Digging Ontology Correspondence Antipatterns

Anselmo Guedes, Fernanda Baião and Kate Revoredo

Department of Applied Informatics. Federal University of the State of Rio de Janeiro – UNIRIO, Rio de Janeiro, Brazil
{anselmo.guedes, fernanda.baiao, katerevoredo}@uniriotec.br

Abstract. A correspondence antipattern is a set of generic correspondences between two ontologies that represents an incorrect alignment. It is useful to help identify incorrect correspondences between two ontologies, thus improving the Ontology Matching process. The specification of a correspondence antipattern requires the identification and correct understanding of a relevant alignment problem, and its representation in a proper modeling language. In this work we investigate the last three editions of OAEI challenge datasets so as to identify correspondence antipatterns from frequent and recurring errors; the resulting antipatterns are formalized and discussed.

Keywords: ontology matching, correspondence antipatterns, inconsistent alignment.

1 Introduction

As the research and practice on Ontology become more popular and evolve, several ontology artifacts arise for the same universe of discourse. However, they differ among each other in several perspectives, such as distinct representation languages (syntactic heterogeneity), variations in names referring to the same entity (terminological heterogeneity), different conceptualizations for the same domain (conceptual heterogeneity) and entities being perceived differently (semiotic heterogeneity) [7]. The Ontology Matching area [7][8] deals with all these problems, being considered by many authors the key element for heterogeneity reduction between ontologies.

The Ontology Matching task consists in identifying the correct correspondences among entities of multiple ontologies, which it is a necessary condition for establishing the interoperability among them [8]. A number of techniques can be used to identify the correspondence between the entities of two ontologies, including the analysis of subsumption between classes and the similarity between the entity names. However, current results from state-of-the-art techniques are neither complete nor precise, i.e., they are not able to identify all existing correspondences between two ontologies and sometimes suggest correspondences that do not exist [9]. With regard to precision errors, suggesting a correspondence that does not exist may lead to either logical or ontological incompatibilities.

On the other hand, in the context of software development, *antipatterns* are considered a valuable tool for the identification of bad or incorrect practices in the software

development process. Antipatterns prevent or hamper the good conduct of the process of development and maintenance of software. In the context of ontology matching, bad solutions consist of incorrect (including missing) or problematic correspondences. A correspondence antipattern is a matching model for identifying problematic correspondences that may occur repeatedly in ontology matching processes.

Looking for correspondence antipatterns, we "dig" the alignments available by OAEI and apply the methodology to assist to build a correspondence antipattern proposed in [11] for the construction of them.

This work is divided as follows: Section 2 shows an overview about ontology correspondence antipatterns, Section 3, presents how we "dig" the correspondence antipatterns, Section 4 presents the related works and, finally, Section 5 presents the final considerations of this work.

2 Correspondence Antipatterns

Ontology matching identifies correspondences between the entities of multiple ontologies, and it is a necessary condition to establish interoperability between them [8]. According to Euzenat [7], technically the ontology matching process occurs by taking two ontologies *O* and *O'* as input, optionally considering a set of resources *r*, a set of parameters *p* and an initial alignment *A*. The result of this process is an alignment *A'* between the ontologies *O* and *O'*, and can be represented as $A' = f(O, O', A, p, r)$. Basically, ontology matching is a process in which semantic links between entities of ontologies are established. As a result of this process, is obtained a set of semantic links, which are called correspondences, and their set is called alignment.

Due to possible precision errors that every ontology alignment tool is subject to, it may be the case that a correspondence included in an ontology alignment is not correct. Take, for example, a real problem illustrated in Figure 1, showing an alignment problem that occurs in the last three OAEI[1] editions, between *ConfOf* and *Edas* ontologies. The Ontology Alignment Evaluation Initiative (OAEI) is a coordinated international initiative whose goal is to evaluate the strengths and weaknesses of the ontology alignment tools. OAEI organizes annual campaigns addressing several domains, and publishes the results of the evaluated tools. The correspondence between the *ConfOf.Conference* and *Edas.Conference* is a problematic one. Let's analyze this case: suppose that *x* is an instance of *Edas.Conference*. Since an equivalent relationship between the entities *Edas.Conference* and *ConfOf.Conference* has been established, we may deduce that there is a possible world *w* in which *x* is an instance of *ConfOf.Conference* as well. Since *ConfOf.Conference* is a specialization of *ConfOf.Event*, *x* is necessarily instance of *ConfOf.Event* in *w*. We also notice that there is an equivalent relationship established between *ConfOf.Event* and *Edas.Conferece_Event*. Thus, *x* is also an instance of *Edas.Conference_Event* in *w*. However, considering that *Edas.Conference_Event* and *Edas.Conference* are disjoint classes, there sould be no possible world in which *x* instantiates both *Edas.Conference* and

[1] http://oaei.ontologymatching.org/

Edas.Conference_Event simultaneously, which leads to a contradiction, thus evidencing an alignment problem.

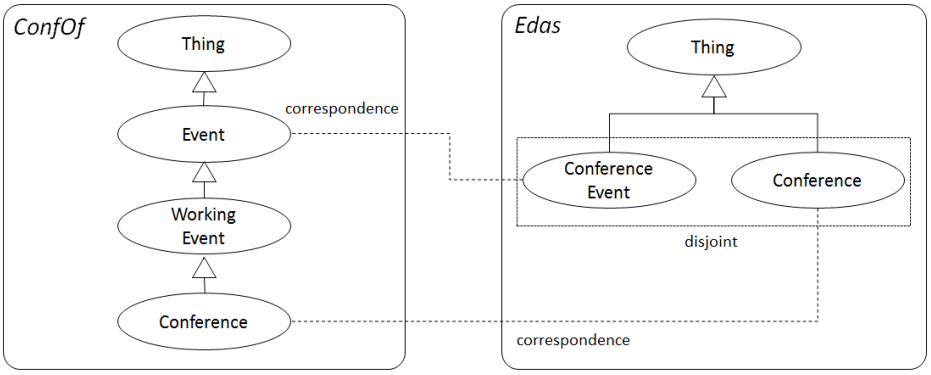

Fig. 1. Fragment of two ontologies and an alignment problem.

Patterns assist in building a collective experience based on the skills of domain specialists. On the other hand, an antipattern is a description of a given solution to a common problem that generates, definitely, negative consequences.

According to Guedes *et al* [11], a correspondence antipattern is a set of generic, domain-independent correspondences and/or non-correspondences which, when occurring combined with specific properties of the ontologies *O* and *O'* being aligned, characterizes an incorrect correspondence. The purpose of a correspondence antipattern is, then, to identify a mismatch in an alignment.

We may generalize the example scenario illustrated in Figure 1 as follows: Consider a class e1 in an ontology o1 that is a subclass of a class e2, which in turn is subclass of a class e3 in o1. If class e3 in the ontology o1 is equivalently matched with class e2 in ontology o2, and classes e1 (from ontology o2) and e2 (from ontology o2) are disjoint, then class e1 from ontology o1 cannot equivalently match class e1 from ontology o2. As shown in [28], this correspondence antipattern can be represented as follows:

$$\{(?o1{:}?e1 \equiv ?o2{:}?e1) \sqcap (?o1{:}?e1 \sqsubseteq ?o1{:}?e2) \sqcap (?o1{:}?e2 \sqsubseteq ?o1{:}?e3) \sqcap (?o1{:}?e3 \equiv ?o2{:}e2) \sqcap (?o2{:}?e1 \sqcap ?o2{:}?e2 \sqsubseteq \bot)\} \quad (1)$$

3 Digging Correspondence Antipatterns

As shown in [11], for the development of correspondence antipatterns, the first step is to have the correct understanding of the problem being treated. When properly understood, the identified problem can result in correspondence antipatterns templates. Figure 2 presents the methodology proposed in [11], which can assist in the construction of a correspondence antipattern. This methodology focuses on responding to key issues which are essential for an antipattern identification.

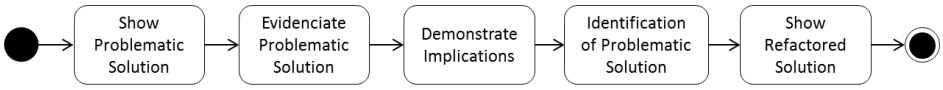

Fig. 2. Methodology to build a correspondence antipattern.

In this work this methodology was applied on the results provided by the OAEI in their last three editions (2011, 2012 and 2013) for the identification of correspondence antipatterns from recurring alignment errors generated by the evaluated tools. Each step of this process will now be briefly explained and illustrated in the OAEI scenario.

First step: *Show problematic solution*. According to Guedes *et al* [11], the first step towards the construction of correspondence antipatterns is the correct understanding of the problem being treated. To start the search for correspondence antipatterns, the first step was the identification of incorrect matches, or false positives, in the set of selected alignments. False positives are the matches found, which are not part of the real set of correspondences (true matches) that exist between two ontologies. Within the universe of identified incorrect matches, we selected those that most frequently occurred within the set of alignments. We selected incorrect matches that occurred over 50% of the maximum number of inconsistent matches identified (53) obtaining 40 inconsistent matches, as shown in Table 1. The columns *Ontology 1* and *Ontology 2* denotes the ontologies being aligned and the columns *Entity 1* and *Entity 2* denoted which entities appeared in the final alignment generated by tools that represent a mismatch.

Table 1. Inconsistent correspondences found in the set of alignments.

Error Nº	Ontology 1	Ontology 2	Entity 1	Entity 2	Total Problems	Total Alignments	Percent
1	Conference	Ekaw	Invited talk	Invited Talk	53	56	95%
2	Cmt	Iasted	Document	Document	53	57	93%
3	Edas	Ekaw	Presenter	Presenter	53	57	93%
4	Iasted	Sigkdd	Document	Document	53	57	93%
5	Conference	Ekaw	Conference participant	Conference Participant	52	56	93%
6	Edas	Iasted	Person	Person	52	57	91%
7	Conference	Iasted	Presentation	Presentation	52	56	93%
8	Conference	ConfOf	Conference	Conference	52	56	93%
9	Edas	Ekaw	Conference	Conference	52	57	91%
10	Cmt	Conference	Reviewer	Reviewer	51	56	91%
11	Conference	Edas	Conference	Conference	51	56	91%
12	ConfOf	Edas	Conference	Conference	50	57	88%
13	Conference	Ekaw	Conference	Conference	49	56	88%

14	Edas	Ekaw	ConferenceSession	Conference Session	48	57	84%
15	Cmt	ConfOf	Paper	Paper	47	57	82%
16	Conference	Ekaw	Paper	Paper	47	56	84%
17	Conference	Sigkdd	Conference	Conference	47	56	84%
18	Cmt	Conference	Paper	Paper	47	56	84%
19	ConfOf	Edas	hasEmail	hasEmail	46	57	81%
20	ConfOf	Ekaw	Paper	Paper	46	57	81%
21	Iasted	Sigkdd	pay	pay	44	57	77%
22	ConfOf	Edas	hasPhone	hasPhone	43	57	75%
23	Cmt	Sigkdd	name	Name	43	57	75%
24	Iasted	Sigkdd	obtain	obtain	42	57	74%
25	Cmt	ConfOf	writtenBy	writtenBy	41	57	72%
26	ConfOf	Edas	hasPostalCode	hasPostalCode	41	57	72%
27	ConfOf	Edas	hasStreet	hasStreet	40	57	70%
28	Cmt	Sigkdd	date	Date	40	57	70%
29	ConfOf	Edas	hasTopic	hasTopic	39	57	68%
30	mouse	human	MA 0000065	NCI C12685	39	45	87%
31	ConfOf	Edas	hasCountry	hasCountry	39	57	68%
31	mouse	human	MA 0000323	NCI C12378	39	45	87%
33	Cmt	Ekaw	writtenBy	writtenBy	38	57	67%
34	ConfOf	Ekaw	writtenBy	writtenBy	38	57	67%
35	mouse	human	UNDEFINED part of	UNDEFINED part of	37	45	82%
36	Conference	Iasted	is given by	is given by	37	56	66%
37	mouse	human	MA 0000003	NCI C12919	36	45	80%
38	Cmt	Edas	email	hasEmail	31	57	54%
39	Conference	Edas	Call for paper	CallForPapers	29	56	52%
40	Conference	Edas	has an email	hasEmail	27	56	48%

Second Step: *Evidentiate problematic solution.* For a solution to be considered problematic, this should in fact occur [11]. Table 1 confirms that these errors are recurrent. The Total Errors column of Table 1 shows the total occurrences of the correspondence in the last three editions of the OAEI.

Third Step: *Demonstrate Implications.* For each mapped incorrect correspondence, we analyzed the error and its implications. Jean-Mary *et al* [10] show semantic checks may be performed in an alignment so that it is validated. From the list present-

ed in [10], some were used in this work to demonstrate the implication of an incorrect alignment, as follows:

- *Disjointness-subsumption contradiction*: Suppose that (o1:e2 • o1:e1) and (o2:e2 ⊥ o2:e1). If an alignment contains both ⟨o1:e1, o2:e2⟩ and ⟨o1:e2, o2:e1⟩, this implies (o1:e2 ⊥ o1:e1) and (o2:e2 • o2:e1), which are both invalid and therefore cannot be verified.
- *Domain and range incompleteness*: Let o1:c1, o2:c1 be classes and o1:p1, o2:p1 be properties in o1 and o2 respectively, let dom(o1:p1) denote the domain of a property p1, and suppose o1:c1 ∈ dom(o1:p1). If an alignment contains both ⟨o1:c1, o2:c1⟩ and ⟨o1:p1, o2:p1⟩ this implies (o2:c1 ∈ dom(o1:p1)); domain incompleteness occurs when this axiom cannot be verified. A similar entailment exists for ranges.

Due to space limitation, only some cases are presented as follows.

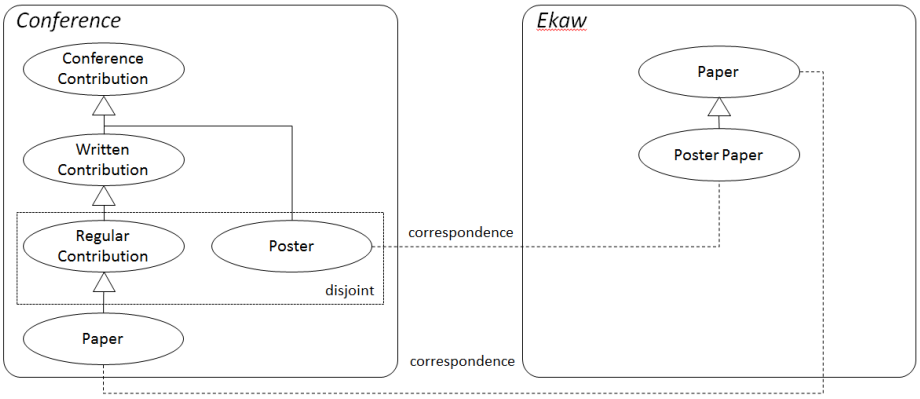

Fig. 3. Alignment problem between Conference and Ekaw ontologies.

Error Number 16: In the set of alignments analyzed, the correspondence ⟨*conference.paper, ekaw.paper,* ≡*,* _⟩ occurs 47 times. By analyzing the correspondence together with the aligned ontologies we established the following problem: let e1 be a class in an ontology o1 which is subclass of a class e2, which in turn is a disjoint class of a class e3, also in ontology o1. If class e1 in ontology o1 equivalently corresponds to class e1 in ontology o2, class e2 in ontology o1 corresponds to class e2 in ontology o1 and class e2 in o2 is a subclass of e1 in ontology o1, then there is a disjointness-subsumption contradiction alignment problem. Figure 3 shows the case identified on the correspondence number 16, where the above problem occurs.

Error Number 20: In the set of alignment analyzed, the correspondence ⟨*confof.paper, ekaw.paper,* ≡*,* _⟩ occurs 46 times. By analyzing the correspondence together with the aligned ontologies we established the follow problem: let e1 be a class in ontology o1 that is disjoint with class e2 in the same ontology o1, and a class e1 in ontology o2 that specializes class e2 in the same ontology o2. If class e1 in o1 equiva-

lently corresponds to class e1 in o2 and class e2 in o1 equivalently corresponds to class e2 in o2, then there is a disjointness-subsumption contradiction alignment problem. Figure 4 shows the case identified on the correspondence number 20, where the above problem occurs.

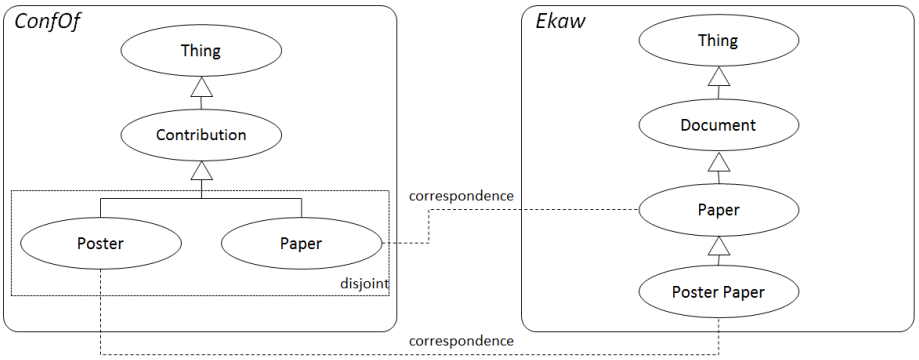

Fig. 4. Alignment problem between ConfOf and Ekaw ontologies.

Error Number 25: In the set of alignment analyzed, the correspondence ⟨*cmt.writtenBy, confof.writtenBy,* ≡*,* _⟩ occurs 41 times. By analyzing the correspondence together with the aligned ontologies we established the following problem: let p1 be a property in ontology o1 that has class e1 as its domain and class e2 as its range, both in ontology o1, and a property p1 in an ontology o2 that has class e1 as its domain class e2 as its range, both in ontology o2. If p1 in o1 equally corresponds to the property p1 in o2, but class e1 in o1 does not correspond to class e1 in o2 or class e2 in o1 does not correspond to class e2 in o2, then there is a domain and range incompleteness alignment problem. Figure 5 shows the case identified on the correspondence number 25, where the above problem occurs.

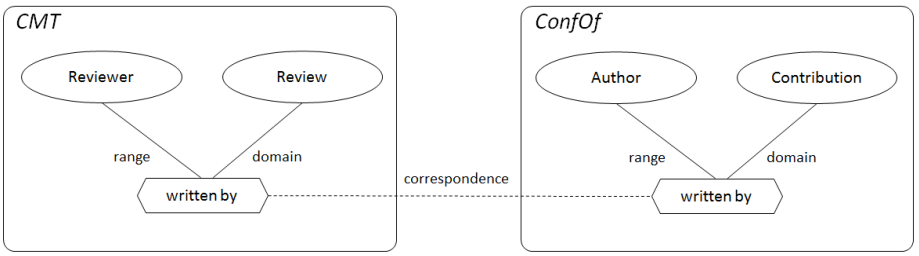

Fig. 5. Alignment problem between CMT and ConfOf ontologies.

Error Number 27: In the set of alignment analyzed, the correspondence ⟨*confof.hasStreet, edas.hasStreet,* ≡*,* _⟩ occurs 40 times. By analyzing the correspondence together with the aligned ontologies we established the following problem: let p1 be a property in an ontology o1 that has classes e1 and e2 as its domain, both in ontology o1, and a property p1 in an ontology o2 that has as its domain a class e1 in ontology

o2. If p1 in o1 equally corresponds to the property p1 in o2 and class e1 in o2 does not correspond to any domain class of p1 in o1, then there is a domain and range incompleteness alignment problem. Figure 6 shows the case identified on the correspondence number 27, where the above problem occurs.

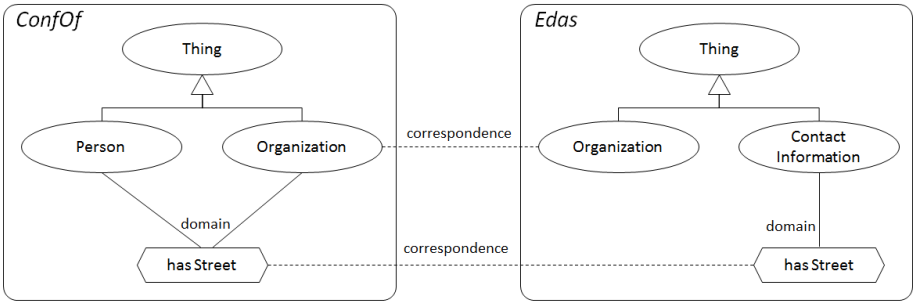

Fig. 6. Alignment problem between ConfOf and Edas ontologies.

Fourth Step: *Identification of the Problematic Solution.* The formal representation of how to identify an alignment problem is what gives life to correspondence antipattern. For each problem analyzed was created one correspondence antipattern, as summarized in Table 2.

Table 2. Antipatterns builded from alignment problems.

Antipattern Item	Short Description
Name	OCA02 - Disjointness-subsumption contradiction with disjoint classes with subclasses.
Antipattern general form	(o1:e1 ≡ o2:e1) ⊓ (o2:e2 ⊑ o2:e1) ⊓ (o1:e1 ⊓ o1:e3 ⊑ ⊥) ⊓ (o1:e2 ≡ o2:e2) ⊓ (o1:e2 ⊑ o1:e3)
Name	OCA03 - Disjointness-subsumption contradiction with disjoint classes without subclasses.
Antipattern general form	(o1:e1 ≡ o2:e1) ⊓ (o2:e2 ⊑ o2:e1) ⊓ (o1:e1 ⊓ o1:e2 ⊑ ⊥) ⊓ (o1:e2 ≡ o2:e2)
Name	OCA04 - Domain and range incompleteness with no correspondence in domains or ranges
Antipattern general form	(o1:p1≡o2:p1) ⊓ ((o1:e1 ∈ domain(o1:p1) ⊓ o2:e1 ∈ domain(o2:p1) ⊓ ∄(o1:e1≡o2:e1)) ⊔ (o1:e2 ∈ range(o1:p1) ⊓ o2:e2 ∈ range(o2:p1) ⊓ ∄(o1:e2≡o2:e2)))
Name	OCA05 - Domain and range incompleteness with no correspondence in domains
Antipattern general form	(o1:p1≡o2:p1) ⊓ (o1:e1 ∈ domain(o1:p1) ⊓ o2:e1 ∈ domain(o2:p1) ⊓ ∄(o1:e1≡o2:e1))

For the construction and computational representation of a correspondence antipattern, we adopt EDOAL (Expressive Declarative Ontology Alignment Language), an open and agnostic language [2] [11]. A fragment of the *OCA02 - Disjointness-subsumption contradiction with disjoint classes with subclasses* correspondence antipattern EDOAL representation is illustrated as follows:

```
<map>
  <cell>
    <entity1><Class rdf:about="?o1:?e1"/></entity1>
    <entity2><Class rdf:about="?o2:?e1"/></entity2>
    <relation rdf:resource="equivalence"/>
  </cell>
  <cell>
    <entity1><Class rdf:about="?o2:?e2" /></entity1>
    <entity2><Class rdf:about="?o2:?e1" /></entity2>
    <relation rdf:resource="subsumedBy"/>
  </cell>
  <cell>
    <entity1><Class rdf:about="?o1:?e1" /></entity1>
    <entity2><Class rdf:about="?o1:?e3" /></entity2>
    <relation rdf:resource="disjoint"/>
  </cell>
  <cell>
    <entity1><Class rdf:about="?o1:?e2" /></entity1>
    <entity2><Class rdf:about="?o2:?e2" /></entity2>
    <relation rdf:resource="equivalence"/>
  </cell>
  <cell>
    <entity1><Class rdf:about="?o1:?e2" /></entity1>
    <entity2><Class rdf:about="?o1:?e3" /></entity2>
    <relation rdf:resource="subsumedBy"/>
  </cell>
</map>
```

Fifth Step: *Refactored Solution.* Refactoring in this case means repairing the alignment. In other words, when an instance of a correspondence antipattern is identified, it should be removed from the alignment.

4 Related Work

In ontology research, Ontology Design Patterns (ODPs) are an emerging approach that favors the reuse of encoded experiences and good practices. ODPs are modeling solutions to solve recurrent ontology development problems [1]. According to Falbo et al. [2], compared with Software Engineering, where patterns have been used for a long period, patterns in Ontology Engineering are still in infancy. The earliest works addressing the issue of patterns in Ontology Engineering are from the beginning of the 2000s. Sales and colleagues present semantic antipatterns for ontology engineering [3]. These antipatterns capture error prone modeling decisions, which can result in the creation of models that allow for unintended model instances (representing undesired state of affairs). The antipatterns presented by [3] have been empirically elicited through an approach of ontology conceptual models validation via visual simulation.

In [12], the authors collect a list of common antipatterns that can be found in ontologies and that cause a large percentage of inconsistency problems Besides, their list some antipatterns that do not have an impact on the logical consequences of the ontology being developed, but are important to reduce the number of errors in the intended meaning of ontologies or to improve their understandability.

Correspondence patterns, proposed by [2], are essentially correspondences and sets of correspondences with generic entities. They act as role models to help find correspondences more precise than simply relate one entity to another one. Each correspondence pattern is a generic solution to a problem of alignment. Author of [2] proposed a library of correspondence patterns for design that represent solutions to different recurrent mismatches which are quite hard for matchers using usual matching techniques. Padilha [4] proposes design patterns and antipatterns for ontology alignment using high-level ontologies. The proposed design patterns were built based on the OntoUML [5], ontology modeling language which considers the ontological distinctions and axiomatic theories proposed in Foundational Ontology Unified (UFO). The patterns described are design patterns modeling, and there is no any kind of implementation thereof.

5 Final Considerations

Ontology matching is a very active research field in the scientific community, where various techniques, approaches and tools have been proposed. However, such methods are still likely to identify incorrect correspondences between the entities of the ontologies that are being aligned. By identifying which errors may occur in the ontology matching process, it is possible that such errors be transcribed in the form of correspondence antipatterns. Correspondence antipatterns assist in identifying incorrect correspondences or set of correspondences between ontologies that are being matched.

For the correct construction of a correspondence antipattern, you must have the correct understanding of the problem being addressed [11]. The methodology proposed by Guedes *et al* [11] for build antipatterns correspondence is to help identify key issues that lead to a correct understanding of the problematic alignment, leading us to the construction of correct and accurate correspondence antipatterns. The wealth of the correspondence antipatterns consists in the fact that they are domain-independent and deal with generic entities, in other words not instantiated entities, besides having a computational representation based on an open and agnostic language.

The OAEI to provide for the community in general the alignments generated based on your edits, provides us with a rich environment for analyzing the mistakes and successes of alignments. From this set of alignment, we will collect and analyze the alignment problems generated by the ontology alignment tools evaluated by OAEI. As a result of this analysis, this paper demonstrates how the methodology for the construction of anti-patterns matching proposed in [11] can be applied over the set of alignment provided by OAEI, to identify recurring alignment problems and transform

these alignments problems in templates of correspondence antipatterns, so they can be reused and auxiliary the ontology alignment tools to identify inconsistent correspondences between two ontologies.

References

1. Presutti, V., Daga, E., Gangemi, A., Blomqvist, E.: eXtreme Design with Content Ontology Design Patterns. In: Proc. Workshop on Ontology Patterns, Washington D.C., USA (2009)
2. Falbo, R., Barcellos, M., Nardi, J., Guizzardi, G.: Organizing Ontology Design Patterns as Ontology Pattern Languages. In: 10th Extended Semantic Web Conference (2013)
3. Sales, T., Barcelos, F., Guizzardi, G.: Identification of Semantic Antipatterns in Ontology-Driven Conceptual Modeling via Visual Simulation. 4th International Workshop on Ontology-Driven Information Systems (2012)
4. Padilha, N.: Padrões e antipadrões de correspondências para melhoria do alinhamento de ontologias bem fundamentadas. Dissertação de Mestrado. Universidade Federal do Estado do Rio de Janeiro (2013)
5. Guizzardi, G.: Ontological Foundations for Structural Conceptual Models, Ph.D. Thesis, University of Twente, The Netherlands (2005)
6. Buschmann, F., Meunier, R., Rohnert, H., Sornmerlad, P., Stal. M.: Pattern-Oriented Software Architecture. A System of Patterns, John'Wiley & Sons Ltd., England (1996)
7. Euzenat, J., Shvaiko, P.: Ontology Matching, Springer-Verlag Berlin Heidelberg (2007)
8. Ehrig, M.: Ontology Alignment: Bridging the Semantic Gap, Springer Science + Business Media, LLC (2007)
9. Grau, B. C., Dragisic, Z., Eckert, K., Euzenat, J., Ferrara, A., Granada, R., Ivanova, V., Jiménez-Ruiz, E., Kempf, A. O., Lambrix, P.: Results of the Ontology Alignment Evaluation Initiative 2013, in *Proc. 8th ISWC workshop on ontology matching (OM)*, pp. 61–100 (2013)
10. Jean-Mary, Y., Shironoshita, E., Kabuka, M.: Ontology matching with semantic verification. Journal Web Semantics: Science, Services and Agents on the World Wide Web archive. Volume 7 Issue 3, pp. 235-251 (2009)
11. Guedes, A., Baião, F., Revoredo, K.: On the Identification and Representation of Ontology Correspondence Antipatterns. 8th International Workshop on Modular Ontologies in 8th International Conference on Formal Ontology in Information System. *To be appear*. (2014)
12. Roussey, C., Corcho, O., Vilches-Blázquez, L.: A catalogue of OWL ontology antipatterns. In Proceedings of the fifth international conference on Knowledge capture (K-CAP '09), pp. 205-206 (2009)