Integrating Ontological Data Sources Using Viewpoints-Based Approach

Bouchra Boulkroun, Fouzia Benchikha and Chahinez Bachtarzi

LIRE Laboratory, Constantine 2 University Abdelhamid Mehri, Constantine, Algeria

Within the development of Internet and intranets, information integration from various data sources becomes increasingly important and more challenging issue. Recently, the trend in data integration has favored the semantic integration using ontologies. However, the existing ontology-based approaches do not support the aspect of data multi-representations, which is important in the development of multi-user applications. The motivation of this paper is to address a novel semantic integration approach based on ontologies and viewpoints paradigms. This contribution combines the advantages of existing ontology-based integration approaches while avoiding their drawbacks. The proposed integration approach is evaluated using query processing. Profiles are introduced to offer answers to users according to their viewpoints and choices.

ACM CCS (2012) Classification: Information systems → Information integration → Mediators and data integration

Keywords: integration, viewpoint, multi-viewpoint, mediation, ontologies, profiles

1. Introduction

Nowadays, due to the success of the World Wide Web and the rapid growth of information exchange over it, there has been an increasing interest in accessing, relating and combining data from multiple heterogeneous data sources. Indeed, developing information integration systems becomes a crucial need. An information integration system provides a uniform interface to efficiently access and use these heterogeneous data sources.

Several works focusing on heterogeneous data integration have been proposed in the literature. Most of them are based on the common mediation architecture [1], [2]. The main challenge consists on resolving data heterogeneity problems on both structural and semantic levels. The structural heterogeneity is often resolved by defining a common data model. The semantic heterogeneity represents the major problem and ontologies have been recently proposed to deal with it.

Several ontology-based integration approaches have been developed [3], [4]. They can be classified into two categories: the "a priori" and the "a posteriori" approaches [5]. In the former, the concepts of local ontologies are a priori articulated with those of a pre-existent global ontology and the integration process is completely automated. The latter attempts to integrate data sources that possess their own local ontology conceived independently. The integration process, in this case, requires human intervention to define the correspondences between the concepts of different ontologies.

On the other hand, another particularity of data integration is that it is a collaborative task and may involve many experts with different degrees of knowledge in the same application domain [6]. However, each expert can have his own vision of the domain according to his own standpoint, expectations and point of view. Unfortunately, existing integration systems do not support the aspect that data, in a particular viewpoint, can be more relevant to answer user's query.

In this paper, we propose an original ontology-based integration approach that benefits from the advantages of both "a priori" and "a posteriori" approaches. It aims to integrate a large number of ontology-based data sources considering at the same time the existing view-
points and perspectives in the domain. The integration framework of our approach holds three principal characteristics: (i) there is a domain ontology that represents the domain vocabulary as well as the experts’ viewpoints, (ii) the data sources keep their autonomy since they are conceived independently treated as local ontologies, and (iii) the integration process is performed in an automatic way.

We propose a query processing method where SPARQL queries expressed on the multi-viewpoint ontology are reformulated into sub-queries and viewpoints of ontologies. This reformulation makes use of the notion of viewpoints and the mappings between the different levels of the integration system defined in an algebraic formalism.

The remainder of the paper is organized as follows. In the next section, we present some research works on ontology-based integration and viewpoints. An overview of the proposed approach is presented in Section 3. In Section 4, we detail the viewpoint-based integration methodology with its advantages, followed by the mediation architecture in Section 5. In Section 6, we give some validation examples and experiments. Finally, as conclusion, we discuss some prospects and future works.

2. Background

In this section, we present some related works dealing with semantic data integration, then we discuss the utilization of the viewpoint paradigm in different research works and in the context of ontologies in particular.

2.1. Ontology-Based Integration

Information integration has been studied in different domains and contexts to reach several goals. Each domain has its particular requirements that lead to new approach for data integration. In [7], the authors outline some of these new ways such as: event-based integration, stream-based integration. Some researchers have been focused on quality assessment where several works have adopted different quality definitions such as information quality and data quality, but these techniques are generally applied at the query answering stage. In this area, we can quote [8], where the authors propose methods for measuring the quality of information retrieved from the data sources. In [9], the authors propose three quality criteria explicitly for the data integration context: schema completeness, data type consistency, schema minimality. WASSIT [10] is an example of ontology-based integration frameworks that exploits data quality as a criterion for data sources selection. In [6], the authors describe a quality framework for data integration that is able to represent different quality requirements arising from different stakeholders.

In this paper, we are particularly interested in the integration of heterogeneous data sources to ensure their interoperability and to provide a unified access to them. However, the first generation of integration systems was represented by the manual integration [2]. In all these propositions, efforts have been essentially focused on automating the syntactic interoperability of data; the semantic conflicts are solved in a manual way. According to the point of view of Hacid et al. [11], the integration raises several problems such as the expressiveness of the operation and the lack of an intelligent interface that assists users in the formulation of their queries.

For this purpose, several researchers have developed new integration methods based on the notion of ontology called "Ontology-based integration approaches". In the proposed approaches, different types of ontologies are used to deal with data heterogeneity. In the MOMIS project [12], a linguistic ontology is employed to semi-automatically integrate data of structured and semi-structured sources. In other research works such as: PICSEL [13], OB-SERVER [14], KRAFT [15], designers articulate the semantics of their data sources with a conceptual ontology.

Ontology-based integration approaches are classified in two main categories: a posteriori semantic integration and a priori semantic integration.

- The a posteriori semantic integration is adopted when data sources possess their own local ontology conceived independently of the developed integration system. In this approach, the integration is performed in a manual or semi-automatic way by establishing correspondences between the concepts of the ontologies and those of the domain ontology [14], [15]. The main advantage of such an approach is the large autonomy of the participants' data sources. However, human intervention is required to reference explicitly the shared ontology and to define the semantic relations between the concepts of local ontologies and the shared ontology. Another drawback of this approach lies in the evolution of the integration system, since adding a new data source may require changes in the global ontology.

- In the a priori semantic integration, each data source builds its local ontology by a priori taking concepts from the pre-existent domain ontology. Thus, the local ontologies can be seen as subsets of the pre-existent global ontology. The integration in this approach is naturally automatic since the semantic relations between the local ontologies and the global ontology are a priori established [13], [16], [17]. Such an approach allows easily integrating a new data source into the system if the semantic relations of this source are defined in the global ontology. On the other hand, this aspect limits the autonomy of data sources. In [17], authors propose an integration approach where each data source is allowed to make its own extensions of the shared ontology.

2.2. Viewpoints and Ontologies

In knowledge acquisition and modeling areas, the viewpoint paradigm has largely contributed to master the complexity of the design of complex systems. Viewpoints help to model knowledge by giving different perceptions of an object with respect to the observer’s position. By opposition to the mono-viewpoint approach, the multi-viewpoint approach allows modeling the same reality according to different viewpoints. Researches on viewpoints were carried out in the seventies with the works of [18]. They considered that an object could be seen by different observers according to diverse viewpoints. Later, the viewpoint concept has been used with diverse senses in various domains such as in databases [19], [20], knowledge representation [21], [22], semantic web [23], [24], [25], etc.

In this paper, we are particularly interested in the use of viewpoints in ontology's modeling and exploitation. Indeed, authors in [26] were the first to show the duality between ontologies and viewpoints: while ontologies are shared models, viewpoints are local models. They highlight the importance of context-ontologies and propose an extension of the OWL language using contexts. In [25], the author also raises the issue of reusing ontologies in a context where only parts of the originally encoded aspects are relevant. He proposes different viewpoints of the same ontology that can be used for multi-viewpoint reasoning.

However, ontology corresponds to a specific domain modeling and requires that the members of the concerned community make a commitment to use it. According to [21], the maintenance of such a consensus becomes a very difficult task. Therefore, an ontology should be built in a multi-viewpoint environment to take into account the diversity of the data sources and the diverse categories of users by keeping certain level of consensus. The constructed ontology in such a way is called "multi-viewpoint ontology". In [24], the authors define a multi-viewpoint ontology (MVPO) as an ontology in which a concept can be associated to several definitions, each corresponding to a particular viewpoint. The place of the concept in the hierarchy depends on its definition. Viewpoints have been used for different purposes to solve various problems. In the context of our work, we use the viewpoint paradigm to integrate a large number of distributed heterogeneous data sources.

3. Approach Overview

Currently, we agree to recognize the interest of the viewpoints in the conception and the development of multi-user applications that require the cooperation of several experts, each with its interests and knowledge. In [27], the authors show the benefits of viewpoints to allow data integration in an intuitive way.

In this section, we first introduce the basic principles of the proposed viewpoint-based integration approach and then we discuss the purpose of using viewpoints for data integration.
Integrating Ontological Data Sources Using Viewpoints-Based Approach

In this paper, we are particularly interested in the use of viewpoints in ontology modeling and exploitation. Indeed, authors in [26] were the first to show the duality between ontologies and viewpoints: while ontologies are shared models, viewpoints are local models. They highlight the importance of context for ontologies and propose an extension of the OWL language using contexts. In [25], the author also raises the issue of reusing ontologies in a context where only parts of the originally encoded aspects are relevant. He proposes different viewpoints of the same ontology that can be used for multi-viewpoint reasoning.

However, ontology corresponds to a specific domain modeling and requires that the members of the concerned community make a commitment to use it. According to [21], the maintenance of such a consensus becomes a very difficult task. Therefore, an ontology should be built in a multi-viewpoint environment to take into account the diversity of the data sources and the diverse categories of users by keeping certain level of consensus. The constructed ontology in such a way is called "multi-viewpoint ontology". In [24], the authors define a multi-viewpoint ontology (MVPO) as an ontology in which a concept can be associated to several definitions, each corresponding to a particular viewpoint. The placement of the concept in the hierarchy depends on its definition. Viewpoints have been used for different purposes to solve various problems. In the context of our work, we use the viewpoint paradigm to integrate a large number of distributed heterogeneous data sources.

3. Approach Overview

Currently, we agree to recognize the interest of the viewpoints in the conception and the development of multi-user applications that require the cooperation of several experts, each with his interests and knowledge. In [27], the authors show the benefits of viewpoints to allow data integration in an intuitive way.

In this section, we first introduce the basic principles of the proposed viewpoint-based integration approach and then we discuss the purpose of using viewpoints for data integration.
3.1. Basic Principles

The proposed approach is based on the viewpoint paradigm. It allows the integration of numerous distributed heterogeneous ontological data sources in an innovative manner. The integration is held in three levels (see Figure 1): local level, viewpoint level and multi-viewpoint level, detailed in the following.

a) Local level: at this level, a set of \( N \) data sources that participate in the integration system are considered, where:
- Each data source is an ontology-based data source. This means that it defines its terms in its own ontology, called local ontology (LO).
- Each data source represents data of the same domain according to a particular viewpoint.

b) Viewpoint level: it is an intermediate integration level which holds the set of viewpoints ontologies (VPO, \( i = 1, p \)). The data sources are clustered into categories. Each category is related to a particular viewpoint, and:
- Data sources of the same category are associated to the same VPO.
- Each VPO represents one viewpoint (VP) and holds the ontological concepts description according to the considered viewpoint.

c) Multi-viewpoint level: it is the global integration level that holds the multi-viewpoint ontology (MVPO), where:
- The MVPO represents the shared ontology and provides a conceptual description of the domain by a set of conceptual terms.
- The MVPO provides a unified vision of the different VPO. It is used as a reference for the classification of the data sources into viewpoint categories.

3.2. Why Using Viewpoints?

In this section, we essentially motivate the choice of the viewpoint paradigm in our approach and present its main advantages. However, existing ontology-based integration approaches have the ability to respond to various problems related to the semantic heterogeneity but they do not take into account the divergence of the experts’ and end users’ viewpoints. Therefore, the development of an integration approach that exploits both ontologies and viewpoints appears as a revolution in this context. Moreover, the use of viewpoints allows avoiding the problem of conflicts related to the different interpretations that may have the same object in the real world. Other advantages are offered:
- Viewpoints can be used as a means for the interpretation of initial knowledge of the domain ontology.
- Viewpoints allow filtering the access to data sources according to the users’ preferences and needs as a way to improve system performances.
- Viewpoints can separate the initial knowledge at different granularity levels. This separation promotes the division of the domain into several sub-domains, which are easier to manage.

4. The Proposed Integration Methodology

In this section, we present the proposed viewpoint-based integration approach, then we compare it with existing techniques and systems.

4.1. The Viewpoint-Based Integration Approach

The integration methodology proposed in this paper allows designers to exploit existing ontology-based data sources for the realization of a unified and integrated system. It breaks the integration process into four steps (see Figure 2): MVPO definition, VPO extraction, correspondences detection and data sources classification.

In the next section, we give more details about each step, and provide formal definitions of the concepts. These definitions aim to reduce the specification ambiguity.

4.1.1. MVPO Definition

Rather than developing a MVPO from scratch, it is more convenient and effortless to reuse existing domain ontology. The goal of this phase is to exploit domain ontology and transform it into a MVPO. Indeed, to achieve a multi-viewpoint representation for different experts’ perspectives, the MVPO must contain all the knowledge and terminologies of their viewpoints. In the following, a formal definition of a MVPO is given. The MVPO-OWL language [21] is adopted for the MVPO representation. In the annex, we succinctly present this model.

Definition 4.1. (Multi-View Point Ontology): a MVPO allows expressing in the same ontology multiple data descriptions according to different experts’ viewpoints, and the consensus between them. A MVPO is a sextuple: \((C, VP, CVP, P, R, I)\) where:
- \(C\): is the set of classes/concepts, defined by the primitives of OWL: owl:Thing and owl:Nothing
- \(VP\): is the set of viewpoints represented in the ontology. They are defined by the primitive vp:viewpoint
- \(CVP\): is the set of concepts defined according to the experts’ viewpoints. \(CVP = \bigcup_{i=1}^{p} CVP_i\), where \(CVP_i\) is the set of concepts defined according to a particular viewpoint \(VP_i\).
- \(P\): is the set of properties, \(P = P_D \cup P_I\), where:
  - \(P_D\): the set of properties having a dual for value.
  - \(P_I\): the set of properties having an individual for value.
- \(R\): the set of relations between the concepts of different viewpoints.
- \(I\): the set of individuals.

Figure 3 depicts a fragment of the MVPO for a “Manufacturing company” expressed using the MVPO-OWL model. It contains three viewpoints: “Accounting”, “Commercial” and “Manufacturing” viewpoints.

4.1.2. VPO Extraction

In this phase, after the identification of the existing viewpoints, the VPOs are generated. They are conceived by a priori extracting concepts, properties and instances which are pertinent in the considered viewpoint from the MVPO, leading to its partitioning into a set of sub-ontologies. The formal definition of a VPO is given in the following.

Definition 4.2. (Viewpoint ontology): Each VPO is considered as a sub-set of the MVPO and is defined as a quadruplet VPO: \((CVP, VP, P, R)\) where:
- \(CVP\): is the set of concepts defined according to the viewpoint \(VP\).
- \(P\): is the set of properties defined according to the viewpoint \(VP\).
- \(R\): is the set of relations between the concepts of the same viewpoint \(VP\).
3.1. Basic Principles

The proposed approach is based on the viewpoint paradigm. It allows the integration of numerous distributed heterogeneous ontological data sources in an innovative manner. The integration is held in three levels (see Figure 1): local level, Viewpoint level and Multi-Viewpoint level, detailed in the following.

c) Multi-viewpoint level: it is the global integration level that holds the multi-viewpoint ontology (MVPO), where:

- The MVPO represents the shared ontology and provides a conceptual description of the domain by a set of conceptual terms.
- The MVPO provides a unified vision of the different VPO. It is used as a reference for the classification of the data sources into viewpoint categories.

3.2. Why Using Viewpoints?

In this section, we essentially motivate the choice of the viewpoint paradigm in our approach and present its main advantages. However, existing ontology-based integration approaches have the ability to respond to various problems related to the semantic heterogeneity but they do not take into account the divergence of the experts’ and end users’ viewpoints. Therefore, the development of an integration approach that exploits both ontologies and viewpoints appears as a revolution in this context. Moreover, the use of viewpoints allows avoiding the problem of conflicts related to the different interpretations that may have the same object in the real world. Other advantages are offered:

- Viewpoints can be used as a means for the interpretation of initial knowledge of the domain ontology.
- Viewpoints allow filtering the access to data sources according to the users’ preferences and needs as a way to improve system performances.
- Viewpoints can separate the initial knowledge at different granularity levels. This separation promotes the division of the domain into several sub-domains, which are easier to manage.

4. The Proposed Integration Methodology

In this section, we present the proposed viewpoint-based integration approach, then we compare it with existing techniques and systems.

4.1. The Viewpoint-Based Integration Approach

The integration methodology proposed in this paper allows designers to exploit existing ontology-based data sources for the realization of a unified and integrated system. It breaks the integration process into four steps (see Figure 2): MVPO definition, VPO extraction, correspondences detection and data sources classification. In the next section, we give more details about each step, and provide formal definitions of the concepts. These definitions aim to reduce the specification ambiguity.

4.1.1. MVPO Definition

Rather than developing a MVPO from scratch, it is more convenient and effortless to reuse existing domain ontology. The goal of this phase is to exploit domain ontology and transform it into a MVPO. Indeed, to achieve a multi-viewpoint representation for different experts’ perspectives, the MVPO must contain all the knowledge and terminologies of their viewpoints. In the following, a formal definition of a MVPO is given. The MVPO-OWL language [21] is adopted for the MVPO representation. In the annex, we succintly present this model.

Definition 4.1. (Multi-View Point Ontology): a MVPO allows expressing in the same ontology multiple data descriptions according to different experts’ viewpoints, and the consensus between them. A MVPO is a sextuple: (C, VP, CVP, P, R, I) where:

- C: is the set of classes/concepts, defined by the primitives of OWL and nothing
- VP: is the set of viewpoints represented in the ontology. They are defined by the primitive vp:viewpoint
- CVP: is the set of concepts defined according to the experts’ viewpoints. CVP = \bigcup_{VP \in VP} CVP, where CVP is the set of concepts defined according to a particular viewpoint VP.
- P: is the set of properties, P = PVP ∪ P1, where:
  - PVP: the set of properties having a data for value.
  - P1: the set of properties having an individual for value.
- R: the set of relations between the concepts of different viewpoints.
- I: the set of individuals.

Figure 3 depicts a fragment of the MVPO for a “Manufacturing company” expressed using the MVPO-OWL model. It contains three viewpoints: “Accounting”, “Commercial” and “Manufacturing” viewpoints.

4.1.2. VPO Extraction

In this phase, after the identification of the existing viewpoints, the VPOs are generated. They are conceived by a priori extracting concepts, properties and instances which are pertinent in the considered viewpoint from the MVPO, leading to its partitioning into a set of sub-ontologies. The formal definition of a VPO is given in the following.

Definition 4.2. (Viewpoint ontology): Each VPO is considered as a sub-set of the MVPO and is defined as a quadruplet VPO: (CVP, VP, PVP, RVP, IVP) where:

- CVP ⊆ CVP, is the set of concepts defined according to the viewpoint VP.
- PVP ⊆ P is the set of properties defined according to the viewpoint VP.
- RVP ⊆ R: is the set of relations between the concepts of the same viewpoint VP.
B. Boulkroun et al.
Integrating Ontological Data Sources Using Viewpoints-Based Approach

- $I_{VP} \subseteq I$: is the set of individuals defined according to the viewpoint $VP$.

The set of VPOs is formally represented using an operational language (OWL). Figure 4 represents an extract of two VPOs: "Accounting" and "Manufacturing". The global mapping table (GMT) is generated to define the correspondences between the MVPo and the VPOs.

4.1.3. Correspondences Detection

The main objective of this step is to identify, through similarity measures, the relations between entities of the MVPo and those of the local ontologies. These relations will be used later for data sources classification. To this end, we exploit the ASCO1 algorithm proposed by [21]. This algorithm allows deducing structural similarities between the entities of different ontologies. We propose the ASCO-VP as an extension of the ASCO1 algorithm. In ASCO-VP, we add some measures to consider both semantic and structural similarities. The Initial Mapping Table (IMT), containing mapping information between the local ontologies and the MVPo, is generated as result.

**WordNet similarity.** WordNet is a semantic network, where every node is a set of synonyms, called "synset" representing a real-world concept. The synsets are connected by links that describe the relation between different concepts. The idea is that two concepts are semantically close if their synsets are at least connected by a path. The WordNet similarity is given by $S_{WN}$.

- **Combinations of properties similarity:** The final results of attributes and roles distances are organized in two matrices ADM (Attribute Distance Matrix) and RDM (Roles Distance Matrix), respectively. These matrices are used to calculate the similarity of two concepts according to their attributes and roles.

**Definition 4.3. (ADM and RDM):** Let ADM and RDM be two matrices that contain all the similarity distance proportions between the attributes/roles of two concepts $C_1$ and $C_2$. The similarity average value for $n$ attributes or roles between $C_1$ and $C_2$ is respectively returned by $||ADM||$ or $||RDM||$.

- **Attributes distance (AD):** The attributes distance $AD$ measures the similarity between two attributes $P_i$ and $P_j$ of two concepts $C_1$ and $C_2$.

- **Roles distance (RD):** The roles distance $RD$ measures the similarity between two roles $R_i$ and $R_j$ of two concepts $C_1$ and $C_2$.


The set of VPOs is formally represented using an operational language (OWL). Figure 4 represents an extract of two VPOs: "Accounting" and "Manufacturing". The global mapping table (GMT) is generated to define the correspondences between the MVPO and the VPOs.

4.1.3. Correspondences Detection

The main objective of this step is to identify, through similarity measures, the relations between entities of the MVPO and those of the local ontologies. These relations will be used later for data sources classification. To this end, we exploit the ASCO1 algorithm proposed by [21]. This algorithm allows deducting structural similarities between the entities of different ontologies. We propose the ASCO-VP as an extension of the ASCO1 algorithm. In ASCO-VP, we add some measures to consider both semantic and structural similarities. The Initial Mapping Table (IMT), containing mapping information between the local ontologies and the MVPO, is generated as result.

WordNet similarity. WordNet is a semantic network, where every node is a set of synonyms, called "synset" representing a real-world concept. The synsets are connected by links that describe the relation between different concepts. The idea is that two concepts are semantically close if their synsets are at least connected by a path. The WordNet similarity is given by $S_{WN}$:

- **Attributes distance (AD):** The attribute distance $AD$ measures the similarity between two attributes $P_i$ and $P_j$ of two concepts $C_1$ and $C_2$, respectively.

- **Roles distance (RD):** The roles distance $RD$ measures the similarity between two roles $R_i$ and $R_j$ of two concepts $C_1$ and $C_2$, respectively.

Properties similarity. The properties similarity value is calculated from two proportions of similarity based on the internal structure of the concepts. These proportions concern the attributes distance and the roles, which are defined using the linguistic similarity $LSim$ of ASCO1.

Combination of properties similarity: The final results of attributes and roles distances are organized in two matrices $ADM$ (Attribute Distance Matrix) and $RDM$ (Roles Distance Matrix), respectively. These matrices are used to calculate the similarity of two concepts according to their attributes and roles.

Definition 4.3. (ADM and RDM): Let $ADM$ and $RDM$ be two matrices that contain all the similarity distance proportions between the attributes/roles of two concepts $C_1$ and $C_2$. The similarity average value for $n$ attributes or roles between $C_1$ and $C_2$ respectively is respectively returned by $||ADM||$ or $||RDM||$.

"Combination of properties similarity: The final results of attributes and roles distances are organized in two matrices $ADM$ (Attribute Distance Matrix) and $RDM$ (Roles Distance Matrix), respectively. These matrices are used to calculate the similarity of two concepts according to their attributes and roles.

Definition 4.3. (ADM and RDM): Let $ADM$ and $RDM$ be two matrices that contain all the similarity distance proportions between the attributes/roles of two concepts $C_1$ and $C_2$. The similarity average value for $n$ attributes or roles between $C_1$ and $C_2$ is respectively returned by $||ADM||$ or $||RDM||$.

Figure 4. An extract of the Accounting VPO and Commercial VPO.

The ASCO-VP algorithm is given in Figure 5: all the extensions are represented in bold characters.

WordNet similarity. WordNet is a semantic network, where every node is a set of synonyms, called "synset" representing a real-world concept. The synsets are connected by links that describe the relation between different concepts. The idea is that two concepts are semantically close if their synsets are at least connected by a path. The WordNet similarity is given by $S_{WN}$:

\[
S_{WN}(C_1, C_2) =
\begin{cases}
1 & \text{if } C_1 \text{ and } C_2 \text{ are synonyms or } \exists \text{ at least a terminological relation in WordNet between the synsets of } C_1 \text{ and } C_2 \\
0 & \text{in the opposite case}
\end{cases}
\]

Properties similarity. The properties similarity value is calculated from two proportions of similarity based on the internal structure of the concepts. These proportions concern the attributes distance and the roles, which are defined using the linguistic similarity $LSim$ of ASCO1.

- **Attributes distance (AD):** The attribute distance $AD$ measures the similarity between two attributes $P_i$ and $P_j$ of two concepts $C_1$ and $C_2$, respectively.

- **Roles distance (RD):** The roles distance $RD$ measures the similarity between two roles $R_i$ and $R_j$ of two concepts $C_1$ and $C_2$, respectively.

\[
AD(C_1, P_1, C_2, P_2) =
\begin{cases}
1 & \text{if } P_1 \text{ and } P_2 \text{ are identical or synonymic} \\
0 & \text{in the opposite case}
\end{cases}
\]

\[
RD(C_1, R_1, C_2, R_2) =
\begin{cases}
1 & \text{if } R_1 \text{ and } R_2 \text{ are identical or synonymic} \\
0 & \text{in the opposite case}
\end{cases}
\]

Figure 5. Correspondences detection algorithm ASCO-VP.
where
\[
\lambda(i) = \begin{cases} 
1 & \text{if } \sum_{j=1}^{n} RD(C_j, R_i, C_j) > 0 \\
0 & \text{in the opposite case}
\end{cases}
\]

Definition 4.4. (Properties similarity): Let A and B be two classes of two ontologies \(O_1\) and \(O_2\), respectively. The properties similarity between A and B is a similarity function of signature \(S_{\text{Property}}: O_1 \times O_2 \rightarrow [0, 1]\), which is calculated as the average of the RDM and the ADM:
\[
S_{\text{Property}} = \text{Average}(||\text{ADM}|| + ||\text{RDM}||)
\]

Correspondences generation. The result of the ASCO\(_{\text{VP}}\) algorithm is a set of correspondences between the entities of the shared ontology and every local ontology. These correspondences are represented by triplets \((A, B, S_{\text{Total}})\), where A and B are two entities of the same type in both ontologies, and \(S_{\text{Total}}\) is the similarity value between them. Thus, the correspondences are generated according to the last result of similarity \(S_{\text{Total}}\) where \(S_{\text{Total}}\) represents the sum of \(S_{\text{Linguistic}}, S_{\text{Property}}, S_{\text{Structural}}\) values.

Definition 4.5. (Total similarity): Let A and B be two entities of the same type of two ontologies \(O_1\) and \(O_2\), respectively. The total similarity between these two entities is a similarity function \(S_{\text{Total}}: O_1 \times O_2 \rightarrow [0, 1]\) such as:
\[
S_{\text{Total}}(A, B) = S_{\text{Linguistic}}(A, B) \times W_{\text{Linguistic}} + S_{\text{Property}}(A, B) \times W_{\text{Property}} + S_{\text{Structural}}(A, B) \times W_{\text{Structural}}
\]
where \(W_{\text{Linguistic}} + W_{\text{Property}} + W_{\text{Structural}} = 1\).

Let \(S_{\text{Simularity}} \geq 0\) be a predefined threshold of similarity. Two entities A and B are considered similar, if and only if: \(S_{\text{Total}}(A, B) \geq S_{\text{Simularity}}\). At the end of this process, a mapping table is generated (Initial Mapping Table IMT). It contains mapping information between local ontologies and the shared ontology including the defined viewpoints. This latter will be used in the definition of correspondences between VPO and local ontologies.

4.1.4 Data sources classification

This step aims to classify data sources into viewpoints categories; each category represents a particular viewpoint. To this end, the similarity results obtained in the previous step are used. We also propose a similarity measure called \(S_{\text{VP}}\) to calculate the similarity between the concepts of a local ontology and the concepts representing a particular viewpoint in the MVPO. The result of this stage is represented by basic similarity matrix (SM), which will be used later to build the IMT. The SM is a set of several sub-matrices (SM-S), each one corresponding to a particular viewpoint (see Figure 6).

![Figure 6. Basic similarity matrix.](image)

Definition 4.6. (Viewpoints similarity): Let the SM be the matrix representing the similarity between the concepts of the two ontologies MVPO and \(LO_1\). \(VP\) is the set of viewpoints represented in the global ontology, and \(S_{\text{SM}}\) is the sub-matrix that contains the similarity values between the concepts of a local ontology and those of a particular viewpoint. The viewpoint’s similarity \(S_{\text{VP}}\) between a local ontology and a particular viewpoint is a similarity function: \(S_{\text{VP}}: LO_1 \times VP \rightarrow [0, 1]\) such as:
\[
S_{\text{VP}}(LO_1, VP) = \sum_{i, j=1}^{n} S_{\text{Total}}(C_i, C_j) \times W_{\text{Total}}
\]
where \(S_{\text{Total}} \geq 0.5\).

At this stage, both the GMT and the IMT are used to build the local mapping table (LMT) which represents the possible links between the concepts of the \(LO_1\) in a same category and those in the correspondent VPO.

4.2. Discussion

Through the literature review of semantic data integration approaches, the advantages and the inconvenience of a posteriori semantic integration and a priori semantic integration are in duality. The a posteriori semantic integration guarantees the autonomy of the local ontologies, but it is performed in a manual or semi-automatic way and thus requires more efforts. In contrast, the a priori semantic integration provides a low autonomy degree for the data sources, but the integration process is completely automated.

Our approach brings an innovative solution that presents the advantages of the two approaches while avoiding their drawbacks (see Table 1). It provides two kinds of ontologies: (1) the VPO, which allows taking into account a limited and controlled consensus sufficient for the integration of a set of data sources related to a particular viewpoint, and (2) the MVPO, which offers the consensual knowledge shared by all the data sources; it ensures the semantic compatibility between the different viewpoint descriptions offered by the VPOs. With its specific characteristics, our integration approach offers the following advantages:

- it helps the exploitation of a unique and shareable ontology by accessing it according to several viewpoints and using it in a decentralized way,
- it ensures a completely automated integration process based on similarity measures used for the classification of the data sources,

<table>
<thead>
<tr>
<th>Characteristics</th>
<th>A priori approach</th>
<th>A posteriori approach</th>
<th>Viewpoint-based integration approach</th>
</tr>
</thead>
<tbody>
<tr>
<td>Local ontologies</td>
<td>Created</td>
<td>Pre-existing</td>
<td>Pre-existing</td>
</tr>
<tr>
<td>Integration automaticity</td>
<td>Automatic</td>
<td>Manual/Semiautomatic</td>
<td>Automatic</td>
</tr>
<tr>
<td>Sources autonomy</td>
<td>Less autonomy</td>
<td>High autonomy</td>
<td>High autonomy</td>
</tr>
<tr>
<td>System dynamicity</td>
<td>Easy integration</td>
<td>Difficult integration of a new data sources</td>
<td>Easy integration of a new data sources</td>
</tr>
</tbody>
</table>
where
\[ \lambda(i) = \begin{cases} 1 & \text{if } \sum_{j=1}^{n} RD(C_i, R_i, C_j, R_j) > 0 \\ 0 & \text{in the opposite case} \end{cases} \]

**Definition 4.4. (Properties similarity):** Let A and B be two classes of two ontologies O₁ and O₂ respectively. The properties similarity between A and B is a similarity function of signature \( S_{\text{Prop}}: O_1 \times O_2 \rightarrow [0, 1] \), which is calculated as the average of the RDM and the ADM:

\[ S_{\text{Prop}} = \text{Average}(\| \text{ADM} \| - \| \text{RDM} \|) \]

**Correspondences generation.** The result of the ASCOVP algorithm is a set of correspondences between the entities of the shared ontology and every local ontology. These correspondences are represented by triplets \((A, B, S_{\text{RDM}})\), where \( A \) and \( B \) are two entities of the same type in both ontologies, and \( S_{\text{RDM}} \) is the similarity value between them. Thus, the correspondences are generated according to the last result of similarity \( S_{\text{Total}} \) where \( S_{\text{Total}} \) represents the sum of \( S_{\text{Structural}} \) and \( S_{\text{Property}} \) values.

**Definition 4.5. (Total similarity):** Let \( A \) and \( B \) be two entities of the same type of two ontologies \( O_1 \) and \( O_2 \), respectively. The total similarity between these two entities is a similarity function \( S_{\text{Total}}: O_1 \times O_2 \rightarrow [0, 1] \) such as:

\[ S_{\text{Total}}(A, B) = S_{\text{Linguistic}}(A, B) \times W_{\text{Linguistic}} + S_{\text{Property}}(A, B) \times W_{\text{Property}} + S_{\text{Structural}}(A, B) \times W_{\text{Structural}} \]

where \( W_{\text{Linguistic}} + W_{\text{Property}} + W_{\text{Structural}} = 1 \).

Let \( S_{\text{Similarity}} \geq 0 \) be a predefined threshold of similarity. Two entities \( A \) and \( B \) are considered similar, if and only if: \( S_{\text{Total}}(A, B) \geq S_{\text{Similarity}} \).

At the end of this process, a mapping table is generated (Initial Mapping Table IMT). It contains mapping information between local ontologies and the shared ontology including the defined viewpoints. This latter will be used in the definition of correspondences between VPO and local ontologies.

**4.1.4 Data sources classification**

This step aims to classify data sources into viewpoints categories; each category represents a particular viewpoint. To this end, the similarity results obtained in the previous step are used. We also propose a similarity measure called \( S_{\text{VP}} \) to calculate the similarity between the concepts of a local ontology and the concepts representing a particular viewpoint in the MVPO. The result of this stage is represented by basic similarity matrix (SM), which will be used later to build the IMT. The SM is a set of several sub-matrices (S-SM), each one corresponding to a particular viewpoint (see Figure 6).

\[ S_{\text{VP}}(LO_i, VP_j) = \frac{\sum_{i,j} S_{\text{Total}}(C_i, C_j)}{\max \{S_{\text{Total}}(C_i, C_j)\}} \]

where \( S_{\text{Total}} \geq 0.5 \).

Where \( \max \{C_i, C_j\} \) is the maximal number of compared concepts between the local ontology \( LO_i \) and a given viewpoint \( VP_j \).

The algorithm that allows data sources classification is given in the following Algorithm 1 while avoiding their drawbacks (see Table 1). It provides two kinds of ontologies: (1) the VPO, which allows taking into account a limited and controlled consensus sufficient for the integration of a set of data sources related to a particular viewpoint, and (2) the MVPO, which offers the consensual knowledge shared by all the data sources; it ensures the semantic compatibility between the different viewpoint descriptions offered by the VPOs. With its specific characteristics, our integration approach offers the following advantages:

- it helps the exploitation of a unique and shareable ontology by accessing it according to several viewpoints and using it in a decentralized way,
- it ensures a completely automated integration process based on similarity measures used for the classification of the data sources,
- it offers the advantages of the two approaches.

**Table 1. Ontology-based integration approaches comparison.**
• it offers a high level of autonomy for the participant sources as the addition of a new data source doesn’t require either antecedent conditions or a priori articulations,
• the integration process is effortless since the integration of a new data source does not affect the global ontology; instead, this task is supported by an existing VPO,
• it ensures the coherence of the system by integrating different involved viewpoints in the field. The consistency of the system can be kept only by the integration of different viewpoints in a single representation (in our approach the MVPO construction).

Our approach is compared with existing semantic integration systems. Table 2 gives some comparison criteria. Note that, in OBSERVER [14], data sources are also grouped. A single local ontology is used for each group. Furthermore, the concept of global ontology is not supported and a unified access is not provided.

After presenting the proposed viewpoint-based integration approach, we outline in the following section the global architecture of the system.

5. Viewpoint-Based Mediation Architecture

The architecture supporting the viewpoint-based integration approach is represented as an arborescence of mediators. In this architecture, three levels are distinguished (see Figure 7): local level including ontology-based data sources, viewpoint-based integration level including a set of viewpoint mediators, and the global level containing the global mediator and the user’s interface. In addition, the viewpoint-based architecture includes an integration module.

• Integration module: The integration module is responsible for the classification of the different data sources for their integration. During the integration process, the MVPO is first constructed using the MVP-OWL language and the ontology editor Protégé. The participant local ontologies are then imported into the system using the charging module. This latter is implemented using the Jena library that contains Java classes allowing the development of Semantic Web applications. The different parts of the ASCOVP algorithm are implemented using Java language and its evaluation is achieved by a set of pertinence measures, such as the precision, recall and F-measure. The VPOs extraction is performed by a specific module that is based on a set of algorithms exploiting the basics of the MVP-OWL language.

• Global mediator: The goal of this mediator is the management and the processing of the end users queries. When a user query is expressed via the user interface, it is sent to the global mediator which uses the GMT to rewrite and optimize the query execution plan (QEP) using the rewriting and optimization modules successively. Afterward, the sub-queries are transferred to the concerned viewpoint mediators.

• The viewpoint mediator: This mediator uses the evaluation and execution modules to rewrite and execute optimized sub-queries over the pertinent data sources. Finally, the recomposition module is used at the viewpoint-based level as well as the global level to reformulate the answers according to the QEP.

Between the different levels of the architecture, mappings (IMT, LMT and GMT) that represent possible links between the different ontologies are created and managed. In order to define a generic process for SPARQL query treatment, the mappings are translated into the SPARQL algebra [26]. An example of both global mappings and local mappings represented in the form of SPARQL assertions is given in Table 3.

6. Evaluation and Validation

In this section, we consider both qualitative and quantitative evaluation of our approach. For the former, we present a simple case study applying the query processing method. For the latter, we give some experiments, which will show that the proposed approach may reduce the execution time of the users’ queries.

<table>
<thead>
<tr>
<th>Systems</th>
<th>Data type</th>
<th>Ontology architecture</th>
<th>Semantic integration</th>
<th>System architecture</th>
<th>Integration module</th>
</tr>
</thead>
<tbody>
<tr>
<td>OBSERVER</td>
<td>HTML, Data bases, Files</td>
<td>Multiple</td>
<td>A posteriori</td>
<td>Adapter, ontology server</td>
<td></td>
</tr>
<tr>
<td>KRAFT</td>
<td>Data bases</td>
<td>Hybrid</td>
<td>A posteriori</td>
<td>Agent-based</td>
<td></td>
</tr>
<tr>
<td>SIMS</td>
<td>Data bases</td>
<td>Unique</td>
<td>A priori</td>
<td>Ontology-based mediation</td>
<td></td>
</tr>
<tr>
<td>PICSEL</td>
<td>Data bases, XML</td>
<td>Hybrid</td>
<td>A priori</td>
<td>Ontology-based mediation</td>
<td></td>
</tr>
<tr>
<td>Proposed system</td>
<td>OHDB</td>
<td>Hybrid</td>
<td>Hybrid</td>
<td>Ontology-based mediation</td>
<td></td>
</tr>
</tbody>
</table>

Table 2. Ontology-based integration systems comparison.

<table>
<thead>
<tr>
<th>Systems</th>
<th>Data type</th>
<th>Ontology architecture</th>
<th>Semantic integration</th>
<th>System architecture</th>
<th>Integration module</th>
</tr>
</thead>
<tbody>
<tr>
<td>OBSERVER</td>
<td>HTML, Data bases, Files</td>
<td>Multiple</td>
<td>A posteriori</td>
<td>Adapter, ontology server</td>
<td></td>
</tr>
<tr>
<td>KRAFT</td>
<td>Data bases</td>
<td>Hybrid</td>
<td>A posteriori</td>
<td>Agent-based</td>
<td></td>
</tr>
<tr>
<td>SIMS</td>
<td>Data bases</td>
<td>Unique</td>
<td>A priori</td>
<td>Ontology-based mediation</td>
<td></td>
</tr>
<tr>
<td>PICSEL</td>
<td>Data bases, XML</td>
<td>Hybrid</td>
<td>A priori</td>
<td>Ontology-based mediation</td>
<td></td>
</tr>
<tr>
<td>Proposed system</td>
<td>OHDB</td>
<td>Hybrid</td>
<td>Hybrid</td>
<td>Ontology-based mediation</td>
<td></td>
</tr>
</tbody>
</table>

Table 3. Correspondences assertions.
it offers a high level of autonomy for the participant sources as the addition of a new data source doesn't require either antecedent conditions or a priori articulations,

- the integration process is effortless since the integration of a new data source does not affect the global ontology; instead, this task is supported by an existing VPO,

- it ensures the coherence of the system by integrating different involved viewpoints in the field. The consistency of the system can be kept only by the integration of different viewpoints in a single representation (in our approach the MVPO construction).

Our approach is compared with existing semantic integration systems. Table 2 gives some comparison criteria. Note that, in OBSERVER [14], data sources are also grouped. A single local ontology is used for each group. Furthermore, the concept of global ontology is not supported and a unified access is not provided.

After presenting the proposed viewpoint-based integration approach, we outline in the following section the architecture of the system.

5. Viewpoint-Based Mediation Architecture

The architecture supporting the viewpoint-based integration approach is represented as an arborescence of mediators. In this architecture, three levels are distinguished (see Figure 7): local level including ontology-based data sources, viewpoint-based integration level including a set of viewpoint mediators, and the global level containing the global mediator and the user's interface. In addition, the viewpoint-based architecture includes an integration module.

- **Integration module**: The integration module is responsible for the classification of the different data sources for their integration. During the integration process, the MVPO is first constructed using the MVP-OWL language and the ontology editor Protégé. The participant local ontologies are then imported into the system using the charging module. This latter is implemented using the Jena library that contains Java classes allowing the development of Semantic Web applications. The different parts of the ASCO\_p algorithm are implemented using Java language and its evaluation is achieved by a set of pertinence measures, such as the precision, recall and F-measure. The VPOs extraction is performed by a specific module that is based on a set of algorithms exploiting the basics of the MVP-OWL language.

- **Global mediator**: The goal of this mediator is the management and the processing of the end users queries. When a user query is expressed via the user interface, it is sent to the global mediator which uses the GMT to rewrite and optimize the query execution plan (QEP) using the rewriting and optimization modules successively. Afterward, the sub-queries are transferred to the concerned viewpoint mediators.

- **The viewpoint mediator**: This mediator uses the evaluation and execution modules to rewrite and execute optimized sub-queries over the pertinent data sources. Finally, the recomposition module is used at the viewpoint-based level as well as the global level to reformulate the answers according to the QEP.

Between the different levels of the architecture, mappings (IMT, LMT and GMT) that represent possible links between the different ontologies are created and managed. In order to define a generic process for SPARQL query treatment, the mappings are translated into the SPARQL algebra [28]. An example of both global mappings and local mappings represented in the form of SPARQL assertions is given in Table 3.

6. Evaluation and Validation

In this section, we consider both qualitative and quantitative evaluation of our approach. For the former, we present a simple case study applying the query processing method. For the latter, we give some experiments, which will show that the proposed approach may reduce the execution time of the users' queries.

---

**Table 2.** Ontology-based integration systems comparison.

<table>
<thead>
<tr>
<th>Systems</th>
<th>Data type</th>
<th>Ontology architecture</th>
<th>Semantic integration</th>
<th>System architecture</th>
</tr>
</thead>
<tbody>
<tr>
<td>OBSERVER</td>
<td>HTML, Data bases, Files</td>
<td>Multiple</td>
<td>A posteriori</td>
<td>Adapter, ontology server</td>
</tr>
<tr>
<td>KRAFT</td>
<td>Data bases</td>
<td>Hybrid</td>
<td>A posteriori</td>
<td>Agent-based</td>
</tr>
<tr>
<td>SIMS</td>
<td>Data bases</td>
<td>Unique</td>
<td>A priori</td>
<td>Ontology-based mediation</td>
</tr>
<tr>
<td>PICSEL</td>
<td>Data bases, XML</td>
<td>Hybrid</td>
<td>A priori</td>
<td>Ontology-based mediation</td>
</tr>
<tr>
<td>Proposed system</td>
<td>OHDB</td>
<td>Hybrid</td>
<td>Hybrid</td>
<td>Ontology-based mediation</td>
</tr>
</tbody>
</table>

---

**Table 3.** Correspondences assertions.

<table>
<thead>
<tr>
<th>SGMT assertions</th>
<th>LMT assertions</th>
</tr>
</thead>
<tbody>
<tr>
<td><em>the namespace prefixes &quot;mvpo:&quot;, &quot;vpoa:&quot; and &quot;vpoc:&quot; refer to the vocabularies of the MVPO, Accounting VPO and Commercial VPO, respectively</em></td>
<td></td>
</tr>
<tr>
<td>Class' assertions:</td>
<td></td>
</tr>
<tr>
<td>1. (?p rdf type mvpo:AccountingProduct) ← (?p rdf type vpoa: Product)</td>
<td></td>
</tr>
<tr>
<td>2. (?p rdf type mvpo:CommercialProduct) ← (?p rdf type vpoa: Product)</td>
<td></td>
</tr>
<tr>
<td>Property' assertions:</td>
<td></td>
</tr>
<tr>
<td>1. (?p mvpo:Average-margin ? Average-margin) ← (?p vpoc: Average-margin ? Average-margin)</td>
<td></td>
</tr>
<tr>
<td>2. (?p mvpo:Sales ?Sales) ← (?pvpoc: Sales ? Sales)</td>
<td></td>
</tr>
</tbody>
</table>

---

*the namespace prefixes "vpoc:," "s1:" and "s2:" refer to the vocabularies of the Manufacturing VPO and data sources S1 and S2 respectively*;
6.1. Qualitative Validation

The query processing method proceeds in four steps: query rewriting, optimization, evaluation and execution (see Figure 8).

Let’s consider a query Q expressed in SPARQL over the MVPO given in Figure 4. The query Q searches the designation, the average margin and the sales of products sold to customers who live in Constantine (see Figure 9).

6.1.1. Query Rewriting

The query Q is firstly rewritten into a query tree describing the involved entities, properties and eventually conditions. Then, using the GMT containing the correspondences assertions presented in Table 4, the query tree is expanded to a combination of sub-queries, where the MVPO concepts are replaced by the VPOs correspondent concepts (see Figure 10). The property Average-margin searched by the user characterizes the class Product in the "Accounting VPO", and the property sales characterizes the class Product in the "Commercial VPO".

6.1.2. Optimization

In this step, the optimal query execution plan is generated using the cost model. In the considered example, the execution plan of Q contains two sub-queries SQ1 and SQ2, which can be executed in parallel on the "Accounting VPO" and the "Commercial VPO". SQ1 extracts designation and average margin of products sold to customers from Constantine, and SQ2 extracts designation and sales of products sold to customers from Constantine. The generated sub-queries are represented in Figure 11, and the execution plan is the union of both sub-queries (SQ1 \( \cup \) SQ2).

6.1.3. Evaluation

In this step, the relevant queries over the data sources are obtained. However, using the LMT, sub-queries sent to the target viewpoint mediators are rewritten in terms of the local ontologies of the same viewpoint. A set of optimal QEPs representing the local QEPs is generated. For example, Figure 12 presents the decomposition of SQ2 using LMT in Table 3 defined between the "Commercial VPO" and the local ontologies.

6.1.4. Execution and Recomposition

After the execution of the sub-queries over the data sources, results are sent back to the combination module at the viewpoint level. This one builds the answers received form data sources according to the defined QEP. These intermediary results are then sent to the global mediator to be combined into a final answer for the end user (see Figure 13).

By the decomposition of the query processing and the selection of relevant data sources ac-

---

Table 4. User query and profile.

<table>
<thead>
<tr>
<th>User</th>
<th>Query</th>
<th>Profile</th>
</tr>
</thead>
</table>
6.1. Qualitative Validation

The query processing method proceeds in four steps: query rewriting, optimization, evaluation and execution (see Figure 8).

Let’s consider a query Q expressed in SPARQL over the MVPO given in Figure 4. The query Q searches the designation, the average margin and the sales of products sold to customers who live in Constantine (see Figure 9).

6.1.1. Query Rewriting

The query Q is firstly rewritten into a query tree describing the involved entities, properties and eventually conditions. Then, using the GMT containing the correspondences assertions presented in Table 4, the query tree is expanded to a combination of sub-queries, where the MVPO concepts are replaced by the VPOs correspondent concepts (see Figure 10). The property Average-margin searched by the user characterizes the class Product in the “Accounting VPO”, and the property sales characterizes the class Product in the “Commercial VPO”.

6.1.2. Optimization

In this step, the optimal query execution plan is generated using the cost model. In the considered example, the execution plan of Q contains two sub-queries SQ1 and SQ2 which can be executed in parallel on the “Accounting VPO” and the “Commercial VPO”. SQ1 extracts designation and average margin of products sold to customers from Constantine, and SQ2 extracts designation and sales of products sold to customers from Constantine. The generated sub-queries are represented in Figure 11, and the execution plan is the union of both sub-queries (SQ1 ∪ SQ2).

6.1.3. Evaluation

In this step, the relevant queries over the data sources are obtained. However, using the LMT, sub-queries sent to the target viewpoint mediators are rewritten in terms of the local ontologies of the same viewpoint. A set of optimal QEPs representing the local QEPs is generated. For example, Figure 12 presents the decomposition of SQ2 using LMT in Table 3 defined between the "Commercial VPO" and the local ontologies.

6.1.4. Execution and Recomposition

After the execution of the sub-queries over the data sources, results are sent back to the combination module at the viewpoint level. This one builds the answers received form data sources according to the defined QEPs. These intermediate results are then sent to the global mediator to be combined into a final answer for the end user (see Figure 13).

By the decomposition of the query processing and the selection of relevant data sources ac-

Table 4. User query and profile.

<table>
<thead>
<tr>
<th>User 1</th>
<th>User 2</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Profile</strong></td>
<td>(Product; 0.50, Supplier; 0.50, AccountingSupplier; 0.80, CommercialSupplier; 0.80, AccountingProduct; 0.80)</td>
</tr>
</tbody>
</table>
Let us consider a part of the MVPO where all the concepts have the same weight:

**MVPO:**

- **Commercial VP** (CommercialProduct; 0.50, CommercialSupplier; 0.50, CommercialCustomer; 0.50)
- **Accounting VP** (AccountingProduct; 0.50, AccountingSupplier; 0.50, AccountingCustomer; 0.50)
- **Manufacturing VP** (ManufacturingProduct; 0.50, ManufacturingSupplier; 0.50)

To read easily these statistics, we homogenize them in the same table (see Table 5). Absent concepts are added in some profiles, while affecting a weight of zero.

To measure the similarity between the various profiles, we use the inner product. Thus, we define a score of similarity to rank the VPOs from the most relevant to the less relevant according to the user profile. We assume that the VPO with the highest score is more relevant to represent the users’ preferences and choice.

**Definition 6.1. (Inner product):** The inner product is algebraically defined between two vectors A \([a_1, a_2, ..., a_n]\) and B \([b_1, b_2, ..., b_n]\) with the following formula:

\[
A \cdot B = \sum_{i=1}^{n} a_i b_i = a_1 b_1 + a_2 b_2 + ... + a_n b_n
\]

Let’s calculate, in Table 6, the similarity scores between the users and all the viewpoints vectors of the previous example.

We observe that the most relevant viewpoints for User1 are the "Accounting VP" and the "Commercial VP" and his query will be sent directly to the specific mediators. Lacking information about User2 viewpoints, his query will be sent to all viewpoint mediators. By specifying the viewpoints notion, the number of data sources is consequently limited and the quality of answers is better. Furthermore, the query processing time is reduced.

7. Conclusion

In this paper, we have presented a novel ontology-based integration approach using the paradigm of viewpoints. The main objective is to benefit from viewpoints to overcome the drawbacks of the "a priori" and the "a posteriori" semantic integration approaches. The proposed

<table>
<thead>
<tr>
<th>Table 5. Homogenizing profile values.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Product</td>
</tr>
<tr>
<td>Supplier</td>
</tr>
<tr>
<td>Accounting-Supplier</td>
</tr>
<tr>
<td>Commercial-Supplier</td>
</tr>
<tr>
<td>Manufacturing-Supplier</td>
</tr>
<tr>
<td>Accounting-Product</td>
</tr>
<tr>
<td>Commercial-Product</td>
</tr>
<tr>
<td>Manufacturing-Product</td>
</tr>
<tr>
<td>Commercial-Customer</td>
</tr>
<tr>
<td>Accounting-Customer</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Table 6. Similarity scores between users and viewpoints profiles.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Similarity score</td>
</tr>
<tr>
<td>-------------------</td>
</tr>
<tr>
<td>User 1</td>
</tr>
<tr>
<td>User 2</td>
</tr>
</tbody>
</table>
Let us consider a part of the MVPO where all the concepts have the same weight:

**MVPO**:

- Commercial VP (CommercialProduct ; 0.50, CommercialSupplier ; 0.50, CommercialClient ; 0.50)
- Accounting VP (AccountingProduct ; 0.50, AccountingSupplier ; 0.50, AccountingCustomer ; 0.50)
- Manufacturing VP (ManufacturingProduct ; 0.50, ManufacturingSupplier ; 0.50)

To read easily these statistics, we homogenized them in the same table (see Table 5). Absent concepts are added in some profiles, while affecting a weight of zero.

To measure the similarity between the various profiles, we use the inner product. Thus, we define a score of similarity to rank the VPOs from the most relevant to the less relevant according to the user profile. We assume that the VPO with the highest score is more relevant to the users’ preference and choice.

**Definition 6.1. (Inner product):** The inner product is algebraically defined between two vectors A \[a_1, a_2, ..., a_n\] and B \[b_1, b_2, ..., b_n\] with the following formula:

\[A \cdot B = \sum_{i=1}^{n} a_i b_i = a_1 b_1 + a_2 b_2 + \ldots + a_n b_n\]

Let’s calculate, in Table 6, the similarity scores between the users and all the viewpoints vectors of the previous example.

We observe that the most relevant viewpoints for User1 are the "Accounting VP" and the "Commercial VP" and his query will be sent directly to the specific mediators. Lacking information about User2 viewpoints, his query will be sent to all viewpoint mediators. By specifying the viewpoints notion, the number of data sources is consequently limited and the quality of answers is better. Furthermore, the query processing time is reduced.

### 7. Conclusion

In this paper, we have presented a novel ontology-based integration approach using the paradigm of viewpoints. The main objective is to benefit from viewpoints to overcome the drawbacks of the "a priori" and the "a posteriori" semantic integration approaches. The proposed

<table>
<thead>
<tr>
<th>Similarity score</th>
<th>Accounting VP</th>
<th>Commercial VP</th>
<th>Manufacturing VP</th>
</tr>
</thead>
<tbody>
<tr>
<td>User 1</td>
<td>0.8</td>
<td>0.8</td>
<td>0</td>
</tr>
<tr>
<td>User 2</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>
In our research, we adopt the model proposed in [21] for the representation of the MVPO. The MVP-OWL model contains an extension of the OWL language. In this model, new primitives are added to the OWL language to integrate the viewpoint notions in ontologies (see Table 7).

Table 7. MVP-OWL: new primitives' description.

<table>
<thead>
<tr>
<th>Class name</th>
<th>rdf:subClassOf</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>vp: onViewpoint</td>
<td>owl:Class</td>
<td>To define classes with viewpoints</td>
</tr>
<tr>
<td>vp:belongToDatePoint</td>
<td>owl:objectProperty</td>
<td>Define the individuals' viewpoint</td>
</tr>
<tr>
<td>vp:characterisedBy</td>
<td>owl:objectProperty</td>
<td>Define a viewpoint' properties and criterion</td>
</tr>
<tr>
<td>vp:extraClass</td>
<td>owl:Class</td>
<td>To define classes with viewpoints</td>
</tr>
</tbody>
</table>

Approach is based on two kinds of mediators, each governed by a kind of ontologies. First, the viewpoint mediators use viewpoint ontologies (VPO) to represent a specific viewpoint of data sources and fulfill the mediation between them. The second kind represented by the global mediator uses a multi-viewpoint ontology (MVPO) as a shared ontology to represent the different viewpoints that exist in the considered domain. Additionally, the integration process is completely automatic.

We currently intend to improve the query optimization process by using the notion of profiles at both levels of the integration system. Future works will especially be concerned with the development of a new data source presenting a new viewpoint that is not considered in the global ontology and with proving that the pile of mediators in the mediation architecture have no impact on the system performances.

Annex

In our research, we adopt the model proposed in [21] for the representation of the MVPO. The MVP-OWL model contains an extension of the OWL language. In this model, new primitives are added to the OWL language to integrate the viewpoint notions in ontologies (see Table 7).

References

approach is based on two kinds of mediators, each governed by a kind of ontologies. First, the viewpoint mediators use viewpoint ontologies (VPO) to represent a specific viewpoint of data sources and fulfill the mediation between them. The second kind represented by the global mediator uses a multi-viewpoint ontology (MVPO) as a shared ontology to represent the different viewpoints that exist in the considered domain. Additionally, the integration process is completely automatic.

We currently intend to improve the query optimization process by using the notion of profiles at both levels of the integration system. Future works will especially be concerned with the development of a new data source presenting a new viewpoint that is not considered in the global ontology and with proving that the pile of mediators in the mediation architecture have no impact on the system performances.

Annex
In our research, we adopt the model proposed in [21] for the representation of the MVPO. The MVP-OWL model contains an extension of the OWL language. In this model, new primitives are added to the OWL language to integrate the viewpoint notion in ontologies (see Table 7).

Table 7. MVP-OWL: new primitives’ description.

<table>
<thead>
<tr>
<th>Class name</th>
<th>rdf:subClassOf</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>vp:Viewpoint</td>
<td>owl:Class</td>
<td>To define classes of type: viewpoint</td>
</tr>
<tr>
<td>vp:ClassWithViewpoint</td>
<td>owl:Class</td>
<td>To define classes with viewpoints</td>
</tr>
<tr>
<td>Property name</td>
<td>Type</td>
<td>Description</td>
</tr>
<tr>
<td>--------------</td>
<td>----------</td>
<td>-------------</td>
</tr>
<tr>
<td>vp:belongsToViewpoint</td>
<td>owl:ObjectProperty</td>
<td>Define the individuals’ viewpoint</td>
</tr>
<tr>
<td>vp:characterisedBy</td>
<td>owl:ObjectProperty</td>
<td>Define a viewpoint’s properties and criterion</td>
</tr>
<tr>
<td>vp:onClass</td>
<td>owl:ObjectProperty</td>
<td>To define classes with viewpoint</td>
</tr>
<tr>
<td>vp:onViewpoint</td>
<td>owl:ObjectProperty</td>
<td>To define classes with viewpoints</td>
</tr>
</tbody>
</table>

References
[27] B. Bouchra et al., "Data Sources Integration Using Viewpoint-Based Approach", in 1st Int. Conf. on Intelligent Information Processing, Security and Advanced Communication, Batna, Algeria, pp. 25, 2015. http://dx.doi.org/10.1109/CIPI.2015.681639

Received: November 2015
Revised: December 2016
Accepted: December 2016
Contact addresses:
Bouchra Boulkroun
LIRE Laboratory
Software Technologies and Information Systems Department
Constantine 2 University Abdelhamid Mehri
Route d’Ain-El Bey, 25000
Constantine, Algeria
e-mail: bouchra.boulkroun@univ-constantine2.dz

Fouzia Benchikha
LIRE Laboratory
Software Technologies and Information Systems Department
Constantine 2 University Abdelhamid Mehri
Route d’Ain-El Bey, 25000
Constantine, Algeria
e-mail: f_benchikha@yahoo.fr

Chahinez Bachtarzi
LIRE Laboratory
Software Technologies and Information Systems Department
Constantine 2 University Abdelhamid Mehri
Route d’Ain-El Bey, 25000
Constantine, Algeria
e-mail: bachtarzi_chahinez@live.fr

Bouchra Boulkroun is pursuing her PhD in Computer Science at Constantine 2 University Abdelhamid Mehri, Algeria. Her research interests include distributed information systems, ontology design, and multi-viewpoint representations. She focuses her doctoral research on applying the viewpoint paradigm to integrate distributed ontology-based data sources using an original ontology-based integration approach.

Fouzia Benchikha is a Professor in Department of Software Technologies and Information Systems at Constantine 2 University Abdelhamid Mehri, Algeria. Her current research activities are conducted at the LIRE Laboratory from the same university. Her research interests are data modeling and integration, information systems interoperability, multi-viewpoint representations, and semantic web technology.

Chahinez Bachtarzi is a Doctor in Department of Software Technologies and Information Systems the Constantine 2 University Abdelhamid Mehri, Algeria. She received her PhD from Constantine 2 University Abdelhamid Mehri, Algeria in 2014. Her current research activities are conducted at the LIRE Laboratory from the same university. Her research interests are ontology design, database technology, and multi-viewpoint ontology.