A Declarative Semantics for P2P Systems

Luciano Caroprese, Ester Zumpano

To cite this version:

Luciano Caroprese, Ester Zumpano. A Declarative Semantics for P2P Systems. 1st International Cross-Domain Conference for Machine Learning and Knowledge Extraction (CD-MAKE), Aug 2017, Reggio, Italy. pp.315-329, 10.1007/978-3-319-66808-6_21 . hal-01677127

HAL Id: hal-01677127
https://hal.inria.fr/hal-01677127

Submitted on 8 Jan 2018

A Declarative Semantics for P2P Systems

Luciano Caroprese and Ester Zumpano

DIMES, University of Calabria, 87036 Rende, Italy
{l.caroprese, e.zumpano}@dimes.unical.it

Abstract. This paper investigates the problem of data integration among Peer-to-Peer (P2P) deductive databases and presents a declarative semantics that generalizes previous proposals in the literature. Basically, by following the classical approach, the objective of a generic peer, joining a P2P system, is to enrich its knowledge by importing as much knowledge as possible while preventing inconsistency anomalies. This basic idea is extended in the present paper by allowing each peer to select between two different settings. It can either declare its local database to be sound but not complete, or declare it to be unsound. In the first case the peer considers its own knowledge more trustable than the knowledge imported from the rest of the system i.e. it gives preference to its knowledge with respect to the knowledge that can be imported from other peers. In the second case the peer considers its own knowledge as trustable as the knowledge that can be imported from the rest of the system i.e. it does not give any preference to its knowledge with respect to the knowledge that can be imported from other peers.

1 Introduction

The interaction and cooperation of peers in a P2P system allows to perform the important task of data integration. Data integration is one of the most fundamental processes in intelligent systems, from individuals to societies. In traditional data integration systems queries are submitted through a central *mediated schema*. Data is stored locally in each source and the two main formalisms managing the mapping between the mediated schema and the local sources are the *global-as-view* (GAV) and the *local-as-view* (LAV) approach [26]. The main drawbacks of traditional integration systems are due to the lack of flexibility: i) the centralized mediated schema, that controls and manages the interaction among distributed sources, must be defined looking at the global system; ii) the insertion of a new source or the modification of an existing one may cause a violation of the mappings to the mediated schema. Divergent or conflicting concepts were selected by hierarchical authorities or by (democratically) applying the majority criteria.

In a P2P system, ideally, there is no selection, but integration of the valuable contributions of every participant. Generally, peers can both provide or consume data and the only information a peer participating in a P2P system has is about neighbors, i.e. information about the peers that are reachable and can provide data of interest. More specifically, each peer joining a P2P system exhibits a set of *mapping rules*, i.e. a set of semantic correspondences to a set of peers which are already part of the system (neighbors). Thus, in a P2P system the entry of a new source, *peer*, is extremely simple as it just requires the definition of the mapping rules. By using mapping rules as soon as it enters the system a peer can participate and access all data available in its neighborhood, and through its neighborhood it becomes accessible to all the other peers in the system.

The possibility for the users for sharing knowledge from a large number of informative sources, have enabled the development of new methods for data integration easily usable for processing distributed and autonomous data.

Due to this, there have been several proposals which consider the integration of information and the computation of queries in an open ended network of distributed peers [3, 6, 9, 8, 20] as well as the problem of schema mediation [24, 27, 25], query answering and query optimization in P2P environments [1, 29, 18]. Many of the approaches proposed in the literature investigate the data integration problem in a P2P system by considering each peer as initially consistent, therefore the introduction of inconsistency is just relied to the operation of importing data from other peers. These approaches assume, following the basic classical idea of data integration, that for each peer it is preferable to import as much knowledge as possible.

In this paper we follow the proposal in [11–14, 16] in which a different interpretation of mapping rules, that allows importing from other peers only tuples not violating integrity constraints, has been proposed. This new interpretation of mapping rules has led to the proposal of a semantics for a P2P system defined in terms of *Preferred Weak Models*. Under this semantics only facts not making the local databases inconsistent can be imported, and the preferred weak models are the consistent scenarios in which peers import maximal sets of facts not violating the integrity constraints. Therefore, the preferred weak model semantics follows the classical strategy of importing as much knowledge as possible, but limiting this to the maximal subset that do not generate inconsistencies.

The following example, introduces the idea of importing in each peer maximal sets of atoms not violating integrity constraints.

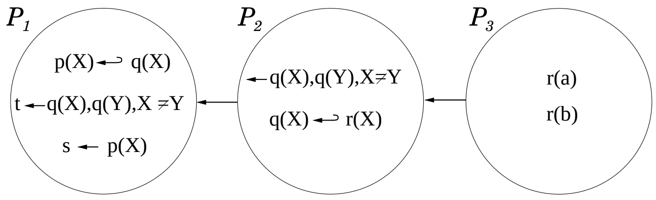

Fig. 1. A P2P system (Example 1)

Example 1. Consider the P2P system depicted in Figure 1 consisting of three peers $\mathcal{P}_1$, $\mathcal{P}_2$ and $\mathcal{P}_3$ where

- $\mathcal{P}_3$ *contains two atoms:* $r(a)$ *and* $r(b)$*,*
- $\mathcal{P}_2$ *imports data from* $\mathcal{P}_3$ *using the (mapping) rule* $q(X) \hookleftarrow r(X)$[1]*. Moreover imported atoms must satisfy the constraint* $\leftarrow q(X), q(Y), X \neq Y$ *stating that the relation* q *may contain at most one tuple, and*
- $\mathcal{P}_1$ *imports data from* $\mathcal{P}_2$*, using the (mapping) rule* $p(X) \hookleftarrow q(X)$*.* $\mathcal{P}_1$ *also contains the rules* $s \leftarrow p(X)$ *stating that* s *is* true *if the relation* p *contains at least one tuple, and* $t \leftarrow p(X), p(Y), X \neq Y$*, stating that* t *is* true *if the relation* p *contains at least two distinct tuples.*

The intuition is that, with $r(a)$ *and* $r(b)$ *being* true *in* $\mathcal{P}_3$*, either* $q(a)$ *or* $q(b)$ *could be imported in* $\mathcal{P}_2$ *(but not both, otherwise the integrity constraint is violated) and, consequently, only one tuple is imported in the relation* p *of peer* $\mathcal{P}_1$*. Note that whatever is the derivation in* $\mathcal{P}_2$*,* s *is derived in* $\mathcal{P}_1$ *while* t *is not derived. Therefore, the atoms* s *and* t *are, respectively,* true *and* false *in* $\mathcal{P}_1$*.*

□

With the preferred weak model semantics in [11–14, 16], each peer gives preference to local data with respect to data imported from the neighborhood. Therefore the basic assumption in this semantics is that *each peer is sound, but not complete.*

In this paper, we extend the framework in [11–14, 16]. In a more general setting each peer joining the system can either (i) assume its local database to be sound, but not complete; in this case the peer considers its own knowledge more trustable than the knowledge imported from the rest of the system i.e. it gives preference to its knowledge with respect to the knowledge that can be imported other peers or (ii) assume its local knowledge to be unsound; in this case the peer considers its own knowledge as trustable as the knowledge imported from the rest of the system i.e. it does not give any preference to its knowledge with respect to the knowledge that can be imported from from other peers.

Let's now introduce another example, that will be used as a running example in the rest of the paper.

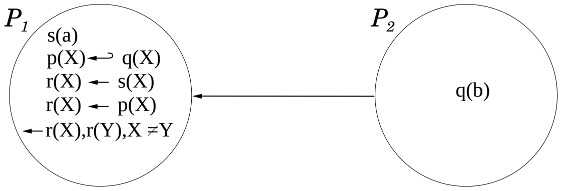

Fig. 2. A P2P system (Example 2)

Example 2. Consider the P2P system $\mathcal{PS}$ *depicted in Figure 2.* $\mathcal{P}_2$ *contains the fact* $q(b)$*, whereas* $\mathcal{P}_1$ *contains the fact* $s(a)$*, the mapping rule* $p(X) \hookleftarrow q(X)$*,*

[1] Please, note the special syntax we use for mapping rules.

the constraint $\leftarrow r(X), r(Y), X \neq Y$ *and the standard rules* $r(X) \leftarrow p(X)$ *and* $r(X) \leftarrow s(X)$.

- *if* $\mathcal{P}_1$ *considers its own knowledge more trustable than the knowledge imported from* $\mathcal{P}_2$ *(that is it gives preference to its knowledge with respect to the knowledge that can be imported from* $\mathcal{P}_2$*), then the fact* $p(b)$ *cannot be imported in* $\mathcal{P}_1$*, as it indirectly violates its integrity constraint. More specifically,* $p(b)$ *cannot be imported in* $\mathcal{P}_1$ *due to the presence of the local fact* $s(a)$.
- *if* $\mathcal{P}_1$ *considers its own knowledge as trustable as the knowledge imported from* $\mathcal{P}_2$ *(that is if it does not give any preference to its knowledge with respect to the knowledge that can be imported from* $\mathcal{P}_2$*), then two possible scenarios are possible: a first one in wich no atom is imported from* $\mathcal{P}_2$*, and a second one in wich* $q(b)$ *is imported from* $\mathcal{P}_2$ *and* $s(a)$ *is removed from* $\mathcal{P}_1$.

In this paper we generalize the definition of P2P system in order to capture previous behavior and allow the possibility for each peer to declare either that its local knowledge is preferred with respect to the knowledge that can be imported from other peers (sound peer), or that it does not give any preference to its local knowledge with respect to the knowledge that can be extracted from the rest of the system (unsound peer). An unsound P2P system is a p2P system in which at least one peer is unsound. The semantics of an unsound P2P system is captured by the weak model semantics of a correspondent standard P2P system obtained by splitting each unsound peer into two peers.

Organization. Preliminaries are reported in Section 2. Section 3 introduces the syntax used for modeling a P2P system and reviews the *Preferred Weak Model* semantics, proposed in [11, 12]. Section 4 proposes a generalization of a P2P system so that each peer can declare either its preference to its own knowledge, or give no preference to its own knowledge with respect to imported knowledge. Section 5 provides results on the computational complexity of computing preferred weak models and answers to queries Related work is discussed in Section 6. Conclusions are drawn in Section 7.

2 Background

We assume there are finite sets of predicate symbols, constants and variables. A *term* is either a constant or a variable. An *atom* is of the form $p(t_1, \ldots, t_n)$ where p is a predicate symbol and $t_1, \ldots, t_n$ are terms. A *literal* is either an atom A or its negation *not* A. A *rule* is of the form $H \leftarrow \mathcal{B}$, where H is an atom (*head* of the rule) and $\mathcal{B}$ is a conjunction of literals (*body* of the rule). A program $\mathcal{P}$ is a finite set of rules. $\mathcal{P}$ is said to be positive if it is negation free. The definition of a predicate p consists of all rules having p in the head. A ground rule with empty body is a *fact*. A rule with empty head is a *constraint*. It is assumed that programs are *safe*, i.e. variables appearing in the head or in negated body literals are range restricted as they appear in some positive body literal. The ground instantiation of a program $\mathcal{P}$, denoted by $ground(\mathcal{P})$ is built by replacing variables with constants in all possible ways. An interpretation is a set of ground atoms. The truth value of ground atoms, literals and rules with respect to an interpretation M is as follows: $val_M(A) = A \in M$, $val_M(not\ A) = not\ val_M(A)$, $val_M(L_1, \dots, L_n) = min\{val_M(L_1), \dots, val_M(L_n)\}$ and $val_M(A \leftarrow L_1, \dots, L_n) = val_M(A) \geq val_M$ $(L_1, \dots, L_n)$, where A is an atom, $L_1, \dots, L_n$ are literals and $true > false$. An interpretation M is a model for a program $\mathcal{P}$, if all rules in $ground(\mathcal{P})$ are *true* w.r.t. M. A model M is said to be minimal if there is no model N such that $N \subset M$. We denote the set of minimal models of a program $\mathcal{P}$ with $\mathcal{MM}(\mathcal{P})$. Given an interpretation M and a predicate symbol g, $M[g]$ denotes the set of g-tuples in M. The semantics of a positive program $\mathcal{P}$ is given by its unique minimal model which can be computed by applying the *immediate consequence operator* $\mathbf{T}_\mathcal{P}$ until the fixpoint is reached ($\mathbf{T}_\mathcal{P}^\infty(\emptyset)$). The semantics of a program with negation $\mathcal{P}$ is given by the set of its stable models, denoted as $\mathcal{SM}(\mathcal{P})$. An interpretation M is a *stable model* (or *answer set*) of $\mathcal{P}$ if M is the unique minimal model of the positive program $\mathcal{P}^M$, where $\mathcal{P}^M$ is obtained from $ground(\mathcal{P})$ by (i) removing all rules r such that there exists a negative literal $not\ A$ in the body of r and A is in M and (ii) removing all negative literals from the remaining rules [22]. It is well known that stable models are minimal models (i.e. $\mathcal{SM}(\mathcal{P}) \subseteq \mathcal{MM}(\mathcal{P})$) and that for negation free programs, minimal and stable model semantics coincide (i.e. $\mathcal{SM}(\mathcal{P}) = \mathcal{MM}(\mathcal{P})$).

3 P2P Systems: Syntax and Semantics

This section introduces the syntax used for modeling a P2P system and reviews the *Preferred Weak Model* semantics, proposed in [11, 12], in which a special interpretation of mapping rules is introduced.

3.1 Syntax

A *(peer) predicate symbol* is a pair $i : p$, where i is a *peer identifier* and p is a predicate symbol. A *(peer) atom* is of the form $i : A$, where i is a *peer identifier* and A is a standard atom. A *(peer) literal* is a peer atom $i : A$ or its negation $not\ i : A$. A conjunction $i : A_1, \dots, i : A_m, not\ i : A_{m+1}, \dots, not\ i : A_n, \phi$, where ϕ is a conjunction of built-in atoms[2], will be also denoted as $i : \mathcal{B}$, with $\mathcal{B}$ equals to $A_1, \dots, A_m, not\ A_{m+1}, \dots, not\ A_n, \phi$.
A *(peer) rule* can be of one of the following three types:

1. STANDARD RULE. It is of the form $i : H \leftarrow i : \mathcal{B}$, where $i : H$ is an atom and $i : \mathcal{B}$ is a conjunction of atoms and built-in atoms.
2. INTEGRITY CONSTRAINT. It is of the form $\leftarrow i : \mathcal{B}$, where $i : \mathcal{B}$ is a conjunction of literals and built-in atoms.

[2] A *built-in atom* is of the form $X\theta Y$, where X and Y are terms and θ is a comparison predicate.

3. MAPPING RULE. It is of the form $i : H \hookleftarrow j : \mathcal{B}$, where $i : H$ is an atom, $j : \mathcal{B}$ is a conjunction of atoms and built-in atoms and $i \neq j$.

In the previous rules, $i : H$ is called *head* while $i : \mathcal{B}$ (resp. $j : \mathcal{B}$) is called *body*. Negation is allowed just in the body of integrity constraints. The concepts of *ground rule* and *fact* are similar to those reported in Section 2. The definition of a predicate $i\!:\!p$ consists of the set of rules in whose head the predicate symbol $i\!:\!p$ occurs. A predicate can be of three different kinds: *base predicate*, *derived predicate* and *mapping predicate*. A base predicate is defined by a set of ground facts; a derived predicate is defined by a set of standard rules and a mapping predicate is defined by a set of mapping rules.
An atom $i : p(X)$ is a *base atom* (resp. *derived atom*, *mapping atom*) if $i : p$ is a base predicate (resp. standard predicate, mapping predicate). Given an interpretation M, $M[\mathcal{D}]$ (resp. $M[\mathcal{LP}]$, $M[\mathcal{MP}]$) denotes the subset of base atoms (resp. derived atoms, mapping atoms) in M.

Definition 1. P2P SYSTEM. A *peer* $\mathcal{P}_i$ is a tuple $\langle \mathcal{D}_i, \mathcal{LP}_i, \mathcal{MP}_i, \mathcal{IC}_i \rangle$, where (i) $\mathcal{D}_i$ is a set of facts (*local database*); (ii) $\mathcal{LP}_i$ is a set of standard rules; (iii) $\mathcal{MP}_i$ is a set of mapping rules and (iv) $\mathcal{IC}_i$ is a set of constraints over predicates defined by $\mathcal{D}_i$, $\mathcal{LP}_i$ and $\mathcal{MP}_i$. A *P2P system* $\mathcal{PS}$ is a set of peers $\{\mathcal{P}_1, \ldots, \mathcal{P}_n\}$. □

Given a P2P system $\mathcal{PS} = \{\mathcal{P}_1, \ldots, \mathcal{P}_n\}$, where $\mathcal{P}_i = \langle \mathcal{D}_i, \mathcal{LP}_i, \mathcal{MP}_i, \mathcal{IC}_i \rangle$, $\mathcal{D}, \mathcal{LP}, \mathcal{MP}$ and $\mathcal{IC}$ denote, respectively, the global sets of ground facts, standard rules, mapping rules and integrity constraints, i.e. $\mathcal{D} = \bigcup_{i \in [1..n]} \mathcal{D}_i$, $\mathcal{LP} = \bigcup_{i \in [1..n]} \mathcal{LP}_i$, $\mathcal{MP} = \bigcup_{i \in [1..n]} \mathcal{MP}_i$ and $\mathcal{IC} = \bigcup_{i \in [1..n]} \mathcal{IC}_i$. In the rest of this paper, with a little abuse of notation, $\mathcal{PS}$ will be also denoted both with the tuple $\langle \mathcal{D}, \mathcal{LP}, \mathcal{MP}, \mathcal{IC} \rangle$ and the set $\mathcal{D} \cup \mathcal{LP} \cup \mathcal{MP} \cup \mathcal{IC}$; moreover whenever the peer is understood, the peer identifier will be omitted.

3.2 Semantics

This section reviews the *Preferred Weak Model* semantics for P2P systems [11, 12] which is based on a special interpretation of mapping rules.
Observe that for each peer $\mathcal{P}_i = \langle \mathcal{D}_i, \mathcal{LP}_i, \mathcal{MP}_i, \mathcal{IC}_i \rangle$, the set $\mathcal{D}_i \cup \mathcal{LP}_i$ is a *positive normal program*, thus it admits just *one minimal model* that represents the *local knowledge* of $\mathcal{P}_i$. In this paper it is assumed that each peer is *locally consistent*, i.e. its local knowledge satisfies $\mathcal{IC}_i$ (i.e. $\mathcal{D}_i \cup \mathcal{LP}_i \models \mathcal{IC}_i$). Therefore, inconsistencies may be introduced just when the peer imports data from other peers. The intuitive meaning of a mapping rule $i : H \hookleftarrow j : \mathcal{B} \in \mathcal{MP}_i$ is that if the body conjunction $j : \mathcal{B}$ is *true* in the source peer $\mathcal{P}_j$ the atom $i : H$ can be imported in $\mathcal{P}_i$ only if it does not imply (directly or indirectly) the violation of some integrity constraint in $\mathcal{IC}_i$. The following example will clarify the meaning of mapping rules.

Example 3. Consider the P2P system in Fig. 2. If the fact $p(b)$ is imported in $\mathcal{P}_1$, the fact $r(b)$ will be derived. As $r(a)$ is already true in $\mathcal{P}_1$, because it is derived from $s(a)$, the integrity constraint is violated. Therefore, $p(b)$ cannot be imported in $\mathcal{P}_1$ as it indirectly violates an integrity constraint. □

Before formally presenting the preferred weak model semantics, some notation is introduced. Given a mapping rule $r = H \hookleftarrow \mathcal{B}$, the corresponding standard logic rule $H \leftarrow \mathcal{B}$ will be denoted as $St(r)$. Analogously, given a set of mapping rules $\mathcal{MP}$, $St(\mathcal{MP}) = \{St(r) \mid r \in \mathcal{MP}\}$ and given a P2P system $\mathcal{PS} = \mathcal{D} \cup \mathcal{LP} \cup \mathcal{MP} \cup \mathcal{IC}$, $St(\mathcal{PS}) = \mathcal{D} \cup \mathcal{LP} \cup St(\mathcal{MP}) \cup \mathcal{IC}$.

Given an interpretation M, an atom H and a conjunction of atoms $\mathcal{B}$:

- $val_M(H \leftarrow \mathcal{B}) = val_M(H) \geq val_M(\mathcal{B})$,
- $val_M(H \hookleftarrow \mathcal{B}) = val_M(H) \leq val_M(\mathcal{B})$.

Therefore, if the body is *true*, the head of a standard rule *must* be *true*, whereas the head of a mapping rule *could* be *true*.
Intuitively, a *weak model* M of a P2P system $\mathcal{PS}$ is an interpretation that satisfies all standard rules, mapping rules and constraints of $\mathcal{PS}$ and such that each atom $H \in M[\mathcal{MP}]$ (i.e. each mapping atom) is *supported* from a mapping rule $H \hookleftarrow \mathcal{B}$ whose body $\mathcal{B}$ is satisfied by M. A *preferred weak model* is a weak model that contains a maximal subset of mapping atoms. This concept is justified by the assumption that it is *preferable* to import in each peer *as much knowledge as possible.*

Definition 2. (Preferred) Weak Model. Given a P2P system $\mathcal{PS} = \mathcal{D} \cup \mathcal{LP} \cup \mathcal{MP} \cup \mathcal{IC}$, an interpretation M is a *weak model* for $\mathcal{PS}$ if $\{M\} = \mathcal{MM}(St(\mathcal{PS}^M))$, where $\mathcal{PS}^M$ is the program obtained from $ground(\mathcal{PS})$ by removing all mapping rules whose head is *false* w.r.t. M.
Given two weak models M and N, M is said to *preferable* to N, and is denoted as $M \sqsupseteq N$, if $M[\mathcal{MP}] \supseteq N[\mathcal{MP}]$. Moreover, if $M \sqsupseteq N$ and $N \not\sqsupseteq M$, then $M \sqsupset N$. A weak model M is said to be *preferred* if there is no weak model N such that $N \sqsupset M$.
The set of weak models for a P2P system $\mathcal{PS}$ will be denoted by $\mathcal{WM}(\mathcal{PS})$, whereas the set of preferred weak models will be denoted by $\mathcal{PWM}(\mathcal{PS})$. □

The next theorem shows that P2P systems always admit maximal weak models.

Theorem 1. *For every consistent P2P system $\mathcal{PS}$, $\mathcal{PWM}(\mathcal{PS}) \neq \emptyset$.*

Proof. Let us consider a set M such that $\{M\} = \mathcal{MM}(\mathcal{D} \cup \mathcal{LP} \cup \mathcal{IC})$, that is the minimal model of a P2P system obtained from $\mathcal{PS}$ by deleting all mapping rules. As $\mathcal{PS}$ is initially consistent, such a model exists. Moreover, as M does not contain any mapping atoms, all the ground mapping rules have to be deleted from $ground(\mathcal{PS})$ in order to obtain $St(\mathcal{PS}^M)$. It follows that $St(\mathcal{PS}^M) = ground(\mathcal{D} \cup \mathcal{LP} \cup \mathcal{IC})$ and $\{M\} = \mathcal{MM}(St(\mathcal{PS}^M))$. This means that M is a weak model for $\mathcal{PS}$. As there is at least a weak model, then $\mathcal{PWM}(\mathcal{PS}) \neq \emptyset$. □
Observe that in the previous definition $St(\mathcal{PS}^M)$ is a positive normal program, thus it admits just one minimal model. Moreover, note that the definition of weak model presents interesting analogies with the definition of stable model.

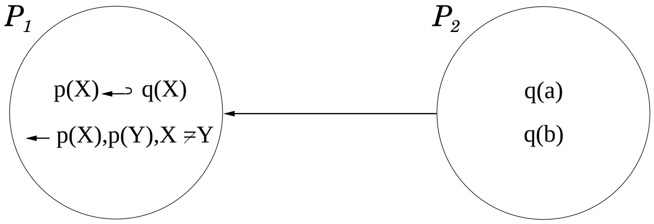

Fig. 3. The system $\mathcal{PS}$

Example 4. Consider the P2P system $\mathcal{PS}$ in Fig. 3. $\mathcal{P}_2$ contains the facts $q(a)$ and $q(b)$, whereas $\mathcal{P}_1$ contains the mapping rule $p(X) \hookleftarrow q(X)$ and the constraint $\leftarrow p(X), p(Y), X \neq Y$. The weak models of the system are $M_0 = \{q(a), q(b)\}$, $M_1 = \{q(a), q(b), p(a)\}$ and $M_2 = \{q(a), q(b), p(b)\}$, whereas the preferred weak models are M_1 and M_2 as they import the maximal set of atoms from $\mathcal{P}_2$. □

We conclude this section showing how a classical problem can be expressed using the preferred weak model semantics.

Example 5. Three-colorability. We are given two peers $\mathcal{P}_1$, containing a set of nodes, defined by a unary relation node, and a set of colors, defined by the unary predicate color, and $\mathcal{P}_2$, containing the mapping rule

$$colored(X, C) \hookleftarrow 1\!:\!node(X), 1\!:\!color(C)$$

and the integrity constraints

$$\begin{array}{l} \leftarrow colored(X, C_1),\ colored(X, C_2),\ C_1 \neq C_2 \\ \leftarrow edge(X, Y),\ colored(X, C),\ colored(Y, C) \end{array}$$

stating, respectively, that a node cannot be colored with two different colors and two connected nodes cannot be colored with the same color. The mapping rule states that the node x can be colored with the color c, only if in doing this no constraint is violated, that is if the node x is colored with a unique color and there is no adjacent node colored with the same color. Each preferred weak model computes a maximal subgraph which is three-colorable. □

An alternative characterization of the preferred weak model semantics, called *Preferred Stable Model* semantics, based on the rewriting of mapping rules into prioritized rules [28] has been proposed in [11, 12].

4 A more general framework

In previous section we introduced the *Preferred Weak Model* semantics for P2P systems [11, 12] that assumes that the knowledge of each peer of a P2P system is sound, but not complete, that is each peer (implicitly) prefers its local knowledge

with respect to the knowledge that can be imported from the rest of the system. Mapping rules are therefore used to enrich the local knowledge only if imported atoms do not cause inconsistencies.
In this section we generalize the definition of P2P system so that each peer can declare either its preference to its own knowledge (sound peer), or give no preference to its own knowledge with respect to imported knowledge (unsound peer). A sound peer trusts its knowledge more than the knowledge of the rest of the system, whereas an unsound peer trusts its knowledge as much as the knowledge that can be extracted from other peers.
We first provides the a definition of *P2P system* that generalizes Definition 1 by introducing a new type of peers - *the unsound peers* - giving the same priority to local data and to imported data.

Definition 3. *An unsound P2P system $\mathcal{UPS}$ is a pair $(\mathcal{PS},\mathcal{U})$, where $\mathcal{PS} = \{\mathcal{P}_1, \ldots, \mathcal{P}_n\}$ is a (standard) P2P system and $\mathcal{U} \subseteq \mathcal{PS}$. Peers in $\mathcal{U}$ are called unsound peers.* □

The semantics of an unsound P2P system $\mathcal{UPS}$ is captured by the weak model semantics of a correspondent standard P2P system obtained from $\mathcal{UPS}$ by splitting each unsound peer $\mathcal{P}_i$ into two peers. The idea is to move the local database $\mathcal{D}_i$ from $\mathcal{P}_i$ to the new peer and to introduce in $\mathcal{P}_i$ a set of mapping rules able to import only portions of $\mathcal{D}_i$ that do not violate its integrity constraits.

Definition 4. *Let $\mathcal{UPS} = (\mathcal{PS},\mathcal{U})$ an unsound P2P system, where $\mathcal{PS} = \{\mathcal{P}_1, \ldots, \mathcal{P}_n\}$, and $\mathcal{P}_i = \langle \mathcal{D}_i, \mathcal{LP}_i, \mathcal{MP}_i, \mathcal{IC}_i \rangle$ a peer in $\mathcal{U}$. Then, Split$(\mathcal{P}_i)$ is the set containing the following peers:*

- $\mathcal{P}_{(i+n)} := \langle \{(i+n) : p(X) \mid i : p(X) \in \mathcal{D}_i\}, \emptyset, \emptyset, \emptyset \rangle$

- $\mathcal{P}_i := \langle \emptyset, \mathcal{LP}_i, \mathcal{MP}_i \cup \{i : p(X) \hookleftarrow (i+n) : p(X) \mid i : p(X) \in \mathcal{D}_i\}, \mathcal{IC}_i \rangle$

Moreover:

$$\mathit{Split}(\mathcal{UPS}) = (\mathcal{PS} \setminus \mathcal{U}) \ \cup \bigcup_{\mathcal{P}_i \ \in \ \mathcal{U}} \mathit{Split}(\mathcal{P}_i)$$

The preferred weak models of $\mathcal{UPS}$ are obtained from those of Split$(\mathcal{UPS})$ by removing each atom $i : p(X)$ such that $i > n$. □

In previous definition, the peer $\mathcal{P}_i$ is redefined by deleting its local database $\mathcal{D}_i$ and inserting mapping rules allowing to import tuples into old base relations (which now are mapping relations) from an auxiliary peer $\mathcal{P}_{(i+n)}$. Observe that *Split*$(\mathcal{UPS})$ is a standard P2P system for which a preferred weak model semantics is adopted.

Example 6. Let us continue our discussion about the P2P system $\mathcal{PS}$ presented in Example 2.

- Assuming that $\mathcal{PS}$ is a sound P2P system, then $\mathcal{P}_1$ trusts its own data more than the data that can be imported from $\mathcal{P}_2$. Therefore it will not import the fact $p(b)$ because it would violate its integrity constraint. The preferred weak models of $\mathcal{PS}$ is: $M_1 = \{q(b), s(a), r(a))\}$

- Assuming that $\mathcal{PS}$ is an unsound P2P system and $\mathcal{P}_1$ is an unsound peer, $Split(\mathcal{PS})$ contains the following peers (for the sake of presentation we omit indexes for peers $\mathcal{P}_1$ and $\mathcal{P}_2$):

  - $\mathcal{P}_1 = \langle \emptyset, \mathcal{LP}_1, \mathcal{MP}_1 \cup \{s(X) \hookleftarrow 3 : s(X)\}, \mathcal{IC}_1 \rangle$
  - $\mathcal{P}_2 = \langle \{q(b)\}, \emptyset, \emptyset, \emptyset \rangle$
  - $\mathcal{P}_3 = \langle \{3 : s(a)\}, \emptyset, \emptyset, \emptyset \rangle$

  In this case, $\mathcal{P}_1$ does not give any preference to its knowledge with respect to the knowledge that can be imported from $\mathcal{P}_2$ and two possible scenarios are possible: a first one in which no atom is imported from $\mathcal{P}_2$ and $s(a)$ is imported from $\mathcal{P}_3$ - corresponding to the preferred weak model $M_1 = \{q(b), s(a), r(a)\}$, and a second one in which $q(b)$ is imported from $\mathcal{P}_2$ and $s(a)$ is not imported from $\mathcal{P}_3$ (this is equivalent to delete $s(a)$ from $\mathcal{P}_1$) - corresponding to the preferred weak model $M_2 = \{q(b), p(b), r(b)\}$. Observe that the auxiliary atom $3 : s(a)$ does not occur in the preferred weak models M_1 and M_2. □

5 Query answers and complexity

We consider now the computational complexity of calculating preferred weak models and answers to queries. As a P2P system may admit more than one preferred weak model, the answer to a query is given by considering *brave* or *cautious* reasoning (also known as *possible* and *certain* semantics).

Definition 5. *Given a P2P system* $\mathcal{PS} = \{\mathcal{P}_1, \ldots, \mathcal{P}_n\}$ *and a ground peer atom* A*, then* A *is* true *under*

- *brave reasoning if* $A \in \bigcup_{M \in \mathcal{PWM}(\mathcal{PS})} M$*,*
- *cautious reasoning if* $A \in \bigcap_{M \in \mathcal{PWM}(\mathcal{PS})} M$*.* □

Lemma 1. $\bigcup_{M \in \mathcal{PWM}(\mathcal{PS})} M = \bigcup_{N \in \mathcal{WM}(\mathcal{PS})} N$ □

The lemma states that for every P2P system $\mathcal{PS}$ an atom is *true* in some of its preferred weak models if and only if it is *true* in some of its weak models.

Theorem 2. *Let* $\mathcal{PS}$ *be a P2P system, then:*

1. *Deciding whether an interpretation* M *is a preferred weak model of* $\mathcal{PS}$ *is co*$\mathcal{NP}$ *complete.*
2. *Deciding whether a preferred weak model for* $\mathcal{PS}$ *exists is in* Σ_2^p*.*

3. *Deciding whether an atom A is* true *in some preferred weak model of $\mathcal{PS}$ is Σ_2^p complete.*
4. *Deciding whether an atom A is* true *in every preferred weak model of $\mathcal{PS}$ is Π_2^p complete.*

Proof

1. **(Membership)** We prove that the complementary problem, that is the problem of checking whether M is not a preferred weak model, is in $\mathcal{NP}$. We can guess an interpretation N and verify in polynomial time that (i) N is a weak model, that is $\{N\} = \mathcal{MM}(St(\mathcal{PS}^N))$, and (ii) either M is not a weak model, that is $\{N\} \neq \mathcal{MM}(St(\mathcal{PS}^N))$, or $N \sqsupset M$, that is $N[\overline{\mathcal{MP}}] \supset M[\overline{\mathcal{MP}}]$ or $N[\overline{\mathcal{MP}}] = M[\overline{\mathcal{MP}}] \wedge N[\underline{\mathcal{MP}}] \subset M[\underline{\mathcal{MP}}]$. Therefore, the original problem is in $co\mathcal{NP}$.
   (Hardness) We will reduce the *SAT problem* to the problem of *checking whether a weak model is not preferred.*
   Let X be a set of variables and F a CNF formula over X. Then the problem that will be reduced is checking whether the QBF formula $(\exists X)\ F$ is *true*.
   We define a P2P system $\mathcal{PS}$ with two peers: $\mathcal{P}_1$ and $\mathcal{P}_2$. Peer $\mathcal{P}_1$ contains the atoms:

   $$\begin{array}{l} 1 : variable(x), \text{ for each } x \in X \\ 1 : truthValue(true) \\ 1 : truthValue(false) \end{array}$$

   The relation $1 : variable$ stores the variables in X and the relation $1 : truthValue$ stores the truth values $true$ and $false$. Peer $\mathcal{P}_2$ contains the atoms:

   $2 : variable(x)$, for each $x \in X$
   $2 : positive(x, c)$, for each $x \in X$ and clause c in F s.t. x occurs non-negated in c
   $2 : negated(x, c)$, for each $x \in X$ and clause c in F s.t. x occurs negated in c

   the mapping rule:

   $$2 : assign(X, V) \hookleftarrow 1 : variable(X), 1 : truthValue(V)$$

   stating that the truth value V could be assigned to the variable X, the standard rules:

   $$\begin{array}{l} 2 : clause(C) \leftarrow 2 : positive(X, C) \\ 2 : clause(C) \leftarrow 2 : negated(X, C) \\ 2 : holds(C) \leftarrow 2 : positive(X, C), 2 : assign(X, true) \\ 2 : holds(C) \leftarrow 2 : negated(X, C), 2 : assign(X, false) \\ 2 : assignment \leftarrow 2 : assign(X, V) \end{array}$$

   defining a *clause* from the occurences of its positive and negated variables (first and second rule), whether a clause holds with a given assignment of values (third and fourth rule) and whether an assignment of values actually exists (fifth rule), and the integrity constraints:

$$\begin{array}{l} \leftarrow 2 : assign(X, true),\ 2 : assign(X, false) \\ \leftarrow 2 : clause(C),\ not\ 2 : holds(C),\ 2 : assignment \\ \leftarrow 2 : variable(X),\ not\ 2 : assign(X, true),\ not\ 2 : assign(X, false), \\ \quad 2 : assignment \end{array}$$

stating that two different truth values cannot be assigned to the same variable (first constraint), that if there is an assignment then there cannot be an unsatisfied clause (second constraint) and cannot be an unevaluated variable (third constraint).
Let $\mathcal{DB}$ the set of atoms in $\mathcal{PS}$, $\mathcal{MP}$ the set of mapping rules in $\mathcal{PS}$, $\mathcal{LP}$ the set of standard rules in $\mathcal{PS}$ and $\mathcal{IC}$ the set of integrity constraints in $\mathcal{PS}$.
Let M be the minimal model of $\mathcal{DB} \cup \mathcal{LP} \cup \mathcal{IC}$, that is the model containing no mapping atom. As $\mathcal{PS}$ is initially consistent, M is a weak model of $\mathcal{PS}$. Observe that the integrity constraints in $\mathcal{PS}$ are satisfied when no mapping atom is imported in $\mathcal{P}_2$ that is if no assignment of values is performed for the variables in X. If F is not satisfiable, then there is no way to import mapping atoms in P_2 preserving consistency because the second constraint will be violated. In this case M is a preferred weak model. If F is satisfiable there is a weak model N whose set of mapping atoms corresponds to an assignment of values to the variables in X that satisfies F. Clearly, as $\mathcal{MP}[N] \supset \mathcal{MP}[M]$, M is not a preferred weak model. Moreover, if M is not a preferred weak model there must be another weak model N whose set of mapping atoms corresponds to an assignment of values to the variables in X that satisfies F. In other words, F is satisfiable if and only if M is not a preferred weak model.

2. Let us guess an interpretation M. By (1), deciding whether M is a preferred weak model can be decided by a call to a $co\mathcal{NP}$ oracle.
3. In [11] has been shown that a $\mathcal{PS}$ can be modeled as a disjunction-free $(\vee - free)$ prioritized logic programs. For this program deciding whether an atom is *true* in some preferred stable model is Σ_2^p complete [28].
4. In [11] has been shown that a $\mathcal{PS}$ can be modeled as a disjunction-free $(\vee - free)$ prioritized logic programs. For this program deciding whether an atom is *true* in every preferred stable model is Π_2^p complete [28]. □

6 Related Work

The possibility for the users for sharing knowledge from a large number of informative sources, have enabled the development of new methods for data integration easily usable for processing distributed and autonomous data. The present paper is placed among the works on semantic peer data managment systems. Among the approaches that are related to ours, we mention [24, 9, 10, 20].
In [24] the problem of schema mediation in a Peer Data Management System (PDMS) is investigated. A formalism, PPL, for mediating peer schemas, which

uses the GAV and LAV formalism to specify mappings, is proposed. A FOL semantics to the global system is proposed and query answering for a PDMS is defined by extending the notion of certain answer. More specifically, certain answers for a peer are those that are true in every global instance that is consistent with local data.

In [9, 10] a new semantics for a P2P system, based on epistemic logic, is proposed. The paper proposes a sound, complete and terminating procedure that returns the certain answers to a query submitted to a peer. The advantage of this framework is that certain answers of fixed conjuntive queries posed on a peer can be computed in polynomial time.

In [20, 19, 21] a characterization of P2P database systems and a model-theoretic semantics dealing with inconsistent peers is proposed. The basic idea is that if a peer does not have models all (ground) queries submitted to the peer are *true* (i.e. are *true* with respect to all models). Thus, if some databases are inconsistent it does not mean that the entire system is inconsistent. The semantics in [20] coincides with the epistemic semantics in [9, 10].

Interesting semantics for data exchange systems that offer the possibility of modeling some preference criteria while performing the data integration process has been proposed in [4–6, 14, 15]. In [4–6] it is proposed a new semantics that allows for a cooperation among pairwise peers that related each other by means of data exchange constraints (i.e. mapping rules) and trust relationships. The decision by a peer on what other data to consider (besides its local data) does not depend only on its data exchange constraints, but also on the trust relationship that it has with other peers. Given a peer P in a P2P system a solution for P is a database instance that respects the exchange constraints and trust relationship P has with its 'immediate neighbors'. Trust relationships are of the form: $(P, less, Q)$ stating that P trusts itself less that Q, and $(P, same, Q)$ stating that P trusts itself the same as Q. This trust relationships are static and are used in the process of collecting data in order to establish preferences in the case of conflicting information.

The introduction of preference criteria among peers is out of the scope of this paper, anyhow we have proposed in recent papers extensions of the maximal weak model semantics that allows to express preferences between peers. More specifically, in [14] it is defined a mechanism that allows to set different degree of reliability for neighbor peers.

Both in [14] and in [4, 6] the mechanism is *rigid* in the sense that the preference among conflicting sets of atoms that a peer can import only depends on the priorities (trust relationship) fixed at design time. To overcome static preferences, in [15] 'dynamic' preferences that allows to select among different scenarios looking at the properties of data provided by the peers is introduced. The work in [15] allows to model concepts like *"in the case of conflicting information, it is preferable to import data from the neighbor peer that can provide the maximum number of tuples"* without selecting a-priori preferred peers.

7 Conclusion

The paper introduces a logic programming based framework and a new semantics for P2P deductive databases.
The presented semantics generalizes previous proposal in the literature by allowing each peer joining the system to select between two different settings: it can either declare its local database to be sound, but not complete; in this case, if inconsistencies arise it gives preference to its knowledge with respect to the knowledge that can be imported from the rest of the system or declare its local knowledge to be unsound; in this case it does not give any preference to its knowledge with respect to the knowledge that can be imported from other peers.

References

1. Abiteboul, S.,Duschka, O. M., Complexity of Answering Queries Using Materialized Views. ACM SIGMOD-SIGACT-SIGAI Symposium on Principles of Database Systems. PODS 1998: 254-263.
2. Arenas, M., Bertossi, L., Chomicki, J., Consistent Query Answers in Inconsistent Databases. *Symposium on Principles of Database Systems*, pp. 68–79, 1999.
3. Bernstein P. A., Giunchiglia, F., Kementsietsidis, A., Mylopulos, J., Serafini, L., and Zaihrayen, I. Data Management for Peer-to-Peer Computing: A Vision, *WebDB*, pp. 89- 94, 2002.
4. Bertossi, L. and Bravo, L., Query Answering in Peer-to-Peer Data Exchange Systems. *EDBT Workshops*, pages 476-485, 2004.
5. Bertossi, L. and Bravo, L., The Semantics of Consistency and Trust in Peer Data Exchange Systems. LPAR 2007: 107-122.
6. Bertossi, L. and Bravo, L.,, Consistency and trust in peer data exchange systems. TPLP 17(2): 148-204 (2017).
7. Calì, A., Calvanese, D., De Giacomo, G., and M. Lenzerini, On the decidability and complexity of query answering over inconsistent and incomplete databases. *PODS*, pp. 260-271, 2003.
8. Calvanese, D., Damaggio, E., De Giacomo, G., Lenzerini M., and Rosati, R., Semantic Data Integration in P2P Systems. DBISP2P 2003: 77-90
9. Calvanese, D., De Giacomo, G., and M. Lenzerini, M. and Rosati, R., Logical foundations of peer-to-peer data integration. *PODS*, pp. 241-251, 2004.
10. Calvanese, D., De Giacomo, G., Lembo D., Lenzerini M., and Rosati, R., Inconsistency tolerance in P2P data integration: An epistemic logic approach. Inf. Syst. 33(4-5): 360-384 (2008).
11. Caroprese, L., Greco, S., Zumpano, E., A Logic Programming Approach to Querying and Integrating P2P Deductive Databases. *The International Florida AI Research Society Conference*, pp. 31-36, 2006.
12. Caroprese, L., Molinaro, C., Zumpano, E., Integrating and Querying P2P Deductive Databases. *International Database Engineering & Applications Symposium*, pp. 285-290, 2006.
13. Caroprese, L., Zumpano, E., Consistent Data Integration in P2P Deductive Databases. SUM 2007: 230-243
14. Caroprese, L., Zumpano, E., Modeling Cooperation in P2P Data Management Systems. ISMIS 2008: 225-235.

15. Caroprese, L., Zumpano, E., Aggregates and priorities in P2P data management systems. IDEAS 2011: 1-7.
16. Caroprese, L., Zumpano, E., Handling Preferences in P2P Systems. FoIKS 2012: 91-106
17. Caroprese, L., Zumpano, E., Generalized Maximal Consistent Answers in P2P Deductive Databases. DEXA (2) 2016: 368-376.
18. Fagin, R., Kolaitis, P. G. Popa, L., Data exchange: getting to the core. ACM Trans. Database Syst. 30(1): 174-210 (2005)
19. Franconi, E., Kuper, G. M., Lopatenko, A., Zaihrayeu,I., Queries and Updates in the coDB Peer to Peer Database System. *VLDB*, pp. 1277-1280, 2004.
20. Franconi, E., Kuper, G. M., Lopatenko, A., Zaihrayeu,I., A Robust Logical and Computational Characterisation of Perto-Peer Database Systems. DBISP2P 2003: 64-76, 2003.
21. Franconi, E., Kuper, G. M., Lopatenko, A., Zaihrayeu,I., A Distributed Algorithm for Robust Data Sharing and Updates in P2P Database Networks. EDBT Workshops 2004: 446-455.
22. Gelfond, M., Lifschitz, V., The Stable Model Semantics for Logic Programming, ICLP/SLP 1988: 1070-1080.
23. Greco, G., Greco, S., and Zumpano, E., Repairing and Querying Inconsistent Databases, *Transactions on Knowledge and Data Engineering*, pp. 1389-1408, 2003.
24. Halevy, A., Ives, Z., Suciu, D., and Tatarinov, I., Schema mediation in peer data management systems. *ICDT*, pp. 505-516, 2003.
25. Alon Y. Halevy, Zachary G. Ives, Dan Suciu, Igor Tatarinov, Schema mediation for large-scale semantic data sharing. VLDB J. 14(1): 68-83 (2005)
26. Lenzerini, M., Data integration: A theoretical perspective. *PODS*, 233-246, 2002.
27. Madhavan, J., and Halevy, A. Y., Composing mappings among data sources. *VLDB*, pp. 572-583, 2003.
28. Sakama, C., Inoue, K., Priorized logic programming and its application to commonsense reasoning. *AI*, No. 123, 185-222, 2000.
29. Tatarinov, I., Halevy., A., Efficient Query reformulation in Peer Data Management Systems, *SIGMOD*, pp. 539-550, 2004.