

Foundational Ontologies for Semantic Integration in EAI: A Systematic Literature Review

Julio Cesar Nardi^{1,2}, Ricardo de Almeida Falbo², and João Paulo A. Almeida²

¹ Research Group in Applied Informatics, Informatics Department,

Federal Institute of Espírito Santo, Campus Colatina, Colatina, ES, Brazil

² Ontology & Conceptual Modeling Research Group (NEMO), Computer Science Department,

Federal University of Espírito Santo, Campus Goiabeiras, Vitória, ES, Brazil

julionardi@ifes.edu.br, {falbo,jpalmeida}@inf.ufes.br

Abstract. Despite (i) the recognized benefits of using ontologies in semantic EAI initiatives, (ii) the benefits of using foundational ontologies for promoting meaning negotiation and common understanding, and (iii) the importance of the semantic integration issue in EAI area, foundational ontologies have not yet become widely adopted in EAI initiatives for dealing with semantic conflicts. This has led us to investigate, through a systematic review of the literature, the adoption of foundational ontologies in EAI initiatives, with the purpose of understanding the current role of these ontologies in EAI and identifying gaps for future research, in which the potential benefits of such ontologies could be explored. We consider: (i) the role of foundational ontologies as part of the integration approach; (ii) the use of ontologies at development time and/or at run time; and (iii) the adoption of systematic approaches for semantic EAI.

Keywords: Enterprise Application Integration, Foundational Ontologies, Semantic Interoperability, Semantic Integration, Systematic Literature Review.

1 Introduction

The Enterprise Application Integration (EAI) research area regards the development and use of plans, methods, and tools to put together distinct information systems, by supporting their ability to exchange information and functionality to accomplish specific parts of a collaborative business process [1][2]. By dealing with distinct software applications, from both intra- and inter-organizational contexts, semantic conflicts arise. These conflicts are caused, among others, because the various heterogeneous applications do not share the same conceptualization [3].

A foundational ontology is a kind of (meta)ontology, independent of a particular problem or domain, that describes a set of real-world categories that can be used to talk about reality. It is constructed using the theories of Formal Ontology in philosophy [4][5]. The main purpose of a foundational ontology is to aid in negotiating meaning and facilitating common understanding [6]. Because of that, foundational ontologies have been acknowledged as an important means for dealing with semantic conflicts [5].

However, despite the recognized benefits of using ontologies in semantic EAI initiatives [3], the benefits of using foundational ontologies for promoting meaning negotiation and common understanding [6], and the importance of the semantic integration issue in EAI area [3], foundational ontologies have not yet become widely adopted in EAI initiatives for dealing with semantic conflicts. This has led us to investigate the adoption of foundational ontologies in EAI, aiming at understanding the current role of these ontologies in this area and identifying gaps for future research, in which the potential benefits of foundational ontologies could be explored.

In this paper we present a Systematic Literature Review (SLR) [7], analyzing EAI initiatives that adopt foundational ontologies to address semantic integration. We have formulated three research questions (see section 3), which aim to investigate three main aspects of the surveyed studies: (i) the role of foundational ontologies as part of the integration approach; (ii) the use of ontologies at development time and/or at run time; and (iii) the adoption of systematic approaches for these EAI initiatives.

This paper is organized as follows. Section 2 presents the theoretical background and clarifies some terminological aspects. Section 3 describes the research method adopted and the review protocol. Section 4 presents the selection process and a brief description of the selected studies. Section 5 presents a synthesis of the data collected from these studies in the light of the research questions. In Section 6, we discuss important points identified during data analysis, which may be useful for the research agenda in EAI area. Finally, in Section 7, we present some final considerations.

2 Background

Ontologies have been acknowledged as an important means to address semantics in EAI [3]. An ontology is a formal representation of a common conceptualization of a universe of discourse [4]. There are different classifications of ontologies. Guarino [4] classifies ontologies into: (i) foundational (or top-level) ontologies, which describe very general concepts independently of a particular problem or domain, such as object, event, action etc., (ii) domain ontologies, which describe the conceptualization related to a generic domain (for instance, law or biology), (iii) task ontologies, which describe the conceptualization related to a generic task (such as diagnosis or planning), and (iv) application ontologies that describe concepts dependent on a particular domain and a task. Scherp et al. [8] extend this classification by admitting the so-called “core ontologies” with a level of generality between that of foundational and domain ontologies. In this sense, core ontologies provide a precise definition of structural knowledge in a specific field that spans across different application domains.

Although these kinds of ontology are important for clarifying concepts used along this work, we are especially interested in foundational ontologies. Examples of foundational ontologies include DOLCE [9], SUMO [10], YAMATO [11], and UFO [5].

In an ontology-driven Information System (IS), an ontology can be used at development time or at run time [4]. At development time, the knowledge represented by a domain/task ontology can be reused to support conceptual analysis and assure the ontological adequacy of the IS. Further, a foundational ontology may be used at development time to support requirements engineers in conceptual modeling [4]. In this case, fundamental ontological distinctions embodied in the foundational ontology

are used to improve the quality of conceptual models [5]. When using ontologies at development time, we should focus on their representation adequacy, since these ontologies, said *reference ontologies*, are used in an off-line manner to assist humans in tasks of meaning negotiation and consensus establishment. On the other hand, once users have already agreed on a common conceptualization, specialized versions of the reference ontology can be created for run-time use. These versions are said *lightweight ontologies*, and sacrifice representation adequacy to guarantee desirable computational properties [5].

The principles that guide ontology-driven IS development can also be applied in ontology-based EAI initiatives. In EAI initiatives, ontologies may be used at integration development time, in the sense that the ontology can be used by humans to negotiate meaning during the integration process, more specifically during an activity of conceptual mapping between applications. So, the two scenarios described by Guarino [4] hold: domain and task ontologies may be used by humans to negotiate meaning when mapping concepts and relations between different applications, as well as foundational ontologies may be used to capture fundamental ontological distinctions in order to categorize applications' concepts when mapping them. On the other hand, ontologies may also be used at integration run time, in the sense that an ontology may be implemented as a lightweight ontology, in a machine readable language, and used to link the applications at run time.

Finally, in the context of this review, it is necessary to clarify some aspects of terminology and scope. Firstly, we are interested in both intra- and inter-enterprise application integration. So, we use the term "enterprise application integration" to cover both categories, since most methods and patterns apply to both of them [2]. Secondly, although the terms "integration" and "interoperability" have been used to refer to different or interrelated concepts, they have also been used in an indistinct way. In this paper, we use the term "integration" in a broad sense, encompassing all which is called integration and interoperability. In order to cover a wide range of intended senses, we consider both "application integration" as well as "application interoperability" in the search string presented in Section 3.

3 Research Method and Review Protocol

The research method for this SLR was defined according to [7]. A systematic literature review is a form of secondary study that uses a well-defined method to identify, analyze and interpret the available evidences in a way that is unbiased and (to a degree) repeatable. A secondary study is a study that reviews primary studies related to specific research questions with the aim of integrating/synthesizing the evidences related to these research questions [7].

A SLR involves three phases [7]: Planning, Conducting and Reporting the review. Planning involves the pre-review activities, which the most important one is defining the review protocol. Conducting the review is concerned with searching and studies selection, and data extraction and synthesis. Reporting is the final phase and involves writing up the results and circulating them to potentially interested parties.

The main parts of the review protocol used in this work are described as follows.

Research Questions. This SLR aims to answer the following research questions:

RQ1. How have foundational ontologies been used as part of EAI approaches?

RQ2. Do the studies use the ontologies at development time, at run time or both?

RQ3. Do the studies follow a systematic approach for performing the integration project? (Do they adopt or propose a method or a process model defining activities, inputs, outputs, guidelines, etc.?)

Question RQ1 aims to investigate if the ontological distinctions of foundational ontologies have been used directly to integrate applications or indirectly for grounding other lower-level ontologies that are, then, used to integrate the application. From that, question RQ2 aims to look if foundational ontologies have been applied at development time and/or at run time; Finally, question RQ3 refers to if the studies that use foundational ontologies present systematic integration approaches for addressing semantic EAI.

Search Strategy. The search strategy was split in two: EAI-FO and FO-EAI strategies. Both focus on finding studies that use foundational ontologies in order to address semantic enterprise application integration. However, the EAI-FO strategy starts looking for semantic EAI initiatives and then focusing on those that use foundational ontologies, whereas the FO-EAI strategy starts looking for the use of foundational ontologies to solve integration problems and then focusing on those that address EAI. We adopted this dual search strategy because, depending on the focus given by the authors, it is difficult to establish a single set of terms that is able to capture the full scope of our review. So, the EAI-FO and FO-EAI strategies are complementary and aim to offer a wider review strategy. As a result, we define two search strings (that were applied to title, abstract and keywords of each study), as follows:

EAI-FO search string

("application integration" OR "application interoperability" OR "enterprise system integration" OR "enterprise system interoperability" OR "integration of information system" OR "interoperability of information system" OR "integration of application" OR "interoperability of application" OR "interoperability of enterprise application" OR "interoperability of enterprise system" OR "integration of enterprise application" OR "integration of enterprise system" OR "interoperability of business application" OR "interoperability of business system" OR "integration of business application" OR "integration of business system" OR "integration of heterogeneous system" OR "integration of heterogeneous application" OR "interoperability of heterogeneous system" OR "interoperability of heterogeneous application" OR "interoperability of information system" OR "integrated application" OR "interoperable application" OR "integrated enterprise system" OR "interoperable enterprise system" OR "information system integration" OR "information system interoperability" OR "enterprise system integration" OR "enterprise system interoperability" OR "business system integration" OR "business system interoperability") AND (semantic OR semantics OR semantically)

FO-EAI search string

("foundational ontology" OR "top-level ontology" OR "top level ontology" OR "upper-level ontology" OR "upper level ontology" OR "upper ontology") AND ("integration" OR "interoperability")

Selection Criteria. The selection criteria are organized in inclusion and exclusion criteria. There is a single inclusion criterion: (IC1) The study presents a semantic EAI

initiative that uses foundational ontology. There are four exclusion criteria: (EC1) The study is not written in English; (EC2) The study is an older version of another study already considered; (EC3) The study is not a primary study (including editorials and summaries of keynotes, workshops or tutorials); and (EC4) The study is just published as an abstract. Moreover, we considered the studies published until December 31st 2011, since we performed the data collection in the beginning of 2012.

Sources. The following sources were searched: IEEEExplore, ACM Digital Library, Springer Link, Thomson Reuters Web of Knowledge, Scopus, Science Direct, and Compendex.

Assessment. In order to avoid biases along the review process, we performed periodic meetings. The review process was conducted by one of the authors, and one of the other two performed activities of verification and validation over samples that represent about 35% of the studies. Possible biases were discussed in these meetings.

4 Selected Studies

An overview of the selection process is presented in Fig. 1. Using the EAI-FO strategy, 702 records were retrieved from the considered sources. After that, we eliminated duplicate records and applied the selection criteria, resulting in 5 studies. Using the FO-EAI strategy, 227 records were retrieved and 3 studies remained after eliminating duplicate records and applying the selection criteria. In the intersection of the two result sets, there was 1 study. So, in the end, the following 7 primary studies remained: [2][8][12][13][14][15][16]. Next, we present a brief description of each one.

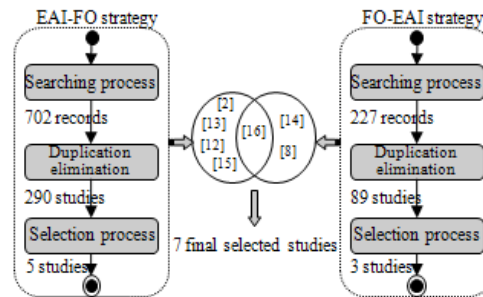


Fig. 1. The searching and selection processes

In [13], a use case scenario for integrating heterogeneous project management applications aims to test the ability of Conceptual Spaces (CSpaces) to handle integration problems. A CSpace is a knowledge container defined as a set of tuples, with a well-defined structure. In the use case, the PSL Ontology (Process Specification Language Ontology) [17] is stored in a CSpace that is shared by several applications. Also, each application has an individual CSpace that stores the extended version of the PSL Ontology together with application data.

The studies [2] and [12] were conducted in the context of the FUSION Project, and thus they share common principles. In [12], Bouras et al. present the ENTERprise

Integration Ontology (ENIO), which is based on the alignment of DOLCE [9] and SUMO [10], with extensions to cover some dimensions of the EAI domain, called facets. There are three facets: data facet is taken as basis in transformations of services' inputs and outputs; functional facet defines the services' capabilities and provides classes for annotating services' operators with functional semantics; and process facet aims at providing a means for defining collaborative business process templates. In [2], Alazeib et al. describe two approaches for semantically-assisted design of collaborative business process (manual and semi-automatic). Both approaches rely on ENIO for solving structural heterogeneities of different business applications.

In [16], Treiblmayr et al. present an initiative for integrating a Geographic Information System (GIS) with a non-GIS application. These applications are integrated in terms of a service-oriented architecture, and the focus is on semantic interoperability. In order to map the conceptual schemes of the services provided by the different systems, ontological distinctions are analyzed at the light of DOLCE [9], or based on a semantic referencing system, also anchored in DOLCE.

In [14], Masuya et al. describe the development of the RIKEN integrated database of mammals. This database integrates heterogeneous mammal-related data stored in multiple individual databases of related research projects. To ensure the consistent classification of the content, they used a top-middle level ontology, YAMATO-GXO Lite. It is the lightened version of the middle-level ontology Genetics Ontology (GXO) to bridge the experimental genetics domain and the top-level ontology YAMATO [11].

Scherp et al. [8] discuss the use of three core ontologies in an integration scenario in the emergency response domain. The Event-Model-F is a core ontology representing different aspects of events. The Core Ontology on Multimedia (COMM) is designed for describing arbitrary digital media data. The Cross-Context Semantic Information Management Ontology (X-COSIMO) is designed for modeling semantic information management and communication. These three core ontologies are based on DOLCE, and follow a pattern-oriented design approach.

Finally, in [15], Paulheim and Probst present an ontology-based framework for integrating applications in the user interface level. For this, they use two separate domain-level ontologies: an ontology of the user interfaces and interactions domain, and a real world domain ontology. The latter is not an integral part of the framework and can be exchanged when integrating an application of a different domain. Any way, Paulheim and Probst argue that a "suitable real world domain ontology" is required, and in the scenario used to illustrate the proposed approach, the domain ontology is grounded in DOLCE. However, the ontology of the user interfaces and interactions domain is not grounded in any foundational ontology.

5 Data Synthesis

After selecting the primary studies, we analyzed each one in order to answer the research questions. Next, we perform the data synthesis regarding these questions.

RQ1. How have foundational ontologies been used as part of EAI approaches?

Analyzing the selected studies, we identified three different ways of using foundational ontologies in EAI initiatives: direct use of a foundational ontology (direct

approach), indirect use of a foundational ontology (indirect approach), and hybrid (both direct and indirect) use of a foundational ontology (hybrid approach).

In the *direct approach*, the ontological distinctions provided by foundational ontologies are directly used to solve semantic conflicts between the applications' concepts. This is the case of [13], where Martín-Recuerda uses the PSL Ontology as an interlingua for application integration.

In the *indirect approach*, a foundational ontology is used as basis for developing a core ontology or a domain ontology, which is, in turn, effectively used in the EAI initiative. This is the most common approach, and it is the case of [2][12][14][8][15], which are described below.

One of the indirect approaches is that employed in the context of the FUSION Project [2][12], in which the alignment of two foundational ontologies is part of the ENterprise Application Integration Ontology (ENIO). ENIO aims to provide an explicit specification of the EAI domain. It comprises, among others, a foundational ontology that is based on the alignment of DOLCE and SUMO. This foundational ontology is used for grounding an upper-level EAI ontology and the facets that cover three dimensions of the EAI domain: data, service and process. ENIO's facets present domain independent elements, which are based on widely adopted standards (and thus can be viewed as core ontologies), and domain dependent elements, which can be viewed as domain ontologies. The elements of each facet are used by EAI initiatives.

Another indirect approach is presented in [14]. Masuya et al. use YAMATO-GXO Lite, which aims to provide an explicit specification of the field of experimental genetics. This specification is used to integrate heterogeneous databases. As a result, 41 classes conveying the key information from each database were classified under 15 upper classes, which were used to classify the content of the integrated databases.

Scherp et al. [8] use and combine three core ontologies (Event-Model-F, COMM and X-COSIMO) to model the knowledge in an emergency response scenario. These core ontologies are aligned with DOLCE+DnS Ultralight2 (DUL), a lightweight version of DOLCE. Considering that the concepts of the core ontologies are used to integrate heterogeneous systems, this study is also classified as using indirect approach.

Finally, to facilitate ontology-based application integration on the user interface level, Paulheim and Probst [15] follow a plugin-based approach, in which each application to be integrated is encapsulated in a plugin. Application developers should formally describe their applications in application ontologies, which have to be based on domain-level ontologies as common ground. Also, the authors present an integration scenario in which a domain ontology is grounded by a foundational one (DOLCE), characterizing the indirect approach.

In the *hybrid approach*, a foundational ontology is used both directly and indirectly to aid solving semantic integration conflicts. This is the case of [16]. Treiblmayr et al. adopt an approach for concept mapping where concepts of each application are described based on both: (i) primitive concepts from a common shared vocabulary given by DOLCE (direct approach); and (ii) what they call a "semantic reference system" (indirect approach). In this context, the semantic reference system can be seen as a core ontology of the geographic information field, and its concepts are also grounded in DOLCE. Thus, the applications' concepts are anchored both in concepts from DOLCE and from the semantic reference system, characterizing a hybrid approach.

RQ2. Do the studies use ontologies at development time, at run time or both?

All the analyzed studies, in some level and stage of the integration process, consider a conceptual perspective when activities related to analysis and conceptual mapping are performed. Thus, all of them are classified as using ontologies at development time, mainly by integration engineers to negotiate the meaning of concepts and relations involved in the applications being integrated. Moreover, most of them also use ontologies at run time, implementing lightweight versions of the ontologies in a variety of machine-readable languages, such as KIF, RDF, OWL and F-Logics.

In [13], mapping and transformation rules between the applications' concepts and PSL are defined at development time and used to link CSpaces. These mapping and transformation rules are described using Distributed First Order Logic, and KIF is the underlying language used for the PSL Ontology at run time.

In the context of the FUSION Project [2][12], concepts from the data and functional facets are used to annotate web services at development time. Alazeib et al. [2] propose two approaches for semantically-assisted design of collaborative business processes. The manual approach uses previously annotated business process models that define control and data flow; the semi-automatic approach deals with business processes whose control and data flow are defined by planning techniques using behavioral descriptions of the web services and composition goals. In [12], the manual approach is applied for demonstrating the usage of ENIO in facilitating the smooth integration of business processes in a real-world B2B scenario. ENIO is also used at run time, being implemented in OWL-DL. "Up-cast" (from service inputs/outputs to ontology concepts) and "down-cast" (from ontology concepts to service inputs/outputs) transformations are used at run time to perform dynamic data mediation.

Masuya et al. [14] examined, at development time, the schemas of the databases to be integrated and, then classified their contents and semantics under YAMATO-GXO Lite, in a job that they consider to be comparable to database annotation. YAMATO-GXO Lite is implemented in RDF and OWL, and the database annotations are made in these languages to support queries at run time.

In [8], Scherp et al. present several object diagrams to exemplify how the structured knowledge in a concrete scenario of a flooding event can be modeled at development time. The ontology design patterns of the core ontologies are combined and used to model this scenario. The core ontologies are implemented in OWL to enable application communication at run time, but this use is not discussed in the paper.

Paulheim and Probst [15] advocate that application developers should describe their applications in application ontologies, which have to be based on domain ontologies. This is done at development time. In the example presented in [15], they annotate class diagrams with concepts and relations from a domain ontology. Moreover, the ontologies are implemented in F-Logic and the proposed framework uses run-time reasoning to determine possible cross application interactions.

Finally, Treiblmayr et al. [16] present the use of ontologies only at development time. Application concepts are grounded in concepts from DOLCE and from the semantic reference system at development time.

RQ3. Do the studies follow a systematic approach for performing the integration project? (Do they adopt or propose a method or a process model defining activities, inputs, outputs, guidelines, etc.?)

Most of the studied initiatives is developed in an ad-hoc manner [13][14][15][8][16], i.e. they do not follow a systematic approach for integrating applications. The exceptions are the works done in the context of the FUSION Project [2][12], mainly [2].

In [13], Martín-Recuerda only revisits the solution for application integration proposed by Cheng and Law [18] to test the suitability of CSpaces to deal with integration scenarios. Treiblmayr et al. [16] do not follow or define an integration method; concerning methodological aspects, they discuss aspects related to conceptual mapping (using a foundational and a core ontology), and how to semantically describe services. Masuya et al. [14] describe how they conducted the integration initiative, discussing the performed steps, without the pretension of proposing that these steps could be generalized to give rise to a method. The focus of [8] is on designing core ontologies. Concerning the integration initiative, this work only exemplifies how to model the knowledge exchanged by the systems. Finally, Paulheim and Probst [15] propose a framework for application integration on the user interface level by encapsulating the applications in plugins, and pinpoint three steps that are necessary for integrating an application into their framework. In the context of the FUSION Project, Alazeib et al. [2] present two approaches (one manual, another semi-automatic) for semantically-assisted design of collaborative business processes in EAI scenarios. Both can be considered as systematic approaches. Table 1 summarizes this section.

Table 1. Summary of the synthesized data, organized by research question

Study	Approach Employed (RQ1)	Use of Ontology (development/run time) (RQ2)	Integration Approach (RQ3)	Foundational Ontology Adopted
[2]	Indirect Approach	Development Time and Run Time	Systematic	DOLCE & SUMO
[8]	Indirect Approach	Development Time and Run Time	Ad-hoc	DOLCE
[12]	Indirect Approach	Development Time and Run Time	Systematic	DOLCE & SUMO
[13]	Direct Approach	Development Time and Run Time	Ad-hoc	PSL Ontology
[14]	Indirect Approach	Development Time and Run Time	Ad-hoc	YAMATO
[15]	Indirect Approach	Development Time and Run Time	Ad-hoc	DOLCE
[16]	Hybrid Approach	Development Time	Ad-hoc	DOLCE

6 Discussion

In this section, we discuss some relevant points that have arisen from data synthesis and discuss limitations of this review.

The Importance of Foundational Ontologies in EAI. Taking into account the small number of selected studies, we can conclude that the use of foundational ontologies in EAI initiatives is relatively underexplored. Nevertheless, foundational ontologies

have gained attention in the latest years, as corroborated by the publishing year of the studies: 2006 ([13]), 2007 ([2][12]), 2010 ([15]), and 2011 ([14][8][16]).

Foundational ontologies are being used in EAI initiatives, especially by offering fundamental ontological distinctions for solving semantic conflicts. We have observed that most of the selected studies use lightweight versions of domain ontologies, and that these versions are grounded directly or indirectly in foundational ontologies. This indicates that these two kinds of ontologies may be used together in EAI.

Regarding the approaches adopted (direct, indirect, and hybrid approaches), we think that the hybrid approach seems to be the most promising, since it puts together the characteristics of the other two approaches, being, therefore, more embracing. In situations where we do not have domain or core ontologies available to address all semantic heterogeneities in an EAI initiative (indirect approach), applying the ontological distinctions directly to analyze the applications' models may be useful. In sum, we claim that common understanding can be achieved by using domain, task or core ontologies, but also by directly applying foundational ontology. Also, we highlight DOLCE as the predominant ontology, being used by different research groups.

Addressing EAI Initiatives at Different Abstraction Levels. All studies included in the review use ontologies at development time, and most of them also use ontologies at run time (except [16]). At development time, ontologies have been used, among others, for negotiating meaning, for defining mappings and transformation rules, for making annotations, as well as for grounding the concepts adopted by the applications being integrated. At run time, most of the studies implement lightweight versions of the ontologies in a variety of machine-readable languages, such as KIF, RDF, OWL and F-Logics. These versions are applied for query processing, concept matching, and message exchange, among other uses.

The use of ontologies at development time indicates a concern related to a conceptual abstraction level in EAI projects, when meaning is negotiated and a common understanding is pursued. The conceptual level has different concerns if compared to a technological one. However, a clear distinction between these two abstraction levels is not present in the analyzed studies. Analogously to other related research areas, such as Software Engineering, a clear separation of concerns related to conceptual and technological abstraction levels could be further explored in EAI.

Moreover, none of the studies have adopted representation languages that incorporate fundamental ontological distinctions in their constructs. In fact, these studies use ontological distinctions given by the foundational ontology, but these distinctions are not part of the language used for building their models. In our view, aligned to what is advocated in [5], the use of well-founded languages is an important issue for adequately addressing the conceptual abstraction level, at the EAI development time.

The Use of Systematic Approaches in EAI Initiatives. Most of the analyzed studies follow an ad-hoc approach to integration (except [2][12]). In our view, there is a gap regarding methodological issues, mainly if we consider the use of foundational ontologies as part of the integration approach. The gaps we have identified suggest that we need to advance towards the establishment of more prescriptive methods and processes. These should clarify the various integration activities, associating them to goals, inputs, outputs and various quality requirements. Systematic approaches can act as a glue to put together, in an organized way, the aspects raised in the previous

discussion points, in particular structuring the integration process into different levels of abstractions and defining guidelines on how to perform the various integration activities under the guidance of ontological distinctions. We believe this is essential for establishing an engineering approach for ontology-based EAI.

Limitations of This Review. Due to the fact that the study selection and data extraction stages were performed by just one of the authors, some subjectivity may have been inserted. To reduce this subjectivity, a second author was responsible for defining a random sample (including about 35% of the studies) and performing the same stages. The results of each reviewer were then compared in order to detect possible bias. Moreover, terminological problems in the search strings may have led to missing some primary studies. In order to minimize these problems, we performed previous simulations in the selected databases. We decide not to search any specific conference proceedings, journals, or the grey literature (technical reports and works in progress). Thus, we have just worked with studies indexed by the selected electronic databases. The exclusion of these other sources makes the review more repeatable, but possibly some valuable studies may have been left out of our analysis.

7 Final Considerations

This work presented a systematic literature review that aims to analyze the adoption of foundational ontologies in EAI initiatives for facing semantic integration problems and, from that, to identify new paths for future research. After searching and selection, seven studies remained, which were analyzed in order to address three research questions. During data discussion and synthesis, some relevant issues were detected and discussed: (i) despite still being underexplored, in the latest years foundational ontologies have gained attention in EAI initiatives; (ii) the need for conducting EAI initiatives in different levels of abstraction (conceptual and technological) as well as the need to adopt more appropriated representation languages, which consider distinct abstraction levels and ontological grounding; and (iii) the need for systematic approaches in EAI to put together all the aspects aforementioned.

Acknowledgments. This research is funded by the Brazilian Research Funding Agencies CAPES, FAPES/CNPq (PRONEX Grant 52272362/11) and CNPq (Grants 483383/2010-4 and 310634/2011-3).

References

1. Lee, J., Siau, K., Hong, S.: Enterprise Integration with ERP and EAI. *Communications of the ACM* 46(2), 54–56 (2003)
2. Alazeib, A., Balogh, A., Bauer, M., Bouras, A., Friesen, A., Gouvas, P., Mentzas, G., Pace, A.: Towards semantically-assisted design of collaborative business processes in EAI scenarios. In: 5th IEEE International Conference on Industrial Informatics, Vienna, Austria, pp. 779–784 (2007)
3. Izza, S.: Integration of industrial information systems: from syntactic to semantic integration approaches. *Enterprise Information Systems* 3, 1–57 (2009)

4. Guarino, N.: Formal Ontology and Information Systems. Formal Ontology and Information Systems 3 (1998)
5. Guizzardi, G.: On Ontology, ontologies, Conceptualizations, Modeling Languages, and (Meta)Models. In: Vasilecas, O., Edler, J., Caplinskas, A. (eds.) *Frontiers in Artificial Intelligence and Applications, Databases and Information Systems IV*, pp. 18–39. IOS Press, Amsterdam (2007)
6. Gangemi, A., Guarino, N., Masolo, C., Oltramari, A., Schneider, L.: Sweetening ontologies with DOLCE. In: Gómez-Pérez, A., Benjamins, V.R. (eds.) *EKA 2002. LNCS (LNAI)*, vol. 2473, pp. 166–181. Springer, Heidelberg (2002)
7. Kitchenham, B., Charters, S.: Guidelines for performing Systematic Literature Reviews in Software Engineering (Version 2.3) - EBSE Technical Report. EBSE-2007-01 (2007)
8. Scherp, A., Saathoff, C., Franz, T., Staab, S.: Designing core ontologies. *Applied Ontology* 6, 177–221 (2011)
9. Masolo, C., Borgo, S., Gangemi, A., Guarino, N., Oltramari, A.: *WonderWeb Deliverable D18. Ontology Library*, Padova, Italy (2003)
10. SUMO (Suggested Upper Merged Ontology), <http://www.ontologyportal.org/>
11. Mizoguchi, R.: Yet Another Top-level Ontology: YATO. In: *Proceedings of the Second Interdisciplinary Ontology Meeting*, Tokyo, Japan, pp. 91–101 (2009)
12. Bouras, A., Gouvas, P., Kourtesis, D., Mentzas, G.: Semantic integration of business applications across collaborative value networks. *International Federation for Information Processing (IFIP)* 243, 539–546 (2007)
13. Martín-Recuerda, F.: Application Integration Using Conceptual Spaces (CSpaces). In: Mizoguchi, R., Shi, Z.-Z., Giunchiglia, F. (eds.) *ASWC 2006. LNCS*, vol. 4185, pp. 234–248. Springer, Heidelberg (2006)
14. Masuya, H., Makita, Y., Kobayashi, N., Ai, E.: The RIKEN integrated database of mammals. *Nucleic Acids Research* 39, D861–D870 (2011)
15. Paulheim, H., Probst, F.: Application integration on the user interface level: An ontology-based approach. *Data & Knowledge Engineering* 69, 1103–1116 (2010)
16. Treiblmayr, M., Scheider, S., Krüger, A., Von der Linden, M.: Integrating GI with non-GI services – showcasing interoperability in a heterogeneous service-oriented architecture. *GeoInformatica* 16, 207–220 (2011)
17. Grüninger, M.: Ontology of the Process Specification Language. In: Staab, S., Studer, R. (eds.) *Handbook on Ontologies*, pp. 575–592. Springer, Berlin (2004)
18. Cheng, J., Law, K.H.: Using Process Specification Language for Project Information Exchange. In: *Proceedings of the 3rd International Conference on Concurrent Engineering in Construction*, Berkeley, USA, pp. 63–74 (2002)