5. An Ontological Framework for Describing the Philosophical World

5.1 Introduction

In this chapter we describe an ontology aimed at the representation of the philosophical world. We show how the ontology builds upon a number of other well-known formal models and, in particular, we focus the analysis on the extensions we created whenever new *philosophy-related* representations were needed.

The chapter is organized as follows: section 2 discusses the most salient features of the philosophical domain with the purpose of translating them into precise ontological requirements; section 3 gives a thorough overview of the ontology by going through the main branches of its top-layer; section 4 deals with the part of the ontology devoted to the representation of *conceptual entities*; section 5 focuses on the analysis of the types of conceptual entities representing *philosophical ideas*; finally, section 6 provides a summary of this chapter's results and introduces the following chapter.

5.2 Philosophy as a domain to represent

In order to identify an initial set of ontology requirements, we used various *informal* knowledge acquisition techniques. Mainly, they consisted of discussions with domain experts, analyses of the implicit curricula formalized in philosophical textbooks, consultation of traditional encyclopedias and online philosophy directories[2]. In general, we recognized the following distinctive features in the philosophical domain:

1) it is very **vast**, both in its *historical* dimension (which comprises at least 2000 years of events related to the lives of thinkers, and to their intellectual productions) and in its *theoretical* dimension (that is, with respect to the various existing philosophical viewpoints, and to the relations they can entertain with other philosophical ideas);

2) it is **dynamic** and **slippery**: there is no general agreement on what are the common denominators in the philosophical work (i.e., problems, approaches or methods?), if not at a very abstract level. Practitioners recognize some major areas of interest, but tend to disagree on the details. Moreover, it is quite a normal practice for a "respectable" philosophy to redefine its own research questions and motivate them in an independent way. We can call this the *recursive* feature of a philosophy, by which a conception, in its radical definition of the world, needs also to define its sense and position within the world (thus, the problems and situations that justify its existence);

[2] A good overview of knowledge acquisition techniques available can be found at http://www.epistemics.co.uk/Notes/63-0-0.htm

3) it is inherently related to **argumentation** practices: activities such as the exegesis of a text, the interpretation of an idea or the discussion about its meaning are what constitutes the core of a philosopher's practice.

A deeper analysis of the materials collected helped us to break down these three generic features into a number of more specific ones (see fig. 5-1). In general, we concluded that a suitable semantic model should provide support for representing:

a. **historical events**, that is, events which are inherently time-dependent (e.g., the publication of a book, or an author's subscription to a viewpoint);

b. generic **uncertainty**, since often we are talking about facts which cannot be located exactly in the time and space dimension (e.g., the birth of Heraclitus);

c. **information objects**, and especially language-based information objects, as they are the traditionally preferred medium philosophical contents are expressed with;

d. **interpretation events**, intended as the process of attributing an abstract content to an information object (e.g., when we say that "Aristotle's fourth book of the Metaphysics states an ontological principle");

e. coexistence of **contradictory information**, which is a direct consequence of point d) (e.g., when people claim different or opposing views on the same subject);

f. **viewpoints**, and other non-material entities ("philosophical ideas"), for they are the objects philosophers are usually involved with, in their everyday practices;

g. **varying granularity:** as previously mentioned, this feature refers to the fact that philosophers normally (re)define the questions and ideas which lie at the centre of their work. As a result, the conceptions of two philosophers can have very little in common, if not at a meta-level (i.e., because they are using the same words, or because both of them can be related to a third conception which acts as a 'common ancestor'). For instance, let us consider Aristotle's doctrine distinguishing four types of "cause" and compare it with Hume's doctrine stating a radical skepticism towards any notion of "cause". Taken singularly, the two doctrines do not have much in common. More precisely, if we were to represent them from the point of view of their *internal* structure, the two notions of "cause" they are talking about would turn out as being essentially different. However, if we consider them from the broader *historical* perspective, it is easier to see a continuity between the two conceptions, for Hume is probably attacking one of the senses of "cause" originally defined by Aristotle. In conclusion, our semantic model needs to be capable of representing philosophical ideas at various levels of granularity, so to support reasoning at different levels of abstraction.

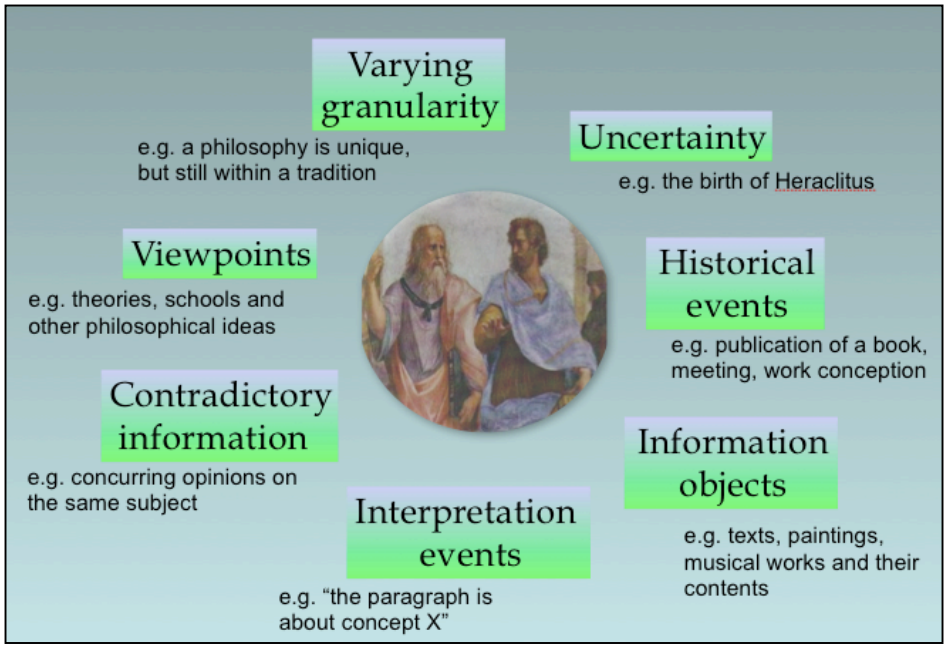

Fig. 5-1. The important dimensions for representing the philosophical domain

5.2.1 Our Approach

The main feature characterizing our approach is the decision to employ the CIDOC Conceptual Reference Model (Doerr, 2003) as a starting point for our formalizations. The CRM ontology started out as an attempt of the CIDOC Committee of the International Council of Museums (ICOM) to achieve semantic interoperability for museum data. Since 1996, the formal model has improved considerably till becoming in 2006 an ISO standard (Crofts et al., 2005). It is now (version 4.2) in a very stable form, and contains 75 classes and 108 properties, both arranged in multiple is-a hierarchies. The choice of using CIDOC-CRM was motivated by two reasons.

Firstly, because of its widely recognized status as a standard for interpreting cultural heritage data. Thus we can facilitate interoperability between our system and a wide range of cultural resources available online.

Secondly, for its extensive event-centered design. This design rationale, in fact, appeared to be appropriate also when trying to organize the history of philosophy. Even if it is common to see it as an *history of ideas*, stressing the importance of the *theoretical* (i.e., meta-historical) dimension, we believe it cannot be examined without an adequate consideration of the *historical* dimension. That is, a history of the events related (directly or indirectly) to those ideas. Thus, with reference to the domain analysis described above, we can say that point a) is directly addressed by CIDOC's generic modeling approach.

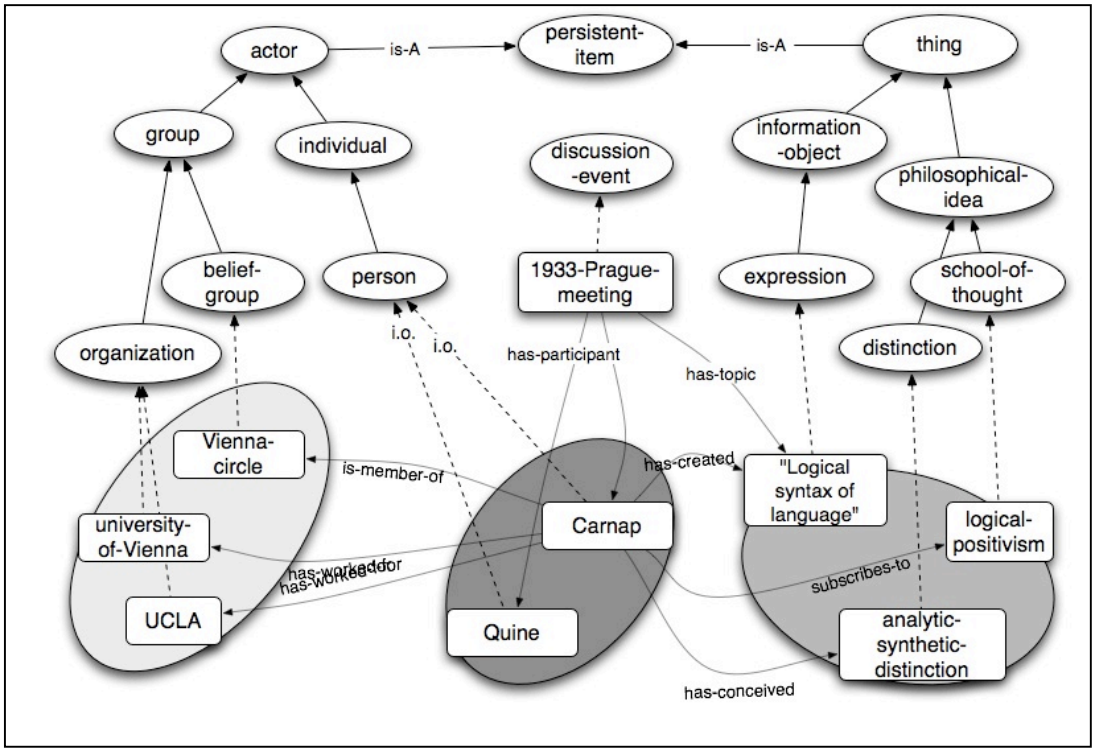

Figure 5-2. A typical event-based modeling in CIDOC

As an example, in figure 1 we can see an event-centered representation in the PhiloSurfical ontology. The `persistent-item` class, which is one of the five classes composing CIDOC's top layer (together with `time-specification`, `dimension`, `place` and `temporal-entity`) subsumes `thing` and `actor`. The two branches of the ontology departing from them can have various instances,

which are related by taking part (in various ways) to the same event (in this example “1933-Prague-meeting”). This kind of modeling, in the context of the PhiloSurfical tool, is extremely useful because of the multiple navigational pathways it can support (e.g., we could move to another event having the same topic, or to another topic treated during the same event, etc.). Please note that in the figure some relations (e.g., *has-worked-for*) are graphical shortcuts for the actual and lengthier formalization of the relevant event (e.g., an event instance stating that an actor worked for an institution at some point in time etc.).

We must remember that CIDOC, in compliance with Allen’s specifications (Allen, 1984), provides all the basic structures needed for time-based reasoning (e.g., relations stating that two events happened at the same time, one after the other etc.). The decision on how to implement the primitives needed for performing such operations is instead up to every specific use of the ontology. Thus, in order to obtain some basic reasoning on philosophers’ timelines and life-events we reused the time specifications found in the AKT reference ontology (AKT, 2002), and some work carried out during a previous project in our department, Cipher (CIPHER, 2008). However, as mentioned in point b), we needed a mechanisms for dealing with the fact that some dates (mainly in ancient philosophy) are uncertain. In such cases we simply decided to represent uncertain dates as time-intervals (cf. section 5.3.1).

Furthermore, we decided to integrate the event-based CIDOC reference model with formalizations from other ontologies, because they provide facilities that are relevant to the points c), d) and e) we have highlighted earlier during the

domain analysis (for a summary of the models imported, please see figure 5-3). In particular, we included knowledge about the domain of publications (e.g., publication-events, references' semantics, publishing-agents) from the AKT reference ontology (AKT, 2002) and knowledge about information objects (e.g., types, structure and content of information objects) from the related module of the Dolce foundational ontology (Gangemi et al., 2002). Moreover, as we are dealing with a domain where bibliographic resources are central, we have also attempted to build a model that is possibly compliant with cataloguing standard. To this purpose, we are providing mappings and reusing notions from the Functional Requirements for Bibliographic Records (FRBR) specifications (IFLA, 1998), which are a very influential standard[3] for librarians .

A large portion of the PhiloSurfical ontology is constituted by a series of new concepts and relations, mostly aimed at the description of philosophical events and ideas. The CIDOC-based modeling of philosophical events satisfy the requirements expressed in points a) and d). The formalizations departing from the `philosophical-idea` class, instead, satisfy the requirements described in f) and g). These concepts are derived from both formal and informal discussions with domain experts, and consultation of dictionaries or existing taxonomies. We will discuss this contribution in detail, in section 5.4.

[3] It is, for example, recommended by the Anglo-American Cataloguing Rules (http://www.aacr2.org/)

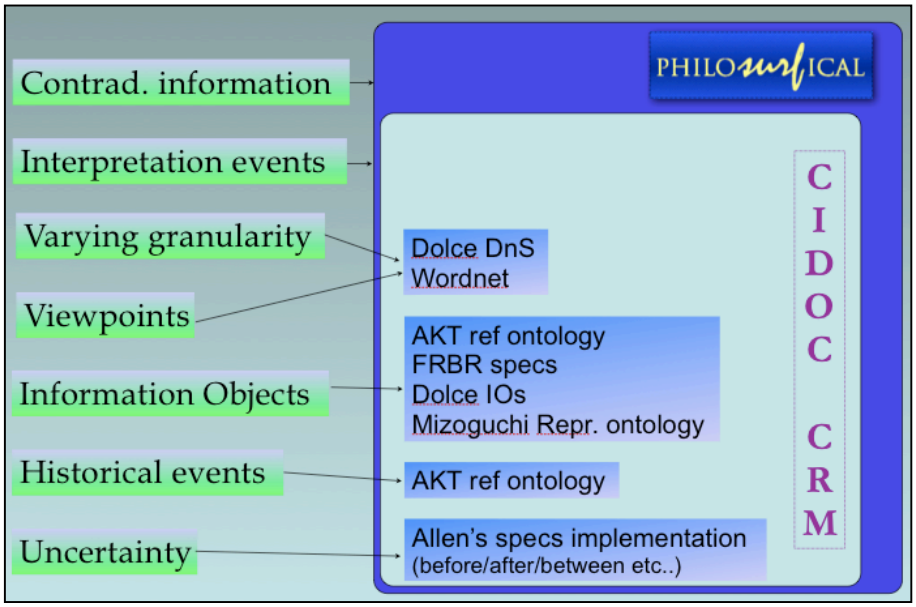

Fig. 5-3. Summary of the models imported in the PhiloSurfical ontology

As a final remark, it is important to remember that, as specified elsewhere (Pasin et al., 2007), the 'modeling choices' resulting from our investigations all share a common principle. That is, they can often be crystallized as 'patterns for navigation'. They could resemble the modeling patterns discussed in other works such as (Gangemi, 2005); however, while these focus on both architectural issues (i.e. regarding the ontology creation process) and content issues (i.e. regarding how to model a domain), we deal only with the second type of problems. In particular, the patterns we are describing represent some modeling decisions that are meant to guide the *interpretation* of philosophical knowledge, thus generating formal models that are *applicable* for providing non-trivial navigation mechanisms.

5.2.2 Technical notes

From the implementation point of view, the ontology (which at the time of writing includes 348 classes) is formalized by means of the Operational Conceptual Modelling Language (OCML) (Motta, 1999), which provides rich support for both specification and execution of knowledge models. Import/export mechanisms from OCML to other languages, such as OWL (W3C, 2004a) and Ontolingua (Farquhar et al., 1996), ensure interoperability with knowledge representation standards.

The decision of using OCML was motivated mainly by the need of having a development platform which was at the same time expressive, robust and easy to use. In particular, when we started our project (2004) we initially considered using the Semantic Web standards proposed by the W3C (W3C, 2008), however we quickly realized that the environments available at the time were not as flexible and reliable as OCML and moreover, they did not yet support adequately the standards for rule-based reasoning, which were being finalized at the time.

The latest version of the ontology can be found online at http://www.PhiloSurfical.open.ac.uk/ontology/. Also, the complete OCML specification of the ontology has been added at the end of this thesis work (see Appendix B).

In the rest of this chapter, when examples from the ontology are provided, they use the OCML syntax for describing classes, instances and rules. Please note that in order to facilitate the reading of this article we used different fonts depending on whether we refer to classes in the ontology (e.g., `event`) or

properties associated to them (e.g., *has-duration*). Instances are instead always double quoted (e.g., “the concept of will”). In the figures, classes are oval-shaped, rounded rectangles stand for instances and arrows represent relations. In particular, if not labeled otherwise, dashed arrows stand for the *instance-of* relation, while solid arrows stand for the *subclass-of* relation. Finally, in order to facilitate the readability of some figures (e.g., see figure 5-7) we grouped instances which belong to the same class using colored shapes and then we use only one arrow to link this group to other entities. In these cases the single arrow represents a set of relations linking each member of the group to the corresponding entity.

5.3 Ontology walkthrough

In this and the next sections we will outline the principal structure of the ontology, highlighting where the most important contributions of this work reside. Briefly, we anticipate that these are mainly related to the integration of different specifications (AKT, CIDOC, FRBR) into a single and consistent model, and to the novel formalization of concepts related to the philosophical domain (e.g., *ideas*, philosophers' *movements* etc.). Please notice that in the following sections we will not describe every single class in CIDOC but only the ones which a more relevant to our intended scope. The interested reader may find additional information about CIDOC in the reference documents made available online (Crofts et al., 2005).

Many of the example-instantiations throughout the article are related to Wittgenstein and his youth work, the Tractatus Logico-Philosophicus (Wittgenstein, 1921), as this is the main topic of the PhiloSurfical tool (cfr. chapter 6).

Moreover, please note that in many occasions we may make reference to philosophical facts or ideas which, although well-known to philosophers, might not be so to non-experts (e.g., Plato's "myth of the cave" as an instance of `rethorical-figure`, in section 5.5.6). In such situations, the reader who would like to gather more information about the examples could do so by consulting a manual, such as Grayling's history of philosophy (Grayling, 1998a,Grayling, 1998b).

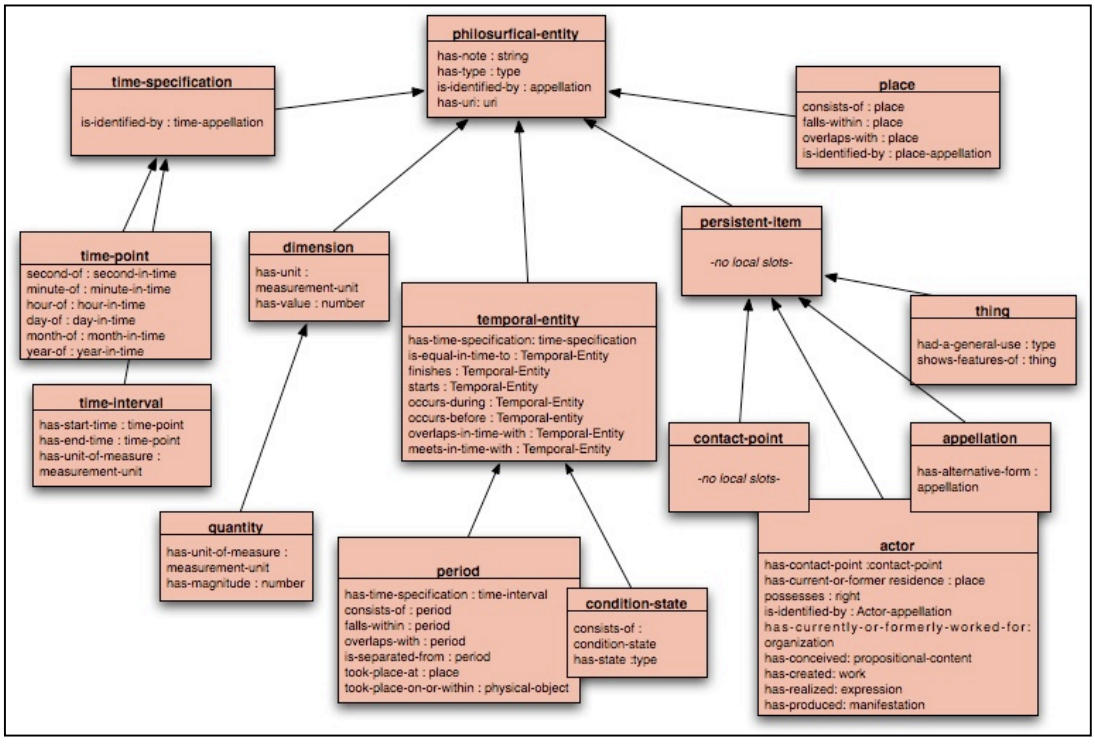

Figure 5-4. The top layer of the CIDOC-PhiloSurfical ontology

The highest layer of the ontology used in PhiloSurfical is the same of the one specified by the CIDOC Conceptual Reference Model (figure 5-4). There are

five main classes, which are `Time-specification`, `Place`, `Dimension`, `Temporal-entity` and `Persistent-item`. These classes map straightforwardly to the highest classes of the CRM. For that regards their subclasses, we kept most of the concepts of the original CRM ontology, modifying and extending it occasionally so to be able to integrate parts of other ontologies and our new concepts. In what follows we go through the major branches of the model, in particular highlighting the changes that we made to the existing formalizations.

5.3.1 Time-specification

The `time-specification` class refers to "abstract temporal extents, in the sense of Galilean physics, having a beginning, an end and a duration". Compared to the original CIDOC one, this class has been further developed by reusing the time specifications found in the AKT reference ontology, and some work carried out during a previous project in our department, Cipher (CIPHER, 2005). As mentioned above, by doing so we managed to deal with the fact that some dates (mainly in ancient philosophy) are uncertain. In such cases we decided to specify dates simply as the time interval within which the person is, for example, supposed to be born or dead.

However, we did not intend to provide a fully-comprehensive time reasoning facility here, so some typical tricky issues such as the julian/gregorian calendar change were not considered. Consequently, we interpreted all time data as implicitly existing as a continuum within the same reference system, which obviously remains unspecified.

A better semantic model of time can be integrated in the future. In OCML, the birth-date of a philosopher like Thales can be represented as follows:

```
(def-instance thales-birth time-interval
 ((has-start-time -623)
  (has-end-time -624)
  (has-unit-of-measure year)))
```

5.3.2 Place

The `Place` class does not differ from the original one in CIDOC. Basically, it provides a layer of abstraction between the definition of a place and the different ways we can refer to it. Places can in fact be simultaneously identified through various types of appellations: common names, addresses, spatial coordinates etc. This facility is supported by the `place-appellation` class, which is in the `persistent-item` branch. So, for example, this abstraction mechanism could serve as a way to map a coordinate-based representation to one based on place names (e.g., names of geopolitical areas), or as an easy way to disambiguate the location of places named differently in time (e.g., Prussia and Germany). With respect to the latter example, it is important to emphasize that the CRM offers also a way to store the information related to the name-change together with its time specification (e.g., who called Prussia Germany, and when it happened). This can be done through the `attribute-assignment` class (which is a subclass of `event`). Moreover, the abstraction of a place from its identifiers seemed to us a promising feature also because it will allow future integrations of the PhiloSurfical tool with other mapping services (e.g., google-maps (Google, 2008)).

So, for example:

```
(def-class ancient-greece place
  ((consists-of ancient-athens ancient-sparta ancient-thebes ancient-
delphi        ancient-argos ancient-corinth ancient-syracusae ancient-
neapolis)
  (falls-within balchanic-peninsula)
  (overlaps-with modern-greece modern-italy modern-turkey)
  (is-identified-by ancient-greece-coordinates)))
```

5.3.3 Dimension

The `Dimension` class "comprises quantifiable properties that are measured by some calibrated means and can be approximated by numerical values". We have not modified it, but in order to integrate several AKT classes (mainly related to time specifications, such as the `duration` class) we added the AKT `Quantity` as a subclass. Its local properties match exactly the properties of `Dimension` (which are *has-unit* and *has-value*), so that a simple slot-renaming declaration (a feature of OCML) guarantees the correct functioning of all the imported specifications from AKT.

```
(def-class QUANTITY (dimension)
 ((has-unit-of-measure :type measurement-unit)
  (has-magnitude :type number))
  :slot-renaming ((has-unit-of-measure has-unit)
                  (has-magnitude has-value)))
```

5.3.4 Persistent item

The `Persistent-item` class refers to what is usually called an *endurant*, in philosophical terms (Johansson, 2005). Among its subclasses we have `Appellation`, `Contact-point`, `Actor` and `Thing`. The first two classes are used to decouple entities from the possible ways we could refer to them (for example, through names or addresses). Here we just added three other concepts: `URI`, `ISBN-number` and `spatial-coordinates`, all of them subclasses of `appellation`.

For that regards `actor` and `thing`, instead, they need to be examined in more details because we have partially modified their specifications. As previously mentioned, in figure 5-2 we can see part of the `persistent-item` hierarchy, together with some interrelated instances.

Actors are divided up into `individual` and `group`. `Person` is the only type of `individual` we have modeled and therefore has no subclasses, while `group` contains `group-of-people` and `legal-body`. The only `group-of-people` we are defining is `belief-group`. In comparison to the original modeling of CIDOC, the classes added here are `individual` and `belief-group`. The former because it seemed to us the most appropriate counterpart to `group` (which was already defined in CIDOC). Moreover, since originally `person` was a direct subclass of `actor`, with our modeling it is also possible to define other individual actors which are not humans: e.g., Orwell's animals, or extraordinary creatures such as Mary Shelley's Frankenstein. The latter because it is needed specifically for the philosophical domain (e.g., to refer to the "stoics" or the "early followers of Shinto").

In order to facilitate the creation of the PhiloSurfical knowledge base, we have extended the properties of `actor` so to be able to say that they work for `organizations` and that they are the authors of different types of `conceptual-objects`. In particular, there are four relations that can possibly link an `actor` to a `conceptual-object` (*has-conceived*, *has-created*, *has-realized*, *has-produced*). This modeling is compliant with the FRBR specifications, and a detailed explanation

of the reasons why we adopted and integrated this design pattern can be found later in the section about information objects (section 5.4).

Let us stress the fact that most of these relations are intended as being 'shortcuts' for the definition of the corresponding events (cfr. CIDOC 4.2 specifications, p. xi). Shortcuts are a handy feature when there is not sufficient information available for the creation of an `event`, but we still want to formalize some piece of information about an entity. For example, the value of the slot *has-currently-or-formerly-worked-for* provides only a plain list of the `organizations` an `actor` has encountered in his life, without any other specification regarding the date or the length of a job. In order to provide this information the work experience should be modeled as an `event` instance (i.e. `working-for-organization`), where *start-time*, *end-time*, *working-position* and also other properties can be specified and stored. Events corresponding to the conceptual-object related slots of `Actor` can also be found in the ontology. We will discuss all of them in section 5.3.5.3. The formalization of the `actor` class is as follows:

```
(def-class actor (persistent-item)
 ((has-contact-point :type contact-point)
  (has-current-or-former-residence :type place)
  (possesses :type right)
  (is-identified-by :type actor-appellation)
  (has-currently-or-formerly-worked-for:type organization)
  (has-conceived :type propositional-content)
  (has-created :type work)
  (has-realized :type expression)
  (has-produced :type manifestation)))
```

In a similar fashion, we extended also the definition of the `person` class, which in CIDOC did not have any local property. `Person` inherits from both the `actor` and the `physical-object` branch. We added several shortcut-properties, such as

has-social-role (which lets us quickly annotate whether a scholar is usually considered as a philosopher, a mathematician, a poet, or all these roles at the same time), *has-date-of-birth* or *has-birth-place*. Also in the case of `person`, since most of these properties can be more exhaustively represented as `event` instances, we encourage annotators to model them fully whenever enough information is available. In fact, a widespread usage of shortcuts would go against CIDOC event-based design principles and, also, may lead to the generation of internal contradictions in the knowledge base.

```
(def-class person (actor biological-object)
 ((has-gender :type gender)
  (has-social-role :type social-role)
  (has-date-of-birth calendar-date)
  (has-date-of-death calendar-date)
  (has-birth-place :type place)
  (has-death-place :type place)))
```

For that regards the `legal-body` class, we have extended the CIDOC ontology with the extensive specifications of `organization` found in the AKT ontology (containing more than 30 concepts). In this way we can properly refer to educational organizations involved in the life of philosophers, or to other organizations responsible for the publishing of their works.

Instead, for that regards the `group-of-people` class, we are mainly interested in its subclass `Belief-group`. Its distinctive property is *share-belief:* it hints at the fact that its members must share one or more existing views. So, for example, we can have a `political-group` (e.g., “the Leninists”), a `religious-group` (e.g., “the early followers of Shinto”) or a `philosophical-school` (e.g., “the stoics”). For example in the last case the group is constrained by its elements having to share a philosophical `view` (for a clearer explanation of the features of the

`field-of-study` and `view` classes, see section 5.5). The OCML formalization reads as follows:

```
(def-class Belief-Group (Group-Of-People)
 ((shares-belief :type View)
  (has-related-intellectual-event :type intellectual-movement)))

(def-class Philosophical-school (Belief-group) ?p
:constraint
      (and (shares-belief ?p ?v)
           (belongs-to-area ?v ?a)
           (or (branch-of-philosophy ?a)
                (and (field-of-study ?a)
                     (has-sub-area branch-of-philosophy ?a)))))
```

We must anticipate that there is a parallelism between belief groups, intellectual events and abstract ideas typifying them: for example, when talking about “the stoics”, depending on the context we might mean the actual people, the philosophy they were pursuing or the overall historical period they characterized. Since the definition of these entities is often ambiguous, we created a modeling pattern for guaranteeing the internal consistency among these entities and help sorting out unclear cases. We will discuss this pattern in more details in the section about temporal items, when describing the `intellectual-movement` class (section 5.3.5.1).

Also, a difficulty may arise from the fact that sometimes it is not easy to decide whether an entity is an instance of a `group-of-people`, or of an `organization`. The discriminating property in this case is the recognized legal status of an `organization`, as opposed to the usually vague boundaries of, for example, a `belief-group` (e.g., “the university of Cambridge” vs. “the circle of Vienna”). But even if this distinction might seem to someone quite straightforward, we

acknowledge that the instantiation of the model will often be quite dependent on individual interpretations.

Finally, another class we have introduced under `organization` is `geopolitical-area` (i.e. continents, cities etc.).

The last branch of persistent-item, starting from `thing`, will not be discussed in detail since it has not been modified at all. It is only worth remembering that it makes an important division between `man-made-thing` (which can be either a physical object, or a conceptual product) and `physical-thing` (such as a stone, or a `biological object` such as a `person`). Another class we will also see later is `information-carrier`, a subclass of `man-made-object` with the essential property of being capable of carrying `information-objects` (see section 5.4.2). Moreover, both *information* and *physical* objects are also subsumed by `legal-object` (that is, they are object of rights). In other words, according to CIDOC the only entities which are not considered `legal-objects` are things such as a theory, a problem or a role. That is, all immaterial man-made products (i.e. instances of `conceptual-object`) which are not direct reifications of objects carrying information (see section 5.4). We can see a picture of the `thing`-hierarchy in figure 5-5 (notice that, as already stated in section 5-2-2, if not labeled otherwise solid arrows always stand for the *subclass-of* relation).

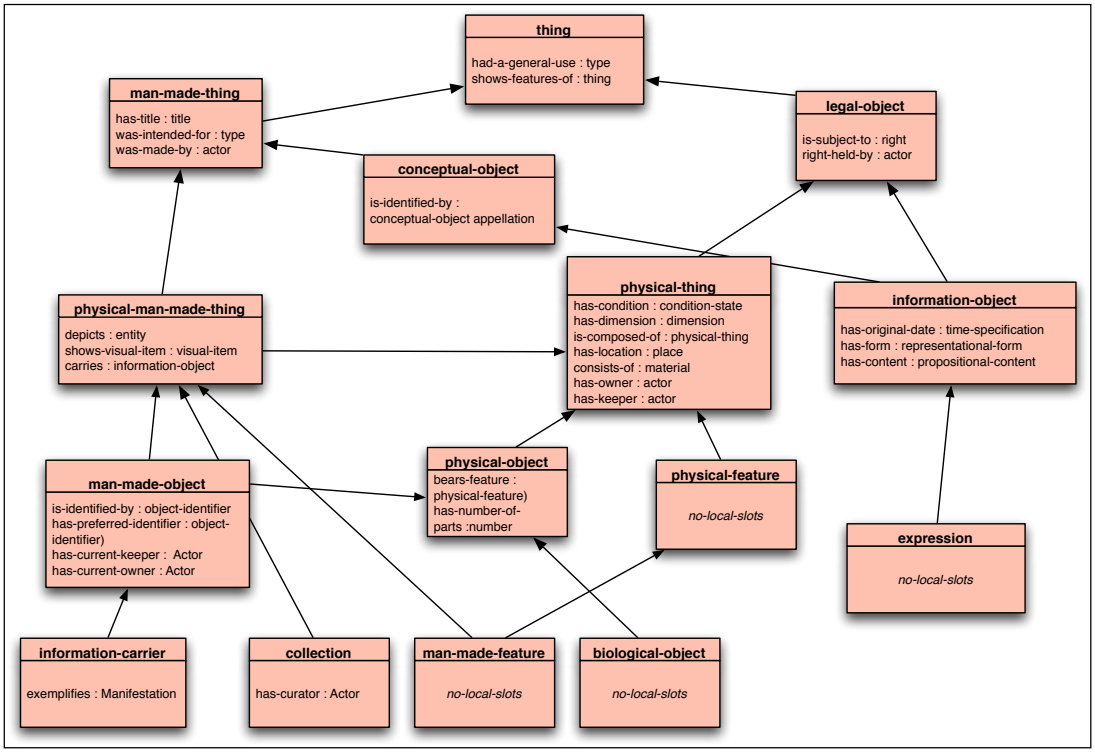

Figure 5-5. The hierarchy of classes departing from 'thing'

5.3.5 Temporal entity

The last of the five classes composing the CIDOC top layer is `Temporal-entity`. It represents all things that have a time span, and among them what we commonly call events. For example, births and deaths of philosophers, journeys which occurred during their lives, creation of intellectual works, conception of ideas or performance of scientific experiments are all entities which can be suitably described using the CIDOC model as a basis. However, it was also necessary to extend it with some new classes. For explanation purposes, we divided them into four groups: intellectual movements; events related to the academic life and to the life of philosophers; events related to the production and modifications of philosophical ideas; interpretation events.

5.3.5.1 Intellectual movements

A direct subclass of `temporal-entity` is `period`, which according to CIDOC should subsume prehistoric or historic periods, or even artistic styles. This is motivated by the fact that "it is the social or physical coherence of these phenomena that identify a Period and not the associated spatio-temporal bounds". This seemed to apply quite neatly also to cultural and philosophical periods, thus we have added `Intellectual-movement` and its subclass `Philosophical-movement` to the hierarchy.

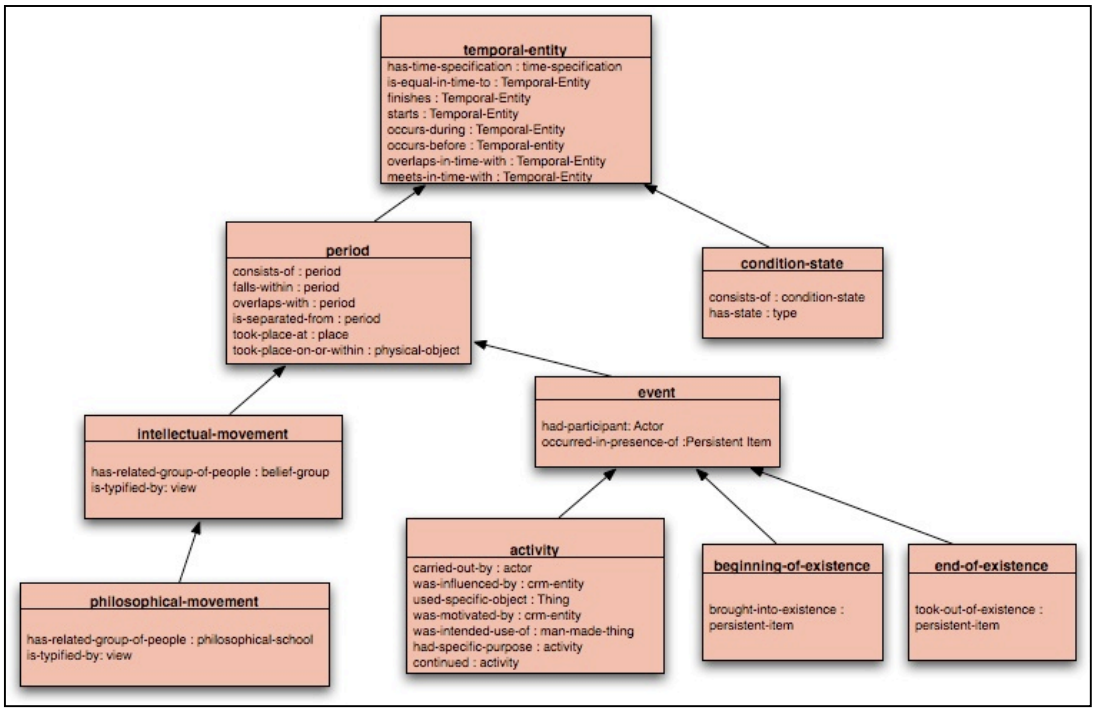

Figure 5-6. The hierarchy of 'temporal entities'

So, for example, we can describe the enlightenment movement in the following way (note that the temporal relations are specified here as slots, but can also be inferred whenever other periods' time specifications were provided):

```
(def-instance enlightenment intellectual-movement
 ((has-time-specification 18th-century)
  (overlaps-in-time-with scientific-revolution renaissance)
    (meets-in-time-with French-revolution American-revolution
romanticism)
  (overlaps-with Age-of-Reason neo-classical-art)
  (took-place-at Germany France  Britain  Spain)
  (has-related-group-of-people enlightenment-group-of-people)
  (is-typified-by enlightenment-conception)))
```

The last two properties in the previous example have a special importance. Periods, in fact, appear to be tightly connected to the abstract ideas defining them and to the group of people that often carries the same name. The slots *has-related-group-of-people* and *is-typified-by* specifically serve this purpose.

This issue is better understood if we just consider how often this feature of intellectual events generates ambiguities, since in natural language expressions it is not clear what entity we are referring to. For example, let us consider the following three statements:

a) “Throughout history, the attacks of *rationalism* against empiricism has diminished”

b) “Descartes was one of the founders of modern *rationalism*”

c) “This theory is clearly a new and re-shaped *rationalism*”

At a first examination, all three sentences refer to “rationalism”. However, a deeper ontological analysis shows that in a) “rationalism” is the label referencing to a group of people, in b) we are denoting an intellectual movement, a type of event, while in c) we are (probably) referring to an abstract idea.

A modeling pattern (figure 5-7) involving `actor`, `period` and `view` (a type of `philosophical-idea`, as we shall see later, expressing a standpoint) attempts to

tidy things up and clarify the ambiguity of a term, such as “rationalism”, by distinguishing between the three different uses of the term in the language. By doing so, we are providing a context of usage for such ambiguous concepts, and a direct way to navigate coherently among entities that are ontologically quite distinct (i.e. from `temporal-entity` to `actor` and `propositional-content`, which belong to separate branches of the ontology). Moreover, such a context-specification could be used by a reasoner to derive inferences from incomplete or inconsistent data sources, or for performing information extraction.

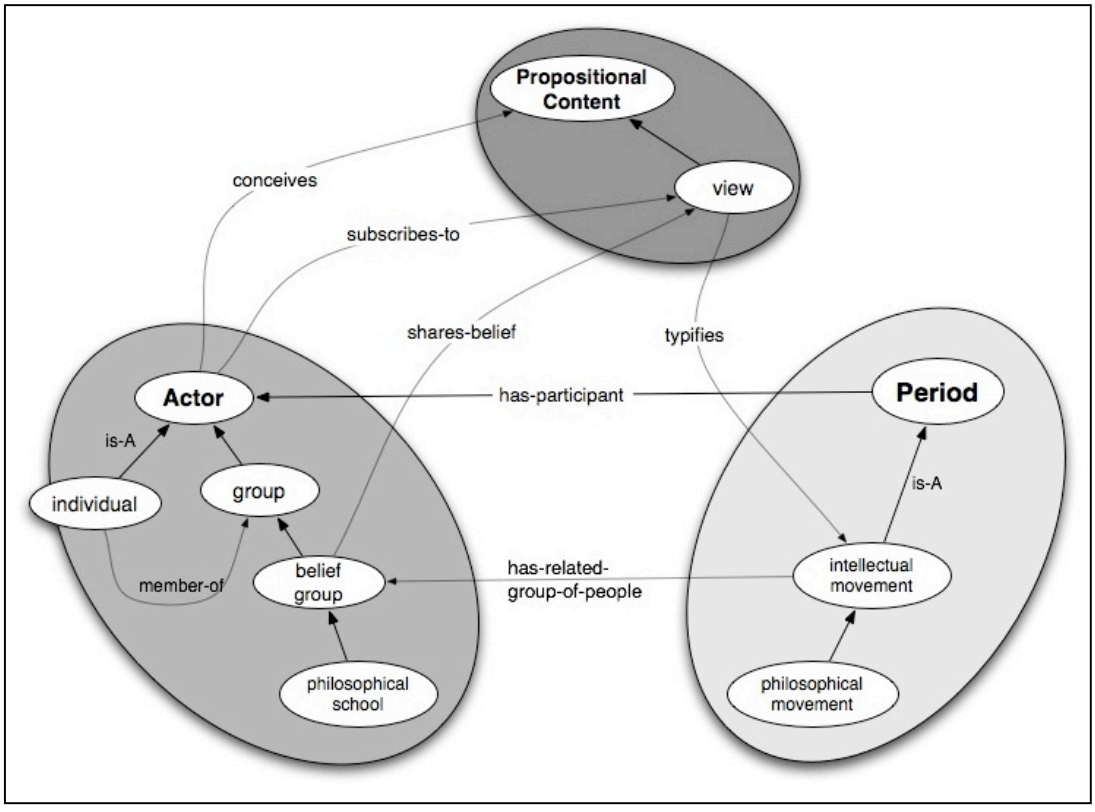

Figure 5-7. The actor-event-view modeling pattern

5.3.5.2 Events related to the academic life, and to the life of philosophers

Among this group of events, we can have births and deaths of philosophers (e.g., the “death of Socrates”), production of physical objects (e.g., “Pascal's

construction of the arithmetic machine"), journeys performed during their lives (e.g., "Wittgenstein's trip to Norway"), production of publications (e.g., the "publication of the first English version of Kant's Critique of practical reason in 1836"), experiments (which, even if not directly performed by philosophers, are used by them in the argumentations - e.g., the "experiment about Dolly, the cloned sheep") and formation events which start the existence of social groups or organizations (e.g., the "formation of the circle of Vienna").

In CRM, `event` differentiates from `period` (of which is a direct subclass) mainly because "viewed at a coarse level of detail, an Event is an 'instantaneous' change of state", as opposed to a much longer change of state, such as an historical period. This distinction is not always easily put into practice, since the boundaries between an `event` and a `period` can often be the result of a subjective interpretation. Nonetheless we decided to use this formalization as it was, because it seemed to mirror well the reality of things in most cases. Moreover, with respect to the examples above, CIDOC provides quite a good support for their representation. In particular, this is done through the branches departing from `activity`, `beginning-of-existence` and `end-of-existence`. The only two classes we added here are `publication` and `journey`, respectively as subclasses of `production` and `move`.

More significant extensions were instead needed in order to model the domain of social activities. As we can see in figure 5-8, `social-activity` (which was defined by us) has five subclasses: `discussion`, `joining-a-group`, `educational-activity`, `close-social-contact` and `social-gathering`. The choice of formalizing these classes and not others (such as, for example, the '*marriage*'

activity) was motivated by two reasons. First, because from our analysis of the philosophical literature those type of events appeared as being the most interesting from the point of view of philosophical scholarship. Second, because during the knowledge-base construction phase (see section 6.3) we needed these classes (and only these classes) in order to represent various data automatically extracted from other websites (e.g., information about philosophers' roles as *Ph.D. advisor* or *Ph.D student*).

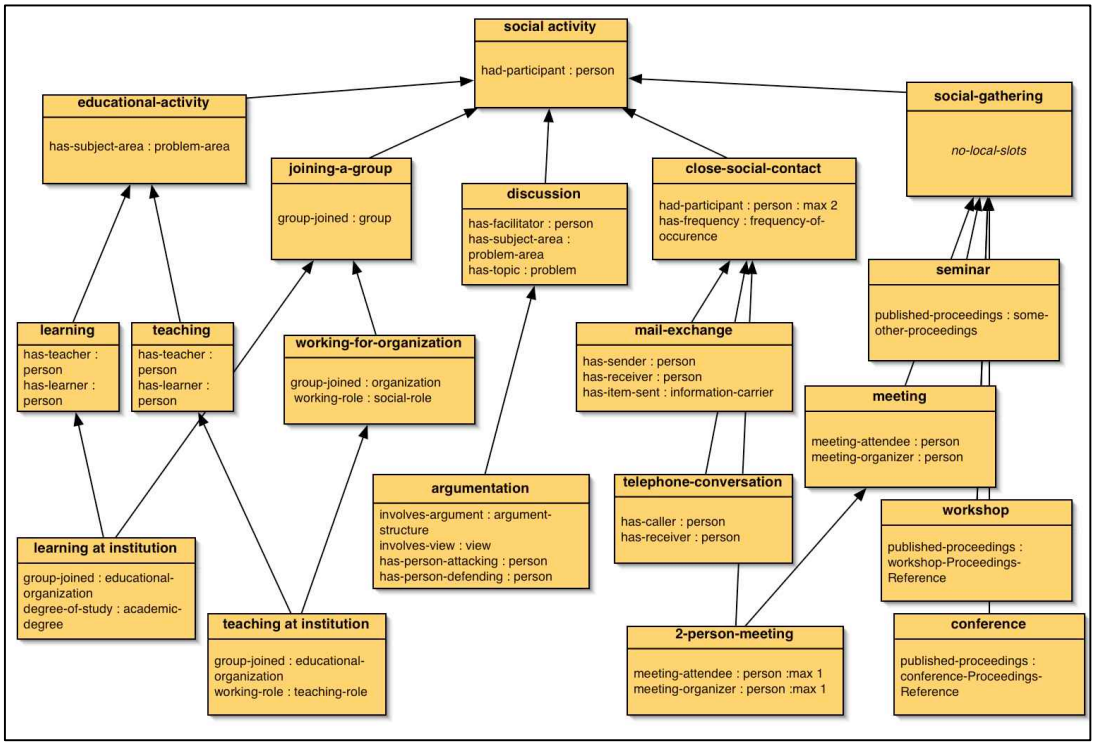

Figure 5-8. The social-activity branch

`Discussion` refers to any generic conversation between persons, with the essential property of having a specific topic (differently from a generic conversation, where the topic could be absent or unspecified). Its subclass `argumentation`, instead, reflects the kind of more structured interactions that often happens among thinkers. Within an `argumentation`, it is always possible to

recognize different competing views about a topic, together with some arguments supporting them. It is important to remember that with this class we just aim at representing an argument from a temporal perspective, without taking into account its internal structure. This is done, instead, through the class `argumentation-structure` (a subclass of `propositional-content`, as discussed in section 5.5.1).

In other words, arguments can have multiple structures (often very different from each other), but they can be part of an `argument` event irrespectively of their specific characteristics. This modeling leaves some space also for future integration of other and more complicated argumentation formalisms, such as the ones presented in (Reed and Rowe, 2004).

`Joining-a-group` refers to distinctive events where a `person` becomes a member of a `group-of-people` or of an `organization` (e.g., "Aristotle joining the Academy of Plato", or "Heidegger joining the Nazi party"). If the institution being joined is an educational institution, the membership could be further defined as a learning activity (`learning-at-institution`) or a teaching activity (`teaching-at-institution`). Of the two, the latter is also a subclass of `working-for-organization`, while both of them inherit from the `educational-activity` branch the specification of the research area treated and of the roles the people involved in the educational event are playing.

Let us remind the reader that all of these classes inherit various properties from CIDOC's `temporal-entity` class; in particular, they inherit the property *has-time-specification* (cf. fig. 5-6), which allows the specification of a `time-specification` instance (i.e., either a `time-point` or a `time-interval`). Therefore, by instantiating a class such as `learning-at-institution` it is also possible to

represent the *entire* time-period a person spent at that institution - that is, in this case we would use a `time-interval` as the temporal 'coordinate' of the instance.

So, for example, we can specify that the book by Kimberley Cornish (titled "The Jew of Linz" (Cornish, 1998)) has as subject the fact that Wittgenstein, while studying at the Linz Realschule, had Hitler as one of his young school-fellows (of course, claiming this to be the subject of the book is ultimately our interpretation, in compliance also with what will be said in section 5.3.5.4). Figure 5-9 depicts graphically this situation.

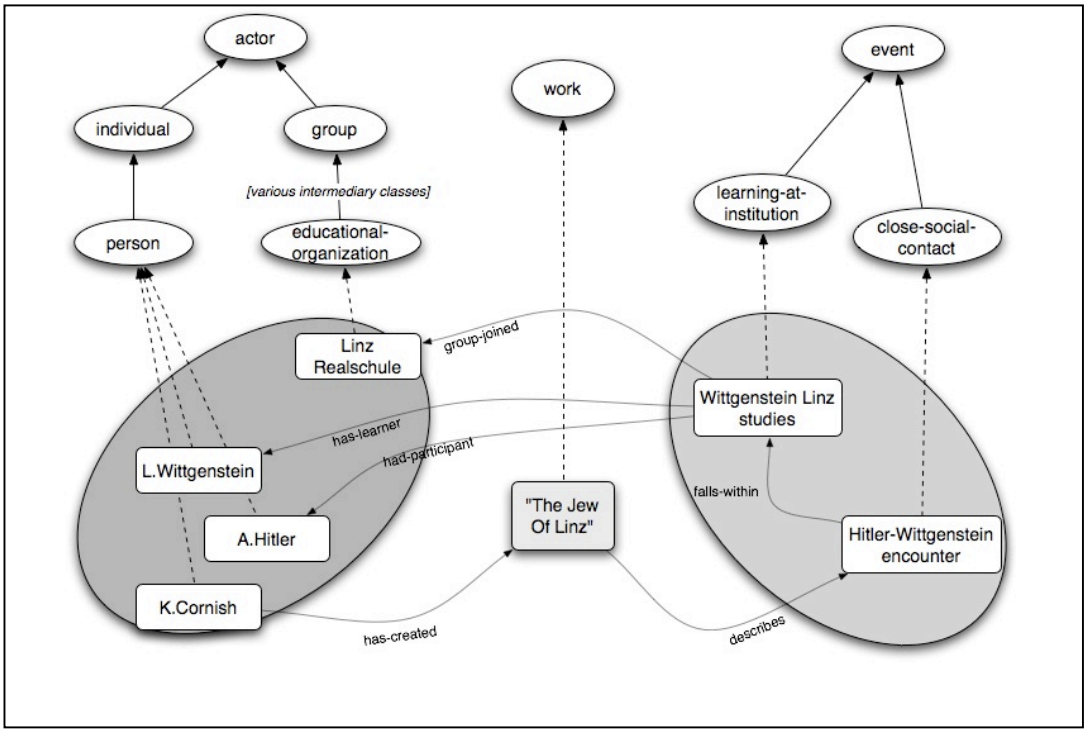

Figure 5-9. Representation of an event described by "The Jew of Linz"

The class `social-gathering` refers to events such as a `seminar` or a `conference`, where the number of people can vary substantially, while the `close-social-contact` class maps "encounters" between two people only (in person, on the phone, or via a mail exchange). In order to store the information

about the content of the mail exchange we need to model an `information-object`, which is then linked to the mail event, as in the example below[4].

```
(def-instance mail-exchange-Witt-Malcolm  mail-exchange
 (has-time-specification 4-1-1948)
 (has-sender Ludwig-Wittgenstein)
 (has-receiver Norman-Malcolm)
 (has-item-sent 4-1-48-letter-to-malcolm)))

(def-instance 4-1-48-letter-to-malcolm  expression
 (was-made-by Ludwig-Wittgenstein)
 (has-form written-english)
 (is-embodied-in original-4-1-48-letter)
 (has-as-subject importance-of-reading-aloud)))
```

The set of classes about social gatherings has been partly imported from the AKT reference ontology (AKT, 2002). By doing so, we achieved the double result of being able to model these types of events (which are quite important in the philosophical domain), and also to easily import the extensive formalizations of publication-references provided by the same ontology. This was possible because in AKT these formalizations rely on various classes describing the academic-events just mentioned.

5.3.5.3 Events related to the production and modification of philosophical ideas

These events are organized in the ontology under the class `intellectual-activity`. As in the case of the academic-events (section 5.3.5.2), the selection of the intellectual activities was motivated by both our initial domain analysis and PhiloSurfical's knowledge-base construction phase.

[4] The letters we are mentioning can be found on http://rabbit.trin.cam.ac.uk/~jon/Witt/Wittgenstein.html

Among them, we identified the following ones: `conceptual-creation`, `idea-modification`, `idea-usage`, `study`, `view-subscription` and `interpretation`.

`Conceptual-creation` reflects the creation of conceptual entities such as ideas and information objects (in the original CRM this class was called “creation”, and was a direct subclass of `activity`). `Idea-modification` reflects the changing of one or more ideas within the context of a `view` (e.g., the “evolution of the meaning of *libido* in the work of Sigmund Freud”); more specifically, `theory-refinement` represents the modification of a `theory` by adding or removing one of its constituent elements (e.g., the “evolution of psychoanalysis theory after the publication of the 1920 essay *Beyond the Pleasure Principle*”).

Often thinkers do not create concepts ex-novo, but reuse old ones: `idea-usage` aims at modelling this event, while `theory-transposition` models the special case when a `theory` is taken out of a context and reused within another one (e.g., “Spencer’s evolutionism”, which extends “Darwin’s evolutionism” from biology to metaphysics). Thinkers are also normally occupied studying things, be them documents (`study-a-document`), ideas (`study-an-idea`) or events (`study-an-event`). With these classes we can model the event of a person being interested and examining something. They are therefore connected, but different, from the activity of interpreting something (as explained in the next section).

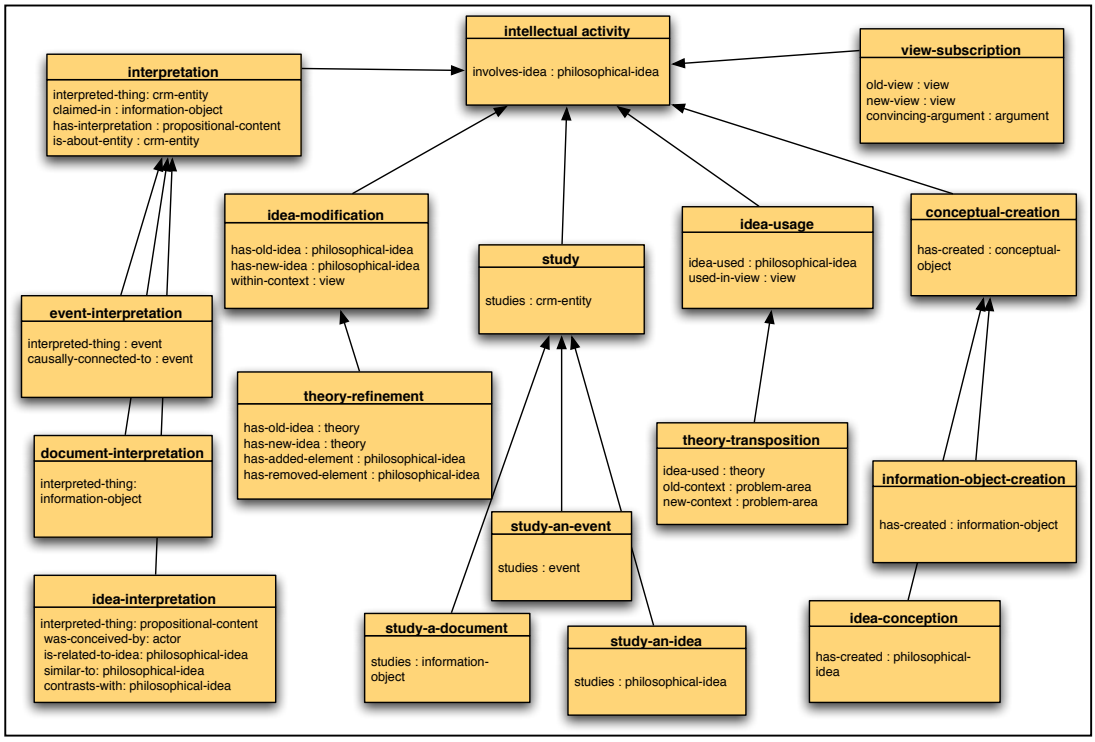

Figure 5-10. The intellectual activities branch

`View-subscription` models the intentional adoption of a view by a person, providing support for the annotation of the main `arguments` that convinced him/her (if existing). Finally, `interpretation` is representing the common-sense process of assigning a 'meaning' to a thing. This class has a fundamental importance in our ontology, in particular because it provides a mechanism to detach an idea-instance from the possible interpretations we may have of it, thus guaranteeing the coexistence of contrasting information within a single knowledge base. In the following section we will explain in more details how this works.

5.3.5.4 Events representing the interpretation process

As mentioned above, an `interpretation` is essentially an attempt to represent the process by which we assign some 'meaning' to a thing. With a more formal

language, this is equivalent to associating an instance of `propositional-content` (i.e. the *interpretation*) to any other instance of the ontology (i.e. the *interpreted-thing*). The OCML formula is the following:

```
(def-class interpretation (intellectual-activity)
 ((interprets :type crm-entity)
  (has-interpretation :type propositional-content)
  (is-about-entity :type crm-entity)))
```

The *is-about-entity* property, differently from *has-interpretation*, may refer to any entity in the ontology. We decided to add it so to provide a less restrictive mechanism to link two entities. For example, it would be possible to state the *aboutness* of two `information-object` (i.e. when we say that 'a text is about another text', such as in "paragraph 5 of the *Tractatus* is related to the last chapter of Carnap's *The Logical Syntax of Language*") without having to instantiate any information concerning the abstract contents of the information objects in question (i.e. the `propositional-content`).

The specific features of the subclasses of `interpretation` may vary depending on the value of the *interprets* slot, that is, depending on what we are interpreting. At the moment, we are providing support for the interpretation of documents, events and ideas.

The `document-interpretation` class lets users correlate a `propositional-content` to an `information-object` (e.g., "paragraph 7 of the Tractatus is about the idea of mysticism"). Also, within a `document-interpretation` it is possible to specify the *pedagogical value* of resources. This is represented by a specific subclass of `type` (see section 5.4.1), `pedagogical-functional-value` (e.g.,

"introduction", "summary", "compendium" etc.). At the moment, this pedagogical knowledge is used simply for organizing the resources' presentation in PhiloSurfical. However, in the future more complex services could benefit from this specification. For example, we could extend it using other related ontologies (e.g., (Ullrich, 2004)) or by linking it to existing LOs metadata (e.g., LOM (IEEE)), so to perform some reasoning also at this level.

A second type of `interpretation` is instead `event-interpretation`. Here a user can claim the connection (possibly causal) between different events, or their being about some abstract content or document (e.g., when we say that "Monk's book is about the life of Wittgenstein", or that, according to Hegel, "the heroic gestures of Napoleon are representing the unfolding of the Absolute Idea").

Finally, with the `idea-interpretation` class we aimed at detaching the specification of the characteristics of an idea-instance (e.g., how it relates to other ideas, who we think authored it, etc.) from the idea-instance itself. Let us exemplify this with an instance from the PhiloSurfical knowledge base:

```
(def-instance interpretation-001 concept-interpretation
 ((interprets law-of-nature-by-wittgenstein)
 (has-related-concept experience-by-wittgenstein prop-of-science-
concept)
 (has-opposite-concept laws-of-logic-concept)
 (is-equivalent-to form-prop-science-concept)
 (is-generalization-of law-of-induction-by-wittgenstein law-of-
causality-by- wittgenstein )
 (is-related-to-idea mesh-metaphor fate-science-analogy)
 (carried-out-by michele-pasin)))
```

Here we are describing the properties of the concept of "law of nature by Wittgenstein" in such a way that these descriptions will be associated only to a

specific user (i.e. the value of the slot *carried-out-by*). This is an important mechanism in PhiloSurfical: in fact by doing so we can have different description of the same concept (and, in general, of any idea) coexisting within the same knowledge base. In other terms, we are providing support for *concurring* and possibly *contradictory* information management. In future versions of our work, this feature is likely to be further developed with more complex mechanism to retrieve, for example, contrasting interpretations, or letting users navigate through alternative views of the same ideas.

In conclusion, in figure 5-11 we can see a graphical example of the idea-instantiation process. In particular, the figure focuses on how the ontology can support the separate specification of various ideas' properties (i.e. appellations, relations to other ideas etc.). Please note that the *has-common-name* relation is represented in the figure by means of a graphical 'shortcut': in reality, each of the `concept` instances should be linked to its correspondent `appellation` instance through a different *has-common-name* relation.

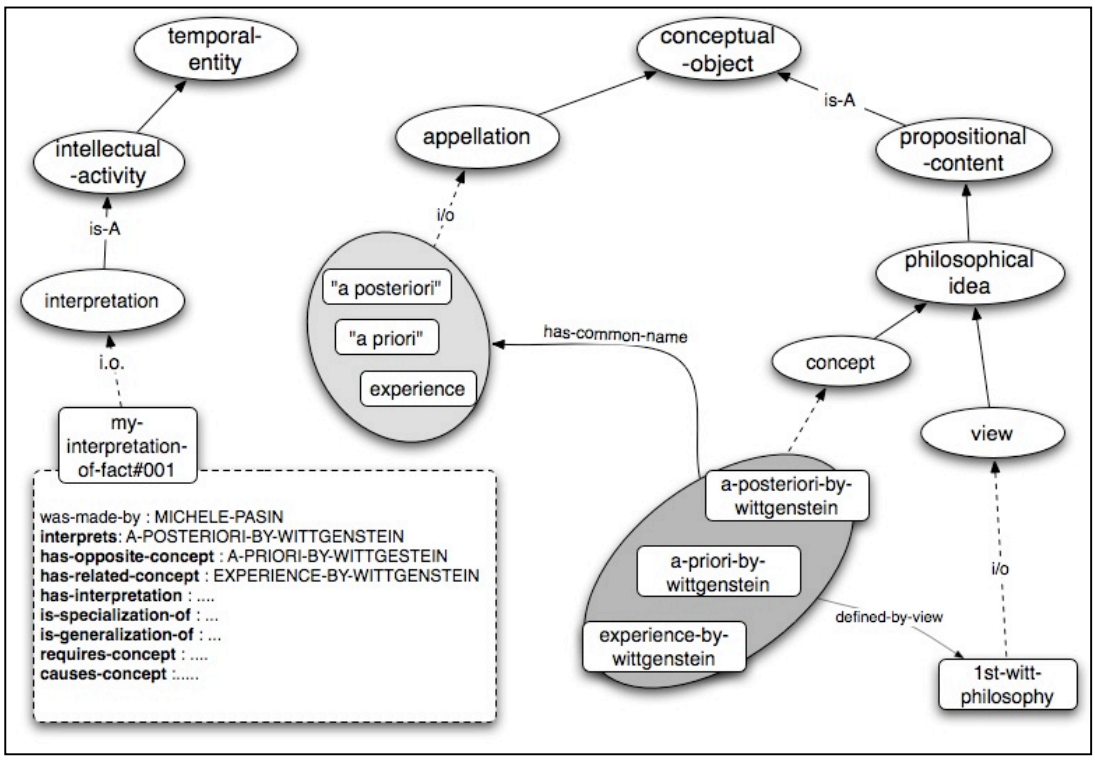

Fig. 5-11. The separation between interpretations and ideas instances

5.4 Conceptual Object

The `conceptual-object` class "comprises the non-material products of our minds, in order to allow for reasoning about their identity, circumstances of creation and historical implications". Compared to the original CIDOC specifications, all the conceptual-object subclasses (`type`, `right` and `information-object`) remained in the hierarchy. However, we made some modifications to them, and we also added several new subclasses.

The main reason for doing this relies on the fact that the domain we are modeling deals eminently with abstract entities, such as philosophical ideas and information objects. Although the CIDOC ontology provides some basic semantic structures for representing things such as books, articles, movies and

their contents, it is clearly more suited for representing various features of physical objects, in particular museum artifacts (e.g., mummies, vases, paintings etc.).

Therefore, we have integrated from other sources some specific modeling patterns targeted at representing information-related entities. One of them is inspired by Dolce's description of information-objects (IO) (Gangemi et al., 2005), which aims at abstracting and separating three aspects of them. That is,

a) the information about the content of a representation,

b) the information about the form used to present it and

c) the information about the physical implementation of an IO, which realizes it in the real world.

This modeling pattern makes it possible to deal with various common-sense concepts, such as the fact that we can have "copies" of representations, different "versions" of them, or representations "about" other representations.

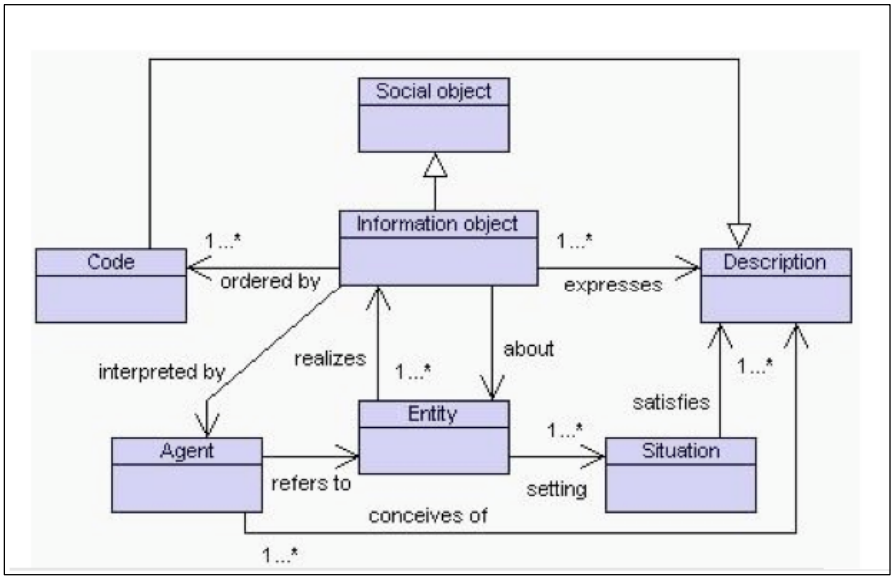

Figure 5-12. Dolce's IO design pattern (from Gangemi, 2005)

Moreover, another source of inspiration for our formalizations have been the FRBR specifications (IFLA, 1998). They are an attempt to separate the description of intellectual works from the various existing versions of them and their physical realization. Although the requirements defined in the FRBR specifications are not as formal as an ontology, they are a widely accepted standard among many libraries and related institutions. Therefore, since the philosophical domain is primarily related to linguistic and textual information objects, we have included in our model a good support for the representation of bibliographic records. This will facilitate the re-usage of our ontology in a context other than the PhiloSurfical tool (e.g., in a digital library).

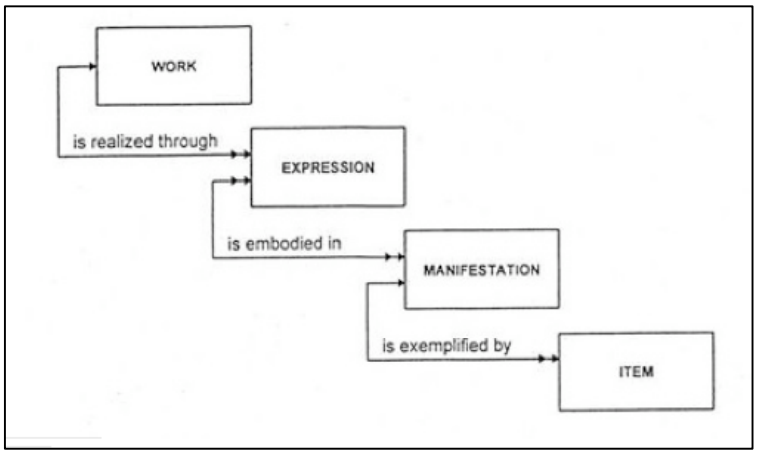

Figure 5-13. Extract from the FRBR specifications

As a further consequence of this library-oriented approach, we also envisaged the need for managing a system of bibliographic records' references. To this purpose, we have integrated a number of classes describing publications and references' structures from the AKT Reference ontology (AKT, 2002).

The resulting model is quite complicated, but neatly organized under seven main classes: `role`, `type`, `right`, `manifestation`, `information-object`, `representational-form` and `propositional-content`. In the next paragraphs we

will go through these classes, often providing an explanation of their usage through some real-world examples. The tree of the main entities subsumed by the conceptual-object class is graphically represented in figure 5-14.

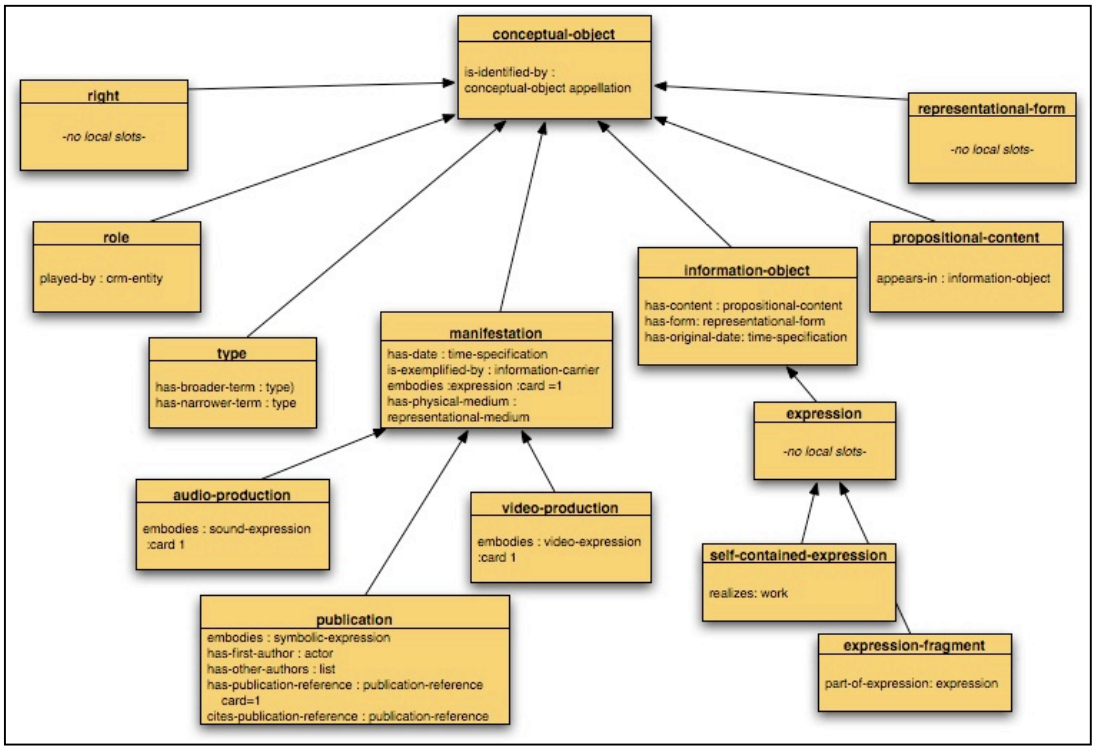

Figure 5-14. The main entities subsumed by conceptual-object

5.4.1 Right, Type and Role

The first two classes come from CIDOC. The first one, `Right`, although likely to be useful in a real-world library of philosophical resources, has not been developed further here.

`Type`, instead, being defined as comprising "arbitrary concepts" and providing a "mechanism for organizing them into a hierarchy", has been extended with definitions which we could not easily fit anywhere else in the ontology. Specifically, we found useful to include here `degree-type` (e.g., “PhD” or “BA”, which is used for further defining educational-events and degree-related references), `gender` (“male” and “female”), `genre` (e.g., a `literary-genre`, such

as “sonnet”, or a `musical-genre` such as “ballad”), `problem-type` (e.g., an “open problem”, or a “dilemma” – we will discuss it more when examining the problem class in section 5.5.3) and `representational-medium` (the types of media which can realize an information object, e.g., “paper-based” or “electronic-based”). The latter class requires a clarification, since it apparently seems to express the same meaning of the class `information-carrier` (a subclass of man-made-object, see fig. 5-5). It is important to remember that an instance of the medium of the `manifestation` (e.g., a paper-based medium, such as "book") is different from an instance of `information-carrier`, which is the specific object carrying information (e.g., "book with ISBN 1234567"). In other words, the last object is one of the items produced as part of a `manifestation`, considered in its actual embodiment in a specific medium (to be precise, in the case of a book, the manifestation would be a `publication`). Thus the medium, under this point of view, results being a type representing classes of physical objects.

We also remodeled the `language` class, which was originally a subclass of type, into a kind of linguistic `representational-form` (see section 5.4.4).

It is worth stressing that the class `type` is quite a tricky one. In fact, since it characterizes entities at a meta-level i.e., because of their being *classes* and not *instances,* `type` should subsume all the other classes in the ontology. However, this is not the case in CIDOC. Here the `type` class is used mainly as a mean to enlarge the ontology with “arbitrary concepts”, such as concepts deriving from other ontologies. As a result `type` becomes the principal way CIDOC employs for achieving ontology-integration. In conclusion, since we too considered this functionality as a useful one, we decided to keep `type` as it is, despite its ontological ambiguity.

Finally, another sibling of `right` and `type` is the class `role`, which is divided into `social-role`, `pedagogical-functional-role` and `textual-structural-role`. Instances of the first kind are "philosopher" or "musician", intended as roles a person can play in his life. Actually, at the moment we do not support any reasoning based on roles, but we just store this information as a handy representation of the commonly accepted ‘professions’ of a thinker (e.g., Hume was an “historian” and a “philosopher”). In future versions of the ontology roles could be attached to activities and problem-areas, so to dynamically infer them even if not explicitly specified (e.g., because Hume has published history-related material, he could have the tag "historian" automatically assigned to him). Some examples of possible role-based reasoning can be found in (Masolo et al., 2004).

The class `pedagogical-functional-role`, instead, abstracts the pedagogical value of a resource (e.g., “introduction”, “exercise”, “summary”). Also in this case not much inferencing results from this class. It is used mostly because it gives the annotator a way of better defining a learning resource (as explained in section 5.3.5.4, during the instantiation of a `document-interpretation` event). In future work we plan to extend this branch of the ontology too.

Finally, `textual-structural-role` gathers the various logical structures we can highlight within a text (e.g., “theme”, “sub-theme”, “vocabulary-lexicon”, “question”, “introduction”, “abstract” etc.). We decided to model all of them as instances, as this is the level of granularity needed in our application (in compliance with the ‘navigation’ centered approach described the final paragraph of section 5.2.1). In particular, these are modeled as instances

because they model individual roles, as opposed to sets of individuals of this nature. For example, we are interested in 'introduction' as an atomic role, not in the class of all introductions. The only exception is `reference`, which has been modeled as class because by doing so we could import all the AKT conceptualizations about references. For example, the `reference` branch includes details about `book-reference` or `workshop-proceedings-reference`, providing also relevant connections to the relevant `publication` subclasses (from the `manifestation` branch of the ontology, cfr. section 5.4.3).

We must remember that the decision of modeling most of these structural roles as instances was dictated by practical reasons. In fact, a different approach could have been to represent them as classes too; by doing so, it would also been possible to benefit from a model that abstracts the chosen *composition of elements* of a presentational structure into a specific class. Such a class would allow the modeling of a certain *sequence* of textual components (e.g., what paragraph/intro/summary comes first etc.) as a separate instance. Instead, as a temporary solution, we 'hardwired' this type of information in the ontology. That is, all text-components have a number indicating their ordering and a slot indicating what `textual-structural-role` they have in the context of an `expression` (see the following section). The decision regarding whether the increasing or decreasing ordering of the components has to be chosen for the final presentational structure, in the case of PhiloSurfical, is hard-wired into the system implementation.

5.4.2 Information Object

`Information-object` is defined in CIDOC as comprising "identifiable immaterial items, such as poems, jokes, data sets, images, text [...] that have an objectively recognizable structure and are documented as single units". This definition was not satisfactory for us, since we needed to separate the *form* of an information object from its *content*. Dolce's definition (Gangemi et al., 2005), instead, reads as follows:

> "[....] a content (information) transferred in any modality is assumed to be equivalent to a kind of social object called information object (IO). Information objects are spatio-temporal reifications of pure (abstract) information as described e.g., in Shannon's communication theory, hence they are assumed to be in time, and realized by some entity."

This definition (and the related IO "design pattern" represented in figure 5-12, whose details will not be entirely reported here for space reasons) seemed much more suitable as a guideline for our modeling choices. Conceiving IOs as reifications of pure informational entities, in fact, would leave space for some reasoning on "copies" of IOs, or different "versions" of them. Therefore, since the original CIDOC information-objects (*design*, *document*, *linguistic object*, *visual item*) were often mixing together form and content, we have substituted them with two new classes, `self-contained-expression` and `expression-fragment`.

The first one refers to self-contained information objects, i.e. IOs that conveys the whole idea of the proposition they represent (e.g., a `text` or a `musical-score`). The second one instead refers to the identifiable parts of an information

object (e.g., a `sentence` or a `word`). Please notice that, as we can see from figure 5-14, both of them are subsumed by a class simply called `expression`: this class does not have any specific function or meaning, but we decide to insert it for facilitating the creation of mappings to the original FRBR entity it derives from.

Moreover, we created two other `conceptual-objects`, `propositional-content` and `representational-form` (described in sections 5.4.5 and 5.4.4), which serve to represent, respectively, the *content* and the *form* of an `information-object` (in a similar fashion as Dolce's 'description' and 'abstract-information-structure' classes). In particular, the content of an `information-object` is represented through an instance of `propositional-content`, while the content of an `expression` is represented through an instance of a subclass of `propositional-content`, called `work`.

In general, these choices let us create a model which is also compliant with the FRBR specifications. In fact, in FRBR a `work` is defined as follows:

> "A work is an abstract entity; there is no single material object one can point to as the work. We recognize the work through individual realizations or expressions of the work, but the work itself exists only in the commonality of content between and among the various expressions of the work."

For example, a `work` is the abstract idea of the Hamlet, which can be realized through a movie, a play or even a musical performance. In other words, `work` reifies the content of famous intellectual creations (e.g., Beethoven's "9th

Symphony", or "The world as will and representation" by Schopenhauer), without giving any specification whatsoever regarding their representational form. However, when instantiating a `work` we can still specify what is supposed to be its original form and genre, through the *has-original-form* and *has-original-genre* properties.

```
(def-instance Tractatus-Logico-Philosophicus work
 ((was-made-by Wittgenstein)
  (has-title "Tractatus Logico-Philosophicus")
  (has-original-genre Treatise)
  (has-original-form written-german)))
```

Being totally abstract entities, `works` can be treated as the content of other information objects. Following FRBR, this is the process by which they are *realized* through various 'expressions' (i.e. versions of the work, such as the original authors' one, a second edition, an interpreted version, a translation etc.). If we consider the "Tractatus-Logico-Philosophicus" as an abstract work, the following formalization is the exemplification of such a modeling:

```
(def-instance Tractatus-Ogden-english-version self-contained-
expression
 ((was-made-by C-K-Ogden)
  (has-title "Tractatus Logico-Philosophicus")
  (has-genre Treatise)
  (has-date 1922)
  (has-form written-english)
  (realizes Tractatus-Logico-Philosophicus)))
```

To sum up, an `expression` is defined as the `information-object` that realizes a `work` through a `representational-form`. The advantage of defining things in such a way, is that it is possible to easily draw connections among the various

expressions of the same intellectual or artistic creation, and to keep track of the differences they have because of the varying forms of representation.

Moreover, the `expression` class is specialized through several subclasses. These are determined by the specific `representational-form` the `expression` is using. Thus we can have `2d-expression`, `3d-expression`, `video-expression`, `sound-expression` and `symbolic-expression`, to which correspond related forms of representation (see figure 5-16 below). Among the symbolic expressions we can find the `linguistic-expression` class (such as `word` or `sentence`), which is particularly important in the modeling of philosophical works. So, for example, a paragraph of the Tractatus can be formalized as follows:

```
(def-instance paragraph-2.13-ogden sentence
 ((part-of-expression Tractatus-Ogden-english-version)
   (has-string-content "In a picture objects have the elements of the
picture    corresponding to them.")
  (has-form written-english)
  (has-number-reference 2.13)))
```

For that regards the other relationships between an `information-object` and its content, let us remind the reader that they ought to be represented using the most appropriate `interpretation` class, as discussed previously in section 5.3.5.4.

In conclusion, in the figure below (fig. 5-15) it is possible to see a summary of the formalizations necessary to represent the various levels of abstractions related to the “Tractatus Logico-Philosophicus”. Please note that in the figure we used a graphical ‘shortcut’: when a relation is attached to a *group* of instances, that is to mean that the relation is repeated over all of those instances. For example, the `work` instance “Tractatus Logico-Philosophicus”

exhibits the property *is-realized-through* three times, corresponding to the three instances of `linguistic-expression` we grouped together.

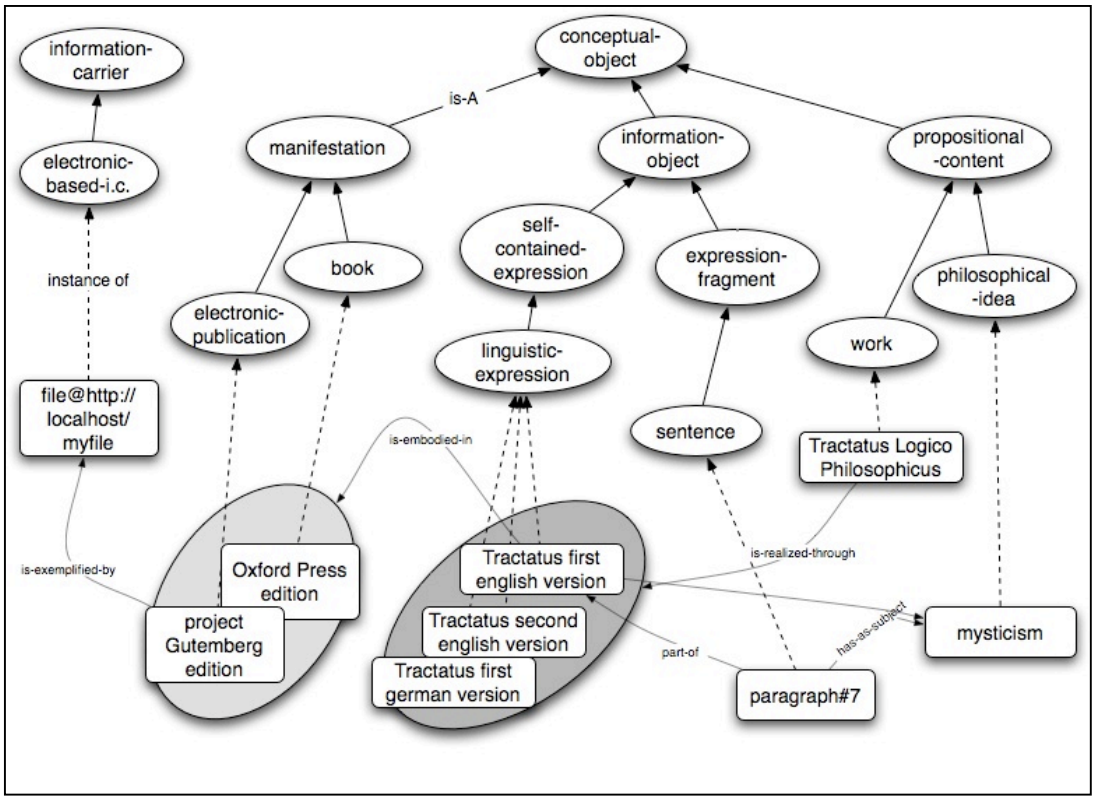

Figure 5-15. The "Tractatus" at various levels of abstraction

5.4.3 Manifestation

This is another class inspired by the FRBR specifications. A `manifestation` is defined there as "the physical embodiment of an expression of a work", through the usage of a physical medium.

Even if in FRBR it is considered an entity reflecting physical form, we acknowledged the fact that it is not a physical entity, but an abstract one (that is, a `conceptual-object`). According to our view, a `manifestation` is the reification of the *embodiment* of an `expression` into a physical entity. As such, it is

representative of the class of the physical entities produced with the same 'publication' process. Therefore, it is an abstraction.

For example, `manifestation` can be an `audio-production`, a `video-production` or a `publication`. This last class, especially, has an important role in the ontology. It refers to linguistic publications (usually on paper-based media, but not necessarily) and it is the junction point with the AKT specifications about documents' types. We have thus imported concepts defining things such as `magazine`, `journal`, and `conference-proceedings` as `manifestations` (in the original AKT model they were instead conceived as tangible-things).

The main relation between manifestation and expression is *embodies*; the relation *is-exemplified-by*, instead, links a `manifestation` to the single `items` that result from the production process. Following the CRM ontology, we decided to represent these `items` by means of a subclass of `information-carrier`. For a clarification of the difference between an `information-carrier` and `physical-medium`, we refer the reader to section 5.3.4

In continuation of the previous section's examples about Wittgenstein's Tractatus, we provide here the formalization of one of its possible manifestations: the electronic publication of the original English version done by the Gutemberg project (GutenbergFoundation, 2008).

```
(def-instance tractatus-project-Gutemberg-transcription publication
 ((is-exemplified-by my-tractatus-file)
  (embodies Tractatus-Ogden-english-version)
  (was-made-by The-Project-Gutenberg-EBook-team)
  (has-date 5-2004)
  (has-title "Tractatus Logico-Philosophicus")
  (has-publication-reference gutemberg-tractatus-reference)
  (has-physical-medium file)))
```

```
(def-instance my-tractatus-file item
 ((has-current-owner michele-pasin)
  (exemplifies tractatus-project-Gutemberg-transcription)
           (has-uri    "http://michelepasin/Desktop/
TractatusLogicoPhilosophicus.txt")))
```

5.4.4 Form of the representation

The `representational-form` is the 'language' the `information-object` uses to convey its meaning. For example, "written-english", "musical-sound" or "cartoon-animations". The hierarchy we are using has at the top layer `symbolic-form`, the class of forms that represents through symbol systems (e.g., languages), and `iconic-form`, the class of forms which represent through a more direct imitation the original signified object (from the greek "eikon", which means "likeness, image, portrait"). This specification does not pretend to be exhaustive, but mainly functional to our needs. In figure 5-16 we can see a graphical version of the tree, together with the relevant information objects.

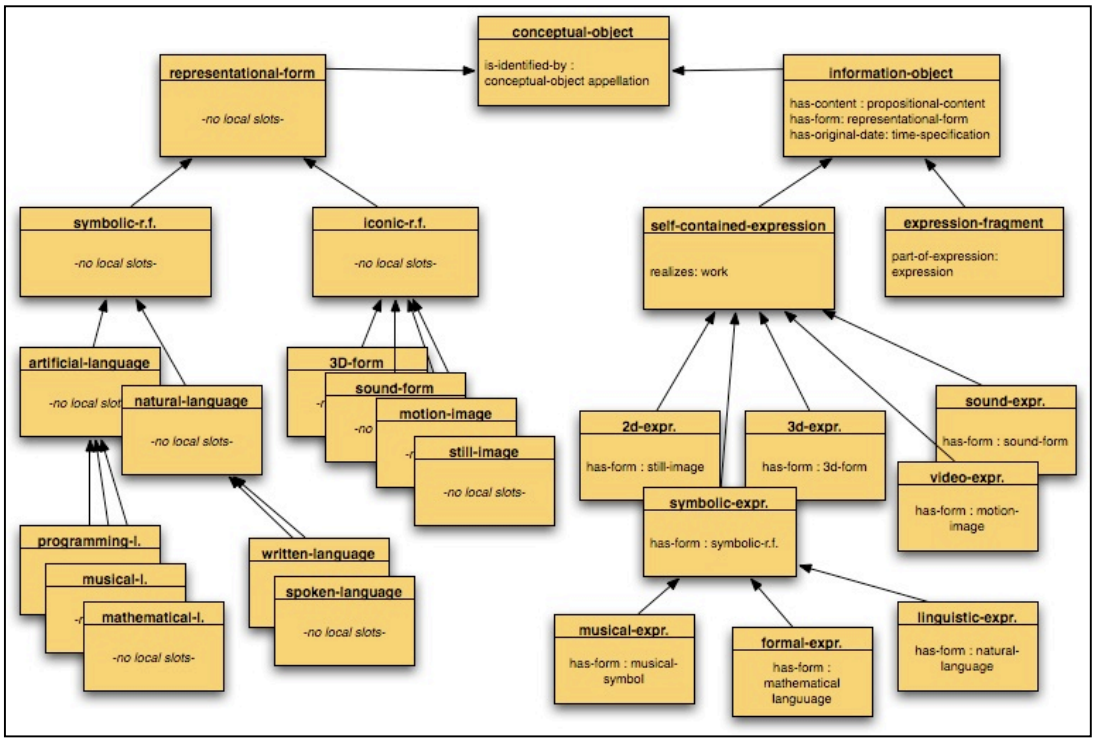

Figure 5-16. Forms of representation and related IOs

5.4.5 Content of the representation

The content of the representation, or `propositional-content`, is essentially what the representation is about. For example, a `philosophical-idea` (such as the "problem of representation", or the "theory of descriptions"), a `drama` (e.g., the “Hamlet”) or an `algorithm` (e.g., the “quicksorting algorithm”).

Modeling propositional contents has proved to us as being a particularly difficult task, also because of the scarce literature available on the subject (see chapter 3). In our opinion, this is caused primarily by the abstract nature of the entities to model, which makes them inherently ambiguous and hard to formalize. Nonetheless, the practical needs linked to the development of the PhiloSurfical’s navigation mechanisms gave us the motivation for creating a comprehensive formal theory, which is also suited for some specific reasoning tasks.

Following an analysis done by Mizoguchi (Mizoguchi, 2004), we decided to have two kinds of propositions: `design-proposition` and `product-proposition`. As he clarifies:

> “The former works as specification of the production of something. The latter itself is the product. For example, a piece of music composed is a specification of the music sound produced by the music player. Procedure is specification of the valid sequence of actions. An execution of the procedure generates a result (product). Novel cannot be specification of anything because it is already a product.”

According to this model, we can have `design-proposition` such as `procedure` (e.g., an algorithm), `piece-of-music` (e.g., the sequence of notes we have to reproduce, so to create the music), `drama` (a sequence of actions), `symbol` (meaning its graphical design) and `specification` (i.e. the technical design of an electronic device). Among the `product-proposition` we can have instead `novel` (i.e. a story), `poem` (as an artistic product) or `painting` (the content of the painting we see).

For that regards the philosophy-specific contents, we decided to model all of them under the class `philosophical-idea`, which is another `product-proposition`. As these entities constitute one of the prominent contributions of our work, we are providing a detailed description of them in the following section.

Finally, let us mention that, as anticipated in section 5.4.2, in order to maintain compatibility with FRBR we also have a further `propositional-content`, which is `work`. This is an entity referring to any self-contained and identifiable *content* of an `information-object`. As such, this class is the result of a conceptualization which is orthogonal to the one used in Mizoguchi's analysis. Thus, many of the classes just mentioned (such as `drama` and `poem`) can also be rightfully considered as subclasses of `work`. Since an extensive classification of `works` types would be beyond the scope of our work, we focused only on the philosophical ideas presented in the next section.

5.5 Philosophical propositional contents

Domain experts and generic practitioners normally use a number of meta-level concepts to classify and organize the propositional contents typical of the philosophical world. So, for example, they talk about a specific *approach*, a *theory*, a *school of thought*, a *doctrine*, a *conception*, etc. It is remarkable how easy it is for people to understand each other, although the meanings of such terms are not explicitly stated, or at least, not shared among the speakers.

Given these premises, where to start in order to formalize the types of abstract entities we talk about in philosophy? This seems a really puzzling question, and probably, to someone, totally nonsensical. Such a slippery and strange domain, in fact, appears to challenge any stable formalization, and defeat any meaning-agreement process.

On the other hand, as previously mentioned, modern days phenomena such as the incredible growth of available information or the increasing need for interoperability standards call for a solution which, although inevitably partial and non-definitive, can bring many more advantages than no solution at all. As claimed by the authors of a recent project for the indexing of the Stanford Encyclopedia of Philosophy (Niepert et al., 2007):

> "while no single ontology can possibly capture the full richness and interrelatedness of philosophical ideas, we are operating on the principle that having (at least) one ontology is better than none"

In the light of this simple but important reflection, we have attempted to model commonly used philosophical concepts without taking any particular philosophical position, that is, for what is possible, trying to remain "outside" specific philosophical stances. Not doing so would have caused a multiplication of ontologies and definitions, each of them reflecting the world according to a single thinker (or to a thinker's view, since thinkers sometimes produce more than one philosophy in their lifetime).

Our approach, which can be related to a "constructivist" epistemology (Bachelard, 1938), sees every philosophy as a system of interrelated conceptual entities which make sense of the world. From this perspective, we can say that such entities are all abstract (non-physical), since they are "what we use" to refer to the physical world. The main consequence of this perspective is that even a common concept like "fire", which would be normally instantiated as a physical entity, in our model becomes an instance of a `concept` (which is possibly related to a physical entity).

In fact, the notion of fire, as any other notion, is socially constructed (Vygotsky, 1978) and often explicitly defined by a viewpoint (e.g., the "Newtonian physics", or the "philosophy of Heraclitus"). The fact that a generic agent happens to be more or less explicitly aware of this viewpoint, in all its aspects and subtleties, constitutes another issue and does not disprove its existence. For us, the problem to tackle is the individuation of the types of non-physical-objects playing a role in the construction of viewpoints, and, more broadly, having a recognizable function in the process of interaction and succession of viewpoints within the whole history of thought.

In conclusion of this overview, we show in figure 5-17 the main classes in the philosophical-idea branch.

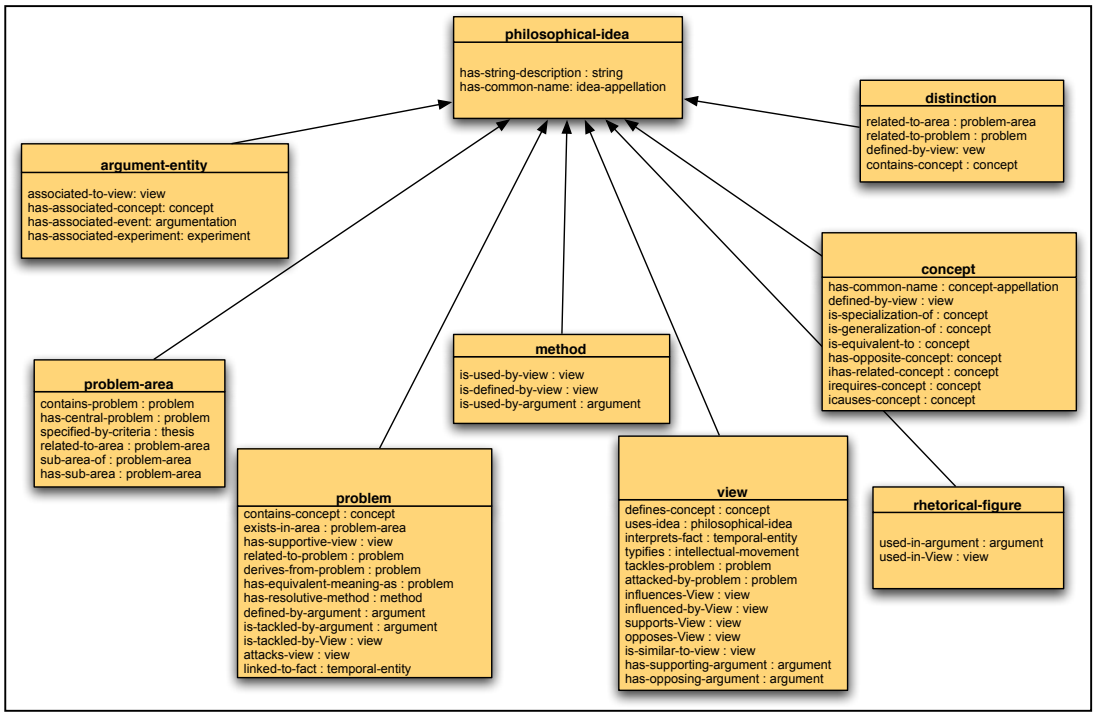

Figure 5-17. The main classes of the philosophical-idea branch

5.5.1 Argument-entity

With the `argument-entity` class we decided to group together two sets of related classes: `argument` and `argument-part` (see figure 5-18).

The first one is the reification of the `argumentation` class (which is a subtype of `event`), as it "freezes" an actual argumentation between two or more thinkers into an abstract idea (i.e. an entity outside space and time). In previous versions of the ontology, we also named it ‘argumentative-knot’. In fact it refers to famous focal points of the philosophical argumentation, where all the main argumentative threads converge and meet. These knots have usually origin in one author, for then being re-called and re-used (maybe in different domains or

for different purposes) by other authors. So, for example, we can have the "third-man argument" of Plato, the Cartesian "cogito-ergo-sum" or the Kantian "transcendental deduction". An important property of this class is *uses-method*, whose range is `argumentative-method` (a subclass of `abstract-method`), because through it we can specify, for example, a `deductive-argument`, an `inductive-argument` or an `abductive-argument`.

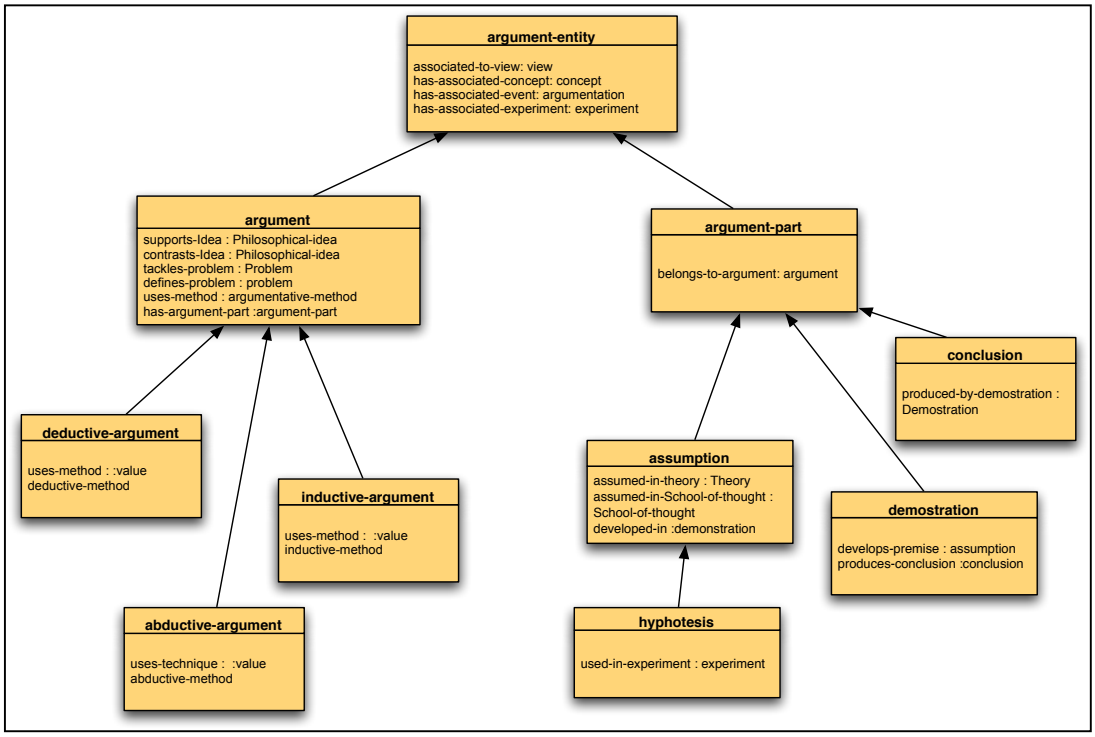

Figure 5-18. Argument and argument-part

The second subclass of `argument-entity` is instead `argument-part`, which precisely serves to map out all the argumentative steps of a standpoint. For the moment, we only defined `assumption`, `demonstration`, `conclusion` and `hypothesis` (a subclass of `assumption` specifically referring to argumentations based on experimental evidence). It is important to note that this is only a simplified classification of the entities that can possibly build up an argument. In the future, other work from the argumentation community (Kirschner et al.,

2003) could be brought in, to represent at a finer granularity the different argument structures.

For example, in the Tractatus we can find the following argument:

```
(def-instance wittgenstein-argument-for-substance argument
 ((is-used-by-view first-witt-philosophy)
  (tackles-problem substance-problem)
  (supports-idea thing independence-of-things)
  (uses-method deductive-method)))
```

5.5.2 Problem-Area

In order to give an account of the distinctive features of fields of study, we decided to use as a starting point a problem-centred approach. This means that we tended to see the activity of philosophers as essentially an ongoing process of specifying and giving solutions to problems. Consequently, we consider any recognized area of study, of whatever type or dimensions, as a `problem-area`. In its simplest version, a problem-area is composed by a set of problems linked by different relational schemas, but in general, tying around a main theme. This theme, in our ontology, can be represented through a `problem` (has-central-problem property) or thanks to a `thesis` functioning as a criteria (*has-criteria* property). For example, “psychology”, when treated as a `problem-area`, can gather problems tied to the “mind-definition” problem, to the problem of “relating human behavior to brain activities”, or to the thesis that "brain and mind can be investigated with the methods of natural sciences".

Other features of problem areas are that they can be *related-to* each other (e.g., “mathematics” and “philosophy of mathematics”) and that they can be organized into simple hierarchies (e.g., “internet-ethics” is a sub-area of “ethics”).

However, we realized soon that "psychology" has a role and significance in our world that goes beyond a mere problem-area. In a similar fashion, "ethics" or "cognitive science" would not be properly characterized only as instances of `problem-area`, for they also refer to theories or methods which have become intrinsically related to the definition of the area.
Moreover, if we consider the history of thought, the topic and description of problem-areas have always been subject of many debates: different views aspire at having the ultimate vision about what the central issues to look at are, or the right methods to take. In this respect, `problem-areas` are not very different from other ideas that can be *defined* by multiple `views`. For example, we can just consider how different was the sense given to “philosophy of language” by the first philosophy of Wittgenstein and the second one.

In order to catch these subtle differences, we defined the class `field-of-study` as a `problem-area` that has been socially and historically recognized as separate from the others (and from being a mere agglomerate of problems). In the ontology, this is reflected by the fact that a `field-of-study` is not just specified by a criteria, but is *defined-by* a `view`. It is also characterized by the fact that it collects not only problems, but also ways to solve or tackle them (i.e. `theories` and `methods`). The distinguishing properties are therefore *defined-by-view*, *has-exemplar-theory* and *has-methodology*.

```
(def-class Field-of-study (Problem-area)
 ((defined-by-view :type view)
  (has-exemplar-theory :type theory)
  (has-methodology :type method)))
```

As an example, we show a possible formalization of an "old-fashioned" `field-of-study`, “phrenology”.

```
(def-instance phrenology field-of-study
 ((has-referred-author Franz-Gall)
  (defined-by-view phrenology-theory)
    (contains-problem what-is-personality what-is-character relation-personality-skull)
  (has-criteria skull-shape-determines-personality-thesis)
  (sub-area-of psychology)
  (related-to-area craniometry physiognomy)
  (has-methodology phrenological-analysis)))
```

Finally, a last tricky issue regarding fields of study must be addressed. This does not emerge when treating relatively isolated entities such as “phrenology”, but it clearly is an issue if we consider, say, “physics”. In our everyday language, and also in the organization of academic programs, we usually refer to “physics”, “psychology” or “philosophy of mind” as *generic* fields of study. What this means, is not really clear. In fact, when we delve into them (or even more, if we ask for clarifications to a practitioner), we discover quickly that there are *many* “physics”, “psychologies” and “philosophies”, at least as many as the views defining them. From our ontological perspective, these would all be separate instance-candidates of the `field-of-study` class. However, we also need to represent the fact that they are all part of a more generic (and probably *emptier*, for that regards its meaning) `field-of-study`.

Our solution to this problem consists in the creation of a `generic-field-of-study` class, which has no defining view but the views defining the specific fields of study that are claimed to be part of it. In other words, we are formalizing the fact that `generic-fields-of-study` such as “physics” or “philosophy: can be defined only extensionally. So we have the following OCML rule:

```
(def-rule generic-field-rule
 (defined-by-view ?GF ?V) if  (generic-field-of-study ?GF)
                              (has-sub-area ?GF ?F)
                              (defined-by-view ?F ?V)
```

In the formula, the variables ?GF, ?V and ?F refer respectively to `generic-field-of-study`, `view` and `field-of-study`. Therefore, doing so we can maintain the interoperability between specific thinkers’ definitions of classic problem areas, and the generic but useful ways to refer to them. In figure 5-19 we give a graphical overview of this modeling pattern, highlighting the important relationships among the classes involved. Please note that also in this figure we used a graphical ‘shortcut’: when a relation is attached to a *group* of instances, that is to mean that the relation is repeated over all of those instances. For example, the `generic-field-of-study` instance “physics” exhibits the property *has-sub-area* three times, corresponding to the three instances of `field-of-study` we grouped together.

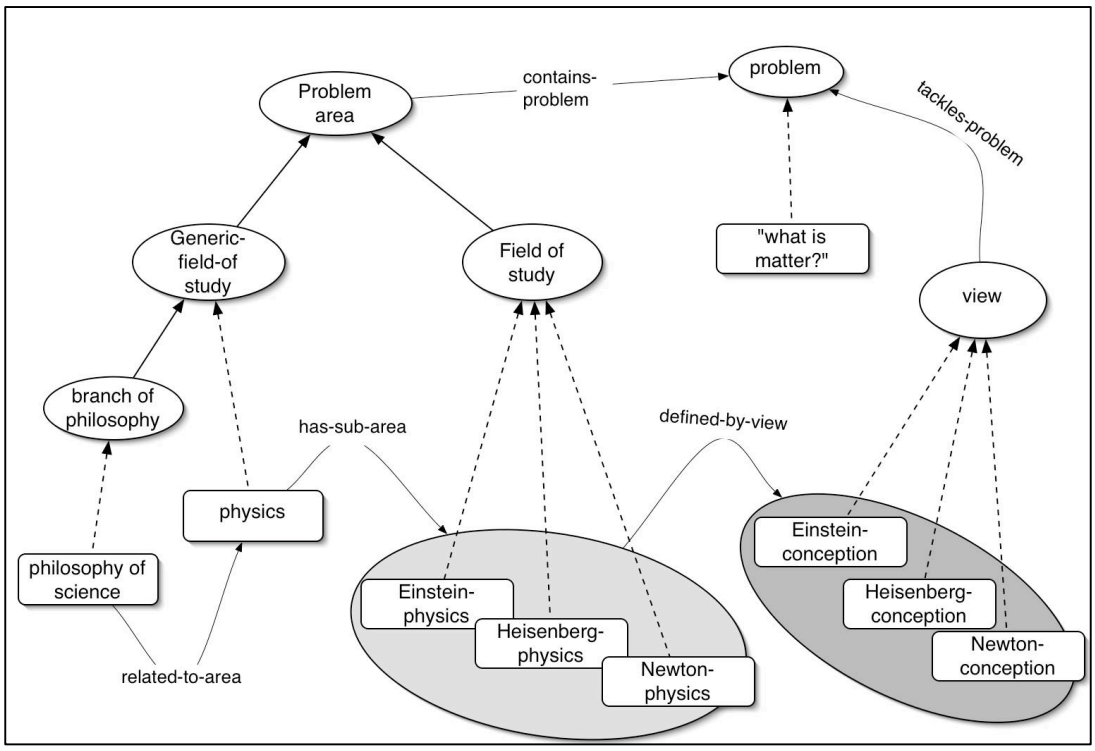

Figure 5-19. Generic and specific fields of study

5.5.3 Problem

The `problem` class represents a very central notion in philosophy, since it is usually the point of departure of any investigation (which often culminates with the creation of a `view`). Examples at hand are many: we talk about the “mind-body” problem, the “alienation” problem or the "problem of the universals". A key feature we can easily recognize is that a problem is always framed within a larger context which gives a more precise connotation to it. So, for example, Marx considered the “alienation problem” to be rooted in “economy”, while Searle treats the “mind-body problem” within the “philosophy of mind”. Therefore, the `problem` exists within a `problem-area`. Moreover, the context which makes us understand a problem is given also by the set of assumptions that justify its existence. Or better, by the `views` and `arguments` that define it (and, conversely, try to solve it). The remaining properties of `problem`, as shown

below, relate them to other problems or to the `view` and `arguments` that tackle them.

A special role is held by the property *has-problem-type*, which can have value “open-problem” (meaning a problem which does not have any solution), “multilemma” (a problem having or allowing multiple solutions), “dilemma” (a problem allowing two solutions only, but neither of the two being satisfactory) and “paradox” (a problem whose solutions seem equally plausible, but when considered together generate a contradiction). Essentially, these concepts describe a `problem` from the viewpoint of the number of solutions it has. We have modelled them as instances of the class `problem-type` (which is not in the `philosophical-idea` branch, but is instead a subclass of CIDOC’s `type`), since they do not appear to be “essential” for the definition of a problem, but just accidentally related to the existence of any solution. In other words, a `definitional-problem` (see below) will always maintain its structure, regardless of being an “open-problem” (i.e. having no solutions) or a “multilemma” (i.e. having various solutions).

From the analysis of the literature we thought it was useful also to provide a classification of problems based on their ‘morphology’. That is, on their external structure, which can be sometimes related to their content, but is usually independent from it. In total, we identified 6 ‘morphological types’ of problems:

1) the `existence-problem` has usually the form "Does X exist?"; specializations are `existence-as-concrete-problem` ("Is X concrete/real?") and `existence-as-abstract-problem` ("Is X abstract?")

2) the `definitional-problem` has usually the form "What is X?". Specializations are `definitional-problem-essence` ("what are the characteristic traits X has?"), `definitional-problem-attribute` ("what are the attributes X has?") and `composition-problem` ("What is X composed of?")

3) the `functional-problem` has usually the form "What is the function of X?"; the only specialization is `purpose-problem` ("What is the purpose of X?")

4) the `relational-problem` has usually the form "What is the relation between X and Y?"; specializations are `dependence-problem` ("Are X and Y dependent?"), `dependence-cause-problem` ("Is X the cause of Y?"), `dependence-effect-problem` ("Is X the effect of Y?"), `independence-problem` ("Is X independent from Y?"), `equality-problem` ("Is X equal to Y?") and `difference-problem` ("Is X different from Y?").

5) the `modality-problem` is a problem about the degree of certainty X is likely to happen (or not). Specializations are `necessity-problem` ("is X necessary?"), `possibility-problem` ("is X possible?"), `contingency-problem` ("is X contingent?") and `impossibility-problem` ("is X impossible?")

6) the `factual-problem` has the form "how, in what way does X happen, or manifests itself?".

At the time of writing, we are instantiating these problem templates by filling the empty spaces in the question with instances of `concept`. For example:

```
(def-instance what-is-virtue definitional-problem
 ((contains-concept virtue)
  (has-problem-type multilemma)
  (exists-in-area ethics)
  (related-to-problem what-is-value)
  (is-tackled-by-View Plato-philosophy Aristotle-philosophy stoic-
philosophy)
  (linked-to-fact death-of-socrates)))
```

A much more interesting solution would be instead letting any instance of `philosophical-idea` be filling those spaces. This would result in a powerful reification mechanism: e.g., we could define a problem about the relation between two other problems. Moreover, we are also investigating how to use these structures for producing inferences (e.g., from a `relational-problem`, we can create a path which links to the `definitional-problems` of the concepts related). These and other issues (such as how to classify problems according to their 'contents' e.g., "moral problem" or "epistemological problem") will be investigated in future research.

5.5.4 Method

Various ontologies introduce a class named 'procedure', with reference to any *sequence-like specification*. Similarly, a heuristic or `method` in philosophy is essentially defined as a series of steps, that is, the 'path to take' from the problem, in order to reach a solution. Depending on whether the method

suggests a practical activity, or an intellectual one, we classified instances as belonging to `abstract-method` or `practical-method` (see fig. 5-20)

The main types of `abstract-method` are `logical-mathematical-method`, `rule-of-inference` and `argumentative-method`. The first one subsumes `algorithm` and comprises instances such as “the quick-sorting algorithm”, Wittgenstein’s “truth-table method” or Leibniz’s “infinitesimal calculus”. The second class refers to rules that are used to justify the steps in a formal proof of the validity of a more complex argument. For example, we can have “modus ponens”, “hypothetical syllogism”, “conjunction”, “double-negation elimination” etc. The class `fallacy`, instead, refers to invalid argumentative steps that may appear convincing at first glance because they closely resemble legitimate patterns of reasoning. For example, fallacies can be the “illicit major”, “affirming the consequent”, “denying the antecedent”, “affirming the alternative” etc. Finally, the class `argumentative-method` categorizes famous and well-established argumentation styles, such as “deductive argument”, “argumentum a fortiori”, “argumentum ad hominem”, “argumentum ad populum” etc.

The other branch of method, `practical-method` is divided into `scientific-method` and `life-praxis`. With the first class we refer to any structured method to investigate reality, in a “scientific” manner (e.g., so to produce and test some explanatory hypotheses). Examples can be “Bacon’s scientific method” or “Galileo’s scientific method”. The second class instead is a method of life conduct, such as the epicurean’s “ataraxia” (e.g., a description of conduct to follow in order to achieve the tranquillity of the soul) or a practice of meditation in eastern philosophies.

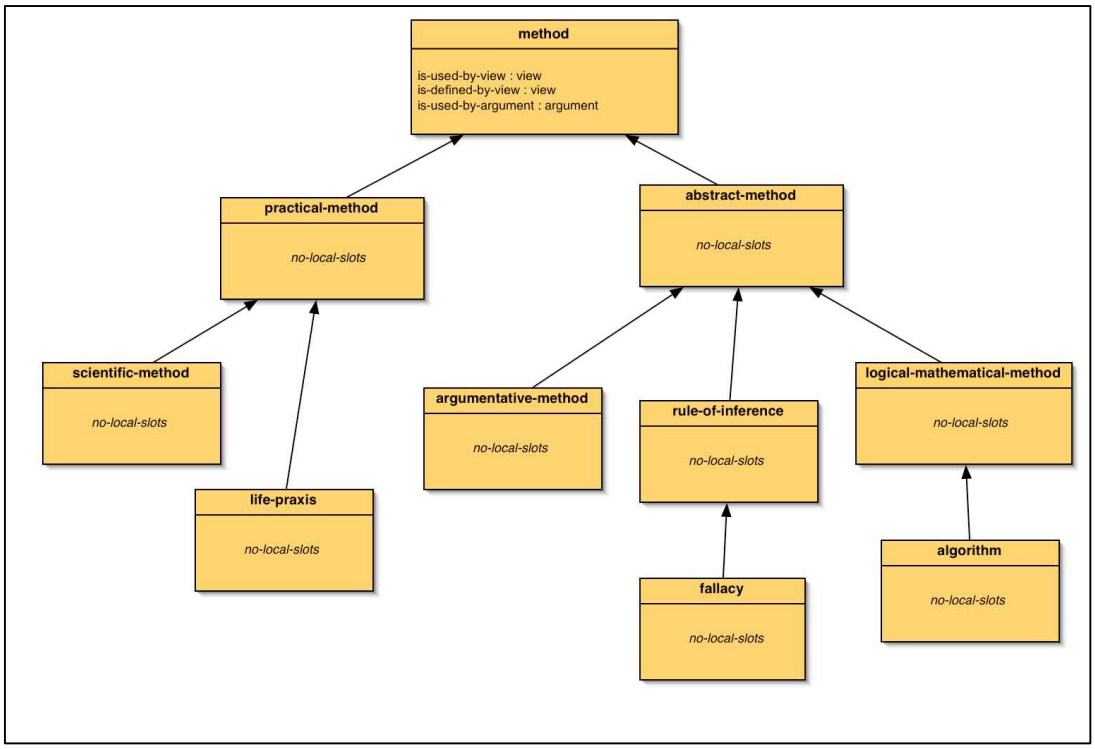

Figure 5-20. The branch of the ontology departing from 'method'

5.5.5 View

This is a generic class referring to propositions expressing a viewpoint, that is, propositions picturing a perspective on the world in the form of more or less structured interpretations of things and events. Examples of `view` are "solipsism", "theory of evolution by natural selection", "philosophy of Plato" or "a name has a meaning only in the context of a proposition" (i.e. Frege's context principle).

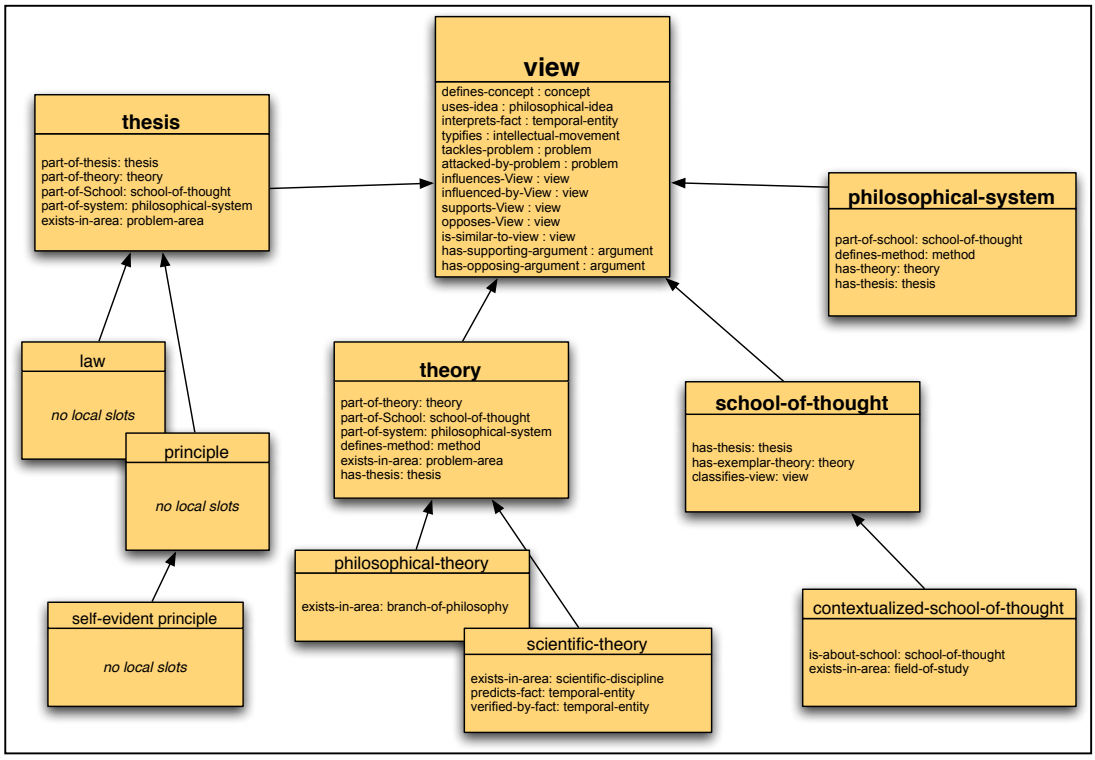

Fig. 5-21. The view-types

Because of their "categorical" attitude, views usually *define* `concepts` and, in general, create the context for the definition of other meanings too (e.g., `problem-areas`, `problems`, `methods` etc.). A number of properties connect views to the other philosophical-ideas: views can *use* other ideas, *tackle* problems, *influence* and *support/contrast* each other, *be-supported* by arguments. Most of them seemed to reflect quite well the common sense understanding of philosophy, so we will not treat them one by one here.

However, the feature we want to highlight here is how views can have varying granularities. From our analysis of the literature, we identified four possible kinds of view: `school-of-thought`, `theory`, `philosophical-system` and `thesis` (see fig. 5-21). The main differences among them depend on the degree of generality they exhibit and the level of complexity they have. In figure 5-22 we

can see a small example including different views and some relations they entertain with each other. In the following four paragraphs we will examine them one by one. Please note that in the figure below we used a graphical 'shortcut' for representing the *conceives* relation: when this relation is attached to a *group* of instances, that is just to mean that the relation is repeated over all of those instances.

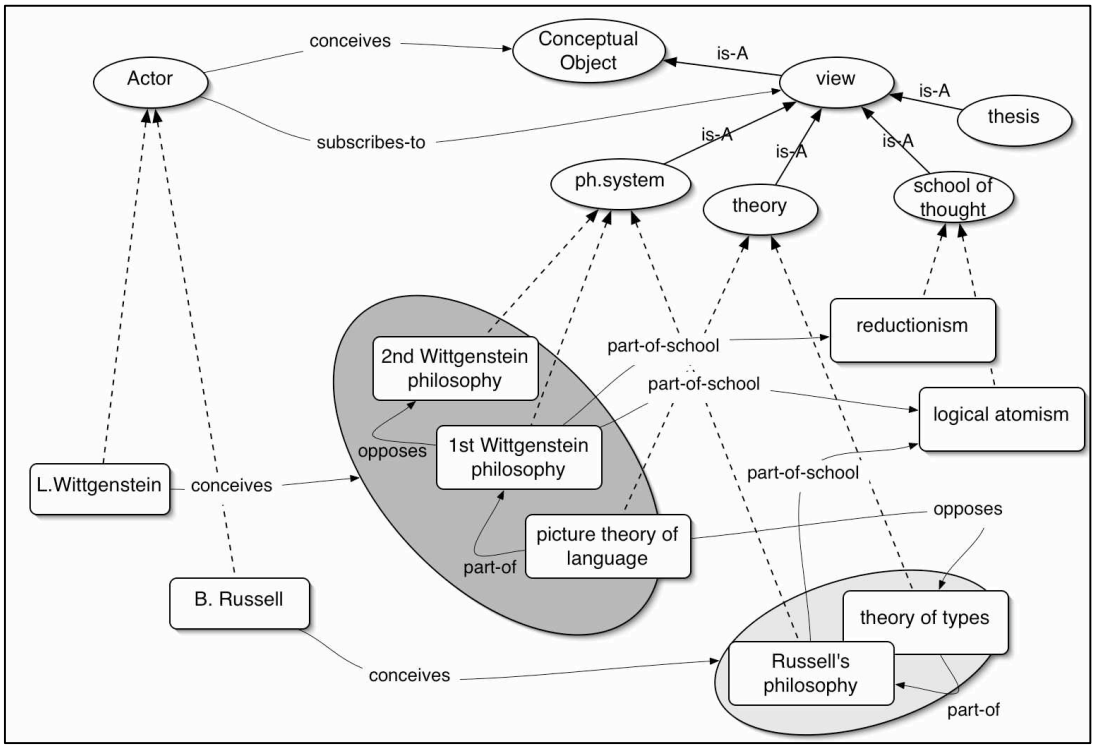

Figure 5-22. The view-types instantiation

5.5.5.1 Thesis

A `thesis` is the least structured `view`, as sometimes it consists only of a standpoint in the form of a statement (i.e. an assertion). So, for example, in the context of Wittgenstein's picture theory of language, a thesis can be the "independence of the state of things" (as recognized by Stenius, one of his

commentators), which is expressed in the original text by a series of paragraphs:

```
(def-instance independence-state-of-things thesis
 ((defines-concept state-of-things independence)
  (part-of-system wittgenstein-first-philosophy)
  (part-of-theory picture-theory-of-language)
    (has-string-description "State of things are independent of one
another")))
```

The local properties of `thesis` are the *part-of* relations linking it to the other subclasses of `view`. Most of its properties are therefore inherited.

However, not all theses have the same status: two subclasses, `law` and `principle`, refer respectively to theses with vast predictive power, especially in scientific areas (e.g., the “law of universal gravitation”), and to theses that play a fundamental role within a view, usually a philosophical one (e.g., a principle in medical ethics). Finally, if the principle is not demonstrable, but self-evident, it becomes a `self-evident-principle`. For example:

```
(def-instance principle-of-contradiction self-evident-principle
 ((defines-concept truth thought)
  (part-of-system aristotle-philosophy)
  (exists-in-area logic)))
   (has-string-description "One cannot say of something that it is and
that it is not in the same respect and at the same time")
  (appears-in Metaphysics-book-IV)))
```

5.5.5.2 Theory

With the class `theory` we refer to a systemic conceptual construction with a coherent and organic architecture. A `theory` explains a specific phenomenon (or a set of phenomena) and typically answers to an already existing `problem`.

Examples can be Darwin's "theory of evolution" or Quine's "verification theory". The first one is a `scientific-theory`, while the second is a `philosophical-theory`. The main difference between the two is that the last one is not necessarily hypothetical and therefore it does not need experimental verification (although it can be provided with it).

The local slots of theory define the following properties: *part-of-theory* expresses the situation where theories are composed by other theories (e.g., Plato's "theory of metempsychosis", which is contained and dependent on the "theory of anamnesis" [..]); *part-of-school* can be used to express that a theory is classified as part of a school of thought (e.g., when we say that the "picture theory of language" is a kind of "reductionism") ; finally *part-of-system* links a theory to an author's philosophy (e.g., the "theory of eternal recurrence" is part of "Nietzsche's philosophy"). Moreover, theories can *define-methods* (e.g., Wittgenstein's "picture theory of language" defines the "truth tables method"), they exist within a specific `problem-area` (*exists-in-area*) and usually within them we can easily identify a set of `theses` (*has-thesis*).

A `philosophical-theory` does not differ much in its formalization from its direct super-class, apart from the fact of having range `branch-of-philosophy` on the property *exists-in-area*. The same property, instead, would have value `scientific-area` in the case of a `scientific-theory`. Moreover, a `scientific-theory` can be further defined as having a more peculiar relationship to the facts it tries to explain, as it is usually required to be verified (proved) by them, and to be able to predict them too.

5.5.5.3 Philosophical-system

A `philosophical-system` might appear as a theory, at first sight, but it differs from it essentially for its generality and breadth. That is, because it spans over various `problem-area`, while a `theory` is usually confined to one `problem-area` only. As a consequence, `theories` are usually *part-of* `philosophical systems`. We can therefore define a system as the set of a person's views (which singularly taken, approach problems coming from different problem areas) which are consistently connected to each other, in such a way to form a unity.
In a way, this class refers to what is normally called the 'philosophy' of a thinker. So, for example, we can have the "Epicurean philosophy", the "Kantian philosophy" or "Hume's philosophy". We must remember, however, that this class does not correspond to the mere sum of an author's theories: in fact, thinkers might produce more than one independent system, during their lifetime (e.g., the first philosophy of Wittgenstein, as opposed to the second one).

Finally, we also recognized how a `philosophical-system` (although being inherently related to various `problem-area`) is often considered as representative of a `school-of -thought` (which, as explained in the next section, is instead usually related to a specific `problem-area`). In other words, it makes sense to say "the philosophy of Hume is skepticism", even if, in such a case, we implicitly refer to only certain aspects of his philosophy (i.e. his epistemology). As this is a normal practice for scholars, we reckoned important also for our ontology users to be able to quickly classify philosophies as part of a school, without having to specify the relevant theories or thesis. In order to

prevent wrong generalizations (e.g., inferring that all the theories of Hume are “skeptical”) we use a set of purposely built rules. Finally, other rules also guarantee the consistency between philosophical-systems and the theories composing them (e.g., if a theory defines a method or a concept, the philosophy comprising the theory is also considered to define them).

5.5.5.4 School of thought

This class refers to the set of theory-types, or generic standpoints, which in the history of thought have acquired a particular significance and, seemingly, a life on their own. They correspond to widely known conceptions, or standardized intellectual trends that hint at typical ways to answer a problem (or a set of problems). Examples are “pacifism”, “animism”, “expansionism”, "empiricism" or "monism".

Sometimes they can be so abstract (as in the case of "monism") that they do not imply anymore a link to a specific problem or area, but refer only to the “formal features” of the view they classify. For example, in the case of “monism”, what is implied is just "a view that admits only one principle as fundamental".

A `school-of-thought`, compared to the other views, is not as formalized and specific as a `theory`, and not as broad and systematic as a `philosophical-system`. Accordingly, in our model we decided to limit its contents to `theses`. Because of this “generic” flavour, we often perceive the meaning of schools as being vague and abstract (e.g., when trying to specify what is a “rationalism”).

On the contrary, we noticed that this is actually not the case when we refer to 1) their “instantiation” within a problem area (e.g., the “ethical rationalism”) and 2)

their specific "expression" within an author's philosophy (e.g., the “rationalism of Kant”). These last two examples seemed to us quite important, therefore we attempted to give an account of them also in the ontology.

According to our analysis, the first case (“ethical rationalism”) relates to the fact that schools of thought often have a "contextualized" version. That is, they assume a different and more specific meaning when associated to a specific `problem-area`. For example, “rationalism”, can be found in “epistemology”, in “ethics”, in “metaphysics” or in “philosophy of religion”. The interesting phenomenon, in this case, is that the contextualized versions do not always have much in common and sometimes are even surprisingly unrelated. E.g., in the case of the meaning of “cognitivism” in “psychology” and in “meta-ethics”.

Therefore, in order to keep separated the meaning of generic schools of thought from their localized ones, we introduced the class `contextualized-school-of-thought`, which has the additional slot *exists-in-area* with range `field-of-study`.

```
(def-class Contextualized-school-of-thought (School-of-thought)
 ((is-about-school :type School-of-thought)
  (exists-in-area :type Field-of-study)))
```

Instead, for that regards the second case (the “rationalism of Kant”), we concluded that it refers to the fact that schools of thought are normally used as ‘classifiers’ of other views. We showed in an earlier paragraph how this relation is already captured by the *part-of-school* property of `theory` and `philosophical-system`. In a similar fashion, we created also the slot *has-exemplar-theory*, which

refers to the `theory` that inspired the `school-of-thought`, and is likely to help in understanding its original sense.

5.5.6 Rhetorical figure

With this class we aimed at grouping figures of speech or statements embodying some rhetoric value; usually these objects of discourse are used for emphasis, for clarity or as a device in the philosophical argumentation. Many of these entities could have been classified as subtypes of `argument-part`, since in most cases they play that role. However, since often they assume a singular significance in the history of thought (i.e. the “myth of the cave”) we decided to represent them separately, so that they could be treated (and re-used) as independent entities.

We have identified three types of `rhetorical-figure`: `metaphor`, which subsumes `myth` and `analogy`, `maxime-motto`, and `thought-experiment`. Examples of the first type is the aforementioned “myth of the cave”, or Hegel’s metaphor of the “night, in which all cows are black” (used in the argument contra Schelling). `Mottos` refer instead to famous and exemplar statements or expressions philosophers used to sum up their position. For example, Descates’ “cogito ergo sum”, Hobbes’ “homo homini lupus” or the ancient maxime “ex nihilo nihil fit”. Finally, `thought-experiments` are mind-simulations used to prove a point: among them, we can remember Searle’s Chinese-room thought-experiment (used to attack strong AI), Putnam’s twin-earth thought-experiment (used to support “semantic externalism”) or David Chalmer’s “unconscious zombies” thought-experiment (used to attack “physicalism”).

5.5.7 Concept

A concept is an atomic element (i.e. not further decomposed) in the ontology. Instances of concepts can be "ego", "evolution" or "god". In determining what is a concept, we are not interested in its cognitive and linguistic features (i.e. the fact that it carries one propositional content, or that it is expressible through one or two words), but mostly in its functional role within the economy of a philosophy or a theory. That is, we tend to see a concept as an element which is defined by a view as primitive, and which is in a net of relations with other concepts.

According to a 'philosophy of minimum commitment', we have chosen not to formalize specific philosophical concepts as classes, but to provide means to create alternative interrelated nets of instances which could resemble (and could be exported as) a small taxonomy. Thus, the creation of a network of interrelated concepts relies totally on the annotator. We expect people to organize the knowledge associated with an author's conception very differently, according to user needs, background and interests.

A concept can be linked to other concepts through various relations: specialization and generalization (*is-specialization-of* and *is-generalization-of* properties); similarity of meaning (*is-equivalent-to*), e.g., for the concepts "inexpressible" and "ineffable" in Wittgenstein; antinomic contrast (*has-opposite-concept*), e.g., when two concepts are part of a dichotomy; generic semantic closeness (*has-related-concept*), e.g., when they concur in explaining the same

phenomena; notional dependency (*requires-concept*), e.g., with concepts such as "buy" and "pay"; causation (*causes-concept*), e.g., with concepts such as "to kill" and "to die". For example, the wittgenstenian concept of "picture" could be defined as follows:

```
(def-instance PICTURE-BY-FIRST-WITTGENSTEIN concept
 ((has-common-name PICTURE)
  (defined-by-view FIRST-WITTGENSTEIN-PHILOSOPHY)
  (is-specialization-of FACT-BY-FIRST-WITTGENSTEIN)
  (is-generalization-of LOGICAL-PICTURE-BY-FIRST-WITTGENSTEIN)
  (has-similar-meaning-as PICTURE-BY-HERTZ)
  (is-in-contrast-with )
  (is-in-relation-with ISOMORPHISM-BY-FIRST-WITTGENSTEIN FORM-OF-
REPRESENTATION-BY-FIRST-WITTGENSTEIN REPRESENTING-RELATION-BY-FIRST-
WITTGENSTEIN))))
```

Finally, the *has-common-name* property (whose range is `idea-appellation`) is used for separating the `concept` object from the name used to identify it (e.g., "picture" in english, "immagine" in italian, "image" in french, etc.). As already mentioned CIDOC provides a useful facility to detach entities from their names, that is the `appellation` class (it is located in the `persistent-item` branch of the ontology). By instantiating this class, for example, we can define multiple names for the same place, or for the same person. Analogously, we added also an `idea-appellation` class in order to support the separation an idea-object from its names.

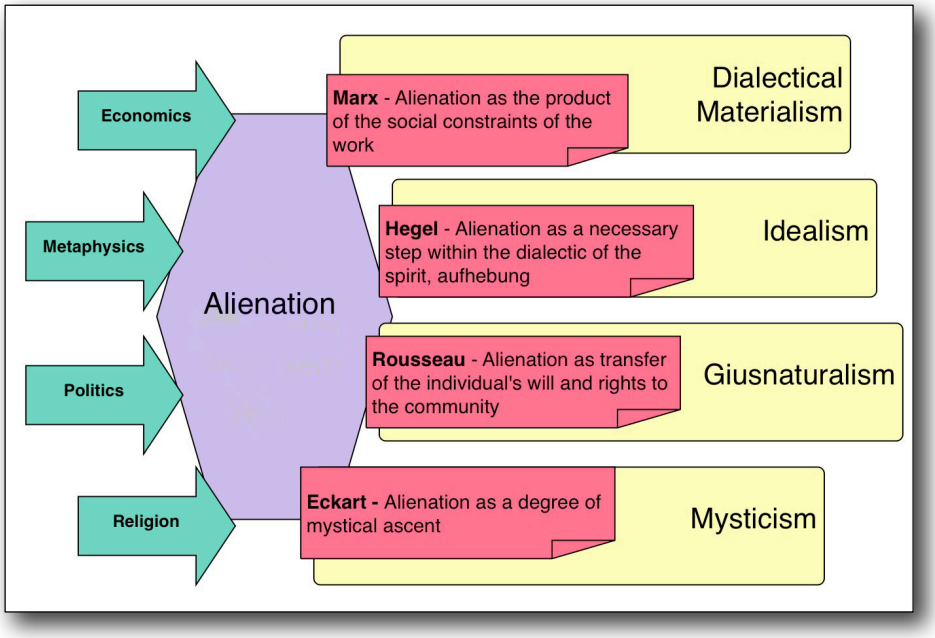

Figure 5-23. Concept of "alienation" with four different view-contexts

This turned out to be a quite handy feature, because often there are no explicit properties stating the relationships between two instances of `concept`, other than the fact that they have the same name. In figure 5-23 we can see an example of how the word "alienation" (which is an `idea-appellation` instance) could be referring to four different concepts. Each of them, in fact, is defined by a different view, categorized by different school-of-thought and typical of different problem-areas.

5.5.8 Distinction

We have a `distinction` when two ideas or more stand out as particularly meaningful in their opposition. That is, the specificity of their sense is obtained or clarified by their being different, but complementary. For example, "Hume's distinction between truth of reason and matters of fact", "Aristotle's distinction between essence and accident", or "Frege's between extension and intension".

Together, the two concepts fill up a whole, with respect to a specific domain of reference e.g., “epistemological” (regarding the limits of human knowledge) or “ontological” (regarding the structure of being). A distinction can have an arbitrary number of concepts (e.g., “Aristotle's four types of causes”), but when comprising two concepts only, is also called `dichotomy`. For example:

```
(def-instance hume-fork dichotomy
 ((has-referred-author david-hume)
  (related-to-area epistemology)
  (related-to-problem what-can-we-know)
  (defined-by-view hume-philosophy)
  (containts-concept relation-of-ideas matter-of-fact)))
```

5.6 Summary

In this chapter we have presented an ontology aimed at the representation of the most important dimensions in the philosophical domain. The model is quite vast, comprising more than 300 classes that deal with entities such as *people, events, ideas* and *documents*.

The ontology builds on various existing models and standards currently used in knowledge representation. In particular, since in many cases the required formalizations were not available, we also introduced a large number of new classes.

The contribution of this ontology is thus twofold: on the one hand, the ontology provides a novel framework that *integrates* several well-known other ontologies, which aims to facilitate the construction of philosophical knowledge-bases. On the other hand, the ontology contains various new classes and modeling

patterns addressing several subtle features of philosophy which, to our knowledge, have never been represented before.

In the next chapter we proceed with a detailed examination of the PhiloSurfical tool. That is, we show the reader how the ontology just presented was populated and used within an educational scenario. In particular, by discussing a number of concrete examples, we aim at showing how the ontological representations can support the creation of various types of navigation mechanisms, which we call *learning narratives*.