Local-As-View Integration of Ontologies in Defeasible Logic Programming

Sergio A. Gómez[†], Carlos I. Chesñevar[†‡], and Guillermo R. Simari[†]

[†]Artificial Intelligence Research and Development Laboratory
Department of Computer Science and Engineering
Universidad Nacional del Sur
Av. Alem 1253, (8000) Bahía Blanca, Argentina
Email: {sag,cic,grs}@cs.uns.edu.ar
[‡]Consejo Nacional de Investigaciones Científicas y Técnicas (CONICET), Argentina

Abstract. Ontology integration is the problem of combining data sources in the Semantic Web. Concepts in source ontologies are captured in terms of concepts defined in a global ontology. When some of these ontologies are inconsistent, the knowledge engineer often has no authority to correct them. In this article we show how to perform local-as-view integration of possibly inconsistent ontologies in terms of Defeasible Logic Programming.

1 Introduction

The *Semantic Web* [1] (SW) is a vision of the Web where resources have exact meaning assigned in terms of ontologies [2], thus enabling agents to reason about them. Ontologies in the SW are defined in the OWL language, whose underlying semantics is based on the *Description Logics* (DL) [3], for which specialized reasoners exist [4]. *Description Logic Programming* (DLP) is an alternative approach to reason with DL ontologies that proposes translating them into the language of logic programming (LP) [5]. Although DLP offers several advantages in terms of efficiency and reuse of existing LP tools (such as Prolog environments), that approach is incapable of reasoning in the presence of inconsistent ontologies. Thus we have developed a framework called δ-*ontologies* [6] for reasoning with inconsistent DL ontologies based on *Defeasible Logic Programming* (DeLP) [7].

As the World Wide Web is constituted by a variety of information sources, in order to extract information from such sources, their semantic integration and reconciliation is required [8]. Indeed, reuse of existing ontologies is often not possible without considerable effort. When one wants to reuse different ontologies together, those ontologies have to be combined in some way. This can be done by integrating the ontologies, which means that they are merged into a single new ontology, or the ontologies can be kept separate. In both cases, the ontologies have to be aligned, which means that they have to be brought into mutual agreement [9]. A particular source of inconsistency is related to the use of imported ontologies when the knowledge engineer has no authority to correct them, and as these imported ontologies are usually developed independently, their combination could also result in inconsistencies. One kind of such integration is known as *local-as-view* (LAV) integration [8], where concepts of the local ontologies are mapped to queries over a global ontology.

In this article, we present an approach for modeling LAV ontology integration when the involved ontologies can be potentially inconsistent. The ontologies are expressed in the language of DL but we give semantics to them in terms of DeLP. The alignments between the local and global ontologies are expressed as DL inclusion axioms that are also interpreted as DeLP sentences. As the ontologies are potentially inconsistent, a dialectical analysis is performed on the interpretation of both the ontologies and the mappings from the local to the global ontology.

The rest of this paper is structured as follows. In Section 2 we present the fundamentals of Description Logics and Defeasible Logic Programming along with a brief introduction to the δ-ontologies framework for reasoning with possibly inconsistent ontologies. In Section 3, we extend δ-ontologies for performing local-as-view integration. Finally Section 4 concludes the paper.

2 Knowledge Representation and Reasoning with δ-Ontologies

2.1 Fundamentals of Description Logics and Defeasible Logic Programming

Description Logics (DL) [3] are a family of knowledge representation formalisms based on the notions of *concepts* (unary predicates, classes) and *roles* (binary relations) that allow to build complex concepts and roles from atomic ones. Let C, D stand for concepts, R for a role and a, b for individuals. Concept descriptions are built from concept names using the constructors conjunction ($C \sqcap D$), disjunction ($C \sqcup D$), complement ($\neg C$), existencial restriction ($\exists R.C$), and value restriction ($\forall R.C$). To define the semantics of concept descriptions, concepts are interpreted as subsets of a domain of interest, and roles as binary relations over this domain. Further extensions are possible including inverse (P^-) and transitive (P^+) roles. A DL ontology consists of two finite and mutually disjoint sets: a *Tbox* which introduces the *terminology* and an *Abox* which contains facts about particular objects in the application domain. Tbox statements have the form $C \sqsubseteq D$ (*inclusions*) and $C \equiv D$ (*equalities*), where C and D are (possibly complex) concept descriptions. Objects in the Abox are referred to by a finite number of *individual names* and these names may be used in two types of assertional statements: *concept assertions* of the type $a : C$ and *role assertions* of the type $\langle a, b\rangle : R$.

Defeasible Logic Programming (DeLP) [7] provides a language for knowledge representation and reasoning that uses *defeasible argumentation* [10] to decide between contradictory conclusions through a *dialectical analysis*. In a DeLP program $\mathcal{P} = (\Pi, \Delta)$, a set Δ of defeasible rules $P \prec Q_1, \ldots, Q_n$, and a set Π of strict rules $P \leftarrow Q_1, \ldots, Q_n$ can be distinguished. An *argument* $\langle \mathcal{A}, H \rangle$ is a minimal non-contradictory set of ground defeasible clauses $\mathcal{A}$ of Δ that allows to derive a ground literal H possibly using ground rules of Π. Since arguments may be in conflict (concept captured in terms of a logical contradiction), an attack relationship between arguments can be defined. Generalized specificity [11] is the criterion used to decide between two conflicting arguments. If the attacking argument is strictly preferred over the attacked one, then it is called a *proper defeater*. If no comparison is possible, or both arguments are equi-preferred, the attacking argument is called a *blocking defeater*. In order to determine whether a given argument $\mathcal{A}$ is ultimately undefeated (or *warranted*), a dialectical process is recursively carried out, where defeaters for $\mathcal{A}$, defeaters for these defeaters, and so on, are taken into account. The answer to a query H w.r.t. a DeLP program $\mathcal{P}$ takes such dialectical analysis into account and can be one of *yes*, *no*, *undecided*, or *unknown*.

2.2 Reasoning with Inconsistent DL Ontologies in DeLP

In the presence of inconsistent ontologies, traditional DL reasoners (such as RACER [4]) issue an error message and stop further processing. Thus, the burden of repairing the ontology (*i.e.*, making it consistent) is on the knowledge engineer. In a previous work [12], we showed how DeLP can be used for coping with inconsistencies in ontologies such that the task of dealing with them is automatically solved by the reasoning system. We recall some of the concepts for making this article self-contained.

Definition 1 (δ-Ontology). *Let C be an $\mathcal{L}_b$-class, D an $\mathcal{L}_h$-class, A, B $\mathcal{L}_{hb}$-classes, P, Q properties, a, b individuals. Let T be a set of inclusion and equality sentences in $\mathcal{L}_{DL}$ of the form $C \sqsubseteq D$, $A \equiv B$, $\top \sqsubseteq \forall P.D$, $\top \sqsubseteq \forall P^{-}.D$, $P \sqsubseteq Q$, $P \equiv Q$, $P \equiv Q^{-}$, or $P^{+} \sqsubseteq P$ such that T can be partitioned into two disjoint sets T_S and T_D. Let A be a set of assertions disjoint with T of the form $a : D$ or $\langle a, b \rangle : P$. A* δ-ontology Σ *is a tuple (T_S, T_D, A). The set T_S is called the* strict terminology *(or Sbox), T_D the* defeasible terminology *(or Dbox) and A the* assertional box *(or Abox).*

Example 1. Consider the δ-ontology $\Sigma_1 = (T_S^1, T_D^1, A^1)$ presented in Fig. 1. The strict terminology T_S^1 says that somebody who is checking mail uses a web browser. The defeasible terminology T_D^1 expresses that those who study usually pass exams, someone who is sitting at a computer is normally studying unless he is web surfing, those who do not study usually do not pass, if someone is using a web browser then he is presumably web surfing unless he is reading Javadoc documentation. The set A^1 asserts that it is known that John, Paul and Mary are sitting at a computer; Paul is using a browser; finally, Mary is also checking mail and reading Javadoc documentation. Notice that the traditional (in the sense of [3]) DL ontology $(T_S^1 \cup T_D^1, A^1)$ is incoherent since somebody who both sits at a computer and web surfs then belongs both to the "Studies" concept and to its complement, rendering the concept empty.

For assigning semantics to a δ-ontology we defined two translation functions $\mathcal{T}_\Delta$ and $\mathcal{T}_\Pi$ from DL to DeLP based on the work of [5]. The basic premise for

Strict terminology T_S^1:
Checks_web_mail $\sqsubseteq$ Uses_browser
Defeasible terminology T_D^1:
Studies $\sqsubseteq$ Pass
Sits_at_computer $\sqsubseteq$ Studies
Sits_at_computer $\sqcap$ Web_surfing $\sqsubseteq$ $\neg$Studies
$\neg$Studies $\sqsubseteq$ $\neg$Pass
Uses_browser $\sqsubseteq$ Web_surfing
Uses_browser $\sqcap$ Reads_javadoc $\sqsubseteq$ $\neg$Web_surfing

Assertional box A^1:
JOHN : Sits_at_computer
PAUL : Sits_at_computer
PAUL : Uses_browser
MARY : Sits_at_computer
MARY : Checks_web_mail
MARY : Reads_javadoc

Fig. 1. Ontology $\Sigma_1 = (T_S^1, T_D^1, A^1)$

achieving the translation of DL ontologies into DeLP is based on the observation that a DL inclusion axiom "$\mathsf{C} \sqsubseteq \mathsf{D}$" is regarded as a First-Order Logic statement "$(\forall x)(C(x) \rightarrow D(x))$," which in turn is regarded as a Horn-clause "$d(X) \leftarrow c(X)$."[1] Naturally "$\mathsf{C} \sqcap \mathsf{D} \sqsubseteq \mathsf{E}$" is treated as "$e(X) \leftarrow c(X), d(X)$." Lloyd-Topor transformations are used to handle special cases as conjunctions in the head of rules and disjunctions in the body of rules; so "$\mathsf{C} \sqsubseteq \mathsf{D} \sqcap \mathsf{E}$" is interpreted as two rules "$d(X) \leftarrow c(X)$" and "$e(X) \leftarrow c(X)$" while "$\mathsf{C} \sqcup \mathsf{D} \sqsubseteq \mathsf{E}$" is transformed into "$e(X) \leftarrow c(X)$" and "$e(X) \leftarrow d(X)$." Likewise axioms of the form "$\exists \mathsf{r}.\mathsf{C} \sqsubseteq \mathsf{D}$" are treated as "$d(X) \leftarrow r(X, Y), c(Y)$." Dbox axioms are treated as *defeasible* and are transformed using the $\mathcal{T}_\Delta$ function (*e.g.*, $\mathcal{T}_\Delta(\mathsf{C} \sqsubseteq \mathsf{D})$ is interpreted as $d(X) \prec c(X)$); Sbox axioms are considered *strict* and are transformed using $\mathcal{T}_\Pi$ (*e.g.*, $\mathcal{T}_\Pi(\mathsf{C} \sqsubseteq \mathsf{D})$ is interpreted as $\{(d(X) \leftarrow c(X)), ({\sim} c(X) \leftarrow {\sim} d(X))\}$).[2] Abox assertions are always considered strict (*e.g.*, $\mathcal{T}_\Pi(\mathsf{a} : \mathsf{C})$ is regarded as a fact $c(a)$ and $\mathcal{T}_\Pi(\langle \mathsf{a}, \mathsf{b} \rangle : \mathsf{r})$ as $r(a, b)$). Formally:

Definition 2 (Interpretation of a δ-ontology). *Let $\Sigma = (T_S, T_D, A)$ be a δ-ontology. The* interpretation *of Σ is a DeLP program $\mathcal{P} = (\mathcal{T}_\Pi(T_S) \cup \mathcal{T}_\Pi(A), \mathcal{T}_\Delta(T_D))$.*

Notice that in order to keep consistency within an argument, we must enforce some internal coherence between the Abox and the Tbox; namely given a δ-ontology $\Sigma = (T_S, T_D, A)$, it must not be possible to derive two complementary literals from $\mathcal{T}_\Pi(T_S) \cup \mathcal{T}_\Pi(A)$. We recall how we interpret the reasoning task of *instance checking* [3, p. 19] in δ-ontologies:

Definition 3 (Potential, justified and strict membership of an individual to a class). *Let $\Sigma = (T_S, T_D, A)$ be a δ-ontology, C a class name, a an individual, and $\mathcal{P} = (\mathcal{T}_\Pi(T_S) \cup \mathcal{T}_\Pi(A), \mathcal{T}_\Delta(T_D))$.*

1. *The* individual a potentially belongs to class C *iff there exists an argument $\langle \mathcal{A}, C(a) \rangle$ w.r.t. $\mathcal{P}$;*
2. *the* individual a justifiedly belongs to class C *iff there exists a warranted argument $\langle \mathcal{A}, C(a) \rangle$ w.r.t. $\mathcal{P}$, and,*

[1] Following standard logic programming notation, in DeLP rules we note constant and predicate names with an initial lowercase and variable names with an initial uppercase.
[2] The function $\mathcal{T}_\Pi$ computes transposes of rules to allow for the application of *modus tollens*.

3. *the* individual a strictly belongs to class C *iff there exists an argument* $\langle \emptyset, C(a) \rangle$ *w.r.t.* $\mathcal{P}$.

Example 2 (Continues Ex. 1). Consider again the δ-ontology Σ_1, which is interpreted as the DeLP program $\mathcal{P}_1$ according to Def. 2 as shown in Fig. 2. From $\mathcal{P}_1$, we can determine that John justifiedly belongs to the concept Pass in Σ_1 as there exists a warranted argument structure $\langle \mathcal{A}_1, pass(john) \rangle$ that says that John will pass the exam as he studies (because he sits at a computer), where $\mathcal{A}_1 = \{(pass(john) \prec studies(john)), (studies(john) \prec sits\_at\_computer(john))\}$. We cannot reach a decision w.r.t. the membership of Paul to the concept "Pass" because there are two arguments attacking each other, so the answer to the query $pass(paul)$ is *undecided*. Formally, there exist two arguments $\langle \mathcal{B}_1, pass(paul) \rangle$ and $\langle \mathcal{B}_2, {\sim} pass(paul) \rangle$, where:

$$\mathcal{B}_1 = \left\{ \begin{array}{l} (pass(paul) \prec studies(paul)), \\ (studies(paul) \prec sits\_at\_computer(paul)) \end{array} \right\}, \text{ and}$$

$$\mathcal{B}_2 = \left\{ \begin{array}{l} ({\sim} pass(paul) \prec {\sim} studies(paul)), \\ ({\sim} studies(paul) \prec sits\_at\_computer(paul), web\_surfing(paul)), \\ (web\_surfing(paul) \prec uses\_browser(paul)) \end{array} \right\}.$$

In the case of Mary's membership to Pass, there is an argument $\langle \mathcal{C}_1, pass(mary) \rangle$. that has two defeaters, $\langle \mathcal{C}_2, {\sim} pass(mary) \rangle$ and $\langle \mathcal{C}_3, {\sim} studies(mary) \rangle$, which are both defeated by $\langle \mathcal{C}_4, {\sim} web\_surfing(mary) \rangle$, where:

$$\mathcal{C}_1 = \left\{ \begin{array}{l} (pass(mary) \prec studies(mary)), \\ (studies(mary) \prec sits\_at\_computer(mary)) \end{array} \right\},$$

$$\mathcal{C}_2 = \left\{ {\sim} pass(mary) \prec {\sim} studies(mary) \right\} \cup \mathcal{C}_3,$$

$$\mathcal{C}_3 = \left\{ \begin{array}{l} ({\sim} studies(mary) \prec sits\_at\_computer(mary), web\_surfing(mary)), \\ (web\_surfing(mary) \prec uses\_browser(mary)) \end{array} \right\}, \text{ and}$$

$$\mathcal{C}_4 = \left\{ \begin{array}{l} ({\sim} web\_surfing(X) \prec uses\_browser(mary), reads\_javadoc(mary)), \\ (uses\_browser(mary) \prec checks\_web\_mail(mary)) \end{array} \right\}.$$

Therefore, Mary belongs justifiedly to the concept "Pass" as the literal $pass(mary)$ is warranted. The dialectical trees for the three queries are depicted graphically in Fig. 3.[3]

3 Local-as-View Integration of δ-Ontologies

Data integration is the problem of combining data residing at different sources, and providing the user with a unified view of those data [13]. There are two main approaches to data integration called *global-as-view* (GAV) and *local-as-view* (LAV). In the LAV approach, we assume we have a global ontology $\mathcal{G}$, a set $\mathcal{S}$ of local/source ontologies, and the mapping between the global and the local ontologies is given by associating to each term in the local ontologies a *view* $V_{\mathcal{G}}$ over the global ontology [8, Sect. 4]. The intended meaning of associating with a term C in $\mathcal{S}$ a view $V_{\mathcal{G}}$ over $\mathcal{G}$ is that such a view represents the best way

[3] In a dialectical tree nodes are labeled as either defeated (D) or undefeated (U). Leaves are always labeled as undefeated; a node is labeled as undefeated iff all of its children are labeled as defeated, otherwise a node is labeled as defeated.

DeLP program $\mathcal{P}_1 = (\Pi_1, \Delta_1)$ obtained from Σ_1:

Facts and strict rules Π_1:

$sits\_at\_computer(john).$ $\quad$ $sits\_at\_computer(paul).$
$uses\_browser(paul).$ $\quad$ $sits\_at\_computer(mary).$
$checks\_web\_mail(mary).$ $\quad$ $reads\_javadoc(mary).$
$uses\_browser(X) \leftarrow checks\_web\_mail(X).$ $\quad$ $\sim checks\_web\_mail(X) \leftarrow \sim uses\_browser(X).$

Defeasible rules Δ_1:

$pass(X) \prec studies(X).$
$studies(X) \prec sits\_at\_computer(X).$
$\sim studies(X) \prec sits\_at\_computer(X), web\_surfing(X).$
$\sim pass(X) \prec \sim studies(X).$
$web\_surfing(X) \prec uses\_browser(X).$
$\sim web\_surfing(X) \prec uses\_browser(X), reads\_javadoc(X).$

Fig. 2. DeLP program $\mathcal{P}_1$ interpreting ontology Σ_1

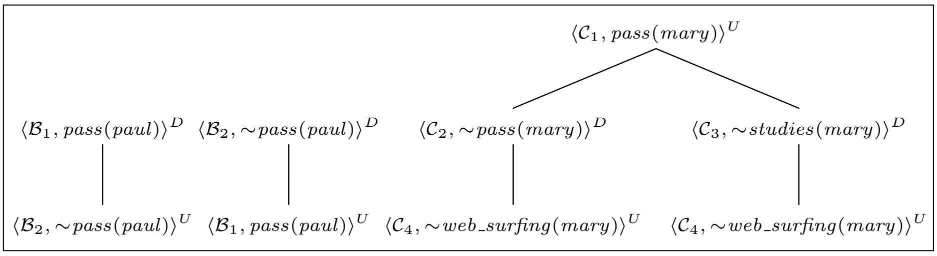

Fig. 3. Dialectical analyses for queries $pass(paul)$ and $pass(mary)$

to characterize the instances of C using the concepts in $\mathcal{G}$. The correspondence between C and the associated view can be sound (all the individuals satisfying C satisfy $V_{\mathcal{G}}$), complete (if no individual other than those satisfying C satisfies $V_{\mathcal{G}}$, and/or exact (the set of individuals that satisfy C is exactly the set of individuals that satisfy $V_{\mathcal{G}}$).

In the GAV and LAV approaches to data integration, the queries w.r.t. the target ontology are reformulated w.r.t. the sources. Hasse & Motik [14] (referring to [13]) explain that in the GAV systems the problem is simply reduced to *unfolding* the views, since the reformulation is explicit in the mappings. In the LAV case, the problem requires more complex reasoning steps as in the case of sound mappings is not clear how to reformulate the concepts of a source ontology in terms of a global ontology. Therefore, in this work, we will restrict the case of LAV integration to complete views.

Definition 4 (Ontology integration system). *An* ontology integration system *$\mathcal{I}$ is a triple $(\mathcal{G}, \mathcal{S}, \mathcal{M})$ where:*

- *$\mathcal{G}$ is a* global ontology *expressed as a δ-ontology over an alphabet $\mathcal{A}_{\mathcal{G}}$.*
- *$\mathcal{S}$ is a* set of n source ontologies *$\mathcal{S}_1, \ldots, \mathcal{S}_n$ expressed as δ-ontologies over alphabets $\mathcal{A}_{\mathcal{S}_1}, \ldots, \mathcal{A}_{\mathcal{S}_n}$, resp. Each alphabet $\mathcal{A}_{\mathcal{S}_i}$ includes a symbol for each concept or role name of the source $\mathcal{S}_i$, $i = 1, \ldots, n$.*
- *$\mathcal{M}$ is a* set of n mappings *$\mathcal{M}_1, \ldots, \mathcal{M}_n$ between $\mathcal{G}$ and $\mathcal{S}_1, \ldots, \mathcal{S}_n$, resp. Each mapping $\mathcal{M}_i$ is constituted by a set of* assertions *of the form $q_{\mathcal{S}_i} \sqsubseteq q_{\mathcal{G}}$, where $q_{\mathcal{G}}$*

and $q_{\mathcal{S}_i}$ are queries of the same arity defined over the global ontology $\mathcal{G}$ and $\mathcal{S}_i$, $i = 1, \ldots, n$, resp. Queries $q_{\mathcal{G}}$ are expressed over alphabet $\mathcal{A}_{\mathcal{G}}$ and queries $q_{\mathcal{S}_i}$ are expressed over alphabet $\mathcal{A}_{\mathcal{S}_i}$. The sets $\mathcal{M}_1, \ldots, \mathcal{M}_n$ are called bridge ontologies.

An ontology integration system will be interpreted as a DeLP program.

Definition 5 (Interpretation of an ontology integration system). *Let $\mathcal{I} = (\mathcal{G}, \mathcal{S}, \mathcal{M})$ be an ontology integration system such that $\mathcal{S} = \{\mathcal{S}_1, \ldots, \mathcal{S}_n\}$ and $\mathcal{M} = \{\mathcal{M}_1, \ldots, \mathcal{M}_n\}$, where $\mathcal{G} = (T_S^{\mathcal{G}}, T_D^{\mathcal{G}}, A^{\mathcal{G}})$; $\mathcal{S}_i = (T_S^{\mathcal{S}_i}, T_D^{\mathcal{S}_i}, A_i^{\mathcal{S}_i})$, and, $\mathcal{M}_i = (T_S^{\mathcal{M}_i}, T_D^{\mathcal{M}_i})$, with $i = 1, \ldots, n$. The system $\mathcal{I}$ is interpreted as the DeLP program $\mathcal{I}_{DeLP} = (\Pi, \Delta)$, with:*

$$\Pi = \left(\mathcal{T}_\Pi(T_S^{\mathcal{G}})\right) \cup \left(\mathcal{T}_\Pi(A^{\mathcal{G}})\right) \cup \left(\bigcup_{i=1}^n \mathcal{T}_\Pi(T_S^{\mathcal{S}_i})\right) \cup \left(\bigcup_{i=1}^n \mathcal{T}_\Pi(T_S^{\mathcal{M}_i})\right), \text{ and}$$
$$\Delta = \left(\mathcal{T}_\Delta(T_D^{\mathcal{G}})\right) \cup \left(\bigcup_{i=1}^n \mathcal{T}_\Delta(T_D^{\mathcal{S}_i})\right) \cup \left(\bigcup_{i=1}^n \mathcal{T}_\Delta(T_D^{\mathcal{M}_i})\right).$$

Possible inferences in the integrated ontology $\mathcal{I}_{DeLP}$ are modeled by means of a dialectical analysis in the DeLP program that is obtained when each DL sentence of the ontology is mapped into DeLP clauses. Thus conclusions supported by warranted arguments will be the valid consequences that will be obtained from the original ontology, provided the strict information in $\mathcal{I}_{DeLP}$ is consistent. Formally:

Definition 6 (Potential, justified and strict membership of individuals to concepts in ontology integration systems). *Let $\mathcal{I} = (\mathcal{G}, \mathcal{S}, \mathcal{M})$ be an ontology integration system. Let a be an individual name, and C a concept name defined in $\mathcal{G}$.*

1. *Individual a is a* potential member *of C iff there exists an argument $\mathcal{A}$ for the literal $C(a)$ w.r.t. DeLP program $\mathcal{I}_{DeLP}$.*
2. *Individual a is a* justified member *of C iff there exists a warranted argument $\mathcal{A}$ for the literal $C(a)$ w.r.t. DeLP program $\mathcal{I}_{DeLP}$.*
3. *Individual a is an* strict member *of C iff there exists an empty argument for the literal $C(a)$ w.r.t. DeLP program $\mathcal{I}_{DeLP}$.*

We will illustrate the above notions with an example. Notice that we label a concept C with the name of the ontology $\mathcal{S}_i$ to which it belongs (as in $\mathcal{S}_i : C$) following the XML name-space convention.

Example 3. Let us consider the problem of assigning reviewers for papers. In Fig. 4, we present a global ontology $\mathcal{G}_3$ interpreted as: a professor with a postgraduate degree can be a reviewer; someone should not be a reviewer unless they are either a professor or have a graduate degree; however, a professor, despite not having a postgraduate degree, is accepted as a reviewer if he is an outstanding researcher. We also present local ontologies $\mathcal{L}_1$ and $\mathcal{L}_2$. $\mathcal{L}_1$ expresses that John is a professor who has a PhD, Paul is also a professor but has neither a PhD nor a MSc, Mary just has a MSc, and Steve is not a professor but has a MSc. The mapping $\mathcal{M}_{\mathcal{L}_1,\mathcal{G}}$ expresses that the terms MSc and PhD from local ontology $\mathcal{L}_1$ are contained in the term postgraduated in the global ontology, and that someone have neither a MSc nor a PhD is not a postgraduate. Ontology $\mathcal{L}_2$ expresses that a is an article, b a book, c a chapter and that Paul has published a, b and c. The mapping $\mathcal{M}_{\mathcal{L}_2,\mathcal{G}}$ expresses that the view corresponding to the individuals who have published an article, a chapter and a book corresponds to the set of outstanding researchers.

The interpretation of above ontologies in DeLP yields the code presented in Fig. 5. We show next the dialectical analyses that have to be performed to compute the justified membership of John, Paul, Mary and Steve to the concept "Reviewer" w.r.t. the ontology integration system $(\mathcal{G}, \{\mathcal{L}_1, \mathcal{L}_2\}, \{\mathcal{M}_{\mathcal{L}_1,\mathcal{G}}, \mathcal{M}_{\mathcal{L}_2,\mathcal{G}}\})$.

The individual John is a justified member of the concept "Reviewer" because the argument $\langle \mathcal{A}, reviewer(john) \rangle$ has no defeaters and is thus warranted (see Fig. 6.(a)), with:

$$\mathcal{A} = \left\{ \begin{array}{l} (reviewer(john) \prec postgrad(john), prof(john)), \\ (postgrad(john) \prec phd(john)) \end{array} \right\}.$$

In Paul's case, we conclude that he is a possible reviewer as he is also a justified member of the concept "Reviewer." Notice that Paul is a potential member of the concept "$\neg$Reviewer" as there is an argument $\langle \mathcal{B}_1, \sim reviewer(paul) \rangle$, with:

$$\mathcal{B}_1 = \left\{ \begin{array}{l} (\sim reviewer(paul) \prec \sim postgrad(paul)), \\ (\sim postgrad(paul) \prec \sim msc(paul), \sim phd(paul)) \end{array} \right\}.$$

However, we see that there is another argument $\langle \mathcal{B}_2, reviewer(paul) \rangle$ that defeats $\mathcal{B}_1$, where:

$$\mathcal{B}_2 = \left\{ \begin{array}{l} (reviewer(paul) \prec prof(paul), \sim postgrad(paul), outstanding(paul)), \\ (outstanding(paul) \prec published(paul, a), article(a), \\ \qquad published(paul, b), book(b), published(paul, c), chapter(c)), \\ (\sim postgrad(paul) \prec \sim msc(paul), \sim phd(paul)) \end{array} \right\}.$$

As $\mathcal{B}_2$ is undefeated, we conclude that the literal $reviewer(paul)$ is warranted (see Fig. 6.(b)-(c)).

Steve is not a reviewer as he is a justified member of the concept "$\neg$Reviewer" (see Fig. 6.(d)). In this case, there exists a unique (undefeated) argument $\langle \mathcal{C}, \sim reviewer(steve) \rangle$. On the other hand, it is not possible to assess Mary's membership to the concept "Reviewer" as no arguments for $reviewer(mary)$ nor $\sim reviewer(mary)$ can be built.

4 Conclusions

We have presented an approach for performing local-as-view integration of Description Logic ontologies when these ontologies can be potentially inconsistent. We have adapted the notion of ontology integration system of [8] for making it suitable for the δ-ontology framework, presenting both formal definitions and a case study. We offer several advantages over previous efforts (such as [8, 14]) as our proposal is capable of dealing with inconsistent ontologies. This work also presents a difference with previous works of ours (such as [15, 12]) as those works focused on the problem of global-as-view integration of ontologies. Despite this advancement, the proposed approach is only useful in the case of complete mappings, and therefore the case for local-as-view integration with sound and exact mappings remains as an open problem and is part of our current research efforts.

Acknowledgments: This research is funded by Projects PIP 112-200801-02798 (CONICET, Argentina), TIN2006-15662-C02-01 (MEC, Spain), PGI 24/ZN10, PGI 24/N006 (SGCyT, UNS, Argentina) and Universidad Nacional del Sur.

Global ontology $\mathcal{G}_3 = (\emptyset, T_D^{\mathcal{G}}, \emptyset)$**:**

$$T_D^{\mathcal{G}} = \left\{ \begin{array}{ll} \mathsf{Prof} \sqcap \mathsf{Postgrad} \sqsubseteq \mathsf{Reviewer}; & \neg\mathsf{Postgrad} \sqcup \neg\mathsf{Prof} \sqsubseteq \neg\mathsf{Reviewer}; \\ \mathsf{Prof} \sqcap \neg\mathsf{Postgrad} \sqcap \mathsf{Outstanding} \sqsubseteq \mathsf{Reviewer} & \end{array} \right\}$$

Local ontology $\mathcal{L}_1 = (\emptyset, \emptyset, A^{\mathcal{L}_1})$**:**

$$A^{\mathcal{L}_1} = \left\{ \begin{array}{llll} \mathsf{JOHN} : \mathsf{Prof}; & \mathsf{JOHN} : \mathsf{Phd}; & \mathsf{PAUL} : \mathsf{Prof}; & \mathsf{PAUL} : \neg\mathsf{Msc}; \\ \mathsf{PAUL} : \neg\mathsf{Phd}; & \mathsf{MARY} : \mathsf{Msc}; & \mathsf{STEVE} : \neg\mathsf{Prof}; & \mathsf{STEVE} : \mathsf{Msc} \end{array} \right\}$$

Mapping $\mathcal{M}_{\mathcal{L}_1,\mathcal{G}}$ **between** $\mathcal{L}_1$ **and** $\mathcal{G}_3$**:**

$$\mathcal{M}_{\mathcal{L}_1,\mathcal{G}} = \left\{ \begin{array}{l} \mathcal{L}_1 : \mathsf{Msc} \sqcup \mathcal{L}_1 : \mathsf{Phd} \sqsubseteq \mathcal{G} : \mathsf{Postgrad}; \\ \mathcal{L}_1 : \neg\mathsf{Msc} \sqcap \mathcal{L}_1 : \neg\mathsf{Phd} \sqsubseteq \mathcal{G} : \neg\mathsf{Postgrad} \end{array} \right\}$$

Local ontology $\mathcal{L}_2 = (\emptyset, \emptyset, A^{\mathcal{L}_2})$**:**

$$A^{\mathcal{L}_2} = \left\{ \begin{array}{lll} a : \mathsf{Article}; & b : \mathsf{Book}; & c : \mathsf{Chapter}; \\ \langle \mathsf{PAUL}, a \rangle : \mathsf{published}; & \langle \mathsf{PAUL}, b \rangle : \mathsf{published}; & \langle \mathsf{PAUL}, c \rangle : \mathsf{published} \end{array} \right\}$$

Mapping $\mathcal{M}_{\mathcal{L}_2,\mathcal{G}}$ **between** $\mathcal{L}_2$ **and** $\mathcal{G}_3$**:**

$$\mathcal{M}_{\mathcal{L}_2,\mathcal{G}} = \{ \mathcal{L}_2 : (\exists \mathsf{published}.\mathsf{Article} \sqcap \exists \mathsf{published}.\mathsf{Book} \sqcap \exists \mathsf{published}.\mathsf{Chapter}) \sqsubseteq \mathcal{G} : \mathsf{Outstanding} \}$$

Fig. 4. LAV ontology integration system

References

1. Berners-Lee, T., Hendler, J., Lassila, O.: The Semantic Web. Scient. American (2001)
2. Gruber, T.R.: A translation approach to portable ontologies. Knowledge Acquisition **5**(2) (1993) 199–220
3. Baader, F., Calvanese, D., McGuinness, D., Nardi, D., Patel-Schneider, P., eds.: The Description Logic Handbook – Theory, Implementation and Applications. Cambridge University Press (2003)
4. Haarslev, V., Möller, R.: RACER System Description. Technical report, University of Hamburg, Computer Science Department (2001)
5. Grosof, B.N., Horrocks, I., Volz, R., Decker, S.: Description Logic Programs: Combining Logic Programs with Description Logics. WWW2003, May 20-24, Budapest, Hungary (2003)
6. Gómez, S.A.: Integración de Argumentación Rebatible y Ontologías en el Contexto de la Web Semántica: Formalización y Aplicaciones. PhD thesis, Universidad Nacional del Sur (2009)
7. García, A., Simari, G.: Defeasible Logic Programming an Argumentative Approach. Theory and Practice of Logic Programmming **4**(1) (2004) 95–138
8. Calvanese, D., Giacomo, G.D., Lenzerini, M.: A framework for ontology integration. In: First Semantic Web Working Symposium. (2001) 303–316
9. Klein, M.: Combining and relating ontologies: an analysis of problems and solutions. In Gomez-Perez, A., Gruninger, M., Stuckenschmidt, H., Uschold, M., eds.: Workshop on Ontologies and Information Sharing, IJCAI'01, Seattle, USA (August 4–5, 2001)
10. Chesñevar, C.I., Maguitman, A., Loui, R.: Logical Models of Argument. ACM Computing Surveys **32**(4) (December 2000) 337–383

DeLP program $(\emptyset, \Delta_{\mathcal{G}})$ obtained from $\mathcal{G}_3$:

$$\Delta_{\mathcal{G}} = \left\{ \begin{array}{l} reviewer(X) \mathrel{-\!\!\prec} postgrad(X), prof(X). \\ \sim reviewer(X) \mathrel{-\!\!\prec} \sim postgrad(X). \\ \sim reviewer(X) \mathrel{-\!\!\prec} \sim prof(X). \\ reviewer(X) \mathrel{-\!\!\prec} prof(X), \sim postgrad(X), outstanding(X). \end{array} \right\}$$

DeLP program $\mathcal{P}_1 = (\Pi^{\mathcal{L}_1}, \emptyset)$ obtained from $\mathcal{L}_1$:

$$\Pi^{\mathcal{L}_1} = \left\{ \begin{array}{llll} prof(john). & phd(john). & prof(paul). & \sim msc(paul). \\ \sim phd(paul). & msc(mary). & \sim prof(steve). & msc(steve). \end{array} \right\}$$

Mapping $\mathcal{M}_{\mathcal{L}_1,\mathcal{G}}$ expressed in DeLP:

$$\Delta_{\mathcal{L}_1,\mathcal{G}} = \left\{ \begin{array}{l} \mathcal{G} : postgrad(X) \mathrel{-\!\!\prec} \mathcal{L}_1 : msc(X). \\ \mathcal{G} : postgrad(X) \mathrel{-\!\!\prec} \mathcal{L}_1 : phd(X). \\ \sim \mathcal{G} : postgrad(X) \mathrel{-\!\!\prec} \sim \mathcal{L}_1 : msc(X), \sim \mathcal{L}_1 : phd(X). \end{array} \right\}$$

DeLP program $\mathcal{P}_2 = (\Pi^{\mathcal{L}_2}, \emptyset)$ obtained from $\mathcal{L}_2$:

$$\Pi^{\mathcal{L}_2} = \left\{ \begin{array}{lll} article(a). & book(b). & chapter(c). \\ published(paul, a). & published(paul, b). & published(paul, c). \end{array} \right\}$$

Mapping $\mathcal{M}_{\mathcal{L}_2,\mathcal{G}}$ expressed in DeLP:

$$\Delta_{\mathcal{L}_2,\mathcal{G}} = \left\{ \begin{array}{ll} \mathcal{G} : outstanding(X) \mathrel{-\!\!\prec} & \mathcal{L}_2 : published(X, Y), \mathcal{L}_2 : article(Y), \\ & \mathcal{L}_2 : published(X, Z), \mathcal{L}_2 : book(Z), \\ & \mathcal{L}_2 : published(X, W), \mathcal{L}_2 : chapter(W). \end{array} \right\}$$

Fig. 5. Ontologies $\mathcal{G}_3$, $\mathcal{L}_1$ and $\mathcal{L}_2$ expressed in DeLP

$$\begin{array}{cccc} & \langle \mathcal{B}_1, \sim reviewer(paul) \rangle^D & & \\ & | & & \\ \langle \mathcal{A}, reviewer(john) \rangle^U & \langle \mathcal{B}_2, reviewer(paul) \rangle^U & \langle \mathcal{B}_2, reviewer(paul) \rangle^U & \langle \mathcal{C}, \sim reviewer(steve) \rangle^U \\ \text{(a)} & \text{(b)} & \text{(c)} & \text{(d)} \end{array}$$

Fig. 6. Dialectical trees for *reviewer*(*john*), *reviewer*(*paul*) and *reviewer*(*steve*)

11. Stolzenburg, F., García, A., Chesñevar, C., Simari, G.: Computing Generalized Specificity. J. of N.Classical Logics **13**(1) (2003) 87–113
12. Gómez, S.A., Chesñevar, C.I., Simari, G.R.: Reasoning with Inconsistent Ontologies Through Argumentation. Applied Artificial Intelligence **1**(24) (2010) 102–148
13. Lenzerini, M.: Data integration: A theoretical perspective. Twenty-first ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems, PODS 2002, Madison, Winsconsin, USA (2002)
14. Haase, P., Motik, B.: A mapping system for the integration of OWL-DL ontologies. In Hahn, A., Abels, S., Haak, L., eds.: IHIS 05: Proceedings of the first international workshop on Interoperability of heterogeneous information systems, ACM Press (NOV 2005) 9–16
15. Gómez, S.A., Chesñevar, C.I., Simari, G.R.: An Argumentative Approach to Reasoning with Inconsistent Ontologies. In Meyer, T., Orgun, M.A., eds.: Proc. of the Knowledge Representation in Ontologies Workshop (KROW 2008). Volume CPRIT 90., Sydney, Australia (2008) 11–20