

An ontological investigation over human relations in linked data

Miroslav Vacura^{a,*}, Vojtěch Svátek^b and Aldo Gangemi^c

^a *Department of Philosophy, University of Economics, Prague, W. Churchill Sq. 4, 130 67 Prague 3, Czech Republic*

E-mail: vacuram@vse.cz

^b *Department of Information and Knowledge Engineering, University of Economics, Prague, W. Churchill Sq. 4, 130 67 Prague 3, Czech Republic*

E-mail: svatek@vse.cz

^c *Paris 13 University – CNRS UMR7030 – Sorbonne Paris Cité, France & STLab, ISTC-CNR, Via Nomentana 56, 00161 Rome, Italy*

E-mail: aldo.gangemi@cnr.it

Abstract. The research presented in this article is motivated by the increasing importance of complex human relations in linked data, either extracted from social networks, or found in existing databases. The FOAF vocabulary, targeted in our research, plays a central role in those data, and is a model for lightweight ontologies largely used in linked data, such as the DBpedia ontology and schema.org. We provide an overview of FOAF and other approaches for describing human relations, followed by a detailed analysis and critique of the FOAF Relationship Vocabulary, the most important FOAF extension. We propose an explicit formal axiomatization of this vocabulary, and an ontological analysis concerning the properties used to describe human relationships. We analyze the distribution of human relations based on their epistemological status, and define an *ontoepistemic* meta-property as characteristic of some of these predicates. Our analysis is generalizable to semantic modeling of social networks. Additionally, the modeling patterns used in other relevant linked data vocabularies are analyzed for comparison.

Keywords: Ontology, FOAF, human relations, linked data

Accepted by: Guido Boella

1. Introduction

The Linked Data initiative was started by Tim Berners-Lee as an architectural vision for the Semantic Web. It explores the idea of Semantic Web understood as a web of data explicitly linked across different datasets, so that both people and machines can integrate and explore them semantically. If data are linked, then “when you have some of it, you can find other, related, data” says Berners-Lee (2009). Just like HTML hyperlinks enable relationships between documents, linked data enable relationships between *entities* by using the RDF abstract data structure. The key requirements for Linked Data are quite simple:

- (1) Use URIs as names for things.
- (2) Use HTTP URIs so people can look up those names.

*Corresponding author. E-mail: vacuram@vse.cz.

- (3) When someone looks up a URI, provide useful information, using standards (RDF, SPARQL, OWL, etc.).
- (4) Include links to other URIs, so that they can discover more things.

Guidance provided by these general points was later extended in technical documents like those by Bizer et al. (2007) and Sauermann et al. (2007), as well as in overview papers by Bizer et al. (2008) and Bizer et al. (2009). Linked data can be queried through SPARQL endpoints like in relational databases, or crawled with appropriate browsers by following RDF links (analogously to HTML links); a search engine can also retrieve RDF triples that are used to annotate HTML pages. RDF also has the nice properties of graph databases to handle high-dimensional sparse data. However, unlike HTML, which only provides a generic linking capability, links in a Linked Data environment can have different types: we can, e.g., specify that one person is *author of* a paper, or that this person *knows* another person.

However, linked data critically need cleaning, as well as the ability to reason on them based on more clearly axiomatized ontologies. Ontology schemata like the DBpedia ontology and schema.org have been developed without caring too much about the axioms that were going to be enforced on linked data, so leading to known issues when querying on those data is performed with inference engines. After the impressive ‘semantic bootstrap’ provided by light-weight linked data vocabularies, it is probably time to introduce some good practices to polish vocabularies and data when serious inference is needed.

In this article we provide a study primarily based on one of the oldest and most central semantic web ontologies, FOAF, as defined by Brickley and Miller (2010), and its extensions. As a matter of fact, FOAF plays a central role in linked data oriented on human relations (HR), and is a model for lightweight ontologies largely used in linked data, such as the DBpedia ontology¹ and schema.org.² For that reason, the results of a study on FOAF would easily be applicable to other vocabularies that are bootstrapping the adoption of semantic-web-like technology in the Web, such as schema.org, now widely embedded in web pages in order to foster search engine optimization.

Our study concentrates on the formal properties of FOAF-related human relation predicates, providing a revised and richer axiomatization, which can inject currently implicit knowledge into human relation linked data, and help maintain its consistency.

While the FOAF project itself has a long history and has been subject of lot of research efforts, it has been for some time put aside of the main research focus. However in the context of rising interest in Linked Data initiative, the importance of FOAF increased, e.g., most linked data vocabularies contain an explicit mapping to the FOAF class `Person`. FOAF is one of the most widely used linked data vocabularies, with millions of data fragments available on the Web as documented by Stuart (2012).

The rest of the article is organized as follows: Section 2 is a basic characterization of the FOAF project, of its history, and an overview of some important extensions directed at describing human relations. This section also provides a short overview of other approaches to modeling human relations. Section 3 analyses the formal ontological characteristics of the Relationship Vocabulary extension of FOAF, and analyses some formal problems. Section 4 provides our (reconstructed) formal axiomatization of the FOAF Relationship Vocabulary, based also on the natural language descriptions provided in the documentation. In Section 5 we proceed with a detailed analysis of the properties defined by this vocabulary, jointly with the introduction of an epistemic/ontologic distinction, leading to the proposition of an ‘ontoepistemic’ predicate. On the grounds of this analysis, we present a revised list of characteristics

¹<http://dbpedia.org/ontology/>.

²<http://schema.org>.

of human relationships. Section 6 contains a revised formal axiomatization that, while respecting the basic assumptions of the original FOAF Relationship Vocabulary, yet makes it more explicit some implicit assumptions in the original axioms. Section 7 provides an empirical survey of human relationships in Linked Open Data/Vocabularies. Finally, Sections 8 provides some conclusions and acknowledgments.

2. FOAF and human relations modelling

In this section we discuss some problems related to the FOAF project and the `knows` relation. The FOAF project is well known in the Linked Data community, and the meaning of the `knows` relation is intuitive. According to Swoogle³ there was 1.2 million FOAF documents at the beginning of year 2015. There are also many other triples using FOAF, since its vocabulary is reused in vast datasets such as DBpedia, schema.org, etc. (cf. LOV⁴ for an assessment of FOAF reuse in linked data). We want to take advantage of the amount of direct FOAF data, as well as of sites that use the format as a standard for data exchange, e.g., blogging sites like LiveJournal⁵ or Drupal 7,⁶ which uses FOAF as one of the vocabularies for its RDF-based core, contributing to their enrichment and consistency management.

The Friend of a Friend (FOAF) project started with the aim of creating a Web of machine-readable pages describing people, the links between them and the things they do, work on, create and like, with an emphasis on the on-line presence of people.⁷ We concentrate on a specific fragment of FOAF: the axioms related to the `knows` property, informally defined in FOAF standard as “a person known by this person (indicating some level of reciprocated interaction between the parties)”. Unfortunately this definition characterizes the members of range of this relation – it is not a definition of the `knows` relation itself. It is understood as a property injectively ranging over the `Person` class, and it seems to be conceptualized as a *symmetric* relation, because the informal definition requires “some level of reciprocated interaction” and stresses that “if someone knows a person, it would be usual for the relation to be reciprocated” (Brickley and Miller, 2010). As we will see, the word *knows* is pretty vague, and its FOAF specification does not resolve this vagueness in any formal way. Figure 1 shows in a UML-like class diagram the axioms of the `foaf:Person` fragment of FOAF, with `foaf:knows` being a simple `owl:ObjectProperty` (a binary relation) ranging over the `foaf:Person` class. Therefore, the formal interpretation only tells us that the universe of the *knows* predicate is a subset of the Cartesian product of the `foaf:Person` set, whose instances are on their turn characterized as agents with own interests, projects, publications, homepages, images, etc.

As a first observation, no axiom tells us that `foaf:knows` is symmetric, therefore, in the generic interpretation of symmetry it should be assumed as not symmetric. The relation is not symmetric if $\neg\forall x\forall y(R(x, y) \rightarrow R(y, x))$. Notice that this is different from the definition of asymmetric relation assumed, e.g., by OWL2 direct semantics: $\forall x\forall y(R(x, y) \rightarrow \neg R(y, x))$ (equivalent to antisymmetry $\forall x\forall y((R(x, y) \wedge R(y, x)) \rightarrow (x = y))$ plus irreflexivity $\forall x\neg R(x, x)$), which we actually adopt in our revised formalization (cf. Section 4). However assuming that `foaf:knows` is not symmetric may conflict with the informal definition requiring “some level of reciprocated interaction”.

³<http://swoogle.umbc.edu/>.

⁴<http://lov.okfn.org/dataset/lov/vocabs/foaf>.

⁵<http://www.livejournal.com/>.

⁶<http://drupal.org/node/574624>.

⁷<http://www.foaf-project.org>.

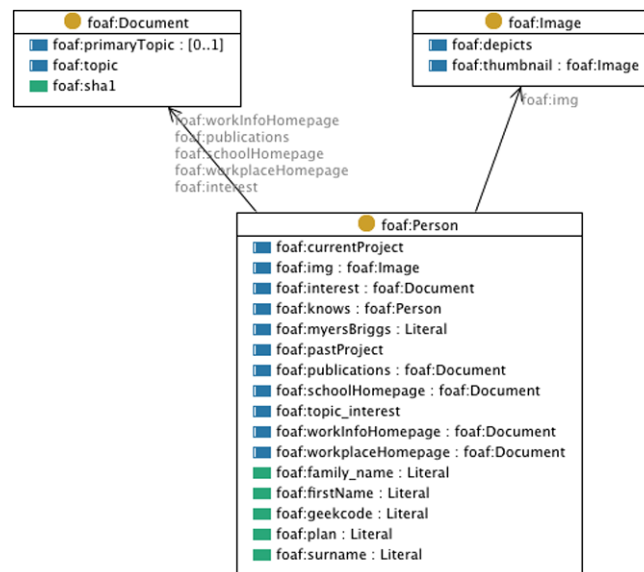


Fig. 1. The axioms related to the `foaf:Person` class. `foaf:knows` is a simple object property ranging over `foaf:Person`.

Certainly reciprocated interaction does not imply symmetry of the specific relationship emerging from reciprocating: I can hate someone that actually loves me. However, at the generic level, both the hater and the lover know each other. It would be very hard to imagine reciprocating without mutual knowing. Probably, FOAF designers have realized that, in many cases, common sense knowing is not symmetrical, as when I know someone that does not know me. The decision not to put symmetry in `foaf:knows` can then be due to pragmatic reasons. Moreover, the original RDFS specification of FOAF does not allow such an axiom, which becomes available only in OWL. A certain amount of evolutionary inconsistency between informal definition and design can be the reason for the conflict. Still, we want to find a design that better fits the common sense and practical requirements of ‘knowing’ networks. Requirements are not specified here with reference to specific applications, but a core set can be devised based on current and prospective usages of the vocabulary. We have intentionally kept the complexity to a minimum by leveraging on common sense intuition and discussions on HR from the literature. For specific applications, additional requirements should be devised against use cases and competency questions:

- in many cases, if I know A then I am known by A;
- in general, knowing someone does not require being known;
- knowing is very specific in practice: special forms of knowing apply to friends, relatives, colleagues, competitors, public people, etc.;
- expressing the actual context of knowing someone (i.e. creating additional entities for knowing contexts, provenance, etc.) creates indirections that one would like to avoid, e.g., for regular social network analysis on graphs whose nodes are persons;
- the conditions, under which I can actually claim/admit to know A, can be unsatisfactory in practice, either for me or others. In other words, there are epistemic aspects of knowing to be considered;
- knowing can be indirect: I can pretend to know someone even if I do not have direct communication with her (e.g., with public persons). Also, someone can claim I know someone even if I do not admit

it, or if there is no evidence of direct connection, e.g., in a legal trial against a politician accused of having liaisons with a mafia boss;

- reciprocated knowing can be physically impossible (e.g., knowing historical figures).

2.1. Relationship FOAF module

Is there anything better than `foaf:knows`? The *relationship FOAF module* developed in 2002 by E. Vitiello⁸ is an extension of FOAF supposed to make that predicate more precise. The RDF schema of this module defines several subproperties of `foaf:knows`, namely: `friendOf`, `acquaintanceOf`, `parentOf`, `siblingOf`, `childOf`, `grandchildOf`, `spouseOf`, `enemyOf`, `antagonistOf` and `ambivalentOf`.

Inclusion of some of these properties seems debatable. For example, if a person describes someone as his/her enemy, then the person surely knows the enemy. However, the opposite may not be true. There may be an unknown enemy of anybody (the same holds, e.g., for children knowing their parents).

Also inclusion of enemies in a *friend of a friend* vocabulary seems counterintuitive, because the general intuition about *knows* in the semantic context of FOAF is a positive (or at least neutral) relation between people.

2.2. FOAF Relationship Vocabulary

Since 2004 the relationship module has been modified to a more general FOAF Relationship Vocabulary and is continually maintained and enhanced.⁹ The following subproperties were added: `ancestorOf`, `apprenticeTo`, `closeFriendOf`, `collaboratesWith`, `colleagueOf`, `descendantOf`, `employedBy`, `employerOf`, `engagedTo`, `friendOf`, `grandparentOf`, `hasMet`, `influencedBy`, `knowsByReputation`, `knowsInPassing`, `knowsOf`, `lifePartnerOf`, `livesWith`, `lostContactWith`, `mentorOf`, `neighborOf`, `participant`, `participantIn`, `Relationship`, `worksWith`, and `wouldLikeToKnow`.

The Relationship Vocabulary is represented in OWL and is presented as a vocabulary for expressing human relations in general. The Relationship Vocabulary is not used as widely as FOAF core, but there are substantial linked datasets that use it, e.g., BBC Music¹⁰ relies on the MusicBrainz database¹¹ that uses some properties of the Relationship Vocabulary for data about persons.¹² The Relationship Vocabulary is still actively developed and maintained (the latest revision has been published in April 2010) and there is an active community around the FOAF website¹³ and FOAF-DEV mailing list.¹⁴ We have not been able to find out whether any particular methodology was used for choosing those relations, whether they were added because of requirements, or whether they were added based on suggestions by participants in the FOAF-DEV mailing list.

The reason why we go into such depth with the analysis of the FOAF project and the Relationship Vocabulary is to analyze the difficulties when we link these data to similar datasets using different

⁸<http://www.perceive.net/schemas/20021119/relationship/>.

⁹<http://purl.org/vocab/relationship>.

¹⁰<http://www.bbc.co.uk/music>.

¹¹<http://musicbrainz.org/>.

¹²See also: <http://kasabi.com/dataset/bbc-music/guide>.

¹³<http://www.foaf-project.org>.

¹⁴<http://lists.foaf-project.org/mailman/listinfo/foaf-dev>.

vocabularies. Subsumption and other logical relations are important because they are used even in the simplest reasoners and information aggregators like Tabulator by Berners-Lee et al. (2006).

2.3. Alternative approaches to modelling human relations

Limitations of the `foaf:knows` relation led to some other proposals. E.g., Matsuo et al. (2004) proposed a different classification of human relations: 1) *relationship by event* – relationships of people based on their participation in the same event (a conference, a baseball game, a walk); 2) *relationship by common property* – relationships based on property like working place or hobby; and 3) *relationship by communication* – these are relationships based on some type of communication: face-to-face conversation, telephone call, e-mail, chat, posting to web blogs etc.

However, even this approach has problems. Firstly, all criteria used in that classification are events: communications, face-to-face meetings, telephone calls, sending e-mail messages. Therefore, it is not clear why relationships based on communication should form a separate top level class from that based on events. Secondly, having a common property in most cases does not imply a `knows` relation. People can work at the same work place, study in the same school, have the same hobbies etc., without knowing each other or having any other significant relationship. It seems that authors actually think about criteria of *plausibility* to be able to infer that someone knows someone else. Indeed, communicating is probably a safer plausibility criterion than working in a same company, which is safer than participating in a same generic event. Coherently, Matsuo's approach is only explicitly tailored to automatic relationship discovery, e.g., from a frequent email exchange between two persons we can conclude that they know each other.

Two other, more general, ontologies are those by Gangemi and Mika (2003) and Masolo et al. (2004), which lay down the foundations for the Neo-Davidsonian representation of human relations proposed by Mika and Gangemi (2004).

Gangemi and Mika (2003) developed the *Descriptions and Situations* ontology design pattern that provides a very generic way to capture relations between a description and a described situation. The authors rely on the distinction between individual (extensional) relationships (i.e. *situations*), and their generic descriptions (intensional relations, i.e. *situation types aka situation descriptions*). These two levels need to be represented separately, in order to support a class of use cases, e.g., the relation between a norm, and a case to which a norm should be applied. The separation is not enough though, because intensional or extensional social relations can have multiple arities (they are *multivaried predicates*), which cannot be known in advance. A famous example is due to Davidson (1967): when representing, e.g., the intensional relation *preparing a coffee*, how many arguments are needed? The agent preparing the coffee, the amount of coffee, its quality, the coffee-pot, the heating, the kind of water, the time and place of preparing it, etc.? In order to support both separation of intension and extension, and multivaried predicates, the generic pattern of Description and Situations applies a double reification strategy, which allows to represent separately (extensional) relationships such as a certain coffee making situation, (intensional) relations with any arity required, such as a coffee making recipe, and to relate them systematically.

Related to work by Gangemi and Mika (2003), paper by Masolo et al. (2004), on a rich axiomatic setting, provides a “general formal framework for developing a foundational ontology of socially constructed entities”, based on formal analysis of *roles* and their descriptions (understood as intensional relations).

These two ontologies share the basic assumptions, and provide the groundwork for paper by Mika and Gangemi (2004), which proposes an extension of social ontologies such as FOAF with reification

(first-order treatment) of social relationships. Mika and Gangemi (2004) introduce a `Social Relationship` class as a subclass of `Context`, and particular social relationships such as `Friendship` as subclasses of `Social Relationship`. As contexts, social relationships can have, e.g., parameters, roles and tasks as components. A typical class of roles is `Relationship Role`, a subclass of the `Social Role` concept introduced by Masolo et al. (2004). An example of a relationship role is `Friend` as in a friendship relation, `Student` and `Professor` as in a student/advisor relationship, and `Uncle` and `Nephew` as kinship roles. This modeling approach also restricts relationship roles to be played by members of the class `Natural Person`, and introduces some characteristics of social relationships that can be associated with it: sign (valence: positive, negative, etc.), strength, provenance, history, etc.

From the point of view of formal ontology, the design pattern proposed by Mika and Gangemi (2004) is more expressive and powerful, and should be preferred whenever a use case requires substantial expressiveness and reasoning capabilities. However the Neo-Davidsonian pattern has not got a wide usage specifically for human relations in the Linked Data domain, since ‘direct’ binary relation patterns are usually preferred, probably because of their simplicity and closeness to the basic RDF data model. The analysis presented in this paper is therefore directed at binary human relations, although the results of the analysis can also be used for ontologies following more expressive patterns.

3. Formal ontological structure of the Relationship Vocabulary

This section focuses on the Relationship Vocabulary extension of FOAF. Its first part reviews definitions of some important ontological concepts, and the second part provides a more detailed analysis of its logical and ontological structure.

3.1. Formal concepts used in analysis of Relationship Vocabulary: Ontology pattern and reification

The meaning of term “ontology pattern” is summarized by Gangemi (2005): historically it is based on latin word “patronus” (patron) – someone proposed for imitation. In knowledge engineering Clark and Porter (1997) define “pattern” as “a theory ‘template’ or ‘schema’, which denotes a structure of objects and relationships, but whose axioms are not directly part of the global KB”. In ontology engineering “ontology pattern” is often defined more flexibly as various schemas and macros for UML, OWL, core ontologies, etc. See works by Reich (2000), Gangemi et al. (2003), Guizzardi et al. (2004) and Svátek (2004).

The term “reification” is historically based on latin words “res” (thing) and “facere” (to make) so it can be translated as “thingmaking”, i.e. turning of something abstract into a thing. Stevens and Lord (2010) claim that in the case of ontology engineering, the term “reification” describes technique that represents a property as an object enabling a richer description of a property.

3.2. Structure of Relationship Vocabulary

At the beginning we will analyze what ontological patterns are used in Relationship Vocabulary to describe various types of human relations.

The first observation is that almost all properties of Relationship Vocabulary have the class `Person` as both domain and range. An example is the property `livesWith` – a relation between two persons, in this case symmetric.

On the other hand, there are also three terms that do not fit within this description: the properties `participant` and `participantIn`, and the class `Relationship`. The domain of `participantIn` is `Person`, while its range is `Relationship`. The opposite holds for `participant`. Intuitively, we would expect these two properties to be inverse of each other, but this is not formally declared in the Relationship Vocabulary.

We therefore observe that Relationship Vocabulary includes two different ontology patterns for human relations.

- The first pattern follows the legacy of the original FOAF, and is depicted in Fig. 2. Human relations such as `parentOf` or `mentorOf` are defined as properties that have the class `Person` as both domain and range. This is formally just an extension of the original property `knows` – based on the idea that the new properties are subproperties of `knows`, thus maintaining ‘backward compatibility’. This pattern is depicted on Fig. 2.
- The second pattern uses the class `Relationship` that is described as a “class whose members are a particular type of connection existing between people related to or having dealings with each other”. Apparently then it seems that members of this class are reifications of intensional relations – so that there is one member per relation type, e.g., one member represents the relation *friendOf*, another *livesWith*, etc. This pattern is depicted on Fig. 3.

Let us look at the first pattern first. Historically the limitation of the original FOAF relationship module was that it consisted of just `rdfs:subPropertyOf` triples (binary relation subsumption axioms). The Relationship Vocabulary extended it and turned it into a generic vocabulary. The description of extended properties now includes some semantics with a richer subsumption structure. Still, probably for backward compatibility, properties like `childOf` are all subsumed by `knows`. A new `knowsOf` property, which is not symmetric, has been introduced, and recently (February 2010) its semantics has been changed so that the `knowsOf` (which is not symmetric) is no longer subproperty of the (symmetric) `knows`. It has been stated that `knows` is subproperty of `knowsOf`.

A recent (February 2010) revision of the Relationship Vocabulary eventually acknowledged that for distant descendants/ancestors it may not be possible to know reciprocally each other, hence the property `descendantOf` is no longer subsumed by `knows`. However, the editors failed to notice that we run into exactly the same problem when we consider the property `grandparentOf` and even `parentOf`.

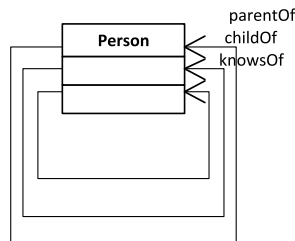


Fig. 2. Ontology pattern of Relationship Vocabulary 1.

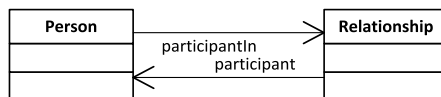


Fig. 3. Ontology pattern of Relationship Vocabulary 2.

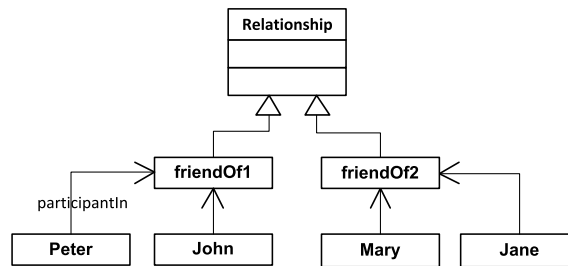


Fig. 4. Instances `friendOf1` and `friendOf2` of class `Relationship` and other instances.

For a grandparent (and even parent) of a person could die before the person was born, so there are real-world cases where people could not know their children or grandchildren.

In many cases the assumption that being `parentOf` implies `knows` may be true. But from strictly ontological point of view these properties are independent.

The second pattern seems a bit puzzling. What is that modeling approach meant for? For a person to be participant in a relationship, we need that members of that class are reifications of *extensional*, rather than *intensional*, relationships.

In practice, each individual relationship becomes an instance of the class `Relationship`. For example the relation `friendOf(Peter, John)` would be reified to, e.g., an instance `friendOf1` and the relation `friendOf(Mary, Jane)` would be reified to, e.g., an instance `friendOf2`. Once reified, one can use them to represent participation, as depicted in Fig. 4.

Notice that even with this interpretation, reified relationships are instances of the general class `Relationship`, but there is no subclass structure for specific relations such as `Friendship` or `Marriage`).

Finally, it is not clear how to express the semantics of the relations, e.g., not being symmetric: how can we use `fanOf` to say that Peter is fan of Beethoven but Beethoven is not fan of Peter?

3.3. Some other questions

The Relationship Vocabulary has some other gaps. For example, it does not include intuitive axioms such as `childOf` and `grandchildOf` being subsumed by `descendantOf`, or define siblings as having common parents, etc. Therefore, it does not take advantage of the expressivity of OWL that could be used for representing some common-sense requirements.

Another problem is related to employment relations like `employedBy`, which has the consequence (because of being subsumed by `knows`) of having both its domain and range set to the class `Person`. In a real-world scenario, an entity that employs other persons is usually a legal entity – company, institute or some other type of organization. This may be called a *legal person*; however, in the FOAF vocabulary such kind of entity is represented by the class `Organization`, which is formally asserted to be disjoint with class `Person`. If we use this formalization then we cannot express that a (physical) person is employed by an organisation without introducing logical inconsistencies into our system. Indeed, employment relationships are usually expressed in FOAF by the property `workplaceHomepage`, which ranges on the class `Document`. Then documents can be related to organizations using the property `homepage`. It would be intuitive to say that if the `workplaceHomepage` of a person is a document that is the homepage of some organisation, then the person is `employedBy` this organisation. This is however impossible in the scope of the Relationship Vocabulary semantics, which defines `employedBy` as a relationship between two (physical) persons.

A clarification of the intended semantics of the Relationship Vocabulary, as far as we can reconstruct it from the known documentation, is provided in Section 4.

4. Formal axiomatization of FOAF Relationship Vocabulary relations

In order to have a neutral basis for improving the design of the FOAF Relationship Vocabulary, we reconstruct here its formal axiomatization (from OWL code¹⁵) in first-order logic with identity. Redundant axioms are omitted, variables are all universally quantified.

$$\text{acquaintanceOf}(x, y) \rightarrow \text{Person}(x) \wedge \text{Person}(y) \wedge \text{knows}(x, y) \wedge (x \neq y) \quad (1)$$

$$\text{acquaintanceOf}(x, y) \leftrightarrow \text{acquaintanceOf}(y, x) \quad (2)$$

$$\text{ambivalentOf}(x, y) \rightarrow \text{Person}(x) \wedge \text{Person}(y) \wedge (x \neq y) \quad (3)$$

$$\text{ancestorOf}(x, y) \rightarrow \text{Person}(x) \wedge \text{Person}(y) \wedge (x \neq y) \quad (4)$$

$$\text{ancestorOf}(x, y) \leftrightarrow \text{descendantOf}(y, x) \quad (5)$$

$$(\text{ancestorOf}(x, y) \wedge \text{ancestorOf}(y, z)) \rightarrow \text{ancestorOf}(x, z) \quad (6)$$

$$\text{antagonistOf}(x, y) \rightarrow \text{Person}(x) \wedge \text{Person}(y) \wedge \text{knows}(x, y) \wedge (x \neq y) \quad (7)$$

$$\text{apprenticeTo}(x, y) \rightarrow \text{Person}(x) \wedge \text{Person}(y) \wedge \text{knows}(x, y) \wedge (x \neq y) \quad (8)$$

$$\text{apprenticeTo}(x, y) \leftrightarrow \text{mentorOf}(y, x) \quad (9)$$

$$\text{childOf}(x, y) \rightarrow \text{Person}(x) \wedge \text{Person}(y) \wedge \text{knows}(x, y) \wedge (x \neq y) \quad (10)$$

$$\text{childOf}(x, y) \leftrightarrow \text{parentOf}(y, x) \quad (11)$$

$$\text{closeFriendOf}(x, y) \rightarrow \text{Person}(x) \wedge \text{Person}(y) \wedge \text{knows}(x, y) \wedge (x \neq y) \quad (12)$$

$$\text{closeFriendOf}(x, y) \leftrightarrow \text{closeFriendOf}(y, x) \quad (13)$$

$$\text{collaboratesWith}(x, y) \rightarrow \text{Person}(x) \wedge \text{Person}(y) \wedge \text{knows}(x, y) \wedge (x \neq y) \quad (14)$$

$$\text{collaboratesWith}(x, y) \leftrightarrow \text{collaboratesWith}(y, x) \quad (15)$$

$$\text{colleagueOf}(x, y) \rightarrow \text{Person}(x) \wedge \text{Person}(y) \wedge \text{knows}(x, y) \wedge (x \neq y) \quad (16)$$

$$\text{colleagueOf}(x, y) \leftrightarrow \text{colleagueOf}(y, x) \quad (17)$$

$$\text{descendantOf}(x, y) \rightarrow \text{Person}(x) \wedge \text{Person}(y) \wedge (x \neq y) \quad (18)$$

$$(\text{descendantOf}(x, y) \wedge \text{descendantOf}(y, z)) \rightarrow \text{descendantOf}(x, z) \quad (19)$$

$$\text{employedBy}(x, y) \rightarrow \text{Person}(x) \wedge \text{Person}(y) \wedge \text{knows}(x, y) \wedge (x \neq y) \quad (20)$$

$$\text{employedBy}(x, y) \leftrightarrow \text{employerOf}(y, x) \quad (21)$$

$$\text{employerOf}(x, y) \rightarrow \text{Person}(x) \wedge \text{Person}(y) \wedge \text{knows}(x, y) \wedge (x \neq y) \quad (22)$$

$$\text{enemyOf}(x, y) \rightarrow \text{Person}(x) \wedge \text{Person}(y) \wedge \text{knows}(x, y) \wedge (x \neq y) \quad (23)$$

¹⁵<http://vocab.org/relationship.rdf>.

$$\text{engagedTo}(x, y) \rightarrow \text{Person}(x) \wedge \text{Person}(y) \wedge \text{knows}(x, y) \wedge (x \neq y) \quad (24)$$

$$\text{engagedTo}(x, y) \leftrightarrow \text{engagedTo}(y, x) \quad (25)$$

$$\text{friendOf}(x, y) \rightarrow \text{Person}(x) \wedge \text{Person}(y) \wedge \text{knows}(x, y) \wedge (x \neq y) \quad (26)$$

$$\text{friendOf}(x, y) \leftrightarrow \text{friendOf}(y, x) \quad (27)$$

$$\text{grandChildOf}(x, y) \rightarrow \text{Person}(x) \wedge \text{Person}(y) \wedge \text{knows}(x, y) \wedge (x \neq y) \quad (28)$$

$$\text{grandChildOf}(x, y) \leftrightarrow \text{grandParentOf}(y, x) \quad (29)$$

$$\text{grandParentOf}(x, y) \rightarrow \text{Person}(x) \wedge \text{Person}(y) \wedge \text{knows}(x, y) \wedge (x \neq y) \quad (30)$$

$$\text{hasMet}(x, y) \rightarrow \text{Person}(x) \wedge \text{Person}(y) \wedge \text{knows}(x, y) \wedge (x \neq y) \quad (31)$$

$$\text{hasMet}(x, y) \leftrightarrow \text{hasMet}(y, x) \quad (32)$$

$$\text{influencedBy}(x, y) \rightarrow \text{Person}(x) \wedge \text{Person}(y) \wedge (x \neq y) \quad (33)$$

$$\text{knows}(x, y) \rightarrow \text{knowsOf}(x, y) \quad (34)$$

$$\text{knowsByReputation}(x, y) \rightarrow \text{Person}(x) \wedge \text{Person}(y) \wedge (x \neq y) \quad (35)$$

$$\text{knowsInPassing}(x, y) \rightarrow \text{Person}(x) \wedge \text{Person}(y) \wedge \text{knows}(x, y) \wedge (x \neq y) \quad (36)$$

$$\text{knowsOf}(x, y) \rightarrow \text{Person}(x) \wedge \text{Person}(y) \wedge (x \neq y) \quad (37)$$

$$\text{knows}(x, y) \rightarrow \text{knowsOf}(x, y) \quad (38)$$

$$\text{lifePartnerOf}(x, y) \rightarrow \text{Person}(x) \wedge \text{Person}(y) \wedge \text{knows}(x, y) \wedge (x \neq y) \quad (39)$$

$$\text{lifePartnerOf}(x, y) \leftrightarrow \text{lifePartnerOf}(y, x) \quad (40)$$

$$\text{livesWith}(x, y) \rightarrow \text{Person}(x) \wedge \text{Person}(y) \wedge \text{knows}(x, y) \wedge (x \neq y) \quad (41)$$

$$(\text{livesWith}(x, y) \wedge \text{livesWith}(y, z)) \rightarrow \text{livesWith}(x, z) \quad (42)$$

$$\text{livesWith}(x, y) \leftrightarrow \text{livesWith}(y, x) \quad (43)$$

$$\text{lostContactWith}(x, y) \rightarrow \text{Person}(x) \wedge \text{Person}(y) \wedge \text{knows}(x, y) \wedge (x \neq y) \quad (44)$$

$$\text{lostContactWith}(x, y) \leftrightarrow \text{lostContactWith}(y, x) \quad (45)$$

$$\text{mentorOf}(x, y) \rightarrow \text{Person}(x) \wedge \text{Person}(y) \wedge \text{knows}(x, y) \wedge (x \neq y) \quad (46)$$

$$\text{neighborOf}(x, y) \rightarrow \text{Person}(x) \wedge \text{Person}(y) \wedge \text{knows}(x, y) \wedge (x \neq y) \quad (47)$$

$$\text{neighborOf}(x, y) \leftrightarrow \text{neighborOf}(y, x) \quad (48)$$

$$\text{parentOf}(x, y) \rightarrow \text{Person}(x) \wedge \text{Person}(y) \wedge \text{knows}(x, y) \wedge (x \neq y) \quad (49)$$

$$\text{participant}(x, y) \rightarrow \text{Relationship}(x) \wedge \text{Person}(y) \quad (50)$$

$$\text{participantIn}(x, y) \rightarrow \text{Person}(x) \wedge \text{Relationship}(y) \quad (51)$$

$$\text{siblingOf}(x, y) \rightarrow \text{Person}(x) \wedge \text{Person}(y) \wedge \text{knows}(x, y) \wedge (x \neq y) \quad (52)$$

$$\text{siblingOf}(x, y) \leftrightarrow \text{siblingOf}(y, x) \quad (53)$$

$$\text{spouseOf}(x, y) \rightarrow \text{Person}(y) \wedge \text{knows}(x, y) \wedge (x \neq y) \quad (54)$$

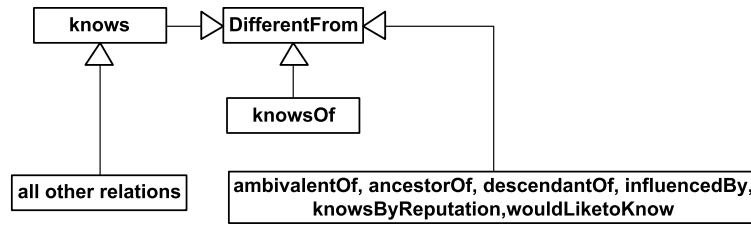


Fig. 5. Ontology of FOAF Relationship Vocabulary.

$$\text{spouseOf}(x, y) \leftrightarrow \text{spouseOf}(y, x) \quad (55)$$

$$\text{worksWith}(x, y) \rightarrow \text{Person}(x) \wedge \text{Person}(y) \wedge \text{knows}(x, y) \wedge (x \neq y) \quad (56)$$

$$\text{worksWith}(x, y) \leftrightarrow \text{worksWith}(y, x) \quad (57)$$

$$\text{wouldLikeToKnow}(x, y) \rightarrow \text{Person}(x) \wedge \text{Person}(y) \wedge (x \neq y) \quad (58)$$

With reference to axiom (54), the domain of `spouseOf` is unrestricted in the source code. The reason is not clear. One suggestion is that cases like nuns who are said to be “spouses of God” are considered. However it is not clear whether in such cases the relation is really symmetric and therefore the unrestricted should be range and not domain. Other possibility is that it is just omission.

Notice also that in the Relationship Vocabulary there is a `differentFrom` relation (explicitly identified with `owl:differentFrom`), which we have represented as \neq in FOL with identity. However, in the original encoding it results to be problematic, since `owl:differentFrom` cannot be used for class-level axioms, and that is probably the reason for introducing `differentFrom`. While in FOL with identity it is straightforward to use \neq , in OWL2 we need irreflexivity to express the same conceptualization.¹⁶ We should therefore make the definition more precise, by saying that `differentFrom` is irreflexive property. In OWL1 it is impossible to assert that the property is reflexive, irreflexive or asymmetric (it allows only assertions that an object property is symmetric or transitive). In OWL2, however, construct `IrreflexiveObjectProperty` allows it to be asserted that an object property expression is irreflexive.¹⁷

An overview of the design of the Relationship Vocabulary is provided in Fig. 5 (with exception of `Relationship`, `participant` and `participantIn` forming a separate pattern described above).

Later in the paper, in Table 1, we present the results of an analysis of the involved relations. The table provides an overview of symmetry (column *RV Symm*) and explicitly assigned super properties for each property (column *RV Super-prop*).¹⁸

¹⁶OWL2 notion of asymmetry is equivalent to antisymmetry plus irreflexivity: $\forall x \forall y (R(x, y) \rightarrow \neg(R(y, x)))$.

¹⁷See http://www.w3.org/2007/OWL/wiki/New_Features_and_Rationale. Note that constructs for expressing reflexive, irreflexive and antisymmetric properties were already available in OWL 1.1 (antisymmetric properties are called asymmetric in OWL2).

¹⁸For a more detailed account of the complexity issues arising from the combination of symmetry, reflexivity, and transitivity with kinship relations, see the work by Longo et al. (2013).

5. Epistemic and ontological state of affairs

The core of some of the problems we have identified could be found in a confusion between epistemic and ontological states of affairs assumed for the relation interpretation. The distinction can be formulated in the following way:

- An **epistemic** state of affairs concerns what is *known* to conscious agents. We may ask, e.g., “Does x know that y is his/her child (enemy, neighbor, ancestor etc.)?” In none of these cases the answer is obvious and it may require further empirical investigation, which in this case consists of *questioning* a person x .
- An **ontological** state of affairs concerns what we assume to be matter of fact *independently of the knowledge* (epistemic state) of particular conscious agents. E.g., we may ask: “Is matter of fact that y is child (enemy, neighbor, ancestor etc.) of x ?” In the case of such questions, the answer could be non-trivial, and may require empirical research, but the state of affairs *does not require questioning* the involved persons: on the contrary, there are external procedures, e.g., performing DNA test, to find out if y is child of x . Eventually it may be found that “ y is child of x ” is true even if there is no knowledge (epistemic state) of this fact in either x or y .

Of course, there is a grey line in the distinction, e.g., when asking involved people is part of the procedures for ascertaining the (ontological) truth of a state of affairs. However, we esteem there is enough intuition to distinguish the mere fact of personally knowing something, vs. desiring to ascertain an objective truth. In some cases, it is an involved person that desires to ascertain that truth, therefore acting according to an ‘ontological mode’.

The distinction between ontological and epistemic state of affairs has two important aspects – philosophical and logical – we will analyse both of them in following sections.

5.1. Philosophical problems relating to human relations

We may start philosophical analysis with some remarks concerning theory of intentionality; its overview was made by Jacob (2010). Theory of intentionality, as founded by Brentano (1874), deals among other topics with the analysis of *intentional inexistence*: the characteristic specific of every mental phenomenon – that it includes something as an object and a direction or reference to it. Mental phenomena that interest us here are mental acts and these can have different types. Any such act has an object and is directed toward this object; e.g., in imagining something is imagined, in judgment something is affirmed or denied, in love something is loved etc.

We must therefore differentiate between three paradigmatic cases represented by these examples of forms of assertions:

- (1) x knows y ;
- (2) x knows/believes that y is his/her child;
- (3) y is child of x .

Notice that while cases 1 and 2 deal with mental acts, assertion 3 concerns no mental phenomena but describes state of affairs. Notice also that the meaning of verb “to know” is in cases 1 and 2 completely different. While the intentional object of assertion 1 is a *person* (person y), the intentional object of assertion 2 is a *proposition* – assertion 2 can be rephrased: “ x knows/believes that proposition ‘ y is child of x ’ is true”. Assertions similar to case 2 are usually called *belief-ascriptions* and some authors like, e.g., Schiffer (1992) interpret them not only using x as person, p as proposition, but introduce even third

component m as *mode of presentation* of under which x believes p (main idea is that x can believe p under m_1 but not under m_2).

Notice also that in assertion 2 we have written “knows/believes”. The reason is that while in common language is the word “to know” used relatively freely with respect to various propositions whose certainty is not assured, in philosophical discourse the verb “to believe” is preferred in such contexts. Usage of the word “to know” is defined usually with reference to “justified true belief” (credited to Plato, see Chisholm (1982)) and there are additional complex requirements to assure certainty of “known” proposition. In philosophical context we should therefore use the verb “to believe” unless these requirements are shown to be fulfilled.

Assertions like 1 are quite different – they still describe intentional acts and their intentional object is a person, however mode of existence of this object is *intentional inexistence*. As already Brentano (1874) noted: while in other relations (like the relation *child of*) both fundament and terminus are real,¹⁹ in cases of mental reference only fundament is certainly real and therefore he suggested we call it *quasi-relational*. We may think of sentences like “ x knows Sherlock Holmes” and even “ x is influenced by Sherlock Holmes”. The point is that using, e.g., natural language processing on non-fictional texts created by really existing persons (like FOAF data) we may end up with relational information about fictional persons who do not exist at all. The fact that x claims to know or be influenced by y does not yet imply existence of y .

Concerning assertions like 1 we have also to ask what is meant by “knowing”? Russell (1911) distinguishes knowing by *acquaintance* and by *description*. Knowing by acquaintance means the first hand knowledge based in case of persons primarily on personal meeting accompanied with sufficient amount of empiric and other experience. The knowledge by description is based only on indirect propositional (verbal, textual, theoretical) information about its object. Notice that only knowledge by acquaintance can guarantee existence of know person.

We can observe also some general problems regarding semantics of names used in these assertions. Frege (1892) noted that one may have some beliefs about “Hesperus” but not about “Phosphorus” because of not knowing that these two names denote the same object (planet Venus). He developed distinction between the *sense* and *reference* of the word and used it to clarify such questions. Similarly one may know “Cicero” but not “Tully” because of not knowing that these two names denote the same person. One may have even different set of beliefs concerning both of these names. Russell (1905) later enhanced this theory with addition of distinction between the name (e.g., “Scott”) and definite descriptions (e.g., “author of Waverley”) that exhibit similar problems.

These questions are also connected with problem of *referential opacity*. According to Quine (1960) we say that two terms are referentially opaque if they cannot be substituted *salva veritate* (i.e. without changing the truth value of the statement). This theory claims that possibility of such substitution depends on context: in context like assertion 1 or 2 are called referentially opaque contexts because we cannot substitute terms *salva veritate*: “John knows Cicero” may be true, but “John knows Tully” may be false. Similarly with assertion 2. However Quine (1960) claims that cases such as our assertion 3 are referentially transparent contexts so truth value of “Marcus is child of Cicero” and truth value of “Marcus is child of Tully” are always the same.

Conclusion of our analysis is that assertion 3 is completely different from assertions 1 and 2. While assertions 1 and 2 concern mental states of individual human beings and also their truth value depends on these mental states, the assertion 3 concerns actual state of affairs, independent of knowledge and beliefs

¹⁹Brentano used words *fundament* and *terminus* to refer to endpoints of binary relation.

of any human being. While assertion 3 represents a real world relation, assertions 1 and 2 represent *quasi-relations*. For relation 3 to obtain, both fundament and terminus (x and y) must exist, assertions 1 and 2 may be true even if terminus (y) does not exist. The conclusion of this analysis is that assertions 1, 2 and 3 are in some very fundamental aspects different and irreducible. Using our terminology introduced above – assertions 1 and 2 concern epistemic state of affairs, while assertion 3 concerns ontological state of affairs.

5.2. Logic of interpersonal relations

Principles of reasoning about knowledge are formalized by *epistemic logic* described by Rescher (2005) and *autoepistemic logic* presented by (Moore, 1985).

Autoepistemic logic is one of formalisms trying to describe nonmonotonic reasoning. It focuses on reasoning of perfectly rational agent about his own knowledge and utilizes classic propositional logic enhanced with *unary* operator K – denoting knowledge or belief. If p signifies proposition, then Kp signifies that p is believed by the rational agent. Every rational agent has a *belief set* of propositions believed by that agent. Because the agent is perfectly rational the believe set is closed under positive introspection ($p \in E \rightarrow (Kp \in E)$) – if agent believes p , he also believes that he believes p – and also closed under negative introspection. Kripke’s modal semantics of possible worlds is used for autoepistemic logic, which corresponds to K45 modal logics (because axioms like $Kp \rightarrow p$ hold). Marek and Truszczyński (1991) demonstrated that autoepistemic logic has many applications related to non-monotonic programming and Lakemeyer (1993) showed that introduction of multi-agent systems led to development of multi-agent autoepistemic logic similar to more general epistemic logic.

Epistemic logic introduces the notational *binary* operator K . Formula $K(x, p)$ is read as “ x knows p ”, where x is a “knower” (i.e. a conscious rational agent) and p is a proposition. Rescher (2005) writes Kxp – we rewrite it for better readability as $K(x, p)$.

On its turn, the *interpersonal* relation “knows”, which we baptize here *p-knows*, is problematic because it is related to epistemic reasoning, but at the same time it refers to the social aspects of knowing certain facts about persons, which lead to the ability of performing certain actions, having rights and duties, etc. In other words, *p-knows* is a consequence of the ‘application’ of epistemic reasoning to interpersonal and social worlds: if I know some facts about a person, then I am entitled to do certain actions; if I have a certain connection (e.g., a kinship relation) to someone then I am expected to know other facts that make me eligible to other actions, etc. So from the point of view of epistemic logic we have to distinguish two relations:

- $knows(x, p)$ – relation from the domain of persons to the range of propositions.
- $p-knows(x, y)$ – relation that has the class of persons both as its domain and range.

Grounding this reflection in logic, what is meant by saying that “person x knows person y ”? What exactly does person x know? Is it possible to define how the relations *knows* and *p-knows* are interrelated?

A comprehensive ontology (“NIC”) of how knowledge, norms, and social behavior are entrenched in the lifecycle of social collectives is presented by Gangemi (2008) by using the Description and Situations pattern (cf. Section 2.3). However, the scope and design of NIC are probably too complex: as we have noticed before, here we want to respect the simplicity of the use case faced by the Relationship Vocabulary, therefore we focus on a rigorous, but logically simple, design.

On the other hand, in line with (Rescher, 2005, p. 6), *p-knows* can be reduced to *knows* by a formula following an axiom schema such as:

$$p\text{-knows}(x, y) \equiv \exists p(\text{identifies}(p, y) \wedge K(x, p)) \quad (59)$$

Notice that a) p denotes a proposition, i.e. another formula, therefore such formulas as (59) actually represent assertions about assertions, b) operator K is intensional (value of $K(x, p)$ is not function of x and p , see paper by Gabbay and Guenther (2003)), c) the identifies relation is a shortcut for a complex of social relations (as e.g. formalised in NIC). These three characteristics make the epistemic approach too complex for e.g. linked data practices.

Eventually, we probably need to live without epistemic extensions, and treat *p-knows* as a plain empirical relation. Still, the considerations about epistemic and ontological states of affairs may provide us some help for a better organization of the Relationship Vocabulary. For each relation R in the Relationship Vocabulary, we may ask whether axiom (60) holds:

$$R(x, y) \leftrightarrow K(x, R(x, y)) \quad (60)$$

That means that the relation R between persons x and y holds if and only if person x knows that relation R holds between persons x and y . This is a non-trivial assumption, because while $(K(x, p) \rightarrow p)$ is the most general principle of epistemic logic, our formula also says the inverse: that for a human relation R , axiom (61) holds:

$$R(x, y) \rightarrow K(x, R(x, y)) \quad (61)$$

It is thus never the case that the assertion $R(x, y)$ evaluates to true without person x also knowing that it evaluates to true. Axiom (60) seems then quite strong: it may be true for some relations but not for others. It means that those relations are equivalent on epistemic and ontological aspects. For this reason, we introduce the metaproperty of “being an *ontoepistemic* relation” when axiom (60) holds for a relation.

5.3. *Ontoepistemic relations*

Ontoepistemic relations are in many cases those describing our mental states. Sometimes, as Robb and Heil (2009) note, they are called *mental properties*. Assuming that ontoepistemic relations are only mental predicates can be debatable for relations such as `apprenticeOf`, which are not purely mental (they may have social or institutional content), but still it seems unlikely that a person could be other person’s apprentice without knowing it.

The domain of an ontoepistemic relation is that of “knowers”, and when a state of affairs occurs, in which $R(x, y)$ is true, then the knower x also knows it. Formally, it is the representation of the ontoepistemicity property based on axiom (60):

$$OE(R) \equiv_{Def} (R(x, y) \leftrightarrow K(x, R(x, y))) \quad (62)$$

E.g., the predicate *hates* is ontoepistemic because if x hates y then x always knows that s/he hates y (this is a simple example because the *hates* property is mental). On the other hand, the predicate *is-FatherOf* is not ontoepistemic because there can be situations when x isFatherOf y , but x does not know it.

It is also quite common that a relation is ontoepistemic while its inverse relation is not. E.g., if a person *likes-to-eat* (R_{lte}) some food then she knows it, however the food does not know that it is the favorite food of some person. This is consistent with the definition of ontoepistemic relation because such predicate is not symmetric and the domain of the relation is different from the domain of its inverse. If this asymmetric relation R_{lte} has domain X and range Y that are disjoint $X \cap Y = \emptyset$ then the following holds:

$$\forall x \forall y R_{lte}(x, y) \leftrightarrow (K(x, (R_{lte}(x, y))) \wedge \neg K(y, (R_{lte}(x, y)))) \quad (63)$$

In the case of symmetric relations then obviously if a relation is ontoepistemic then also inverse relation is ontoepistemic.

This analysis is by no means restricted to the FOAF Relationship Vocabulary: using vocabularies such as SIOC (Semantically-Interlinked Online Communities)²⁰ that describes online communities (e.g., message boards, wikis, weblogs, etc.), we can expect that at least in the time when a user joined some online community, s/he had to know about it. This is implied by the presupposition that everyone is aware of (knows of) his/her voluntary actions.²¹ More strictly, if somebody is “creator of” something (as of SIOC messages), we can assume (at least within a narrow time frame), that s/he knows to have created something.

Other examples include the Sport Ontology,²² where we can argue that if somebody “participates in” some football match with other persons, s/he has to know them, and the Theatre Ontology,²³ where it is hardly possible for the actors not to know each other, at least for the main characters.

All these examples are taken from vocabularies considered to be part of the Linked Data initiative and listed, e.g., on the LOV (Linked Open Vocabulary) portal.²⁴ Moreover, in Sport and Theatre ontology cases, we observe that the ontoepistemic character of some predicates goes beyond purely interpersonal relationships. E.g., someone participating in a competition or in a performance has to know about them.

We can now determine what relations from the Relationship Vocabulary are ontoepistemic. We have also performed an analysis of these predicates from the point of view of general formal ontology, in order to improve their formalization beyond the ontoepistemic properties. The results are presented along definitions from the original FOAF Relationship Vocabulary in Table 1. We have also determined which of these properties are symmetric, asymmetric and irreflexive, and compared the results to the Relationship Vocabulary definitions. Irreflexivity is implied by asymmetry, but there are also relations that are symmetric but also irreflexive (e.g. `engagedTo`). We did not assign irreflexivity or reflexivity to properties, which we considered questionable (e.g. it is psychological question, whether one can be `enemyOf` himself and it depends on application, whether one `hasMet` himself). The table has the following columns:

- (1) *Property* – the name of the property.
- (2) *RV Super-prop* – the super-properties of a property as from the FOAF Relationship Vocabulary. We omit `differentFrom` because it is defined as super-property for all properties. We use *k* for knows.

²⁰<http://rdfs.org/sioc/spec/>.

²¹We do not want to enter here the debate about free will, awareness, and knowing, and assume this presupposition as a common sense one.

²²<http://www.bbc.co.uk/ontologies/sport/2011-02-17.shtml>.

²³<http://lukeblaney.co.uk/semweb/theatre>.

²⁴<http://labs.mondeca.com/dataset/lov/index.html>.

- (3) *Super-prop* – the super-properties of a property as from our analytic revision. We use k for knows and kO for knowsOf.
- (4) *Ontoepist* – whether a property is ontoepistemic.
- (5) *RV Symm* – symmetry as defined in the FOAF Relationship Vocabulary.
- (6) *Symm* – symmetry as from our analytic revision.

It might be interesting to ask which relations are irreflexive. Irreflexivity is implied by asymmetry so we may conclude that all relations that are asymmetric are also irreflexive. For other relations the question is puzzling and in many cases difficult to answer from the point of view of formal ontology. It may belong to psychology or other disciplines and is matter of an interpretation. Can a person be influenced by itself? Can a person lose contact with itself? Can a person be employed by itself in true sense of term employment? Because these questions do not have definite answers we did not include irreflexivity in the Table 1.

For obvious reasons, relations that are not ontoepistemic (e.g., `parentOf`) should not be subproperties of `knowsOf`:

$$\neg OE(\phi) \rightarrow \neg \forall x \forall y (\phi(x, y) \rightarrow kO(x, y)) \quad (64)$$

6. Revised formal axiomatization of the Relationship Vocabulary

In this section we present a revised axiomatization of human relations from the FOAF Relationship Vocabulary. While we respect the original conceptualization (as evidenced in code and textual definitions), we provide refinements that enable to work around some problems, as well as to extend the reuse of data in different contexts. Our axiomatization is minimal, in the sense that we only cover a minimal set of ontological assumptions and do not provide a complete axiomatic theory of the domain. We want to use ontological analysis to make the FOAF Relationship Vocabulary more robust and intuitive, but we also want to respect the basic design choices, which were tailored to *lightweight* semantic web applications. The axioms will then be useful for data injection to and extraction from FOAF Relationship Vocabulary, i.e. as a tool for understanding, processing and transformation of FOAF data, also enabling previously impossible reasoning capabilities.

We introduce here revised axioms for human relations from the Relationship Vocabulary, ordered by type.

For brevity, we do not include here axioms that are already in Table 1, i.e. subsumption axioms by `knows`, ontoepistemicity axioms, and symmetry axioms (except for `knows` that is not included in that table).

As we have discussed in Section 4, the relation `differentFrom` from the FOAF Relationship Vocabulary needs to be revised in terms of asymmetry (in the sense of antisymmetry + irreflexivity): the cases in which asymmetry holds after revision are also indicated in Table 1.

A (possibly intuitive) transitivity axiom for `livesWith` is not generally assumed because, as a binary relation, it cannot take time into account, while persons can live with different people in different times. Generally speaking, temporal interpretation is left to the application and data. Usually it is presumed that a dataset of relations represents true relations at some point of time, usually time of creation of the dataset (e.g., FOAF files). In cases of continually updated datasets their content usually represents relations true at current time. However more complex handling of temporal structuring of human relations can be part of future work on this topic.

Table 1
Properties of relations as from FOAF Relationship Vocabulary and as from our revision

Property	RV Super-prop	Super-prop	Ontoepist	RV Symm	Symm
acquaintanceOf	k	k, kO	yes	sym	sym
ambivalentOf	–	kO	yes ^a	–	not sym
ancestorOf	–	–	no	–	asym
antagonistOf	k	kO	yes	–	not sym
apprenticeTo	k	k, kO	yes	–	asym
childOf	k	–	no	–	asym
closeFriendOf	k	k, kO	yes	sym	sym
collaboratesWith	k	k, kO	yes	sym	sym
colleagueOf	k	–	no ^b	sym	sym
descendantOf	–	–	no	–	asym
employedBy	k	kO	yes	–	not sym ^c
employerOf	k	– ^d	no	–	not sym ^e
enemyOf	k	kO	yes	–	not sym
engagedTo	k	k, kO	yes	sym	sym
friendOf	k	k, kO	yes	sym	sym
grandchildOf	k	–	no	–	asym
grandparentOf	k	–	no	–	asym
hasMet	k	k, kO	yes ^f	sym	sym
influencedBy	–	–	no ^g	–	not sym
knowsByReputation	–	kO	yes	–	not sym
knowsInPassing	k	kO	yes	–	not sym
knowsOf	–	kO	yes	–	not sym
lifePartnerOf	k	k, kO	yes	sym	sym
livesWith	k	k, kO	yes ^h	sym	sym
lostContactWith	k	kO	yes	sym	not sym
mentorOf	k	k, kO	yes	–	asym
neighborOf	k	–	no	sym	sym
parentOf	k	–	no	–	asym
siblingOf	k	–	no	sym	sym
spouseOf	k	k, kO	yes	sym	sym
worksWith	k	– ⁱ	no	sym	sym
wouldLikeToKnow	–	kO	yes	–	not sym

^aThe definition says that x “has mixed feelings or emotions” towards y . We suppose that a conscious agent is aware of his/her feelings or emotions. Therefore s/he also `knowsOf` the person towards whom s/he has these emotions.

^bThe definition says: “A property representing a person who is a member of the same profession as this person”. We suppose that usually people do not necessarily know all people who are members of the same profession. It is also different from relation `collaboratesWith`, which requires symmetric knowledge of both persons involved.

^cIf we accept concept of self-employed person then relations `employedBy` and `employerOf` may be considered antisymmetric: $\forall x \forall y ((R(x, y) \wedge R(y, x)) \rightarrow (x = y))$. See page 242.

^dNote that an employer who has thousands of employees usually does not know each of them. See page 242.

^eSee footnote ^c.

^fWe assume that a relevant meaning of `hasMet` requires at least *having been introduced to*, i.e. not just *having occurred at the same place at the same time*.

^gA person does not necessarily know that s/he was (in his/her work etc.) been influenced by someone else.

^hWe understand it as a social relation, so it is ontoepistemic.

ⁱThis relation is defined as “a property representing person who works for the same employer as this person”. This does not imply that they know each other.

Social knowing relations:

$$\text{knows}(x, y) \leftrightarrow \text{knows}(y, x) \quad [\textit{symmetry}] \quad (65)$$

$$\text{knows}(x, y) \rightarrow (\text{knowsOf}(x, y) \wedge \text{knowsOf}(y, x)) \quad (66)$$

$$\text{knowsByReputation}(x, y) \rightarrow \text{knowsOf}(x, y) \quad (67)$$

$$\text{knowsInPassing}(x, y) \rightarrow \text{knowsOf}(x, y) \quad (68)$$

$$\text{wouldLikeToKnow}(x, y) \rightarrow \text{knowsOf}(x, y) \quad (69)$$

Family relations:

$$\text{parentOf}(x, y) \leftrightarrow \text{childOf}(y, x) \quad (70)$$

$$\text{grandparentOf}(x, y) \leftrightarrow \text{grandchildOf}(y, x) \quad (71)$$

$$\text{childOf}(x, y) \rightarrow \text{descendantOf}(x, y) \quad (72)$$

$$\text{grandchildOf}(x, y) \rightarrow \text{descendantOf}(x, y) \quad (73)$$

$$\text{parentOf}(x, y) \rightarrow \text{ancestorOf}(x, y) \quad (74)$$

$$\text{grandparentOf}(x, y) \rightarrow \text{ancestorOf}(x, y) \quad (75)$$

$$\text{descendantOf}(x, y) \leftrightarrow \text{ancestorOf}(y, x) \quad (76)$$

$$(\text{descendantOf}(x, y) \wedge \text{descendantOf}(y, z)) \rightarrow \text{descendantOf}(x, z) \\ [\textit{transitivity}] \quad (77)$$

$$\text{sibling}(x, y) \rightarrow \exists z(\text{parentOf}(z, x) \wedge \text{parentOf}(z, y)) \quad (78)$$

$$\text{grandparentOf}(x, y) \rightarrow \exists z(\text{parentOf}(x, z) \wedge \text{parentOf}(z, y)) \quad (79)$$

Friendship and enmity relations:

$$\text{closeFriendOf}(x, y) \rightarrow \text{friendOf}(x, y) \wedge \text{hasMet}(x, y) \quad (80)$$

$$\text{friendOf}(x, y) \rightarrow \text{knows}(x, y) \wedge \neg \text{enemyOf}(x, y) \quad (81)$$

$$\text{acquaintanceOf}(x, y) \rightarrow \text{knows}(x, y) \quad (82)$$

$$\text{lostContactWith}(x, y) \rightarrow \text{knows}(x, y) \quad (83)$$

$$\text{enemyOf}(x, y) \rightarrow \text{knowsOf}(x, y) \wedge \text{antagonistOf}(x, y) \quad (84)$$

$$\text{ambivalentOf}(x, y) \rightarrow \text{knowsOf}(x, y) \wedge \neg \text{enemyOf}(x, y) \quad (85)$$

$$\text{antagonistOf}(x, y) \rightarrow \text{knowsOf}(x, y) \quad (86)$$

Living together relations:

$$\text{livesWith}(x, y) \rightarrow \text{knows}(x, y) \wedge \text{hasMet}(x, y) \quad (87)$$

$$\text{lifePartnerOf}(x, y) \rightarrow \text{livesWith}(x, y) \quad (88)$$

$$\text{lifePartnerOf}(x, y) \rightarrow \text{knows}(x, y) \wedge \text{hasMet}(x, y) \quad (89)$$

$$\text{spouseOf}(x, y) \rightarrow \text{lifePartnerOf}(x, y) \quad (90)$$

$$\text{engagedTo}(x, y) \rightarrow \text{knows}(x, y) \wedge \text{hasMet}(x, y) \quad (91)$$

Work connection relations:

$$\text{apprenticeTo}(x, y) \leftrightarrow \text{mentorOf}(y, x) \quad (92)$$

$$\text{apprenticeTo}(x, y) \rightarrow \text{influencedBy}(x, y) \wedge \text{knows}(x, y) \wedge \neg \text{enemyOf}(x, y) \quad (93)$$

$$\text{mentorOf}(x, y) \rightarrow \text{knows}(x, y) \wedge \neg \text{enemyOf}(x, y) \quad (94)$$

$$\text{collaboratesWith}(x, y) \rightarrow \text{knows}(x, y) \quad (95)$$

Axiom (66) implies that `knows` is sub-property of `knowsOf`. This is consistent with observation that property `knows` is symmetric while property `knowsOf` is not.

About close friendship (axiom (80)), we assume that it is not possible to be close friend with someone without meeting him/her (in some physical or virtual sense). About `antagonistOf`, we assume it as weaker than `enemyOf` (axiom (84)).

7. Survey of human relationships in Linked Open Data/Vocabularies

The fact that FOAF (and probably also RV, rooted in it) is undoubtedly the best known semantic vocabulary allowing to describe human relationships does not necessarily mean that it is the one most massively used for providing semantics to instances of linked data on the web. We thus performed an empirical analysis, taking advantage of the results of the linked data statistics observatory LODStats as presented by Auer et al. (2012), as available through the Linked Open Vocabularies (LOV) portal, at <http://lov.okfn.org/dataset/lov/stats/>, as well as of our own investigations of semantic patterns from within DBpedia – see works by Nuzzolese et al. (2011) and Presutti et al. (2011).

7.1. The DBpedia and schema.org HR case

The most used HR vocabulary is actually a general vocabulary, the DBpedia ontology,²⁵ which uses (at version 3.9) 52 relations with a person class in both domain and range positions, on at least 100 RDF triples (simple facts), for a total of 519869 triples (Table 2).²⁶

We have also applied our ontoepistemic property to DBpedia (and schema.org, see below) vocabulary (see again Table 2). The practical impact of the property becomes evident as soon as we would like to use HR relations to reconstruct social networks of public persons. For example, Lloyd Osborne was influenced by Robert Louis Stevenson, spouse of Fanny Stevenson, who is a relative of Lloyd Osborne. In principle, we cannot be sure that R.L. Stevenson was aware of Osborne being a relative of his as well, or that he influenced him. Osborne on its turn was probably aware of the influence of R.L. Stevenson's, but

²⁵<http://dbpedia.org/ontology/>.

²⁶The actual number of human relations and triples used in all DBpedia datasets is higher, probably a bit more than 1 million triples and about 1 thousand relations. Most of those relations come from a less controlled schema that is being refactored, and most of them are used in a small number of triples.

Table 2
The 30 most used DBpedia HR relations (> 1000 triples)

DBpedia relation	Person # triples	All # triples	Ontoepistemic?
currentMember	232898	233481	n/a ^a
successor	57324	70429	yes ^b
associatedBand	27275	99824	no ^c
associatedMusicalArtist	27275	99824	no ^d
spouse	23735	24540	yes
parent	22322	22822	no
influencedBy	19489	21504	yes
creator	11167	22855	n/a ^e
influenced	9625	10650	no ^f
child	9606	10118	no
primeMinister	8180	8211	yes
president	7768	10091	yes
relative	6052	6409	no
doctoralStudent	5400	5532	yes
relation	4967	5208	yes
predecessor	4942	17160	yes ^g
portrayer	4381	4544	no
monarch	4153	4178	yes
doctoralAdvisor	4070	4166	yes
trainer	3221	6051	yes
governor	2247	2264	yes
voice	1995	8284	no ^h
formerCoach	1872	1874	yes
appointer	1482	1506	yes
coach	1427	12941	yes
lieutenant	1394	1417	yes
deputy	1376	1387	yes
formerPartner	1301	1305	yes
team	1194	1427116	yes
associate	1173	1182	no

^aThis relation is one of the projections of the (reified) n -ary relation used in DBpedia to represent the situation of an athlete to be enrolled in a team. From a formal ontology perspective, the entities reifying the individual relationships may be considered as temporal slices of a person, or as tropes/qua-entities of a person playing a certain role. It is massively used in DBpedia data, but is not of interest for the HR relations we are considering here.

^bThis relation seems mostly used for persons that play similar roles at different times. This is a typical case of binary relations that should be actually modeled as n -ary ones, in order to make room for roles and time intervals.

^cThe observed ambiguity between musical artists and bands causes the application of this relation to persons, not only bands.

^dIt is the inverse of associatedMusicBand.

^eThe DBpedia `Person` class includes also fictional characters, which of course can have a creator.

^fIt is the inverse of influencedBy. This relation is ontoepistemic only on the side of the influenced, which can be hardly influenced without knowing the influencer.

^gIt is the inverse of successor.

^hThe DBpedia `Person` class includes also fictional characters, which of course are given a voice from a (natural) person. This relation can also be used for natural persons that are doubled in other languages however.

still we cannot infer he was aware of being a relative of his. On the contrary, in a regular social network analysis scenario, we would easily flatten these relations, assuming the Stevensons and Osborne form a social subgraph.

The benefits of a foundational axiomatization can be described empirically for human relationships. When considering an axiomatization for large, crowdsourced, and partly uncontrolled data, new interesting problems appear which show the importance of ontological and logical methods to improve data quality, but also pose a challenge for the sustainability of data quality assumptions.

DBpedia is quite interesting for ontology design because its ontology schema is partly misaligned with the real usage in data. In practice, the domain and range axioms declared in the schema are not based on the actual distribution of types of entities referred to in the triples, either because of data dirtiness, or because of missing data analysis. For example, the relation `almaMater` is declared to have domain `Person` and range `EducationalInstitution`, and it results to be used in 64928 triples; however, 87 of them are actually used with range `Person`, data that should be probably cleaned up. On the other hand, the `appointer` relation is declared with range `Person`, but no domain is provided, although all its 1506 triples have a `Person` in their domain position.

Even worse, in some cases misalignment with data generates ontological conflicts that in accurate logical representations would generate also logical inconsistencies. For example, the `associatedBand` relation, declared with no specific domain, and range `Band`, is consequently used in 64193 triples with instances of the class `Band` (a subclass of `Organisation`) in the range, but in 31187 triples with instances of the class `MusicalArtist` (a subclass of `Person`). This is not just dirty data (about 2/3 against about 1/3), but an inaccuracy in creating the class taxonomy or relation axioms. The DBpedia ontology schema does not define disjointness axioms extensively, but if these data and schema are used in a more rigorous context, and experts would intuitively make `Person` and `Organisation` disjoint classes, all musical artists declared in the range of `associateBand` would be inferred to be also bands. Now, since the two superclasses are disjoint, the consequence would be a massive inconsistency in the ontology.

A different problem can be seen at the schema level. If the DBpedia ontology schema is aligned to DOLCE,²⁷ some properties become incoherent. For example, the relation `brand` is declared with domain `WrestlingEvent` (a subclass of `Event`, aligned to DOLCE class `Event`), and range `TelevisionShow` (a subclass of `Work`, aligned to DOLCE class `InformationEntity`), in addition, the relation itself is aligned to the DOLCE relation `coparticipatesWith`, which has the DOLCE class `Object` for both domain and range. Since in DOLCE `Event` and `Object` are disjoint, an incoherence emerges for the domain of `brand` as subproperty of `coparticipatesWith`. In this case, the stronger axiomatization of DOLCE reveals a possible problem for the DBpedia ontology, but in principle even a possible problem for foundational ontologies in general: is it correct to assume that only objects can participate in events, and therefore co-participate?

A recent result by Paulheim and Gangemi (2015) on cleaning up DBpedia ontology and data after the linking to DOLCE-Zero (a simplified version of DOLCE + DnS, downloadable from <http://www.ontologydesignpatterns.org/ont/dul/D0.owl>) proves the practical feasibility as well as the substantial gain when using foundational axioms in order to improve linked open data: more than 3 million triples have been found inconsistent, anti-patterns emerged, and design solutions have been suggested. The trade-off between advantages of foundational axioms, and the difficulty of enforcing all potential distinctions, is also discussed.

²⁷An OWL version is <http://www.ontologydesignpatterns.org/ont/dul/DUL.owl>.

Table 3
Schema.org HR relations

Relation	Universe	Gloss	Ontoepistemic?
children	Person	<i>A child of the person.</i>	no
colleague	Person	<i>A colleague of the person.</i>	no
follows	Person	<i>The most generic uni-directional social relation.</i>	yes
knows	Person	<i>The most generic bi-directional social/work relation.</i>	yes
parent	Person	<i>A parent of this person. Supercedes parents.</i>	no
relatedTo	Person	<i>The most generic familial relation.</i>	no
sibling	Person	<i>A sibling of the person. Supercedes siblings.</i>	no
spouse	Person	<i>The person's spouse.</i>	yes

Schema.org is also a source of human relations, and it is becoming a de facto ontology standard for search engine optimization, being maintained and enforced by Google, Yahoo, Microsoft, and Yandex. There is no easy detection of the size of available data for schema.org, but it is interesting to report on what HR are currently included in it, as shown in Table 3.²⁸

7.2. Other LOD HR vocabularies

Besides HR fragments of general ontologies like DBpedia and schema.org, several domain-oriented vocabularies also focus on HR relations. From the of approx. 2800 vocabulary entities listed in LOV,²⁹ we manually selected possible suspects for human relationships. Where in doubt, we checked the underlying definitions in the source vocabularies (for example, the term ‘parent’ in entity names sometimes denotes a thesaurus relationship rather than a human relationship, and the term ‘family’ is also used in the sense of biological taxon).

In Table 4 we can see that the most massively used HR vocabulary at the domain level is GNDO, the German National Library ontology. Since library linked data presumably cover human relationships of authors (or other encyclopedically relevant persons), this use of HR vocabulary is mostly complementary to the use of FOAF/RV, which is typically used by ‘ordinary’ persons in their personal profiles.

Additionally, we also examined in detail all the LOV vocabularies that had been arranged, by the LOV curators, into a ‘cluster’ named ‘People’. There are 14 vocabularies aside FOAF and RV. Of these, 8 contain entities with some relevance to the HR topic. The vocabularies are (from this restricted point of view) reviewed below as roughly ordered from those pre-dominantly focused on officially recognised relationships, through those dealing with mere ‘knowing’, to ontologies that focus on modelling inner feelings of persons. Note that GNDO is not member of this cluster as its coverage is much broader; there is however the Agrelon ontology coming from the same provider (German National Library).

*Bio ontology*³⁰. The vocabulary defines parenthood via direct relationships (the first two also being in Table 4, presumably due to an exemplifying instance): Father, Mother and Child.³¹ Note that, e.g., the last is commented as “a strict definition of child that does not include adopted children, step-children or similar non-biological relationships. The Relationship vocabulary may be more suitable for broader

²⁸Taken from <http://schema.org/Person>.

²⁹The list does not include all entities defined in more than 400 vocabularies registered by LOV, but only those referenced either in a LOD dataset or in another LOV vocabulary.

³⁰<http://purl.org/vocab/bio/0.1>.

³¹The usual lowercase convention for properties is not followed.

Table 4
LOV entities with non-zero reuse in LOD data, as of March 10, 2014

Vocabulary entity	Use in LOD	Reuse: vocabs.	Reuse: entities
GNDO: http://d-nb.info/standards/elementset/gnd			
acquaintanceshipOrFriendship	100706	0	0
familialRelationship	69034	0	0
relatedPerson	4583	0	0
professionalRelationship	954	0	0
FOAF: http://xmlns.com/foaf/0.1			
knows	242	5	30
RV: http://purl.org/vocab/relationship			
siblingOf	226	0	0
childOf	173	1	2
parentOf	173	1	1
apprenticeTo	101	0	0
mentorOf	101	0	0
spouseOf	90	0	0
closeFriendOf	2	0	0
BIO: http://purl.org/vocab/bio/0.1			
Father	1	0	0
Mother	1	0	0

types of parent/child relation”. The remaining relationships are implicit in certain events: *Adoption*, *Marriage*, *Divorce* and the like. Persons are connected to these events via the *Partner* property. Interesting is the notion of *GroupEvent*, which is defined as: “A type of event that is principally about one or more agents and their partnership. Other agents may be involved but the event is most significant for the partner agent”. It seems that such an event is principally about ‘one or more agents and their partnership’: the current agent, about whom the bio data are, is always implicitly involved.

*PoderVocab*³². The vocabulary seems to have been originally designed as a proxy to the Bio ontology, for the sake of Poderopedia, a collaborative data base of public people and organizations in Chile. Its coverage is similar to Bio, but it extends its less developed parts. In particular, it introduces various kinds of formal and informal human relationships, divided into social connections (*Acquaintance*, *Classmate*, *Friendship*), ‘sentimental relationship’ connections (including both legal relationships – *Marriage*, *DomesticPartnership*, *CivilUnion* – and informal *Dating*). Note that *Marriage* is, unlike in Bio, truly meant as a relationship (indirect subclass of *bio:Relationship*) and not as an event of establishing the relationship.

*VCard ontology*³³. The new vocabulary specification from May 2013 includes a large collection of ‘type code sets’. Among them is the *Related Type* set, containing, e.g., *Acquaintance*, *Child*, *Colleague*, *Contact*, *Coresident*, *Coworker*, *Friend*, *Kin*, *Me*, *Met*, *Muse*, *Neighbor*, *Parent*, *Sibling*, *Spouse* and *Sweetheart*. In the VCard context, these are presumably types of visit card meant for easier management of the database by its owner.

³²<http://dev.poderopedia.com/documentation/index>.

³³<http://www.w3.org/TR/2013/WD-vcards-rdf-20130502/>.

*Agrelon ontology*³⁴. The ontology creators followed the pattern of systematically coupling (by *rdfs:seeAlso*, i.e., at extra-logical level) pairs of plain property and reified property (i.e., class). For symmetric properties they only differ in capitalization, e.g., *hasAdoptiveSibling* (property) vs. *HasAdoptiveSibling* (class). For non-symmetric properties the reified property name mixes both inverse properties, e.g., *HasAdoptiveChildParent* (for *hasAdoptiveChild* and *hasAdoptiveParent*). The collection is rather large (it includes relationships that do not occur in other ontologies, such as ‘multiple birth sibling of’ or ‘physician of’), but (consciously, as the authors note) incomplete. The model does not link to any other ontology, nor to related concepts in the GNDO ontology.

*Gen ontology*³⁵. The ontology for genealogical modelling³⁶ is an extension of the Bio ontology, focused on connecting the agent entities with event entities. Unlike Bio, agents are connected with events via specialised properties such as *death* or *spouse* and not by the general *bio:Partner* property.

*Online presence ontology*³⁷. In this ontology, closely connected to the popular SIOC ontology,³⁸ two properties indirectly model certain kinds of human relationships, both in the context of a *SharingSpace*, understood as a ‘group of people belonging to a space for sharing online presence data’. The first is *commonInterest*, which links the sharing space to a document representing the common interest of members of the sharing space. The second is *closestFriendsOf*, which links to people ‘whose closest friends belong to the sharing space’.

*Appearances ontology*³⁹. The ontology has originally been designed for recording physical characteristics of individual historical persons (Great War soldiers) from their medical files. However, its coverage of characteristics (sex, gender, skin, eye and hair colour) makes it suitable for general-purpose description of persons. It contains four ‘relationship’ properties: *hasRomanticAversion*, *hasRomanticPreference*, *hasSexualAversion* and *hasSexualPreference*, relating a person to an appearance feature as such (rather than to a real person holding such appearance). Its relevance to the HR topic is thus indirect.

*Emoca ontology*⁴⁰. The ontology allows to describe various kinds of emotions. Its class *Trait* refers to ‘any kind of philia and phobia’, which can, in turn, be linked by a *relatesTo* property to any resource, i.e. possibly to another human. Through a complex modeling pattern borrowed from a psychological theory, the trait is then connected to an *Emotion* such as joy, fear or disgust.

8. Conclusions

Our research is motivated by the growing importance of the Linked Data initiative that is naturally connected with renewed interest in broadly used vocabularies such as FOAF. We have provided an overview of some FOAF extensions, as well as of alternative approaches for describing human relationships. We then focused on the FOAF Relationship Vocabulary, attempting at its revision on formal ontological grounds. The analysis lead us to compare a reconstructed formal axiomatization of that

³⁴<http://d-nb.info/standards/elementset/agrelon.owl>.

³⁵<https://github.com/joshhansen/vocab-gen/blob/master/gen.ttl>.

³⁶It is not directly accessible from LOV at the moment, but only via its github repository.

³⁷<http://online-presence.net/ontology.php>.

³⁸<http://sioc-project.org/>.

³⁹<http://rdf.muninn-project.org/ontologies/appearances>.

⁴⁰<http://ns.inria.fr/emoca/>.

vocabulary to a proposed revised axiomatization that solves many issues and improves the reasoning potential, without changing the basic architectural choices of FOAF (lightweight expressivity, directness of relations, etc.). Our axiomatic revision is based on several theoretical remarks concerning the nature of properties used to describe human relationship, as well as on a newly defined *ontoepistemic* meta-property as characteristic of some of these predicates. The provided systematization, even if limited to basic properties of knowing, knowing of, and knowing of knowing, can already be used to perform safer reasoning on human relationship data, since linked data do not even address those basic properties.

A broader survey of human relations (and general patterns used to express them) in existing linked vocabularies has also been presented as complement, and an overview of the state of their usage in DBpedia has been proposed, with some evidence of how a stronger axiomatization can be helpful for the human relations domain.

Acknowledgements

This work has been partially supported by the VŠE IGA projects No. F4/34/2014 and F4/90/2015. We also acknowledge long-term institutional support of research activities by Faculty of Informatics and Statistics, University of Economics, Prague.

References

- Auer, S., Demter, J., Martin, M. & Lehmann, J. (2012). LODStats – An extensible framework for high-performance dataset analytics. In *Proceedings of the 18th International Conference on Knowledge Engineering and Knowledge Management (EKAW'12)*, Galway City, Ireland. Berlin, Heidelberg: Springer.
- Berners-Lee, T. (2009). *LinkedData*. <http://www.w3.org/DesignIssues/LinkedData.html>.
- Berners-Lee, T., Chen, Y., Chilton, L., Connolly, D., Dhanaraj, R., Hollenbach, J., Lerer, A. & Sheets, D. (2006). Tabulator: Exploring and analyzing linked data on the semantic web. In *Proceedings of the 3rd International Semantic Web User Interaction Workshop*, Athens, Georgia, USA, November 6, 2006.
- Bizer, Ch., Cyganiak, R. & Heath, T. (2007). How to publish linked data on the web, <http://www4.wiwi.fu-berlin.de/bizer/pub/LinkedDataTutorial/>.
- Bizer, Ch., Heath, T. & Berners-Lee, T. (2009). Linked data – The story so far. *International Journal on Semantic Web and Information Systems*, 5(3), 1–22. doi:10.4018/jswis.2009081901.
- Bizer, Ch., Heath, T., Idehen, K. & Berners-Lee, T. (2008). Linked data on the web. In Huai et al. (Eds.), *Proceedings of the 17th International Conference on World Wide Web (WWW 2008)*, Beijing, China, April 21–25, 2008 (pp. 1265–1266). doi:10.1145/1367497.1367760.
- Brentano, F. (1874). *Psychology from an Empirical Standpoint* (pp. 88–89). London: Routledge and Kegan Paul.
- Brickley, D. & Miller, L. (2010). FOAF Vocabulary Specification 0.97. <http://xmlns.com/foaf/spec/>. January 2010.
- Chisholm, R. (1982). *Knowledge as Justified True Belief. The Foundations of Knowing*. Minneapolis: University of Minnesota Press.
- Clark, P. & Porter, B. (1997). Building concept representations from reusable components. In *AAAI'97* (pp. 369–376). CA: AAAI Press.
- Davidson, D. (1967). The logical form of action sentences. In *The Logic of Decision and Action*. University of Pittsburgh Press.
- Frege, G. (1892). Über Sinn und Bedeutung. *Zeitschrift für Philosophie und philosophische Kritik*, 100, 25–50. (Translation: On sense and reference. In P. Geach and M. Black (Eds.), *Translations from the Philosophical Writings of Gottlob Frege* (3rd ed.). Blackwell. 1980).
- Gabbay, D.M. & Guenther, F. (Eds.) (2003). *Handbook of Philosophical Logic* (Vol. 10). Springer.
- Gangemi, A. (2005). Ontology design patterns for semantic web content. In *Proceedings of the Fourth International Semantic Web Conference 2005* (pp. 262–276).
- Gangemi, A. (2008). Norms and plans as unification criteria for social collectives. *Autonomous Agents and Multi-Agent Systems*, 17(1), 70–112. doi:10.1007/s10458-008-9038-9.

- Gangemi, A. & Mika, P. (2003). Understanding the semantic web through descriptions and situations. In R. Meersman, Z. Tari, D. Schmidt et al. (Eds.), *On The Move 2003 Conferences (OTM2003)*. Springer.
- Gangemi, A., Navigli, R. & Velardi, P. (2003). The OntoWordNet Project: Extension and axiomatisation of conceptual relations in WordNet. In *International Conference on Ontologies, Databases and Applications of Semantics (ODBASE 2003)*, Catania, Italy.
- Guizzardi, G., Wagner, G., Guarino, N. & van Sinderen, M. (2004). An ontologically well-founded profile for UML conceptual models. In A. Persson and J. Stirna (Eds.), *Advanced Information Systems Engineering*. Proceedings of 16th CAiSE Conference. Riga: Springer.
- Jacob, P. (2010). Intentionality. In E.N. Zalta (Ed.), *The Stanford Encyclopedia of Philosophy* (Fall 2010 ed.), <http://plato.stanford.edu/archives/fall2010/entries/intentionality>.
- Lakemeyer, G. (1993). All they know: A study in multi-agent autoepistemic reasoning. In *Proc. of the 13th International Joint Conference on Artificial Intelligence (IJCAI-93)* (pp. 376–381).
- Longo, C., Gangemi, A. & Cantone, D. (2013). Representing Kinship relations on the semantic web. In *Proceedings of OWLED 2013*. <http://ceur-ws.org/Vol-1080/>.
- Marek, W. & Truszczyński, M. (1991). Autoepistemic logic. *Journal of the ACM*, 38(3), 588–619. doi:10.1145/116825.116836.
- Masolo, C., Bottazzi, E., Catanacci, C., Ferrario, R., Gangemi, A. & Guarino, N. (2004). Social roles and their descriptions. In *Proceedings of the Ninth International Conference on the Principles of Knowledge Representation and Reasoning*. AAAI Press.
- Matsuo, Y., Hamasaki, M., Mori, J., Takeda, H. & Hasida, K. (2004). Ontological consideration on human relationship vocabulary for FOAF. In *Proceedings of the 1st Workshop on Friend of a Friend, Social Networking and Semantic Web*, Galway, Ireland.
- Mika, P. & Gangemi, A. (2004). Descriptions of social relations. In *Proceedings of the 1st Workshop on Friend of a Friend, Social Networking and the Semantic Web*, Galway, Ireland.
- Moore, R.C. (1985). Semantical considerations on non-monotonic logic. In *Artif. Int.* (Vol. 28, pp. 75–94).
- Nuzzolese, A.G., Gangemi, A. & Presutti, V. (2011). Encyclopedic knowledge patterns from Wikipedia page links. In *Proceedings of ISWC2011, the Ninth International Semantic Web Conference*. Springer.
- Paulheim, H. & Gangemi, A. (2015). Serving DBpedia with DOLCE – More than just adding a cherry on top. In *Proceedings of ISWC2015, the Thirteenth International Semantic Web Conference*. Springer.
- Presutti, V., Aroyo, L., Adamou, A., Schopman, B., Gangemi, A. & Schreiber, G. (2011). Extracting core knowledge from linked data. In *Proceedings of the Second Workshop on Consuming Linked Data, COLID2011, Workshop in Conjunction with the 10th International Semantic Web Conference 2011 (ISWC 2011)*. <http://ceur-ws.org/Vol-781/>.
- Quine, W.V.O. (1960). *Word and Object*. MIT Press.
- Reich, J.R. (2000). Ontological design patterns: Modelling the metadata of molecular biological ontologies, information and knowledge. In *DEXA 2000*.
- Rescher, N. (2005). *Epistemic Logic: A Survey of the Logic of Knowledge*. Univ. of Pittsburgh Press.
- Robb, D. & Heil, J. (2009). Mental causation. In E.N. Zalta (Ed.), *The Stanford Encyclopedia of Philosophy*. Stanford University. Summer 2009 ed.
- Russell, B. (1905). On denoting. In *Mind* (Vol. 14). Basil Blackwell.
- Russell, B. (1911). Knowledge by acquaintance and knowledge by description. In *Proceedings of the Aristotelian Society (New Series)* (Vol. XI (1910–1911), pp. 108–128).
- Sauermann, L., Cyganiak, R. & Völkel, M. (2007). *Cool URIs for the Semantic Web*. Germany: DFKI GmbH, Deutsches Forschungszentrum für Künstliche Intelligenz GmbH. Technical Memo TM-07-01.
- Schiffer, S. (1992). Belief ascription. *The Journal of Philosophy*, 89(10), 499–521. doi:10.2307/2941005.
- Stevens, R. & Lord, P. (2010). Reification of properties in an ontology. In *Ontogenesis*. <http://ontogenesis.knowledgeblog.org/993>.
- Stuart, D. (2012). FOAF Within UK Academic Web Space: A Webometric Analysis of the Semantic Web. In G. Widén and K. Holmberg (Eds.), *Social Information Research*, Library and Information Science, Vol. 5, pp. 173–191. Emerald Group Publishing Limited.
- Svátek, V. (2004). Design patterns for semantic web ontologies: Motivation and discussion. In *7th Conference on Business Information Systems*, Poznan, Poland.