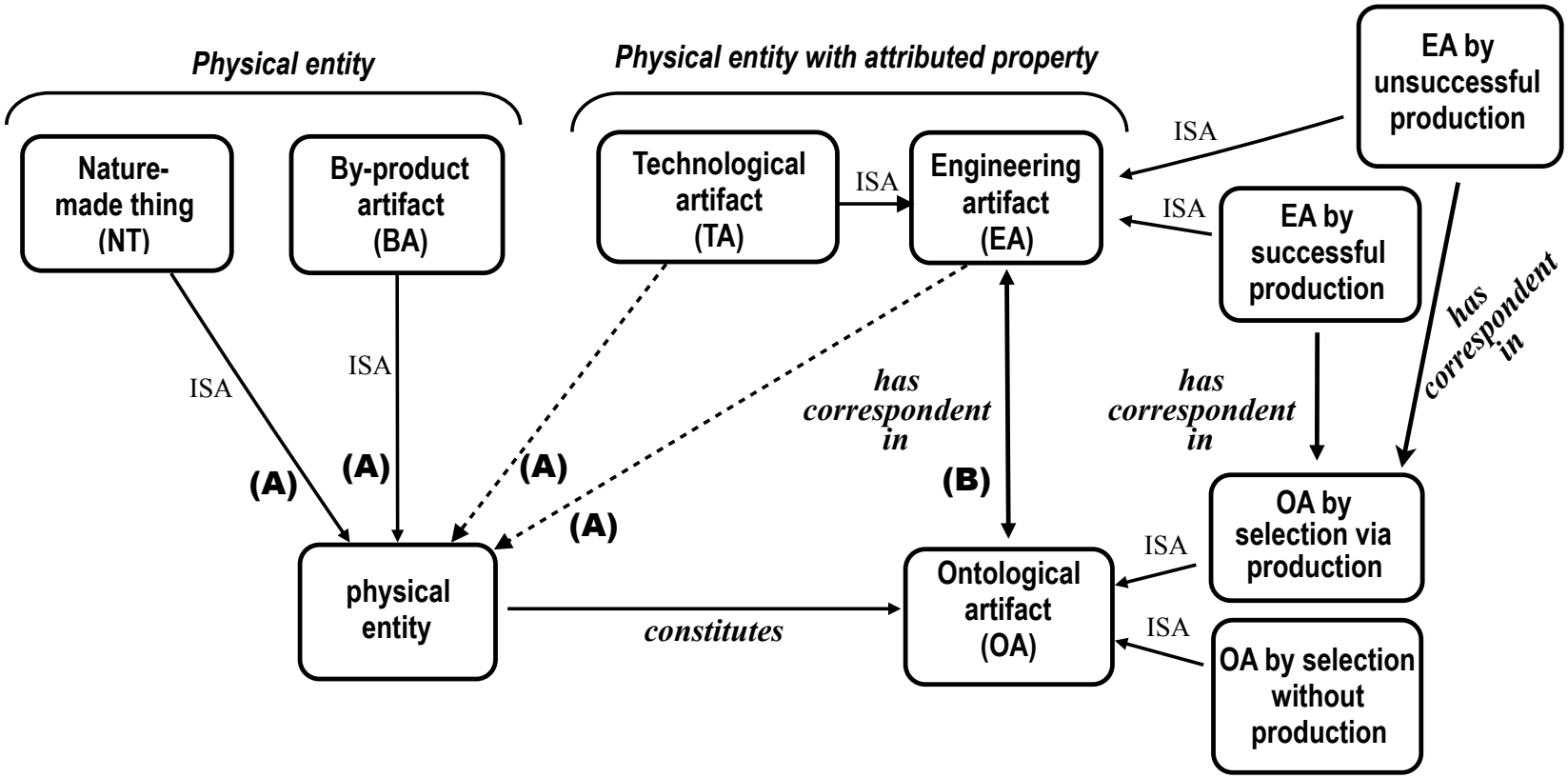


Figure: Relations among types of artefacts.

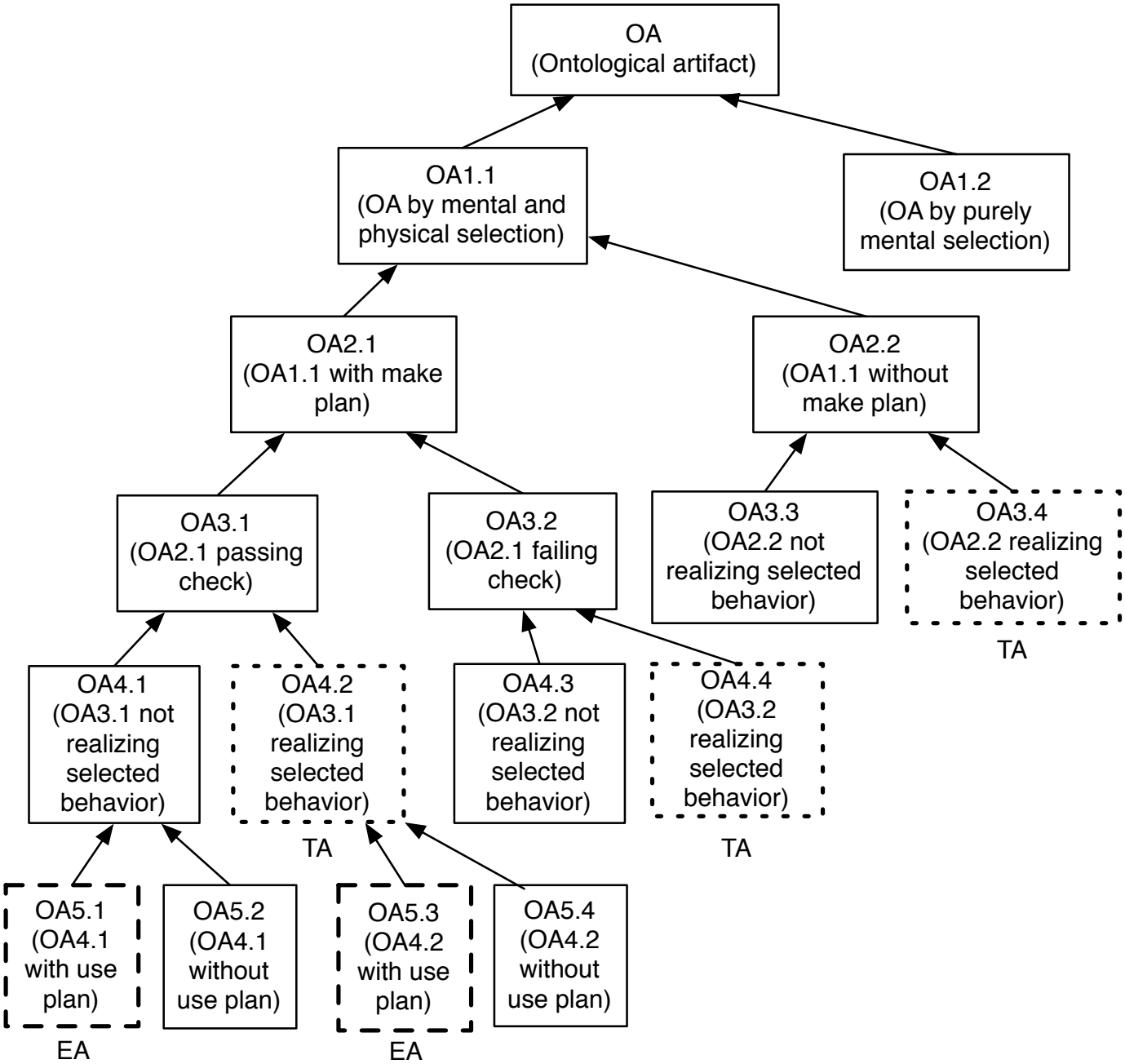


Figure: Artefact categories

Exercise: CASE A

- **CASE A**: An airplane is designed to fly at a maximum velocity of 600 mph but after repeated attempts to achieve that maximum, is only able to achieve at most 585 mph.

- **CASE A (simplified)**: An airplane is designed to fly at a maximum velocity of 600 mph but after repeated flights, is only able to achieve at most 585 mph.

Exercise: CASE A (simplified)

CASE A (simplified): An airplane is designed to fly at a maximum velocity of 600 mph but after repeated flights, is only able to achieve at most 585 mph.

Suggested vocabulary:

- Physical Object (POB)
- Artefact (A)
- Concept (C)
- Time (T)
- Quality_of (QT)
- Quale_of_at (QL)

Exercise: CASE B

CASE B: The IAOA contains a sub-organisation called Executive Council (EC), responsible for electing the IAOA President and regulating its powers. At time t1, no president has been elected yet. At time t2, the EC elects Laure as president. At time t3, the powers of the IAOA President are extended: they now have the additional power to nominate a Vice President.

Suggested vocabulary:

- Social Agent (SAG)
- Concept (C)
- Role (RL)
- Agentive Ph Object (APO)
- Person (subclass of APO)
- Time (T)
- Classified_by (CF)
  (CF(x, y, t) = x is classified by concept y at time t)
- Description (DS)
  (DS(x) = x is a description; DS is a subclass of NAPO)
- Defined_by (DF) (DF(x, y) = concept x is defined by description y]

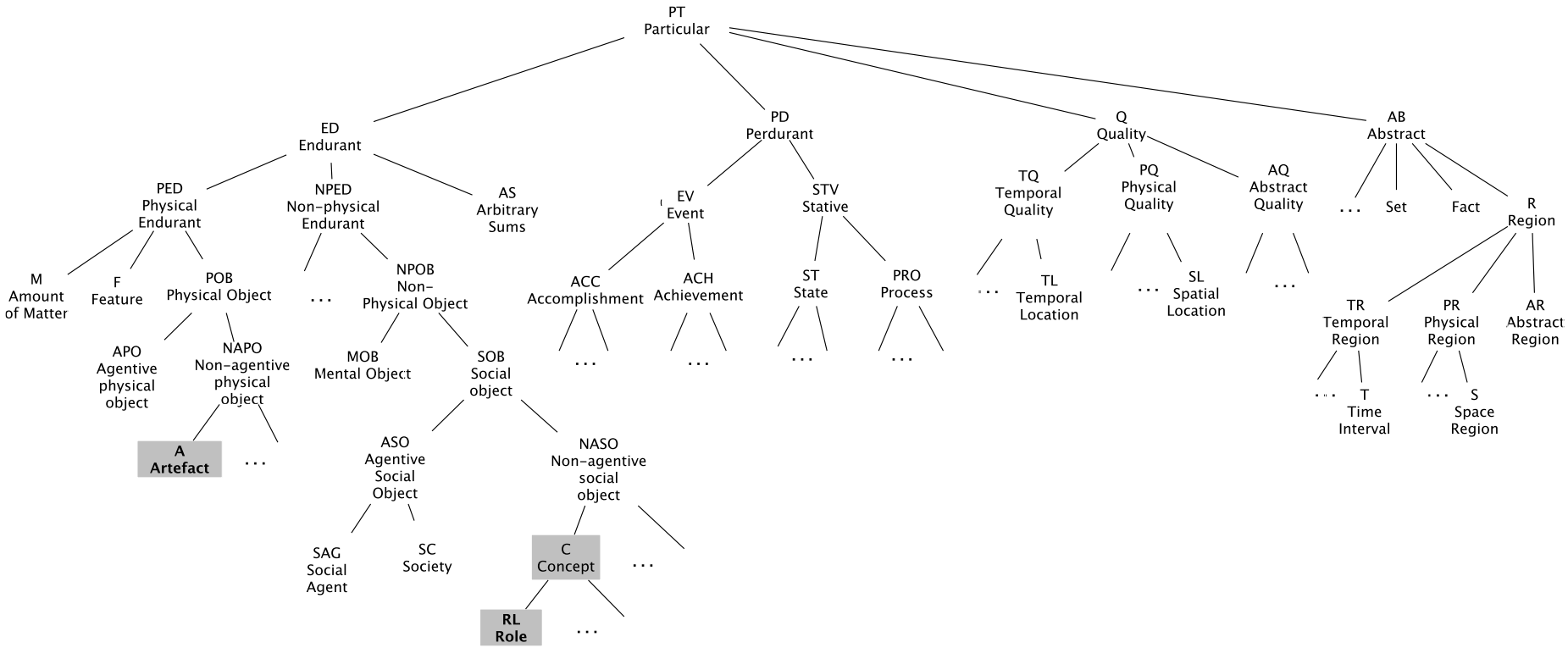
Exercise: CASE C

CASE C: There is a lump of clay that, at time t1, is used to make an aesthetically valuable statue. At time t2, the statue is destroyed. At time t3, the same lump of clay is reshaped to make a different statue. This statue is aesthetically valuable too.

Suggested vocabulary:

- Accomplishment (ACC)
- Non-Agentive Ph. Object (NAPO)
- Amount of Matter (M)
- Time (T)
- Being present (PRE)
- Constituted_by (K)
- Participation (PC)
- Depends_on (D)
- Quality_of (QT)
- Quale_of_at (QL)
- Society (SC)

The *DOLCE* taxonomy



End?

Exercise: CASE A (simplified)

CASE A (simplified): An airplane is designed to fly at a maximum velocity of 600 mph but after repeated flights, is only able to achieve at most 585 mph.

Formalisation

Artefact(AirplaneA)
[it follows that: PhysicalObject(AirplaneA)]
PhysicalQuality(fly-capability) (qt of a Ph. Obj.)
PhysicalQuality(attr-fly-capability) (qt of an Artefact)
TemporalQuality(max-speed) (qt of PD)
TemporalQuality(attr-max-speed) (qt of PD)
Time(T1) (time of the statement)

Let e be the life of AirplaneA, and e_i its flights before time T1.
Let t_e and t_i be the temporal extensions of e and of e_i, risp.ly.
Qt(fly-capability,AirplaneA)
Ql(yes,fly-capability,e_i)
Qt(attr-fly-capability,AirplaneA)
Ql(yes, attr-fly-capability,e)
Qt(max-speed$_i$, e_i)
Ql(585mph,max-speed$_i$)
Qt(attr-max-speed,e)
Ql(600mph,attr-max-speed)

Exercise: CASE A (simplified)

CASE A (simplified): An airplane is designed to fly at a maximum velocity of 600 mph but after repeated flights, is only able to achieve at most 585 mph.

Formalisation

Artefact(AirplaneA)
[it follows that: PhysicalObject(AirplaneA)]

PhysicalQuality(fly-capability)	(qt of a Ph. Obj.)
PhysicalQuality(attr-fly-capability)	(qt of an Artefact)
TemporalQuality(max-speed)	(qt of PD)
TemporalQuality(attr-max-speed)	(qt of PD)
Time(T1)	(time of the statement)

Let e be the life of AirplaneA, and e_i its flights before time T1.
Let t_e and t_i be the temporal extensions of e and of e_i, risp.ly.
Qt(fly-capability,AirplaneA)
Ql(yes,fly-capability,e_i)
Qt(attr-fly-capability,AirplaneA)
Ql(yes, attr-fly-capability,e)
Qt(max-speed$_i$, e_i)
Ql(585mph,max-speed$_i$)
Qt(attr-max-speed,e)
Ql(600mph,attr-max-speed)

Note 1: To model "attempts to" add execution of a plan (description of activities, resources, constraints, goals and intentional execution)
Note 2: For subtler modelling, distinguish the artefact AirplaneA and the ph. obj. AirplObj connecting them via constitution K(AirplObj,AirplaneA,t).

Exercise: CASE B

CASE B: The IAOA contains a sub-organisation called Executive Council (EC), responsible for electing the IAOA President and regulating its powers. At time t1, no president has been elected yet. At time t2, the EC elects Laure as president. At time t3, the powers of the IAOA President are extended: they now have the additional power to nominate a Vice President.

Formalisation

Role(IAOApresident)
DF(IAOApresident,IAOAPresidentDescription)
DS(IAOAPresidentDescription) (NAPO)
P(IAOAPresidentDescription-A,IAOAPresidentDescription,t1)
$\neg$ P(IAOAPresidentDescription-B,IAOAPresidentDescription,t1)
P(IAOAPresidentDescription-A,IAOAPresidentDescription,t2)
$\neg$ P(IAOAPresidentDescription-B,IAOAPresidentDescription,t2)
P(IAOAPresidentDescription-B,IAOAPresidentDescription,t3)
Person(Laure)
Time(t1) $\wedge$ Time(t2) $\wedge$ Time(t3) $\wedge$ t1<t2<13
$\forall x$ $\neg$CF(x,IAOApresident,t1) (at t1 there is no president)
CF(Laure,IAOApresident,t2) (at t2 Laure is president)
CF(Laure,IAOApresident,t3) (at t3 Laure is president)

Exercise: CASE C

CASE C: There is a lump of clay that, at time t1, is used to make an aesthetically valuable statue. At time t2, the statue is destroyed. At time t3, the same lump of clay is reshaped to make a different statue. This statue is aesthetically valuable too.

Formalisation

ACC(creation1)
ql$_T$(t1,creation1)
ACC(destruction)
ql$_T$(t2,destruction)
ACC(creation2)
ql$_T$(t3,creation2)
NAPO(statue1)
NAPO(statue2)
M(clay)
Time(t1) $\wedge$ Time(t2) $\wedge$ Time(t3)
t1<t2<t3
K(clay,statue1,t1)
K(clay,statue2,t3)

PC(statue1,creation1,t1)
PC(statue1,destruction,t2)
PC(statue2,creation,t3)
qt(aestheticValue1,statue1)
ql(V1,aestheticValue1,t) $\wedge$ V1>0
qt(aestheticValue2,statue2)
ql(V2,aestheticValue2,t) $\wedge$ V2>0
SC(societyA)
PRE(societyA,t1) $\wedge$ PRE(societyA,t2) $\wedge$ PRE(societyA,t3)
depends-on(aestheticValue1,societyA)
depends-on(aestheticValue2,societyA)