Scholarly publishing and Linked Data: describing roles, statuses, temporal and contextual extents

Silvio Peroni
Dept. of Computer Science
University of Bologna (Italy)
essepuntato@cs.unibo.it

David Shotton
Dept. of Zoology
University of Oxford (UK)
david.shotton@zoo.ox.ac.uk

Fabio Vitali
Dept. of Computer Science
University of Bologna (Italy)
fabio@cs.unibo.it

ABSTRACT

Recently, several ontologies have been introduced for semantic publishing. However, scholarly publishing, like other real-world domains, needs to be described also in terms of precise temporal durations and the particular contexts in which the relevant processes take place. For instance, a document changes status during its publication process, e.g., from "draft" to "submitted" to "under review" to "accepted for publication", and so on. Similarly, one's roles may change with time: one's affiliation with an academic institution or one's role as a journal editor are likely to change over time. Existing well-known ontologies used to describe individuals and bibliographic entities in the Linked Data are currently not able to model situations of temporary or context-dependent possession (e.g., the holding of a status or of a role). In this paper, we address this issue by introducing two ontologies for semantic publishing, the *Publishing Roles Ontology* and the *Publishing Status Ontology*, that define the roles of people and the statuses of documents in the scholarly publishing domain.

Categories and Subject Descriptors

I.2.4 [**Artificial Intelligence**]: Knowledge Representation Formalisms and Methods—*Representation languages*

General Terms

Theory

Keywords

Linked Data, OWL, SPAR ontologies, context-dependancy, ontology pattern, semantic publishing, time-dependancy

1. INTRODUCTION

In recent years, interesting relationships between Semantic Web and scholarly publishing have arisen and have recently converged into one precise research discipline, *Semantic Publishing*. Semantic publishing is thus the use of Web and Semantic Web technologies to enhance a published document, such as a journal article, so as to enrich its meaning, facilitate its automatic discovery, enable linking and data integration with other semantically related articles, provide actionable access to its data, etc. [17, 18]. One of the most important research area in semantic publishing is the development of semantic models (i.e., ontologies) that enable the publication of related semantic data into the Linked Data arena according to appropriate vocabularies and requirements of scholarly authoring and publishing.

We have recently developed the suite of *Semantic Publishing and Referencing* (*SPAR*) ontologies[1] to describe all aspects of the scholarly publishing domain as comprehensive machine-readable RDF metadata. For this purpose, we looked at many existing ontologies to reuse for our needs, but most of what we found was deemed insufficient for addressing all the requirements we collected to properly describe the publishing domain.

The most serious obstacle we encountered relates to how the description of personal roles and of the statuses of documents needs to vary in time and according to changing contexts, a crucial requirement in the publishing domain. The contextual and time-dependent aspect of such roles and statuses need to be handled formally by ontological models. Unfortunately, many existing ontologies are unable to describe situations in which a status or a role is held only for a particular period, or only with respect to special contexts.

In this paper we introduce and illustrate PRO (the *Publishing Roles Ontology*) and PSO (the *Publishing Status* Ontology), two of the SPAR ontologies we developed for the description of the roles of people involved in publishing and the various statuses of documents within the publishing process. These ontologies have been developed on an ontological pattern called *time-indexed value in context* (*TVC*), also illustrated in this paper. This pattern allows one to relate an *entity* (e.g., a person) to a particular *value or property* (e.g., a role held by that person) in a particular *period* and according to a particular *context* – e.g. the social entity (e.g. an organisation) or physical entity (e.g. a book) to which person's role refers to. TVC integrates smoothly with existing ontologies, and can be used as an intermediate model when converting domain-specific assertions (e.g., concerning a person's roles) between ontologies.

The basic assumptions of TVC are grounded on specific and well-known works in ontology engineering – e.g. [2, 7, 12, 16, 20] – which were taken into account with the aim of

I-SEMANTICS 2012, 8th Int. Conf. on Semantic Systems Sept. 5-7, 2012, Graz, Austria

[1]SPAR, the Semantic Publishing And Referencing ontologies: http://purl.org/spar.

providing a pragmatic and concrete application on solving the problem of how to describe in the Linked Data environment, and in a particularly compact and expressive way, the situation in which an entity has a specific property value within a specific time and according to a specific context and domain.

The rest of the paper is organised as follows. In Section 2 we illustrate design approaches commonly used in the Linked Data world for modelling situations in the publishing domain that change over time, namely agents' roles and document statuses. In Section 3, we introduce a general pattern used as a starting point for the development of the TVC pattern that properly addressing these situations (Section 4). In Section 5 we present our two SPAR ontologies, PRO and PSO, which were developed by strictly following the prescription given by TVC. Finally, in Section 6, we draw out some conclusions for our work.

2. EXISTING ONTOLOGIES

When modelling a domain, we often need to describe scenarios in which an *entity* has some *value* only within a specific temporal interval and/or contextual (e.g. social, cultural, physical) environment, and a different one (or none at all) otherwise. For instance, in the publishing domain, we may want to describe the *status* of a document at any given moment (e.g., *draft*, *under review*, *accepted*, *published*), the institution of which an author is a member, or the role held by people in the publishing process. All these scenarios involve an *entity*, a *value*, and in particular a *time* and a *context* within which the entity is associated to the value.

Most ontologies are unable to model such scenarios effectively, for different reasons. Three techniques in particular have been used in attempts to address this modelling issue - *class subsumptions*, *property links* and *inter-linked classes*, but each falls short in some aspect.

2.1 Subsumptions and time-dependent classes

To clarify this design technique and the issues that arise from it, let us consider the agent/role relations as described in the Portal Ontology[2] of the AKT Reference Ontology. This ontology defines the class *Student* as a person (class *portal:Person*) who studies at (property *portal:studies- at*) some institution (class *portal:EducationalOrganization*), as follows (in Turtle):

```
portal:Student a owl:Class
  ; rdfs:subClassOf [ owl:intersectionOf (
    portal:Person
    [ a owl:Restriction
      ; owl:onProperty portal:studies-at
      ; owl:someValuesFrom
        portal:EducationalOrganization ] ) ] .
```

The fact of being a person is *time-independent* – Silvio Peroni is a living person while Kurt Vonnegut is a dead person, but we still describe both as persons. On the contrary, the fact of being a student is strictly *time-dependent* – Silvio Peroni was recently a graduate student, but is one no longer.

Thus the subsumption model shows a clear design problem: a class having time-dependent characteristics, namely *portal:Student*, has been placed in the same *is-a* hierarchy (i.e., defined through a *rdfs:subClassOf* relation) as a class having time-independent characteristics, namely *portal:Person*. As suggested in [7], which calls them respectively *anti-rigid* and*rigid* classes, we believe they should be part of two separated hierarchies, and we conclude that descriptions of time-dependant entities cannot be satisfactorily achieved using plain subsumption.

2.2 Property links

A solution that properly takes into account anti-rigid characteristics is the use of a specific property for defining each time-dependent *value* that an entity has (e.g., the role of a person), while continuing to express the *entity* itself as an individual of a particular class (e.g. a document). Many ontologies for describing bibliographic resources, such as DC-Terms [6] and BIBO [5], use object properties to model this, linking the document to the persons who are its authors by the use of a specific property (namely, *dcterms:creator* and *bibo:authorList*, respectively), as shown here:

```
# Using DCTerms
:paper1 a dcterms:BibliographicResource
  ; dcterms:creator :peroni , :vitali .
:paper2 a dcterms:BibliographicResource
  ; dcterms:creator
    :peroni , :motta , :daquin .
# Using BIBO
:paper1 a bibo:Article
  ; bibo:authorList ( :peroni , :vitali ) .
:paper2 a bibo:Article
  ; bibo:authorList
    ( :peroni , :motta , :daquin ) .
```

This approach has at least two problems. The first is that we need as many properties as there are roles. Thus if the requirements are not fully known in advance or change with time, the TBox of the ontology will require extensions to include new properties on a case-by-case basis, requiring continuous maintenance and increasing the risk of inconsistencies. For instance, the number of roles in the publishing domain (author, editor, publisher, etc.) has been continually increasing (see, for example, the list of MARC relators[3] dated 7th December 2010), and the current technological and cultural evolution will surely lead to the creation of new ones (for instance, any time now, that of *linked-data manager*).

Alternatively, one could use data properties rather than object properties. For instance, the W3C specification of the ontology for describing vCard objects in RDF [11] prescribes the use of a (very general) data property, *vcard:role*, for an individual's roles. But while this allows easy extensions to the ontology by adding arbitrary literals to represent new roles, it also lacks a clear and well-defined vocabulary for existing ones, causing potential ambiguities (e.g., with the literals "Graduate student" and "Ph.D. student" being used to refer to the same role but being formally different within the model).

A second problem, that affects the scenario regardless of whether we use object properties or data properties, is to discern the context in which an *entity-value* association holds. For instance, consider an author having different institutional affiliations in the context of different publications (e.g., because he/she moved from one to the other). Using

[2]Portal Ontology: http://www.aktors.org/ontology/portal. The prefix *portal* refers to entities defined in it.

[3]MARC Code List for Relators: http://www.loc.gov/marc/relators/relaterm.html.

the *Semantic Web Conference Ontology*[4] [14], we can define affiliations for *:peroni* as follows:

```
:peroni swrc:affiliation :cs-unibo . # Paper 1
:cs-unibo a foaf:Organization
  ; dcterms:description
    "CS Dept., University of Bologna" .
:peroni swrc:affiliation :kmi . # Paper 2
:kmi a foaf:Organization
  ; dcterms:description
    "KMi, Open University" .
```

This specification, although straightforward, does not differentiate between associations. Namely, in OWL author *:peroni*, is associated indifferently to both *:cs-unibo* and *:kmi*, and it is not possible to determine the affiliation of an author within the context of a particular paper – e.g., "give me the institutional affiliation of the person *:peroni* as author of paper 2" (although in [13] Masolo *et al.* propose an approach to deal with this issue by using *qua-individuals*). Finally, the approach proposed in CIDOC CRM allows one to use the meta-property *P14.1 in the role of* [4] (a sub-property of property *P14 carried out by*) so as to specify the role that an agent has in the context of a particular event (such as being affiliated to an institution) through an instance of the class *E55 Type*. However, the official RDFS ontology of CIDOC CRM[5] does not implement any meta-property, and in reality RDF lacks the expressive power needed to define meta-properties.

2.3 Using inter-linked classes

Another way to address the time-dependent association of entities to values is to consider both as classes (with no declared or inferable subsumption), and to link them through object properties. For instance, the *Semantic Web Conference Ontology* [14] implements this through two classes, *foaf:Person* and *swc:Role*, and the property *swc:holdsRole* linking them.

The extensibility of the ontology is thus guaranteed, reducing the possibility of undesirable inferential side effects. In fact, adding new roles simply involves adding new individuals to the class *swc:Role,* requiring no modification to the TBox. However, as before, this solution is still unable to describe the context or time frame in which the person holds the particular role. For instance, Silvio Peroni was an undergraduate student at the University of Bologna between 2005 and 2008, a graduate student at the same university between 2009 and 2012, and also an intern at the University of Oxford in 2010.

The SWC ontology, used together with FOAF [3], gives only a partial description of this, as in the following:

```
:undergraduateStudent a swc:Role .
:graduateStudent a swc:Role .
:intern a swc:Role .
:peroni a foaf:Person
  ; swc:holdsRole :undergraduateStudent
    , :graduateStudent , :intern .
```

This description cannot answer the question "Was *:peroni* a graduate student in 2008?", because it lacks information about time. By adding this using a specific model, such as the Time ontology[6] [10], we manage to describe it as follows:

```
:atTime a owl:ObjectPropery
  ; rdfs:domain swc:Role
  ; rdfs:range time:TemporalEntity .
:undergraduateStudent :atTime [
  a time:TemporalEntity
  ; time:hasBeginning [ a time:Instant
    ; time:inDateTime [
      ; time:year "2005" ] ]
  ; time:hasEnd [ a time:Instant
    ; time:inDateTime [
      ; time:year "2008" ] ] ] . ...
```

The problem here is that time-related information is associated to the roles, rather than to the person holding them. Being an undergraduate student is associated to the 2005-2008 time interval. This of course creates problems once we add another person with the same role, since it will become impossible to reuse the same role unless he/she also happens to have been an undergraduate in the same years. Things becomes even more complicated if we need also to describe the social or cultural *context* within which the agent-role relation holds, for example, by specifying in which institution *:peroni* was an intern on a given date. It would force one to multiply the instances of role by the number of contexts and time intervals for which the different roles are relevant.

3. TIME-INDEXED SITUATIONS

Some ontological patterns [1, 9, 15] have been developed that partially address these issues. For example, through the *time-indexed situation* pattern[7], shown in Fig. 1[8], it becomes possible to link a subject to a time-dependent description of a situation[9].

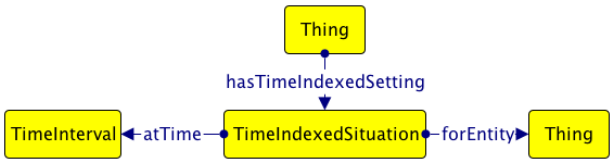

Figure 1: A graphical representation of the *time-indexed situation* ontological pattern.

Using this pattern, the scenario presented in Section 2.3 can be defined as follows:

[4]The Semantic Web Conference Ontology: http://data.semanticweb.org/ns/swc/ontology. The prefixes *swc* and *swrc* refer to entities defined in it.

[5]RDFS ontology of CIDOC CRM: http://www.cidoc-crm.org/rdfs/cidoc-crm-english-label.

[6]The Time Ontology: http://www.w3.org/2006/time. The prefix *time* refers to entities defined in it.

[7]Time-indexed situation pattern: http://ontologydesignpatterns.org/cp/owl/time indexedsituation.owl. The prefixes *tisit*, *sit* and *ti* refer to entities defined in it.

[8]This and all the following graphical representations of ontologies are drawn using Graffoo, the Graphical Framework for OWL Ontologies, available at http://www.essepuntato.it/graffoo. Yellow rectangles represent classes (solid border) and restrictions (dotted border), green parallelograms represent datatypes, arrows starting out of a filled circle refer to object property definitions, arrows starting out of an open circle refer to data property definitions, while other arrows represent assertions between resources.

[9]In this context, a *situation* is defined as a view on a set of entities. It can be seen as a "relational context", reifying a relation.

```
# University of Bologna
:unibo a foaf:Organization .
# University of Oxford
:oxac a foaf:Organization .
:peroni tisit:hasTimeIndexedSetting
  :peroniAsGraduateStudentInUnibo
  , :peroniAsIndernInOxAc .
:peroniAsGraduateStudentInUnibo
  a tisit:TimeIndexedSituation
  ; tisit:atTime [ a ti:TimeInterval
    ; ti:hasIntervalStartDate
      "2009"^^xsd:gYear ]
  ; tisit:forEntity :unibo , :graduateStudent .
:peroniAsInternInOxAc a
  tisit:TimeIndexedSituation
  ; tisit:atTime [ a ti:TimeInterval
    ; ti:hasIntervalStartDate
      "2010-06"^^xsd:gYearMonth
    ; ti:hasIntervalEndDate
      "2010-12"^^xsd:gYearMonth ]
  ; tisit:forEntity :oxac , :intern .
```

Although this pattern correctly describes our scenario, it is still too abstract, both as a model and in terms of its terminology. In particular, the *tisit:forEntity* object property provides little or no guidance as to the interpretation of the relation with the entity. In fact, for instance in *:peroniAsInternInOxAc*, the way the various entities were involved in the situation is not clear, and we do not know what are the relations linking *:peroni*, *:oxac* and *:intern*. Given that, we could come to different conclusions just permuting the entities involved, e.g., a) that the person *:peroni* related at a particular period to the institution *:oxac* that has the role of *:intern* (quite incorrect), or b) that the person *:peroni* worked at a particular period with somebody having the role of *:intern* within the institution *:oxac* (still incorrect), or c) that the person *:peroni* has the role *:intern* within the institution *:oxac* during a particular period (as intended, finally). Thus, using this pattern we would need to apply additional (unwarranted and "smart") steps to infer the correct interpretation of the situation.

To resolve this ambiguity, we decided to extend this pattern so as to specify the relations held by individuals in a situation involving a time-indexed value in a context.

4. TIME-INDEXED VALUE IN CONTEXT

What emerges from the preceding discussion is the needs for a model to describe time-dependant and contextualised entities. In particular, we identified four different things involved in these kinds of scenarios:

1. the *entity* having some value, e.g. a person or a document possessing a role or a status;
2. the *value* had by someone, e.g. a role or a status;
3. the *time period* during which the entity *has* that value, e.g. from April 2008 to September 2008;
4. the particular *context* that characterises the act of *having that value*, e.g. being a member of an institution or the editor of a particular journal.

In Section 3 we introduced a pattern that is able to describe this scenario at an abstract level but missing a mechanism to describe the reciprocal relations of the entities involved. Using that as a starting point, we now wish to define a new ontological pattern called *time-indexed value in context* (*TVC*), summarised in Fig. 2 and available as an OWL ontology[10].

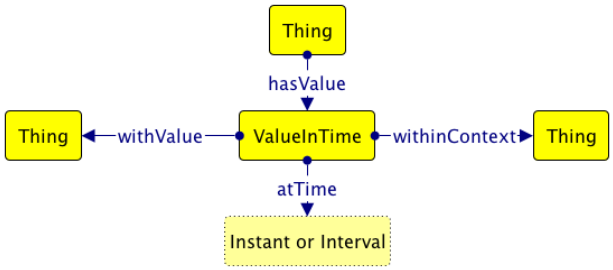

Figure 2: The Graffoo diagram of the *time-indexed value in context* ontological pattern.

This pattern is composed by two different classes and four object properties:

- the class *ValueInTime* is a particular kind of time-indexed situation (i.e. a subclass of *tisit:TimeIndexedSituation*) as shown in Section 3 that represents a hub linking the *entity* having a particular value, the *value* itself and the *temporal and contextual extents* on which the entity-value relationship depends;
- the object property *hasValue* (sub-property of *tisit:hasTimeIndexedSetting*) links an entity (e.g., a Person) to a particular *ValueInTime* situation;
- the object property *withValue* (sub-property of *tisit:forEntity*) gives the value held by the entity taking part inthe situation;
- the classes *Instant* and *Interval* are used, respectively, to specify the particular temporal instant or time period in which the situation takes place. This is done through the object property *atTime*, which is not defined as a sub-property of *tisit:atTime* since it can be used to describe instants in addition to intervals;
- the object property *withinContext* (sub-property of *tisit:forEntity*) links to the specific social, cultural or physical context within which the fact of the entity-having the valueis relevant.

Using the TVC ontology, the *:peroniAsInternInOxAc* excerpt introduced in the previous section can be re-written as follows:

```
:peroni tvc:hasValue :peroniAsInternInOxAC .
:peroniAsInternInOxAc
  a tvc:ValueInTime
  ; tvc:atTime [ a ti:TimeInterval
    ; ti:hasIntervalStartDate
      "2010-06"^^xsd:gYearMonth
    ; ti:hasIntervalEndDate
      "2010-12"^^xsd:gYearMonth ]
  ; tvc:withinContext :oxac
  ; tvc:withValue :intern .
```

[10] *Time-indexed value in context* pattern: http://www.essepuntato.it/2012/04/tvc. The prefix *tvc* refers to entities defined in it.

where *:intern* is the value (i.e. the role) held by *:peroni* during that particular time period, and *:oxac* is the context. In the following sections we expand these concepts, introducing use cases and explaining the benefits of TVC.

4.1 Querying a TVC-based model via SPARQL

In principle, the TVC pattern allows a large number of SPARQL 1.1 queries [8] to return intuitively correct answers. In this section, we discuss as examples three queries of increasing difficulty. For instance, we can ask for all the values had by a person (e.g., the roles held by *:peroni*):

```
SELECT DISTINCT ?value WHERE {
  :peroni a foaf:Person
    ; tvc:hasValue/tvc:withValue ?value }
```

This query can be refined to consider, for instance, only those values that are defined in a particular context, e.g. the University of Bologna (entity *:unibo*):

```
SELECT DISTINCT ?value WHERE {
  :peroni a foaf:Person
    ; tvc:hasValue [ a tvc:ValueInTime
      ; tvc:withValue ?value
      ; tvc:withinContext :unibo ] }
```

This will return both the undergraduate and the graduate student roles of *:peroni*. We can further filter the previous results to return just those roles that are applicable at a particular date such as 24 August 2010:

```
SELECT DISTINCT ?value WHERE {
  :peroni a foaf:Person
    ; tvc:hasValue [ a tvc:ValueInTime
      ; tvc:withValue ?value
      ; tvc:withinContext :unibo
      ; tvc:atTime [ a ti:TimeInterval
        ; ti:hasIntervalStartDate ?start
        ; ti:hasIntervalEndDate ?end ] ]
  FILTER(
    xsd:dateTime(?start)
      <= "2010-08-24T00:00:00Z" &&
    xsd:dateTime(?end) > "2010-08-25T00:00:00Z"
  ) }
```

This will return just the role of graduate student. If the condition *tvc:withinContext :unibo* was omitted, the query would return both *:peroni*'s role as a graduate student at the University of Bologna, and his concurrent role on that date as an intern at the University of Oxford. More complicated and domain-specific queries are introduced in Section 5.1.

4.2 Reusing external classes as values

It is possible, by means of the meta-modelling features of OWL 2 (i.e., OWL punning), to define classes of external ontologies as objects of *tvc:withValue* assertions. In this way, we can use them interchangeably either as instances, when we want to directly associate them with some entity, or as classes when we want to understand hierarchical relationships between them. In addition to opening up the TVC ontology for reuse, this may be very useful for inferring new data for specific categories, even when, in a query, we use their more abstract generalisations (i.e., superclasses).

Consider for example the following dataset defined according to TVC and including entities from the Portal Ontology:

```
:person1 tvc:hasValue [ a tvc:ValueInTime
  ; tvc:withValue portal:Affiliated-Person ] .
:person2 tvc:hasValue [ a tvc:ValueInTime
  ; tvc:withValue portal:Student ] .
:person3 tvc:hasValue [ a tvc:ValueInTime
  ; tvc:withValue portal: PhD-Student ] .
# Statements defined in the Portal Ontology
portal:Student rdfs:subClassOf
  portal:Affiliated-Person .
portal:PhD-Student rdfs:subClassOf
  portal:Student .
```

In this way, it is possible to query the dataset through SPARQL, asking for all the people affiliated with the University of Bologna (the entity *:unibo*), independently from the roles they may hold as a student, a Ph.D. student, or other subclass of *portal:Affiliated-Person*:

```
SELECT DISTINCT ?person WHERE {
  ?person tvc:hasValue [ a tvc:ValueInTime
    ; tvc:withValue ?aff
    ; tvc:withinContext :unibo ] .
  { SELECT ?aff WHERE {
    { ?aff a owl:Class .
      FILTER(?aff = portal:Affiliated-Person) }
    UNION
    { ?aff rdfs:subClassOf+
      portal:Affiliated-Person } } } }
```

TVC makes it possible and useful to reuse specific parts of other ontologies describing categories in the form of classes, thus taking advantages of the OWL 2 punning.

4.3 Constructing second-order inferences

Of course, it is sometimes desirable to reuse ontologies that specify categories (e.g., roles) through properties rather than classes, as introduced in Section 2.2. Consider, for example, BIBO [5], that associates agent roles with documents through particular sub-properties (e.g., *bibo:translator*, *bibo:director*, *bibo:editor*) of the general property *dcterms:contributor*. Using the BIBO ontology with TVC, these object properties can be used as objects of *tvc:withValue* assertions, by means of OWL 2 punning. Moreover, it is possible to construct second-order inferences using the objects of *tvc:withValue* assertions as properties:

```
CONSTRUCT { ?doc ?property ?person } WHERE {
  ?person a foaf:Person
    ; tvc:hasValue [ a tvc:ValueInTime
      ; tvc:withValue ?property
      ; tvc:withinContext ?doc ]
  { SELECT ?property WHERE {
    { ?property a owl:ObjectProperty .
     FILTER(?aff = dcterms:contributor) } UNION
    { ?property rdfs:subPropertyOf+
      dcterms:contributor } } } }
```

Through a model that combines TVC and an ontology defining categories as property links, such as BIBO, it becomes feasible to infer second-order logical statements. More generally, TVC can be used as an intermediate model for the conversion of entity-value relationships from one ontology into another, independent of the particular design technique used by each ontology (i.e., class subsumptions, property links, inter-linked classes or n-ary relationships).

5. USE OF TVC IN SPAR ONTOLOGIES

The pattern TVC is a fundamental ingredient of the *Publishing Roles Ontology* (*PRO*) and the *Publishing Status Ontology* (*PSO*) that we developed to describe roles and statuses in the publishing domain. TVC made their design process simple and quick. In the following subsections we introduce these ontologies, showing how they are integrated with the TVC pattern.

5.1 Identifying publishing roles with PRO

The ability to define publishing roles in SPAR was crucial for the completeness of this suite of ontologies. The problems associated with the adoption of external ontologies to handle this particular requirement has been discussed above. None of them were fully able to satisfy the modelling requirements imposed by SPAR, particularly need for ease of *extendibility* and for the simultaneous representation of *time periods* and *contexts*.

Using TVC as the basis, we implemented *PRO*, the *Publishing Roles Ontology*[11]. This ontology, shown in Fig. 3, permits characterization of the roles of agents – people, corporate bodies and computational agents – in the publication process. It permits one to specify the role an agent has in relation to a particular bibliographic entity (as author, editor, reviewer, etc.) or to a specific institution (as publisher, librarian, etc.), and the period during which each role is held.

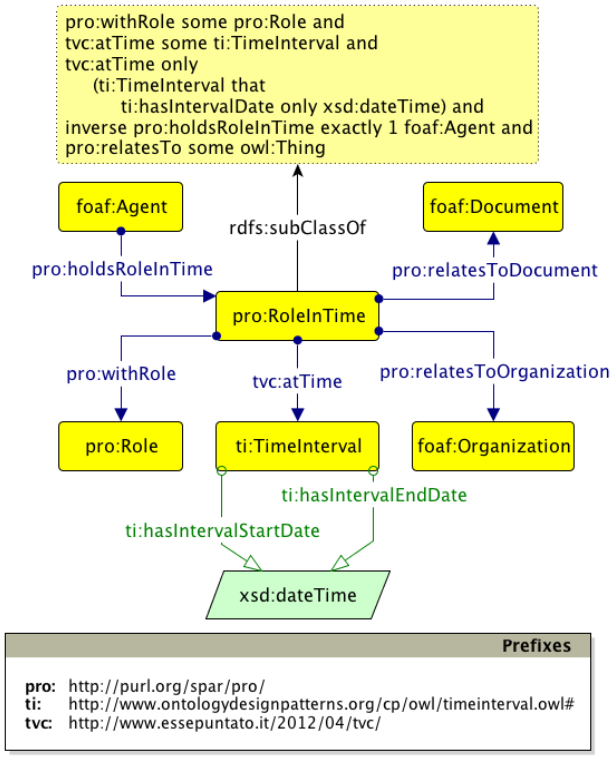

Figure 3: Graffoo representation of the Publishing Roles Ontology (PRO).

Using PRO and its TVC-compliant structured as illustrated in Table 1, it is possible to describe all the scenarios discussed in Section 2.2 and Section 2.3, as follows:

```
:peroni pro:holdsRoleInTime
  # as author of two different papers
  [ a pro:RoleInTime
    ; pro:withRole pro:author
    ; pro:relatesToDocument
      :earmark-paper , :kce-paper ]
  # as affiliate of UniBo CS Dept
  , [ a pro:RoleInTime
    ; pro:withRole pro:affiliate
    ; pro:relatesToDocument :earmark-paper
    ; pro:relatesToOrganization :cs-unibo ]
  # as affiliate of OU KMi
  , [ a pro:RoleInTime
    ; pro:withRole pro:affiliate
    ; pro:relatesToDocument :kce-paper
    ; pro:relatesToOrganization :kmi ] .
```

[11] PRO, the Publishing Roles Ontology: http://purl.org/spar/pro. The prefix *pro* refers to entities defined in it.

Table 1: Alignments between TVC and PRO.

TVC entity	PRO entity	Description
Value InTime	Role InTime	The class of the particular situation that describes the role an agent has within a particular time interval.
has Value	holdsRole InTime	The object property linking any *foaf:Agent* (e.g., a person, a group, an organization or a software agent), to a *pro:RoleInTime* situation.
with Value	with Role	The object property linking the situation to the role the agent has. Currently, 31 roles are defined in the PRO ontology as individuals of the class *pro:Role*.
Instant or Interval	Time Interval	Two (starting and ending) points in time that define a particular period related to (object property *tvc:atTime*) a *pro:RoleInTime* situation.
within Context	relatesTo Document and relatesTo Organization	Object properties linking any kind of bibliographic work (*foaf:Document*) or publishing organization (*foaf:Organization*) taking part in a *pro:RoleInTime* as contextual extent.

As seen, through PRO we can model very rich scenarios, and thus answer complex queries, such as the previously introduced "give me the institutional affiliation of the person *:peroni* as author of the paper *:earmark-paper*":

```
SELECT ?aff WHERE { :peroni pro:holdsRoleInTime
  [ a pro:RoleInTime ; pro:withRole pro:author
    ; pro:relatesToDocument :earmark-paper ]
  , [ a pro:RoleInTime
    ; pro:withRole pro:affiliate
    ; pro:relatesToDocument :earmark-paper
    ; pro:relatesToOrganization ?aff ] }
```

5.2 Specifying document statuses with PSO

The second subdomain of publishing handled in SPAR and based on the TVC pattern is that of the *status* of documents. In this case, the entity is a document holding a

particular status at a certain time as a direct consequence of a particular event. For instance, a document is under review until all reviewers send in their comments and the editor decides whether to accept or reject the paper. After the acceptance/rejection decision is made, the status "under review" is no longer valid: this should be formally describable using an appropriate ontology. Moreover, it is sometimes useful to link documents to the decisions or events that cause the acquisition or loss of a particular status.

Pre-existing ontologies describing the status of documents (e.g., BIBO [5], the *Project Documents Ontology*[12] [19]) and the *Document Status Ontology*[13]) rely for this on specific property links. As discussed in Section 2.2, this approach prevents proper descriptions of scenarios that involve a temporal duration for each status. With the exception of the Document Status Ontology, which describes status changes as events, the other ontologies cannot allow time-dependent data, or can do so only partially.

In order to address these issues in a more satisfactory manner, we developed *PSO*, the *Publishing Status Ontology*[14]. This ontology (shown in Fig. 4) characterises the publication status of a document or any other publication entity at each of the various stages in the publishing process (e.g. draft, submitted, under review, rejected for publication, accepted for publication, version of record, peer reviewed, open access, etc.). As with PRO, PSO was developed following the TVC pattern, as shown in Table 2. Using PSO, it is possible to describe the statuses of documents and how they change over time. For instance, consider the following description:

> The paper *:earmark-paper* was submitted to DocEng 2009 on 24 April 2009 at 13:18. At noon on 26 April, when the authors received acknowledgement of safe receipt of the paper from the conference editorial committee, the paper was considered "under review" until 27 May at 17:38.

PSO can be used to represent this description, as follows:

```
:earmark-paper
  pso:holdsStatusInTime [ a pso:StatusInTime
    ; pso:withStatus pso:submitted
    ; tisit:atTime [ a ti:TimeInterval
      ; ti:hasIntervalStartDate
        "2009-04-24T13:18:21Z"^^xsd:dateTime ]
    ; pso:isAcquiredAsConsequenceOf [
      a part:Event
      ; dcterms:description "An author
      submitted the paper through the online
      conference submission system."] ]
  , [ a pso:StatusInTime
    ; pso:withStatus pso:under-review
    ; tisit:atTime [ a ti:TimeInterval
      ; ti:hasIntervalStartDate
        "2009-04-26T12:00:00Z"^^xsd:dateTime
      ; ti:hasIntervalEndDate
        "2009-05-27T17:38:01Z"^^xsd:dateTime ]
    ; pso:isAcquiredAsConsequenceOf [
      a part:Event
      ; dcterms:description "The editorial
      committee sent the paper to reviewers for
      consideration." ]
    ; pso:isLostAsConsequenceOf [ a part:Event
      ; dcterms:description "The reviewers
      completed their reviews of the paper." ].
```

[12]Project Documents Ontology:http://ontologies.smile.deri.ie/pdo#.
[13]Document Status Ontology: http://ontologi.es/status#.
[14]PSO, the Publishing Status Ontology: http://purl.org/spar/pso. The prefix *pso* and *part* refer to entities defined in it.

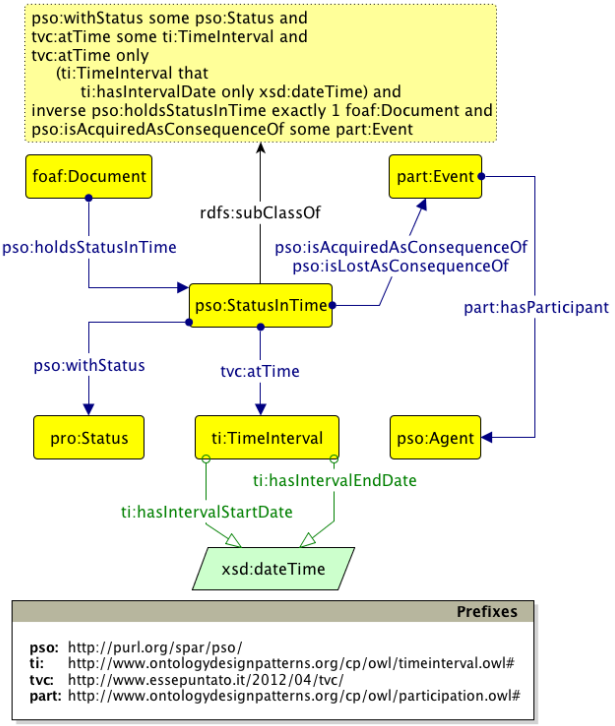

Figure 4: Graffoo representation of the Publishing Status Ontology (PSO).

6. CONCLUSIONS

Time changes everything. Linked Data describing humans acting in specific roles or things processed in phases would be considerably improved by using time-based specifications for the roles and statuses of these humans and things. In this paper we introduced two ontologies, the *Publishing Roles Ontology* and the *Publishing Status Ontology*, that are part of the *Semantic Publishing and Referencing* ontology set. We illustrated how they provide lightweight and easy integration of time-related features such as time- and context-dependent roles and statuses by implementing the *time-indexed value in context* (*TVC*), the ontological pattern proposed in this paper to deal with temporal and contextual issues. The use of the TVC pattern provides PRO and PSO with a model that overcomes most of the limitations in expressivity and conceptual errors that we found in the other formalizations we have reviewed (including well-known ontologies such as SWC and BIBO). The TVC pattern adopted in PRO and PSO can easily be applied in other contexts. In the future, we plan to study integration paths between other well-known ontologies and time- and context-dependent features as made possible through this pattern.

7. REFERENCES

[1] Aranguren, M E., Antezana, E., Kuiper, M., Stevens, R. (2008). Ontology Design Patterns for bio-ontologies:

Table 2: Alignments between TVC and PSO.

TVC entity	PSO entity	Description
Value InTime	Status InTime	The class of the particular situation of the status a document has at a particular time as consequence of one or more events.
has Value	hasStatus InTime	The object property linking a *foaf:Document* (i.e., any bibliographic work) to a *pro:StatusInTime*.
with Value	with Status	The object property linking the situation to the status a document has. Currently, 26 statuses are defined in the PSO ontology as individuals of the class *pro:Status*.
Instant or Interval	Time Interval	Two (starting and ending) points in time that define a particular period related to (object property *tisit:atTime*) a *pso:StatusInTime* situation.
within Context	isAcquiredAs ConsequenceOf and isLostAs ConsequenceOf	Object properties linking a situation to the events (*part:Event*) of any publishing process that changes the status of the document (e.g., writing a draft, submitting a preprint for publication, or publishing the final paper).

a case study on the Cell Cycle Ontology. In BMC Bioinformatics 2008, 9(Suppl 5): S1. doi:10.1186/1471-2105-9-S5-S1.

[2] Boella, G., van der Torre, L., Verhagen, H. (2007). Roles, an interdisciplinary perspective. Special Issue of the Journal of Applied Ontology, 2 (2): 81-207. ISSN: 1570-5838.

[3] Brickley, D., Miller, L. (2010). FOAF Vocabulary Specification 0.98. Namespace Document, 9 August 2010 - Marco Polo Edition. http://xmlns.com/foaf/spec/[15]

[4] Crofts, N., Doerr, M., Gill, T., Stead, S., Stiff, M. (2011). Definition of the CIDOC Conceptual Reference Model. Version 5.0.4, November 2011. ICOM/CIDOC CRM Special Interest Group. http://www.cidoc-crm.org/docs/cidoc_crm_version_5.0.4.pdf

[5] D'Arcus, B., Giasson, F. (2009). Bibliographic Ontology Specification. Specification Document, 4 November 2009. http://bibliontology.com/specification

[6] Dublin Core Metadata Initiative (2010). DCMI Metadata Terms. DCMI Recommendation. http://dublincore.org/documents/dcmi-terms/

[7] Guarino, N., Welty, C. (2002). Evaluating ontological decisions with OntoClean. In Communications of the ACM, 45 (2): 61-65. doi:10.1145/503124.503150.

[8] Harris, S., Seaborne, A. (2012). SPARQL 1.1 Query Language. W3C Working Draft, 05 January 2012. World Wide Web Consortium. http://www.w3.org/TR/sparql11-query

[9] Hayes, P., Welty, C. (2006). Defining N-ary Relations on the Semantic Web. W3C Working Group Note, 12 April 2006. World Wide Web Consortium. http://www.w3.org/TR/swbp-n-aryRelations/

[10] Hobbs, J .R., Pan, F. (2006). Time Ontology in OWL. W3C Working Draft, 27 September 2006. World Wide Web Consortium. http://www.w3.org/TR/owl-time/

[11] Iannella, R. (2010). Representing vCard Objects in RDF. W3C Member Submission, 20 January 2010. World Wide Web Consortium. http://www.w3.org/TR/vcard-rdf/

[12] Kozaki, K., Endo, S., Mizoguchi, R. (2008). Instance Management Problems in the Role Model of Hozo. In Proceedings of the 10th Pacific Rim International Conference on Artificial Intelligence: Trends in Artificial Intelligence (PRICAI 08). doi:10.1007/978-3-540-89197-0_56.

[13] Masolo, C., Vieu, L., Bottazzi, E., Catenacci, C., Ferrario, R., Gangemi, A., Guarino, N. (2004). Social Roles and their Descriptions. In Proceedings of the 9th International Conference on the Principles of Knowledge Representation and Reasoning (KR2004). https://www.aaai.org/Papers/KR/2004/KR04-029.pdf

[14] Moller, K., Bechhofer, S., Heath, T. (2009). Semantic Web Conference Ontology. http://data.semanticweb.org/ns/swc/ontology

[15] Presutti, V., Gangemi, A. (2008). Content Ontology Design Patterns as practical building blocks for web ontologies. In Proceedings of the 27th International Conference on Conceptual Modeling (ER 2008). doi:10.1007/978-3-540-87877-3_11.

[16] Serafini, L., Homola, M. (2011). Contextual Representation and Reasoning with Description Logics. In Proceedings of the 2011 International Workshop on Description Logics (DL2011). http://ceur-ws.org/Vol-745/paper_30.pdf

[17] Shotton, D. (2009). Semantic Publishing: the coming revolution in scientific journal publishing. Learned Publishing, 22 (2): 85-94. doi:10.1087/2009202.

[18] Shotton, D., Portwin, K., Klyne, G., Miles, A. (2009). Adventures in Semantic Publishing: Exemplar Semantic Enhancements of a Research Article. PLoS Computational Biology, 5 (4): e1000361. doi:10.1371/journal.pcbi.1000361.

[19] Varma, P. (2010). Project Documents Ontology. http://vocab.deri.ie/pdo

[20] Welty, C. (2010). Context Slices: Representing Contexts in OWL. In Proceedings of the 2nd Workshop on Ontology Patterns (WOP 2010). http://CEUR-WS.org/Vol-671/pat01.pdf

[15] All the URLs were last visited 08/06/2012.