Data & Knowledge Engineering 68 (2009) 905–925

Contents lists available at ScienceDirect

Data & Knowledge Engineering

journal homepage: www.elsevier.com/locate/datak

Reusing ontologies on the Semantic Web: A feasibility study

Elena Simperl

STI Innsbruck, University of Innsbruck, Technikerstr. 21 a, 6020 Innsbruck, Austria

ARTICLE INFO

Article history:
Received 23 October 2007
Received in revised form 23 January 2009
Accepted 9 February 2009
Available online 21 February 2009

Keywords:
Ontology reuse
Methodology
Requirements
Ontology engineering
Semantic Web

ABSTRACT

Technologies for the efficient and effective reuse of ontological knowledge are one of the key success factors for the Semantic Web. Putting aside matters of cost or quality, being reusable is an intrinsic property of ontologies, originally conceived of as a means to enable and enhance the interoperability between computing applications. This article gives an account, based on empirical evidence and real-world findings, of the methodologies, methods and tools currently used to perform ontology-reuse processes. We study the most prominent case studies on ontology reuse, published in the knowledge-/ontology-engineering literature from the early nineties. This overview is complemented by two self-conducted case studies in the areas of eHealth and eRecruitment in which we developed Semantic Web ontologies for different scopes and purposes by resorting to existing ontological knowledge on the Web. Based on the analysis of the case studies, we are able to identify a series of research and development challenges which should be addressed to ensure reuse becomes a feasible alternative to other ontology-engineering strategies such as development from scratch. In particular, we emphasize the need for a context- and task-sensitive treatment of ontologies, both from an engineering and a usage perspective, and identify the typical phases of reuse processes which could profit considerably from such an approach. Further on, we argue for the need for ontology-reuse methodologies which optimally exploit human and computational intelligence to effectively operationalize reuse processes.

1. Introduction

The term "*ontology*" was introduced to computer science as a means of formalizing the kinds of things that can be talked about in a system or a context. With a long-standing tradition in philosophy, where "*Ontology*" denotes "*the study of being or existence*" [12,42], ontologies provide knowledge engineering and artificial intelligence support for modeling some domain of the world in terms of labeled concepts, attributes and relationships, usually classified in specialization/generalization hierarchies.[1] With applications in fields such as knowledge management, information retrieval, natural language processing, eCommerce, information integration or the emerging Semantic Web, ontologies are part of a new approach to building intelligent information systems [31]: they are intended to provide knowledge engineers with reusable pieces of declarative knowledge, which can be – together with problem-solving methods and reasoning services – assembled into high-quality systems in an economical fashion [52]. According to this idea, ontologies are understood as means to *share and reuse* knowledge. They are at the core of a new strategy for the development of knowledge-based systems, in which application or domain knowledge

E-mail addresses: elena.simperl@sti2.at, elena.simperl@deri.at

[1] Guarino and Giaretta propose the of use different notations for the philosophical and the knowledge engineering views regarding ontologies [40]. According to this distinction, the philosophical sense is denoted by the capitalized word "*Ontology*" while the second one is written without capitals and may be used in plural form.

0169-023X/$ - see front matter © 2009 Elsevier B.V. All rights reserved.
doi:10.1016/j.datak.2009.02.002

is strictly separated from software implementations and can thus be efficiently reused across heterogeneous software platforms [39].

The emergence of the Semantic Web has marked an important stage in the evolution of ontologies. Initially introduced by Berners-Lee [5], the originator of the World Wide Web, the idea of providing the current Web with a computer-processable knowledge infrastructure in addition to its actual, semi-formal and human-understandable content foresees the usage of knowledge components which can be easily integrated into and exchanged among arbitrary software environments in an operationalized manner. In this context the knowledge components, that is, the ontologies, are formalized using Web-suitable, semantically unambiguous representation languages; are pervasively accessible; and (at least theoretically) are shared and reused across the Web.

Paraphrasing the general understanding of reuse in related engineering disciplines (cf., for instance, [33]), *ontology reuse* can be defined as *the process in which existing ontological knowledge is used as input to generate new ontologies*. The ability to efficiently and effectively perform reuse is commonly acknowledged to play a crucial role in the large-scale dissemination of ontologies and ontology-driven technologies, thus being a prerequisite for the mainstream uptake of the Semantic Web. The sharing and reuse of ontologies increases the quality of the applications using them, as these applications become interoperable and are provided with a deeper, machine-processable and commonly agreed-upon understanding of the underlying domain of interest. Secondly, analogously to other engineering disciplines, reuse, if performed in an efficient way, reduces the costs related to ontology development because it avoids the reimplementation of ontological components, which are already available on the Web and can be directly – or after some additional customization – integrated into a target ontology. Furthermore, it potentially improves the quality of the reused ontologies, as these are continuously revised and evaluated by various parties through reuse.

This article surveys case studies in ontology reuse published in the knowledge-/ontology-engineering literature from the early nineties. This survey is complemented by two self-conducted case studies in the areas of eHealth and eRecruitment in which we developed Semantic Web ontologies for different scopes and purposes by resorting to existing ontological knowledge on the Web. Our goal is not to provide a state-of-the-art overview of the technologies and tools which have been developed in the last decade to solve specific ontology-reuse issues, but rather to give an account, based on empirical evidence and real-world findings, of the way ontology-reuse processes are typically being carried out. Based on the analysis of the case studies, we are able to identify a series of research and development challenges which should be addressed to ensure that reuse becomes a feasible alternative to other ontology-engineering strategies such as development from scratch. In particular, we emphasize the need for a context- and task-sensitive treatment of ontologies, both from an engineering and a usage perspective, and identify the typical phases of reuse processes which could considerably profit from such an approach. Further on, we argue for the need for ontology-reuse methodologies which optimally exploit human and computational intelligence to effectively operationalize reuse processes.

2. Ontology reuse to date

Ontologies are expected to play a significant role in various application domains on the emerging Semantic Web [31,70]. Confirming these expectations, to date we have recorded an increasing number of industrial project initiatives choosing to formalize application knowledge using ontologies and Semantic Web representation languages RDFS [8], OWL [57], or WSML [9]. The emerging ontologies, however, seldom reflect a consensual or at least application-independent view of the modeled domain. Even when using Semantic Web representation languages, these ontologies are, in the sense of knowledge management, simple means of representing knowledge, without any claim of being *formal*, *application-independent* or the result of a *common agreement* within a community of practice. Paraphrasing the well-known definition from Gruber [38], they might indeed be *specifications of conceptualizations*, but they are rarely built to be *shared* or *reused*. The current state is confirmed by existing statistical studies on the size of the Semantic Web. For instance, according to the Semantic Web search engine Swoogle, the number of ontologies publicly available is estimated at approximately 10,000.[2] Some of them, such as FOAF, are already used widely – a high percentage of the RDF data on the Web is FOAF data [23]. Others, particularly general ontologies such as DOLCE[3] or SUMO [58], are meanwhile used in different settings as the upper-level grounding of various domain ontologies. The remaining, relatively large number of qualitatively heterogeneous ontologies, modeling the same or related domains of interests, are hardly being used beyond the boundaries of their originating context, though they are ubiquitously accessible. Their limited impact seriously impedes the uptake on semantic technologies in industrial settings. This situation is the result of a series of factors, some specific to the Semantic Web context, others related to more general knowledge- representation and management problems.

A common precondition for the reusability of an ontology is that it should be conceived of and developed independently from its usage context [39]. Consequently, reusable ontologies tend to be overgeneralized and to omit relevant domain knowledge, thus requiring considerable modifications before they can be reutilized for specific purposes. On the other hand, the more committed an ontology is to a specific domain and task, the less its terminological elements can be generalized and reused beyond this scope [1,16,44]. A reusable ontology ideally achieves a balance between specification and

[2] http://swoogle.umbc.edu/.
[3] http://www.loa-cnr.it/DOLCE.html.

overgeneralization without compromising its cross-application usability. Furthermore, the difficulties associated with the realization of a *shared* ontological commitment are inherently related to the fact that the domain experts involved in the conceptualization process possess, even when they belong to the same community of practice, different, personal or organization-specific views about the domain to be modeled, or are not necessarily willing to exchange and communicate the domain knowledge across the engineering team [7]. A third factor contributing to the described state of affairs is related to the still young nature of the Semantic Web field. Even if knowledge-representation structures have been around in computer science for many decades, ontologies, and in particular *Semantic Web ontologies*, emerged so far prevalently in stand-alone systems – targeted at closed user communities – or in proof-of-concept applications. In addition, adequate methodological and technological support for ontology-reuse processes, as well as for ontology engineering in general, is still under development. A comprehensive analysis of current ontology-usage bottlenecks is given in [43].

Ontology discovery is facilitated by the use of URIs and Web-accessible ontological sources, which are core principles of the Semantic Web. Semantic Web search engines such as Swoogle, Watson [21] or OntoSearch [79] and ontology repositories such as the DAML ontology library, the Protégé OWL library, the SchemaWeb directory,[4] or the FIPA ontology service [73] are equally important [24]. As a second step, ontology designers and users need means to *understand and evaluate existing ontologies*, which, as previously mentioned, are based neither on application-independent nor on community-agreed conceptualizations. These issues are vital, especially for large ontologies describing knowledge-intensive domains such as medicine, biology, or legislation. In such cases the evaluation cannot be performed completely manually due to the complexity and size of the modeled field. At the same time, the usage of existing sources in these domains is inevitable because of the high costs associated with a new implementation. Initial ideas on how to assess the usability of an ontology in a target application scenario have emerged in a number of research projects [48,60,13]. A third step towards ontology reuse is related to the *manipulation* of existing ontologies in the new setting. Once ontology designers have determined the relevance of the candidate source ontologies with respect to a set of requirements, those selected for reuse might require some form of customization in order to fit into the final application ontology. This necessitates not only methods to match, align and merge overlapping ontologies, but also tools to extract consistent and complete fragments from ontologies which are only partially relevant to the application setting. The former is one the most mature areas of research in the Semantic Web community (cf. [27,29,35,49–51,53,63,72] and many more). Ontology modularization is a comparatively recent research topic, and preliminary results are presented, for instance, in [22,30,68].

Reusing Semantic Web ontologies requires coping with many of the problems encountered for years in software or knowledge engineering and should hence make use of the impressive body of expertise already available in these disciplines. Compared to these research fields, the reuse issue in the Semantic Web context does, however, present some particularities, which under certain circumstances might encourage or endanger its feasibility. The importance of reuse in relation to ontologies is increased by the fact that ontologies are *per default* intended to be deployed and extended in a variety of contexts in order to mediate communication between humans and machines. Without being reused, ontologies are restricted to classical knowledge bases, and are thus not able to contribute to the realization of the Semantic Web which requires commonly agreed ontologies to mediate between services accessing semantically annotated information spaces. On the other hand, finding ontological resources – one of the core prerequisites for reuse – could be considerably enhanced in the Semantic Web era: ontologies, as well as ontological entities such as concepts or properties, are globally identified by means of URIs and could be discovered by dedicated crawlers and accessed ubiquitously on the Web. Language incompatibilities are expected to be coped with through the enforcement of Web-suitable knowledge-representation languages such as RDF(S) and OWL. Some elaborated methodologies covering major stages of the ontology life cycle complemented by a plethora of ontology-management tools (cf. for instance [70,74] for an overview) constitute a solid inventory for developing a Semantic Web-suitable ontology-reuse platform. Finally, in contrast to the software engineering field, the formal nature of ontologies and their representation using standardized machine-understandable languages open up new vistas for the operationalization of the reuse process. Reuse-relevant activities such as those related to component classification systems, retrieval of reusable components, metadata generation, automatic pattern recognition or component integration are likely to be operated (semi)automatically using semantics-enabled technologies. Compared to classical knowledge bases, the importance of the ontology-maintenance problem has been recognized in a timely manner in the ontology-engineering community, resulting in a series of methodologies and (automatized) methods and tools supporting the systematic and consistent evolution of ontological contents, at both schema and data levels [46,61,71]. Last but not least, the advent of Web 2.0 can be seen as a first step towards the realization of widely accepted ontologies. Enhancing simple tagging structures with formal semantics provides an efficient means for exploiting the impressive amount of user-generated content for the realization of community-driven ontologies Klamma et al.[45], Rattenbury [64], Damme [20].

These factors let us assume that the reuse of knowledge in the Semantic Web context can benefit from the open, ubiquitous nature of the Web, from the emergence of standards for machine-processable knowledge representation, and from the methodological and technological ontology-engineering infrastructure, thus helping to alleviate some of the major obstacles so far impeding reuse in related engineering disciplines. However, in order to tap the full potential of these favorable circumstances, we see a need to revise and refine current ontology-reuse solutions towards a new level of feasibility. This need is clearly evidenced in the case studies described in Sections 3–5.

[4] DAML ontology library: http://www.daml.org/ontologies, Protégé OWL library: http://protegewiki.stanford.edu/index.php/Protege_Ontology_Library, SchemaWeb directory: http://www.schemaweb.info.

3. Case studies in ontology reuse

In this section we provide an overview of the most prominent case studies on ontology reuse, which have been published in the knowledge-/ontology-engineering literature from the early nineties to the present. We analyze every case study from two perspectives:

(1) In order to find out whether ontology-reuse processes are performed systematically and to detect which are the core stages of the reuse process, we examine the *methodology* (implicitly) applied in the case study.
(2) In order to identify methods and techniques which proved to be helpful for the accomplishment of particular reuse activities we take a look at the *technological infrastructure* supporting the ontology-engineering team during the case study. Our aim here is not to analyze the state of the art in the vast landscape of methods and tools for solving particular issues during the ontology life cycle, but merely to spot those stages of a reuse process which could benefit from additional automatic instruments, or which are inherently human-driven and could thus benefit from a combination of human and computational intelligence.

3.1. Goméz-Pérez and Rojas-Amaya's case study

Goméz-Pérez and Rojas-Amaya describe a case study in which an ontology for standard units and a chemical ontology are reused for the purpose of developing an ontology for environmental pollutants [37]. The reuse process focuses on a method for ontology reengineering which attempts to capture the conceptual model of the implemented source ontologies in order to transform them into a new, more correct and more complete ontology. The reengineering methodology proposed by the authors consists of three steps:

(1) *Reverse engineering*: On the basis of the code of the source ontology, that is, its implementation in a particular representation language, one derives a possible conceptual model. This step is performed iteratively, by extracting models with increasing complexity: the taxonomic structure, followed by relations between concepts and instances, and finally more expressive constructs such as axioms or functions.
(2) *Restructuring*: The objective of this step is to evaluate the correctness of the extracted model, correct the detected errors, and refine the model in conformity with the requirements of the new application setting.
(3) *Forward engineering*: The ontology is reimplemented on the basis of the revised conceptual model.

The reuse process was performed according to the following stages:

(1) *Select reuse candidates*: Ontologies stored on the Ontolingua[5] and the Cyc[6] servers were manually selected and evaluated with respect to their relevance to the target domain and with respect to a series of general-purpose modeling guidelines.
(2) *Reengineering*: Relevant ontologies were reengineered as described above.
(3) *Merging*: The ontologies were merged into a final product.

The focus of the work is to demonstrate the applicability of the reengineering approach with the help of a case study. Therefore, the reported results do not provide evidence on the feasibility of current semantic technologies as regards ontology reuse, but rather concentrate on the validation of the self-developed methodology. Nevertheless, the authors admit the limitations of their approach with respect to the complexity of the ontological sources employed, and the need for automatic means. The experiment was restricted to taxonomical ontologies containing a manageable number of at most several hundred concepts. Nevertheless, the proposed reengineering workflow was executed manually, with considerable effort (18 months) in relation to the small size of the ontologies.

3.2. Uschold and Healy's case study

Uschold and Healy report on an experiment in which an engineering mathematics ontology is reused to detail the specification of a simple software tool, and to enforce units conversion and dimensional consistency-checking capabilities to this application [77]. In this attempt they tackle some of the most important issues related to the question of reusability:

(1) *Understand ontology and find the reuse kernel*: In this step the ontology is "read through" by engineers and an initial reuse kernel is identified. This preliminary selection is intended to cover the core reuse requirements of the target engineering application.
(2) *Translate the ontology*: The ontology is converted from Ontolingua into a knowledge-level representation.

[5] http://www.ksl.stanford.edu/software/ontolingua/.
[6] http://www.opencyc.org/.

(3) *Specify the task and refine into executable code*: The ontology is refined in order to satisfy the requirements of a software specification which allows automatic code generation. This step is of course application-dependent, but it could be associated with the customization of the source ontology if new application needs arise.
(4) *Verify refinement*: Verify that the generated executable code corresponds to the original ontology-based specification. Again, the verification step is not application-independent, since ontology reuse is not related per se to the production of executable code. However, this step could be assimilated with a preintegration evaluation of the target ontology.
(5) *Integrate into application*: The reused ontology is embedded in the application environment. In the overall ontology-engineering process model this typically corresponds to an ontology-merging step among reused and/or manually built ontologies (cf. Section 2).

The case study reported in [77] reveals several important limitations of present reuse technologies, although it does not investigate the complete reuse cycle at the same level of detail. While the authors declare that, in this experiment, reusing existing ontologies was subjectively profitable, they admit that further systematic investigations are required in order to alleviate the large-scale reuse of ontologies. In particular, they mention the difficulties of automatic translation between representation formats and the need for a context-oriented approach to performing this task:

> *Intrinsic problems (...) arise when design decisions required to make a good translation depend on information not present in the original ontology. In particular, one must consider the tasks to which the ontology is intended to serve.* [77].

3.3. Russ et al. case study

Russ and colleagues describe a case study in which an ontology covering the air-campaign domain was built by reusing existing ontologies partially covering its context. The reuse process, which does not adhere to a specific methodology, consists of the following steps:

(1) *Select candidate ontologies*: While this step is not elaborated in the case study, the authors identify a general time ontology and two domain ontologies as relevant candidates for reuse.
(2) *Translation*: The ontology of time is translated from Ontolingua to Loom.
(3) *Merge domain ontologies*: Two ontologies modeling the aircraft domain are merged, and the results are evaluated and refined.
(4) *Integrate time ontology*: The final ontology is obtained by aligning the domain knowledge component with the time ontology.

The conclusions of this case study are comparable to those of Uschold and Healy. While the reuse of the three ontologies is perceived to be beneficial for the target application, the authors emphasize the limitations of the techniques available so far, particularly of those related to language translators and ontology merging. Again, the lessons learned from this empirical experiment focus on the necessity of a task-oriented approach to reuse, as a means to improve the usability of general-purpose methods and methodologies in real-world scenarios. With respect to automatic translation the authors argue that "*translators need to take into account not only the 'meaning' of the descriptions or definitions in the ontology, but how these constructs are going to be used*" [65]. This point of view is reinforced by the analysis of existing merging approaches:"*While certain parts are inherently manual, the process can be made much easier if a user is able to express in general terms how the mapping should occur, e.g., this concept maps to this instance, this relation's fillers are mapped into that relation's restrictions, etc. This calls for a tool that incorporates a language to talk about ontologies, their relations and relations among their components.*" [65].

3.4. Peralta and Pinto's case study

Peralta and Pinto describe the development of the ONTO-SD ontology for a natural language dialogue system for a ticket vending machine [59]. The ontology comprised several subontologies, capturing knowledge from different domains related to traveling and ticket vending, and ranging from commercial translation to location information and time. In particular, the time subontology was developed by reuse, and the paper reports on the experiences and lessons learned from the reuse process.

This case study was performed according to the following steps:

(1) *Choice of ontologies to reuse*: The authors looked for ontologies in ontology repositories popular at that time, notably the Ontolingua Server. A second option was large, general-purpose ontologies such as WordNet, SUMO and upperCyc, which, through their very nature, are likely to capture time information. The Simple-Time ontology residing on the Ontolingua Server was selected for reuse due to technical issues, the most important being the availability of an import feature within the ontology-development environment used and the lack of automatic tools for extracting scattered knowledge out of the general-purpose ontologies surveyed.
(2) *Reverse engineering*: A conceptual model for the Simple-Time ontology was created manually by identifying classes, instances, relations and functions and interlinking these ontological primitives accordingly.

(3) *Import from Ontolingua*: This technical step mainly involved the use of the OKBC plug-in available within the ontology editor Protégé 2000.[7]
(4) *Analysis and revision*: The Simple-Time ontology was evaluated at the conceptual and the technical level against a number of generic evaluation criteria. Further on, errors that occurred during the import step were corrected, and refinements of the ontology were made to accommodate the natural language requirements of the application scenario, which requested that terms in the application ontology be in Portuguese.

The main conclusions of this case study refer to the advantages of adequate tool support in several stages of the reuse process. The authors based their choice of the ontology to be reused on the availability of automated tools for extracting knowledge in a particular domain from a large, general-purpose ontology. Another particularity of the case study is the way in which the authors carried out the last step of the process. The analysis and revision were performed in an iterative fashion, alternately on the conceptual and the imported sources, in order to overcome the limitations of the individual views.

3.5. Capellades' case study

Capellades aimed to build an application ontology by reusing ontologies available at the Ontolingua Server. The reuse process covered two main stages [14]:

(1) *Select candidate ontologies*: The selection step did not have to cope with the issue of discovering potential reuse candidates, as the set of reusable ontologies was limited to the Ontolingua repository. However, this step covers a detailed report on the evaluation procedure which unsuccessfully attempted to apply existing reusability-assessment approaches such as [36,37]. This process step resulted in the selection of a single ontology subjectively perceived to be useful for the application context.
(2) *Customize and integrate relevant ontologies*: Due to the poor application relevance results obtained in the previous step, the integration was restricted to extracting particular fragments of the selected ontology, which were subsequently embedded in the application system.

The main conclusions of this experiment refer to the first process stage. The author accounts for the nontrivial nature of the ontology-selection task, which was additionally impeded by the lack of feasible methodologies and methods for comparing or evaluating ontologies.

3.6. Arpírez at al. case study

Arpírez and colleagues give an account of a case study in which the $(KA)^2$ ontology [3] was reused in order to build the Reference ontology, a metaontology intended to capture information about ontologies and ontology-engineering projects [2]. The activities performed in the case study, covering three phases, are not representative for a complete reuse life cycle:

(1) *Choosing candidate ontologies*: In this step the $(KA)^2$ ontology was evaluated with respect to its relevance and usability for the desired purpose. The reuse candidate fulfilled many of the evaluation criteria, ranging from domain to representation formalism.
(2) *Analysis of the candidate ontologies*: The ontology was analyzed as regards the quality of its modeling decisions and its validity.
(3) *Integration*: The $(KA)^2$ ontology was extended and revised in order to adapt it to the requirements of the new Reference ontology.

Reusing the $(KA)^2$ ontology was perceived as beneficial by the case study authors, who mention cost and interoperability as two of the major advantages of this engineering strategy. However, they also identify the circumstances which contribute to the efficient operation of the reuse process: the availability of the reused ontology in an appropriate representation form, the availability of documentation, and the extensive knowledge of the ontology engineers with respect to the domain of the ontology.

3.7. Laresgoiti et al. case study

In order to illustrate the use of the KACTUS framework in real-world situations, Laresgoiti and colleagues set up an experiment in which an existing electrical network ontology was intended to be reused in an application which automatically tested equipment at fault in a Spanish electricity company [47]. The experiment was not performed in explicit compliance with a particular reuse methodology as it tackled only a single aspect of this challenging process: the direct usage of the electrical network ontology in the new application context. While the authors report on the benefits of performing reuse in this

[7] http://protege.stanford.edu/plugins/okbctab/okbc_tab.html.

setting, they are also aware of the limited applicability of their conclusions in arbitrary scenarios and emphasize the fundamental role that the original purpose of the reusable knowledge components plays in the success of this challenging process. Furthermore, they state the need for a fine-grained reuse methodology in order to allow widespread dissemination of ontologies beyond the boundaries of the knowledge-engineering community.

3.8. Bernaras et al. case study

Bernaras et al. combine a domain ontology of electrical transmission networks with a task ontology for service recovery planning on the same domain for application purposes [4]. Again, the case study does not cover the complete range of activities required to perform reuse in arbitrary settings, from discovering the reuse candidates to embedding them in the target application system. It is restricted to experiences in merging the previously mentioned ontologies at a conceptual level. Nevertheless, the conclusions of the case study clearly acknowledge the difficulties related to ontology reuse and emphasize the need for an in-depth cost-benefit analysis of reuse-oriented knowledge engineering against other development alternatives. These conclusions are based on the experiences gained during the case study; abstraction and standardization, design principles considered per default to enhance reusability, implied considerable costs for adapting the abstract ontological primitives of the two reused ontologies to the level of detail imposed by the application system.

3.9. Zhao et al. case study

Zhao et al. report on two case studies in ontology reuse in which they attempted to reuse established eBusiness and eCommerce classification systems to build Semantic Web ontologies [80,81]. In the first case study, an eCommerce ontology was developed based on existing XML standards such as xCBL, UBL, RosettaNet, and OAGIS.[8] The authors identified several challenges, at both a technical and a conceptual level, related to the reuse of such standards: the heterogeneity of the sources with respect to the quality of the modeling; the level of detail of the captured knowledge and the modeling decisions; the access policies and distribution formats; and the different degrees of maturity. As a means to facilitate interoperability, the authors propose a unifying vocabulary comprised of reusable generic components which are covered by several of the surveyed standards. The second case study was in the eProcurement domain. It was based on the same four standards, but it focused more on translation issues from XML schemas to OWL ontologies. As a result of their empirical investigations, the authors specify requirements for tool support at various stages of the reuse process:

(1) Libraries and repositories to facilitate uniform access to eBusiness and eCommerce ontologies and ontology-like structures.
(2) Automatic means to translate XML data to OWL.
(3) OWL validators.
(4) Ontology visualization software.

3.10. Coulet et al. case study

Coulet et al. present a case study in which a domain ontology was developed by reuse in order to enhance knowledge-discovery-in- databases (KDD) processes in the pharmacogenomics domain [19]. The ontology was expected to be useful in several tasks in the KDD context: to resolve heterogeneities among different data sources, to provide a common domain conceptualization in which discovery results could be stored and reused, and to semantically guide the mining of the biological data. Around 20 knowledge resources were identified as relevant to the application domain during the requirements analysis phase. They were further evaluated with respect to several criteria, the most important being the format in which they were available – in this case OBO was preferred – and the context in which they were developed, here the OBO-Foundry project. The OBO-Foundry project has defined a series of quality principles for ontology development, which provide additional guarantees for the ontologies originating from this project. During the implementation phase most of the highly heterogeneous source ontologies, including UML models, XML schemas, databases, controlled vocabularies and so on, were manually coded to OWL. In the case of the OBO ontologies this translation was carried out in an automated fashion. The evaluation of the target ontology is still ongoing.

From an ontology-reuse point of view, this case study provides several interesting insights with respect to the way reuse processes are being performed in one of the most important application areas for semantic technologies. On the one hand it states once more the need for automatic tool support for ontology engineering, most notably ontology assessment and alignment, and for translators from UML and XML schemas to OWL. On the other hand, it presents ontology reuse as a potentially viable alternative or a useful complementary activity to manual ontology development.

[8] xCBL: http://www.xcbl.org/, UBL: www.oasis-open.org/committees/ubl/, RosettaNet: http://www.rosettanet.org/, OAGIS: www.xml.gov/documents/completed/oagi/oagis.htm.

3.11. Ding et al. case study

Ding et al. present an ontology-development method based on ontology reuse and NLP-driven knowledge acquisition Ding et al. [25]. A novelty of the approach is the mechanization of the former. Mechanization is understood as a viable alternative to the costly human-driven ontology-reuse processes, which involve tedious tasks such as ontology assessment. As inputs, the method uses source ontologies and natural language documents in the relevant domain. It consists of three steps, concept selection, relation retrieval, and constraint discovery, which correspond to the definition of classes, relationships, and restrictions within an OWL ontology. The authors carried out a series of experiments to improve the results of their automated method, from which they derived several interesting rules of thumb applicable to classical ontology reuse also. In principle, reuse is considered to be a viable alternative to development from scratch, especially if performed in an automated fashion. The experiments furthermore confirm the positive impact of knowledge acquisition from texts and ontology modularization on the efficiency of the reuse process, while issues such as ontology assessment, either performed automatically or with human intervention, are still open.

3.12. Conclusions

In this section we have presented some of the most important case studies published in the last decade in the literature on ontology reuse. They describe experiments in which ontologies have been developed by reuse, and discuss the empirical findings of these experiments. The case studies have been analyzed in terms of both the methodology followed and the technical details of the activities carried out. They characterize reuse as a primarily manual process assisted by tools which perform very specific tasks such as translation between representation formats or extraction of structured information from texts. Activities such as ontology assessment, integration, translation, and customization seem to be relevant across case studies, though the way they are performed varies significantly. The need for tool support is also a finding shared by all surveyed case studies. The availability, functionality, scalability and performance of an individual technology are likely to improve rapidly as semantics gain more industrial adopters. Putting these potential developments aside, the questions of which tasks and activities within the reuse process can feasibly be automated, or are intrinsically human-driven, and what a truly semiautomatic approach to ontology reuse should look like from a methodology and a tool perspective still require further investigation.

The conclusions of the previously mentioned investigations were confirmed by the experiences we gained during two case studies focusing on reusing existing ontologies in the areas of medicine and job recruitment, respectively. Sections 4 and 5 are dedicated to a detailed description of these studies and the lessons learned during their operation. Section 6 identifies a series of requirements for practical ontology reuse.

4. Case study eHealth

4.1. A semantic web for pathology

The project "A Semantic Web for Pathology" aimed to build a retrieval system for pathology information *based on Semantic Web technologies*. The core of the retrieval system is a knowledge base consisting of several domain and generic ontologies and a set of rules describing decision processes in routine pathology. The knowledge base can be used to improve the retrieval capabilities of the archive of pathology information items. These are pathology reports in textual form, containing the observations of the domain experts (pathologists) with respect to medical cases, and digital histological slides, which are digital images obtained through high-quality scans of glass slides with tissue samples.

A complete description of the project is beyond the scope of this article, which focuses only on ontology-engineering aspects. The architecture of the system as well as further information on the system components are addressed in more detail in [66,75].

4.2. Reusing medical ontologies

The medical knowledge base was built upon UMLS, which in 2003 contained over 1.5 million concepts from over 100 medical libraries. Due to the complexity of the thesaurus and the limitations of Semantic Web tools at that time, it had to be customized with respect to two important axes:

(1) *Evaluation of the UMLS ontologies*: This task focused on the selection of libraries and concepts corresponding to "lung pathology" from UMLS.
(2) *Customization of the relevant sources*: Candidate ontologies were adapted to the particularities of language and vocabulary of the case-report archive.

This two-phase approach was justified by the application-oriented character of the case study. Its aim was not to build a general Semantic Web knowledge base for pathology, or even lung pathology, but rather one which was tailored for, and which could be efficiently used in that application setting. Additionally, building the knowledge base also implies a subsequent adaptation of the content, performed by domain experts. Therefore, the experts should be able to evaluate and modify

the ontology. Along with their technical drawbacks, very large ontologies have the additional disadvantage that they cannot be used efficiently by humans.

4.2.1. Evaluation of the UMLS Ontologies

The straightforward method for addressing the evaluation of the UMLS ontologies is to use the UMLS Knowledge Server, which provides the MetamorphoSys tool and an additional API to tailor the thesaurus to specific application needs. However, both allow mainly syntactic filtering methods (for example, they exclude complete UMLS sources, languages, or term synonyms) and do not offer means to analyze the semantics of particular libraries or to use only relevant parts of them. In a preselection phase domain experts reduced the huge amount of medical information from UMLS to the domain *pathology*. They identified potentially relevant UMLS libraries. The large number of partially overlapping libraries and the complexity of their interdependencies made this process time-consuming and error-prone, so that the final goal of the preselection phase was to identify libraries which were definitively irrelevant to our application domain. This approach corresponds to the recommendations in [60], which foresees a two-step selection process that starts by eliminating ontological sources which are not relevant to the application scope.

Approximately 50 percent of the UMLS libraries, containing more than 500,000 concepts, were assessed as potentially relevant. Managing an ontology of such dimensions with Semantic Web technologies involved still-unsolved issues with respect to the scalability and performance of the system. In the second step, the case-reports archive was used as input for identifying those concepts, which actually occur in medical reports. Such concepts are really used by pathologists when putting down their observations and therefore will also occur as search parameters. The vocabulary of the reports archive was compared to the content of the preselected UMLS libraries by means of an open source information retrieval program. The retrieval program assessed the relevance of each UMLS library – or its vocabulary – to the "query" issued. The queries were generated out of the medical reports and contained the terms used in these documents.[9] The result of this task was computed by calculating the differences between the term vectors in the vector space. We obtained a list of 10 UMLS libraries, still containing approximately 350,000 different concepts.

The size of the concept set can be explained if we consider the fact that the UMLS knowledge is concentrated in a few major libraries which cover important parts of the complete thesaurus and therefore contain most of the concepts in our lexicon. To differentiate among the concepts within the 10 resulting libraries, pathology experts selected four central concepts in lung anatomy ("*lung*", "*pleura*", "*trachea*" and "*bronchia*") and extracted similar or related concepts from the UMLS ontologies. They considered the list of all distinct concepts related through a relation of any kind[10] to the four initial concepts. The result was a set of approximately 1,000 concepts describing the anatomy of the lung and lung diseases; this served as initial input for the domain ontology.

4.2.2. Customization of the relevant sources

The linguistic analysis of the patient-report corpus demonstrated the content-related limitations of UMLS with respect to the concrete vocabulary of the report archive. Additional pathology-specific concepts, such as the components and typical content of a medical report, were added to the available ontology library. Besides content-related adaptation needs, the analysis of the generated ontology outlined several "*syntactic*" issues for further adaptations:

- The absence of naming conventions in UMLS: concepts across UMLS ontologies are not denominated using predefined naming conventions (for example, "*RESECTION OF LUNG WITH RECONSTRUCTION OF CHESTWALL WITHOUT PROSTHESIS*"*). The lack of linguistically predictable concept labels made the usage of the ontology in linguistics-related tasks such as text annotation significantly more difficult.*
- The absence of explicitly represented semantics: concepts such as "*ARF-smaller-than-2*", "*RESECTION OF LUNG WITH RECONSTRUCTION OF CHEST WALL WITHOUT PROSTHESIS*", "*Unspecified injury of lung with open wound into thorax*" should be modeled as concepts with corresponding properties and not directly as single concepts, whose names denote their meaning.
- The absence of concept names in the German language: due to the predominance of English in denominating UMLS concepts and the predominance of German terms in the pathology report archive in our application setting one needs to translate the English labels into German in order to achieve an efficient retrieval.

The comparison of the vocabulary of the medical reports archive with the generated ontology also emphasized the need to extend the knowledge base with non-medical content. In particular, part-whole and spatial relationships are often encountered in medical findings and were therefore included in the ontology library.

4.2.3. Implementation

After identifying the relevant knowledge sources and the list of concepts which could be used as input for our application, the UMLS data was translated to OWL. A Java-based module, which reads the UMLS data from a relational database and

[9] Stop words and abbreviations were not taken into account.

[10] The UMLS Metathesarus contains 7 core relations between concepts: "*parent*", "*child*", "*sibling*", "*narrower*", "*broader*", "*related-other*", "*source-synonymy*".

generates the corresponding OWL constructs, was implemented using Jena2.[11] The resulting ontologies are published server-side and can be accessed by all components in the system. The UMLS consists of two main parts [76]: the UMLS Semantic Network and the UMLS Metathesaurus. The Semantic Network contains generic medical categories and relationships (approximately 150 "*semantic types*" and 50 "*semantic relations*" in the 2003 version). It is used as a "*metalevel*" for the information in the Metathesaurus, which brings together the particular UMLS libraries. The Metathesaurus consists of a list of uniquely identified concepts and several generic relations. Every concept in the Metathesaurus references at least one semantic type, and the relations between concepts are usually typed by means of the semantic relations from the Semantic Network.

A peculiarity of the UMLS data format is the meaning of the "*relation attributes*" used for some of the Metathesaurus relations. A relation attribute references a semantic relation from the Semantic Network, but its exact meaning in the context of a given pair of concepts depends on the associated Metathesaurus relation. For example, the combination `associated_with` (a relation from the Semantic Network) and `parent` (a relation from the Metathesaurus) means a *direct* relationship between the concepts, while the same attribute together with the Metathesaurus relation `broader` implies an indirect relationship between the concepts, that is something like a path of length greater than 1 between the concepts. The absence of a relation attribute reduces the Metathesaurus relations to their original meaning, for instance, a relation `child` with no attribute is interpreted as `subClassOf`.

The list of application-relevant concepts is part of the Metathesaurus and therefore, each of the concepts is subsumed by semantic types. The ontology-engineering team first translated the UMLS Semantic Network to OWL and created a taxonomy of semantic types as classes and a taxonomy of semantic relations as properties. A second ontology contains the UMLS concepts; every UMLS concept was transformed into an OWL class. The Metathesarus relations `parent` and `child` were formalized as OWL `subClassOf` constraints. The `narrower` and `broader` relations, which define indirect subsumption relations, were formalized as `ancestor` and `descendant` in the OWL ontology. These relations could also be ignored, since their meaning could be inferred from the ontology using a reasoner. The semantics of the rest of the Metathesaurus relations is not precisely defined. Therefore, they were merged into a single `related_to` property in our ontology. The connection between relations and relation attributes was also considered. Since the relation attribute points to the semantics of a relationship between two concepts, the engineering team used this information when available. They considered the Metathesarus relations only for the cases where a relation attribute was missing. Further on, they stored the list of alternative names for every concept together with language specifications as `rdfs:label` and connected every concept to the corresponding UMLS libraries containing it. A list of all available UMLS libraries was also formalized in a separate ontology, which was imported by the core ontology.

After translating the UMLS data to OWL, we performed a consistency check of the resulting ontologies. The analysis of the inferred classification hierarchy revealed few differences in related to the original UMLS hierarchy. The UMLS contains several problematic modeling decisions which have frequently been described in research projects aiming to integrate it into knowledge-based applications. Still, a comprehensive analysis of the quality of UMLS in such a setting or especially for Semantic Web applications has not delivered an optimal solution to this problem. A possible starting point could be the Semantic Network, since every Metathesaurus concept is related to it. Additionally, the Semantic Network is supposed to be independent of a particular area in medicine. [67,11] describe some of the deficiencies of the Semantic Network at the ontological level. [62] analyzes the same issue for the UMLS Metathesaurus. However, these issues have proven to be secondary for the quality of the retrieval system, which finally made use of a relatively compact fragment of the developed ontology. Representing medical knowledge using Description Logics is not a trivial task [15]. Although translating the UMLS data format to OWL was a straightforward procedure, the expressivity limitations of the OWL language became clear after a detailed analysis of the semantics of the medical knowledge. Reasoning beyond subsumption hierarchies and extended support for concrete domains are very important for an efficient semantic retrieval system. The lack of automatic methods to deal with these issues was compensated for in our system by a semiautomatic approach to ontology-based search: for a given search query the user has the possibility of choosing semantic relationships which should be taken into account during the retrieval of the relevant documents. The processing of these additional relationships is handled using common query languages for RDFS and OWL.

4.2.4. Conclusions and lessons learned

The complexity of the application domain made the building of a lung pathology ontology from scratch extremely costly. Reusing existing sources theoretically increased interoperability, since the target ontology is, at least partially, aligned to UMLS, which is used by several medical information systems. However, though it contains a huge amount of domain information, the reuse of UMLS and integrated libraries in our application setting was not trivial, due to the often ambiguous modeling decisions and an error-prone integration schema [34,41,67]. The task specificity of each UMLS library, the complexity of the complete thesaurus, and the heterogeneous coverage degree for specific medicine subdomains such as ours, *lung pathology*, made a high-quality customization for specific application needs difficult. Besides, most of the available medicine ontologies are available in a proprietary representation format which hampers sharing and reuse. A translator from UMLS to OWL ontologies is, to the author's knowledge, not available.

[11] http://jena.sourceforge.net.

The most challenging tasks carried out in this experiment can be easily identified by examining the time distribution of the development efforts. The customization of the source ontologies required over 45 percent of the time necessary to build the overall target ontology. A further 15 percent of the engineering time was spent on translating the input representation formalism to OWL. The reuse-oriented approach required considerable effort to evaluate and extend the outcomes (approximately 40 percent of the total engineering time). According to our experiences in this project, the benefits of reuse were outweighed by the costs, because of the difficulties related to the evaluation and (technical) management of large-scale ontologies and because of the costs of the subsequent refinement phase.

Another important issue was the usage of the ontology by the community of domain experts, who reported serious acceptance problems with respect to the UMLS-based ontology: domain experts seemed to have difficulties in trusting the content of the ontology and in systematically extending it for a more detailed representation of pathology-specific knowledge. This was the main motivation for building a second application ontology on the basis of the domain corpus of patient records provided by our healthcare partner [56]. The engineering process relied on the same engineering methodology as the first experiment, while XML-based medical reports were employed as an input for the conceptualization phase (cf. [56] for a detailed description and evaluation of the methods employed). The main advantages of the latter experiment compared to the UMLS-based one were the significant cost savings and the increased fitness for use of the generated ontology with respect to the semantic annotation task. From a resource point of view, building the first ontology involved four times as many resources as the second approach (5 person-months for the UMLS-based ontology with 1,200 concepts versus 1.25 person-months for the ‘ ‘text-close” ontology of a similar size). The evaluation of the suitability of the two ontologies to semantically annotating medical documents confirmed the results of the resource-based evaluation. In addition to the technical and economical benefits the ontology derived from the medical reports had a considerably higher acceptance rate among its users: the results of the methodology were easily understandable to the domain experts, who were able to rapidly evaluate and refine the ontology.

The lessons learned during this case study have been summarized in [55] in an inventory of guidelines which are valid for similar expert domains and for application scenarios related to information retrieval and automatic semantic annotation. As a starting point, we used a set of domain-independent guidelines from the European project OntoWeb, which focus less on technical aspects, and more on ‘‘*issues that relate to the business environment that affects the deployment, integration and acceptance of the ontology-based application*” [54]. The initial checklist contains 13 items covering both organizational and ontology-specific issues. Since the engineering team did not encounter any problems related to the organizational setting (satisfactory user involvement, no legacy systems or license problems, etc.), we elaborated on the topics which relate directly to the ontology-engineering process and adapted them to the medical domain. This list, illustrated in Table 1, could be complemented with the addition of modeling best practices, which are equally important in complex domains such as medicine. Such best practices are emerging as a result of the W3C Semantic Web Best Practices and Deployment Working Group.[12]

The guidelines contributed significantly to the ontology-reuse requirements specification in Section 6.

5. Case study eRecruitment

5.1. Knowledge nets

The ‘‘Knowledge Nets” project explores the potential of Semantic Web from a business and a technical perspective by examining the effects of the deployment of Semantic Web technologies on particular application scenarios and market sectors. The aim of the case study was to analyze the potential of Semantic Web technologies, especially ontologies, in coping with the bottlenecks of present eRecruitment solutions, particularly with the limitations of keyword- or statistics-based information retrieval techniques used in job search engines. In doing so, domain-relevant ontologies termed as ‘‘human resources/HR ontologies” would be used as semantic indices, by which job descriptions and applications in the selected sector could be classified and matched, thus enhancing the system with semantics-aware search functionalities.

Just as in the eHealth scenario, a complete description of the application underlying the case study is beyond the scope of this article, which is concerned with ontology-engineering aspects (cf. for example [6] for a detailed description of the scenario).

5.2. Reusing human resources ontologies

The usage of commonly agreed-upon ontologies has a long tradition in the human resources field. The need for comprehensive classification systems describing occupational profiles has been recognized at an early stage of the eRecruitment era by many interested parties. In particular, and *in contrast to the medical sector*, major governmental and international organizations drove the emergence of standard classifications comprising unambiguous and well-documented descriptions of occupational titles and associated skills and qualifications. The result is an impressive inventory of classification systems, mostly at the national level, ready to be deployed in job portals in order to simplify the management of electronically available job postings and job-seeker profiles and to encourage application interoperability. Standards such as O∗NET

[12] http://www.w3.org/2001/sw/BestPractices/.

Table 1
Guidelines for reusing medical ontologies.

Process step	Guidelines
Domain analysis	Specify the tasks the ontology will be involved in. They have consequences on the content and on the representation of the target ontology. Different tasks imply different relevance criteria for selecting potentially reusable resources: *Semantic annotation task* • Concepts should be denominated in natural language. • The natural language used in the ontology labels should be the same as the one used by the users and in the documents to be annotated if user queries are to be processed automatically based on the ontology. • Concepts should be denominated using naming conventions and in a linguistically predictable form to facilitate the automatic document annotation task. • Modeling decisions should be recorded during the conceptualization phase in order to simplify the human-driven document annotation. *Information retrieval task* • The ontology should be formal to enable automatic reasoning. • Concepts should be denominated in natural language to enable ontology-based query formulation. • The ontology should provide a rich semantic representation of the domain to refine the retrieval algorithm.
Ontology reuse	Despite of the large number of very comprehensive medical ontologies, reusing them entails significant costs, which might outweigh the costs of a new implementation. Knowledge resources which will be reused to create the target ontology potentially necessitate considerable modifications in order to fulfill the application requirements: • Concepts are denominated in an ad hoc manner even within the same ontology. • The semantics of the concepts are sometimes encoded in their names. • Many medical ontologies are stored in proprietary forms, and there are no translation tools. • Many ontologies are modeled in an ambiguous way. Existing medical ontologies have a considerable size, but a relatively simple structure. Adapt your reuse methodology to their particularities: • A complete assessment of their application relevance is rarely possible. • The same domain is covered to a similar extent by several ontologies. There are no fundamental differences among them with respect to their suitability in the Semantic Web context. Eliminating candidate ontologies which are definitely not relevant is sometimes more feasible than attempting a complete evaluation. • Even when an ontology is assigned a high relevance score, its usage in the application setting might depend on the availability of tools which are able to handle it and on user acceptance. Scalability and expressivity of the ontology model are two important factors in this context. • Matching and merging ontologies with overlapping domains imposes serious scalability and performance problems on the tools available. Nevertheless, using simple algorithms (such as linguistic matchers) considerably increases the efficiency of this activity. • The merging results are to be evaluated by human experts. Due to the size of the ontologies, the merging methodology should foresee a flexible and transparent involvement of the users during the process in order to reduce the complexity of the merging evaluation. Combining human and computational intelligence is likely to achieve optimal results in the merging task. • Reasoning about these models requires scalable inference engines.
Ontology management	The size of the target ontology requires powerful storage mechanisms with adequate reasoning support (for example, for automatically checking inconsistencies) Elaborate a detailed evaluation framework to control ontology evolution. The maintenance of large-size ontologies requires additional effort in documenting modeling decisions.
Updates	Medicine is a dynamic domain; most of the ontologies change within a relatively short time. Updating the target ontology under these circumstances can be very tedious, especially if the source ontologies were not directly integrated into the new application.
Ontology-learning	The success of an ontology-learning attempt depends on the quality of the document corpus; domain-focused documents are expected to perform better. Data noise (telegraphic writing style, the intensive usage of nonstandard abbreviations, etc.) is common to medical texts such as medical findings. The ontology-learning algorithm should be able to deal with these particularities. The knowledge acquisition process should be performed incrementally, because of the complexity of the domain to be modeled. Again, an approach exploiting human and computational intelligence is likely to achieve feasible results.

(Occupational Net), ISIC (International Standard Industrial Classification of Economic Activities), SOC (Standard Occupational Classification), or NAICS (North American Industry Classification System), to name only a few, are feasible building blocks for the development of eRecruitment information systems. At the same time, they are valuable knowledge resources for the development of application-specific ontologies, which could inject domain semantics-awareness into classical solutions in this field, as described below.

The reuse process was performed according to the following three phases:

(1) *Discovery of the reuse candidates*: In this step the ontology-engineering team conducted a survey of potentially reusable ontological sources.
(2) *Selection of the relevant ontological sources*: The results of the previous step were analyzed with respect to domain and application relevance, as well as general quality and availability.
(3) *Customization and integration of the ontologies to be reused*: The relevant fragments of the (to some extent) very comprehensive sources were extracted and integrated into a single target ontology.

5.2.1. Discovery of the reuse candidates

In order to compute a list of existing ontologies or ontology-like structures potentially relevant for the human resources domain, we carried out a comprehensive search with the help of currently available ontology-location support technologies:

General-purpose search engines: We used conventional search tools and predefined queries combining implementation and content descriptors such as "filetype:xsd human resources" or "occupation classification".
Ontology search engines and repositories: Resorting to existing dedicated search engines and ontology repositories clearly pointed out the immaturity of these technologies for the Semantic Web.
Domain-related sites and organizations: A third search strategy focused on international and national governmental institutions which might be involved in standardizations efforts in the area of human resources. These organizations make their work, which is proposed for standardization, publicly available in form of domain-relevant, lightweight HR ontologies.

The result of the discovery procedure – which was performed as manual Google-based searches on preselected keywords in correlation with the investigation of the Web sites of international and national employment organizations – was a list of approximately 24 resources covering both descriptions of the recruitment process and classifications of occupations, skills, or industrial sectors in English and German. Semantic Web search engines (Swoogle) and repositories (the DAML library, the Protégé OWL library, the SchemaWeb directory) available at that time did not return any results.

5.2.2. Selection of the relevant reuse candidates

The engineering team decided to reuse the following resources:

(1) *HR-BA-XML*: The official German translation of Human Resources XML, the most widely used standard for process documents such as job postings and applications.[13] HR-XML is a library of more than 75 interdependent XML schemas defining particular process transactions, as well as options and constraints ruling the correct usage of the XML elements.
(2) *BKZ*: Berufskennziffer, which is a German version of SOC System, classifying employees into 5,597 occupational categories according to occupational definitions.[14]
(3) *SOC*: Standard Occupational Classification, which classifies workers into occupational categories (23 major groups, 96 minor groups, and 449 occupations).[15]
(4) *WZ2003*: Wirtschaftszweige 2003, which is a German classification standard for industrial sectors.[16]
(5) *NAICS*: North American Industry Classification System, which provides industry-sector definitions for Canada, Mexico, and the United States to facilitate uniform economic studies across the boundaries of these countries.[17]
(6) *KOWIEN*: Skill Ontology from the University of Essen, which defines concepts representing the competencies required to describe job position requirements and job applicant skills.[18]

The selection of the six sources was performed manually without the use of a predefined methodology or evaluation framework. The documentation of the 24 potential reuse candidates was consulted in order to assess the relevance of the modeled domain to the application setting. The decision for or against a particular resource was very effective due to the small number of reuse candidates covering the same or similar domains and the simplicity of the evaluation framework, which focused on provenance and natural language aspects. Nevertheless, the resulting ontologies required intensive post modifications in order to adapt them to the requirements of the tasks they were expected to be involved in at the application level. The importance of these application-oriented constraints was underestimated by the engineering team at that point. They were not taken into account during the evaluation in the absence of a methodology for assessing the usability of the six reuse candidates against them.

For the German version of the ontology, the BKZ and the WZ2003 were the natural choice for representing occupational categories and industrial sectors, respectively. The same applies for the English version, which reused the SOC and NAICS classifications. As for occupational classifications in the English language, the SOC system was preferred to alternative such as NOC or O∗NET due to the availability of an official German translation.[19] The same applies for the choice between industry-sector classifications: in contrast to ISIC[20] the NAICS system includes a German version and is used in various applications and classifications in the human resources area.

[13] HR-BA-XML: http://www.arbeitsagentur.de/zentraler-Content/A04-Vermittlung/A045-Dritte/Publikation/White-paper.pdf, HR-XML: http://www.hr-xml.org.
[14] http://www.arbeitsamt.de/hst/markt/news/BKZ_alpha.txt.
[15] http://www.bls.gov/soc/.
[16] http://www.destatis.de/allg/d/klassif/wz2003.htm.
[17] http://www.census.gov/epcd/www/naics.html.
[18] www.kowien.uni-essen.de/publikationen/konstruktion.pdf.
[19] http://www23.hrdc-drhc.gc.ca/2001/e/generic/matrix.pdf, http://www.onetcenter.org/.
[20] http://unstats.un.org/unsd/cr/registry/regcst.asp?Cl=17&Lg=1.

5.2.3. Customization and integration of the relevant sources

The main challenge of the eRecruitment scenario was the adaption of the six reusable ontologies to the *technical* requirements of the job portal application. From a content-oriented perspective, five of the sources were included 100 percent in the final setting, due to the generality of the application domain. The focus on a particular industrial sector or occupation category would require a customization of the source ontologies in the form of an extraction of the relevant fragments. To accomplish this task for the KOWIEN ontology, we compiled a small conceptual vocabulary (of approx. 15 concepts) from various job portals and job-procurement Web sites and matched these core concepts manually to the source ontology.

The candidate sources vary with respect to the represented domain, the degree of formality, and the granularity of the conceptualization. They are labeled using different natural languages and implemented in various formats: text files (BKZ, WZ2003), XML schemas (HR-XML, HR-BA-XML), DAML+OIL (KOWIEN). While dealing with different natural languages complicated the process, human-readable concept names in German and English were required in order to make the ontology usable in different job portals and to avoid language-specific problems. Another important characteristic of the candidate ontologies is the absence of semantic relationships among concepts. Except for the KOWIEN ontology, which contains relationships between skill concepts, the remaining ones are confined to taxonomical relationships at most. Consequently we had to focus on how vocabularies (concepts and relations) could be extracted and integrated into the target ontology. The usage of the ontology in semantic matching tasks requires that it is represented in a highly formal representation language. For this reason the implementation of the human resources ontology was realized by translating several semi structured input formalisms and manually coding text-based classification standards to OWL.

5.2.4. Conclusions and lessons learned

A wide range of standards for process modeling and classification schemas for occupations, skills, and competencies have been developed by major organizations in the human resources field. Using these standards was a central requirement for the simplification of the communication between international organizations accessing the portal and for interoperability purposes. Additionally, reusing classification schemas such as BKZ, WZ2003 and their English variants, which were completely integrated into the target ontology, meant significant cost reductions. They guaranteed a comprehensive conceptualization of the corresponding subdomains and saved the costs that would normally be incurred through a collaboration with domain experts.

Further on, due to the generality of the application domain – the final application ontology provided a (high-level) conceptualization of the *complete* human resources domain – and to the manageable number of preselected reuse candidates, the ontology-evaluation step did not entail major development costs. This is indicated by the distribution of effort in each of the enumerated process stages. Solely 15 percent of the total engineering time was spent on searching and identifying the relevant sources. Approximately 35 percent of the overall efforts was spent on customizing the selected source ontologies. Due to the heterogeneity of the knowledge sources and their integration into the final ontology, up to 40 percent of the total engineering costs was necessary to translate these sources to the target representation language OWL. Lastly, the refinement and evaluation process required the remaining 10 percent. The aggregation of knowledge from different domains proved to be a very time-consuming and tedious task because of the wide range of classifications available so far. The second cost-intensive factor was related to technological issues. Though nontrivial, the manual selection of relevant parts from the KOWIEN ontology and the HR-BA-XML standard was possible in our case thanks to the high connectivity degree of the pruned fragments and to relatively simple, tree-like structure of the sources. However, we see a clear need for tools which assist the ontology engineer during this kind of task on real-world, large-scale ontologies with many thousands of concepts and more complicated structure. Despite the problems mentioned, our experiences in the eRecruitment domain make us believe that reusability is both desirable and possible. Even though the ontology is still under development, it already fulfills the most important requirements of the application scenario, which are related to interoperability and knowledge share among job portals. Reusing available ontologies requires, however, a significant amount of manual work, even when using common representation languages such as XML schema or OWL. The reuse process would have been significantly optimized in terms of costs and quality of the outcomes with the necessary technical support.

The case study emphasized once more the need for extensive methodological support for domain experts with respect to ontology reuse. In the absence of fine-granular, business-oriented process descriptions, the domain experts – possessing little to no knowledge on ontologies and related topics – were not able to perform any of the process steps without continuous guidance from the ontology engineers.

Just as for the medical case study, the lessons learned in the eRecruitment scenario were summarized in the form of a set of guidelines for ontology reuse which might aid ontology developers in similar situations. These are depicted in Table 2.

6. Requirements for ontology reuse

Typically, ontology reuse starts with the selection of a set of ontological resources assumed to be relevant to the application setting. Once their usability has been positively evaluated, these ontologies are subject to various technology-driven customization and integration activities. In an arbitrary application scenario the reuse candidates differ with respect to many aspects: content, implementation, provenance, level of formality, maturity, etc. They might model a domain of interest from a multitude of viewpoints, or they might tailor the domain representation to particular scopes and purposes. They might not

Table 2
Guidelines for building HR ontologies through reuse.

Process step	Guidelines
Ontology discovery	Finding an appropriate ontology is currently associated with considerable effort and is dependent on the level of expertise and intuition of the engineering team. Real-world ontologies, which can be used to build applications creating added value for businesses cannot be discovered using existing Semantic Web search engines or in ontology repositories In the absence of full-fledged ontology repositories and mature ontology search engines, the following strategies could be helpful: • Use conventional search engines with queries containing core concepts from the domain of interest and terms such as ontology, classification, taxonomy, controlled vocabulary, glossary. For instance, "classification AND skills". • Identify institutions which might be interested in developing standards in the domain of interest and visit their Web sites in order to check whether they have published relevant resources. • Large amounts of domain knowledge are available in terms of lightweight models, whose meaning is solely human-understandable and whose representation is in proprietary, sometimes unstructured formats. These conceptual structures can be translated into more formal ontologies if appropriate parsing tools are implemented, and are therefore a useful resource for building a new ontology. • Dedicated libraries, repositories, and search engines are still in their infancy. The majority of ontologies stored in this form are currently not appropriate for the human resources domain. This applies also for other institutional areas such as eGovernment or eHealth.
Ontology selection	Due to the high number of classifications proposed for standardization in the HR domain, the evaluation methodology should take into consideration the high degree of content overlap between the reuse candidates and the impact of the originating organization in the field. Furthermore, the evaluation methodology should be aware of the following facts: • A complete evaluation of the usability of the reuse candidates is extremely tedious, if not impossible. The HR domain is covered to a similar extent by several ontologies, while there are no fundamental differences among them with respect to their suitability in a semantic job portal. Eliminating candidate ontologies which are definitely not relevant is sometimes more feasible than attempting a complete evaluation. • An important decision criterion is the provenance of the ontology, since this area is dominated by several emerging standards. Many standards situated within international institutions such as the EU or the UN are likely to be available in various natural languages. • Many high-quality standards are freely available. • as the majority of HR ontologies are hierarchical classifications, the evaluation process requires tools supporting various views on the vocabulary of the evaluated sources. • These considerations apply for further application scenarios such as eGovernment, eCommerce and eHealth.
Ontology integration	Existing HR ontologies have a considerable size, but a relatively simple structure. Adapt your integration methodology to their particularities: • Integrating ontologies with overlapping domains imposes serious scalability and performance problems to the tools available at present. Nevertheless, using simple algorithms such as linguistic and taxonomic matchers considerably increases the efficiency of this activity. • The integration results should be evaluated by human experts. Due to the size of the ontologies, the underlying methodology should foresee flexible and transparent involvement of the users during the process in order to avoid the complexity of a monolithic evaluation and to benefit of human intelligence. • Dedicated means for extracting lightweight ontological structures from semistructured textual documents are required. Approaches such as [10,17] introduce novel algorithms and tools for this purpose. • The customization of these structures with respect to particular domains of interest (for example, a HR ontology for the chemical domain) causes additional efforts as all HR standards are independent of any industrial sector.

have the same level of formality or the same implementation language; might be usable only in accordance with specific license conditions; or might still be under development, or at least subject to frequent updates and changes.

An automatic integration of the source ontologies into an application ontology means not only the translation to a common format, but also the matching, merging and alignment of the resulting schemas and the associated instance data. Our findings during the case studies presented clearly showed that none of the mentioned activities can be performed without human intervention. Two reasons have been identified for this: First, ontology matching, merging, and alignment are knowledge-intensive tasks, which are, through their very nature, human-driven, in the sense that they can be accomplished considerably easier by a human than by a computer program. Second, while constantly improving in terms of user-friendliness, performance, and scalability, existing tools in the previously mentioned areas cannot handle different types of ontologies equally well. Ontology engineers need in-depth knowledge of their pros and cons in order to be able to use them to create an integrated ontology through reuse. Furthermore, while most of the tools are used in a semiautomatic fashion, they are not designed to optimally exploit human and computational intelligence. Their primary aim is to automatize a specific ontology-management task as a means to lower costs and improve productivity. While the quality of such automated approaches has consistently improved, it is still far from outweighing the value of the manual effort required.

At the process level, ontology engineers are provided with a minimal inventory of methodologies for ontology reuse, which are usually restricted to providing a generic, high-level description of the process while concentrating on the technical integration of the ontologies involved. Selecting appropriate ontologies is a nontrivial task, not only because of the lack of flexible and fine-grained evaluation frameworks, but also because of the difficulties attested by humans when dealing with the extreme heterogeneity of the resources assessed. Practical guidelines and best practices would provide additional support in the operation of the process.

In the following discussion, we will take a closer look at the empirical findings gathered during the feasibility study. We start by specifying requirements related to the methodological support for ontology reuse. These are related primarily to the task of assessing the usability of existing ontologies in new application contexts. We then concentrate on a series of design principles for ontology-management technology, particularly for ontology integration.

6.1. Requirements for an ontology-reuse methodology

As a starting point for the specification of the requirements for the planned reuse methodology we revised the analysis benchmark in [32], which addresses the issue of aligning and evaluating counterpart approaches in the more general field of ontology engineering. The proposed framework consists of nine criteria, as follows:

(1) Inheritance from knowledge engineering.
(2) Detail of the methodology.
(3) Recommendations for knowledge formalization.
(4) Strategy for building ontologies.
(5) Strategy for identifying concepts.
(6) Recommended life cycle.
(7) Differences from the IEEE Software Development.
(8) Recommended techniques.
(9) Application to current projects.

We have adapted this framework to ontology reuse. The original criteria 2, 4, 6, 8, and 9 have been revised in accordance with the particularities of the new ontology-engineering setting. The remaining ones have been discarded; they were not relevant in our case since an ontology-reuse methodology is intended to be only one part of a more complex ontology-engineering framework.

(1) Level of detail of the methodology.
(2) Relation to application scenarios.
(3) Recommended life cycle.
(4) Support methods and tools.
(5) Methodology validation.

We now elaborate on each of the five criteria.

6.1.1. Level of detail of the methodology

From a usability perspective, it is important that the methodology provides a fine-grained and precise description of the process, assigns particular activities to roles, and predefines the inputs and outputs of each process phase. Further on, the methodology should avoid recommending activities and tasks whose purpose might be intuitively understood by the majority of methodology applicants but whose implementation is not clearly explained or is even debatable in particular classes of application scenarios. The most prominent example here is probably ontology-evaluation/selection. Assessing the correctness and completeness of an ontology, two criteria which can be found in many ontology-evaluation approaches to date, is a task which cannot be properly accomplished in the absence of a description of the domain to be modeled to which the ontology can be compared. How such comparisons can be carried out is in turn a matter of the format in which this domain description is available and of the ontology-engineering methodology adopted. All this information, and much more, considerably increase the practical added value of an ontology-reuse methodology.

While it does not cover the complete range of reuse activities, examining the workflow underlying each of the reported experiments is a good starting point for the creation of a complete, detailed description of the ontology-reuse process. Table 3 provides an overview of the alternative reuse models encountered during the feasibility study. Apart from the heterogenous terminology, an aggregation of the implicitly utilized process models results in a four-stage reuse workflow, as follows:

(1) *Discovering the ontologies*: Generally, the engineering team starts the reuse process by searching for ontological resources which are superficially perceived as application relevant. As reported in the eRecruitment case study in Section 5, this technical step is to date based on the level of experience and intuition of the process participants as no ontology-location tools have been established. From a terminological point of view, this phase is also refereed to as "searching", "finding" or "locating" ontologies.
(2) *Selecting those to be reused*: Some of the case studies argued about the difficulties related to this step, proposing high-level activities such as understanding the ontology, checking whether the ontology is application-relevant, proofreading the ontology, etc. The case study by Peralta and Pinto reports on reengineering and translation activities performed as part as this selection step. These activities were required as different ontology-evaluation methods can be applied optimally at different levels of an ontology (conceptual, implementation) and for tool-support reasons. Translation is also relevant for the case study by Uschold and Healy; however, in this case the term refers to the a reverse engineering

Table 3
Reuse process as performed in the analyzed case studies.

	Scope of the case study	Ontology discovery	Ontology selection	Ontology customization and integration
Goméz-Pérez & Rojas-Amaya	Ontology reengineering	Ontolingua server	Reverse engineer, restructure	Forward engineer, merge
Uschold & Healy	Ontology reuse	–	Understand, translate	Refine, verify, integrate
Russ et al.	Ontology reuse	–	Select	Translate, merge, integrate
Peralta & Pinto	Ontology reuse	Ontolingua server, upperCyc, WordNet, SUMO	Choose, reengineer	Import, analyze, revise, integrate
Capellades	Ontology reuse	Ontolingua server	Understand, select	Prune
Arpírez at al.	Ontology reuse	–	Choose, analyze	Extend, refine
Laresgoiti et al.	Ontology reuse	–	–	Translate, integrate
Bernaras et al.	Ontology reuse	–	–	Reverse engineer, merge
Zhao et al.	Ontology reuse	–	–	Translate, integrate through unifying vocabulary
Ding et al.	Ontology reuse	–	–	Select concepts, retrieve relations, discover constraints
Paslaru et al.	Ontology reuse	–	Understand, assess usability	Prune, translate, merge, integrate
Paslaru & Mochol	Ontology reuse	Web	–	Prune, translate, integrate

activity which had the purpose to derive the conceptual model underlying the engineering mathematics ontology to be reused.

(3) *Customization of relevant ontologies*: Once the set of reusable ontologies has been determined, they are translated into a new representation language, extended or simplified/pruned. The customized ontologies form the basis of a new, integrated ontology which is then used within an application system [60]. As previously mentioned, customization activities can be part of the selection step as well if requested by the ontology-evaluation methods applied. If reuse includes reengineering, this phase is also associated with forward engineering efforts.

(4) *Integration into an application ontology*: A new ontology is created from two or more existing ontologies. Depending on the amount of change necessary to derive the integrated ontology from the source ontologies, different levels of integration ranging from alignment to merging can be distinguished [69].

6.1.2. Relation to application scenarios

As repeatedly mentioned in the literature survey and in accordance with our own observations, the prospective reuse methodology should pay particular attention to application-narrow aspects and to feasible support methods and tools in order to enhance its real-world usability.

The success of ontology reuse is significantly influenced by a careful analysis of the requirements induced by the context the target ontology is intended to be used in. This primarily means that certain classes of tasks typically carried out using ontologies, such as semantic annotation, semantic search, data mediation, etc, impose particular constraints on the properties of the ontologies to be reused or constructed. These constraints need to be taken into account in the first two stages of the reuse process, in which the engineering team looks for potential reuse candidates and selects those which are estimated to be relevant.

A second application-narrow dimension which impacts the way a specific reuse process is performed is the methodology applicants. In order to ensure the wide-scale dissemination of semantic technologies, it is essential that the proposed techniques minimize the amount of expert knowledge required for their operation. Hence, the methodology should account for this required low barrier of entry and adapt its content and structure to the needs of its users.

6.1.3. Recommended life cycle

Due to the complexity of the reusability issue in conjunction with the practical problems encountered when applying existing methods and tools in arbitrary scenarios, the ontology-reuse methodology should propose an incremental process model. This would allow methodology applicants to monitor and improve intermediary process outcomes, and to have a direct control over the way reuse is being executed. In situations in which a particular activity cannot be carried out automatically without considerable manual intervention, the engineering team should have the possibility to flexibly alternate tool-supported with human-driven process phases and to perform these phases iteratively. In this context it is important to understand the interplay between human and computational intelligence not only as a means of improving (or evaluating) automatically generated results but, also, and much more, as an opportunity to use both types of intelligence for the tasks they are known to be optimal for.

6.1.4. Support methods and tools

Helpful for the usability of the proposed methodology are precise tool and method recommendations. In every one of the examined case studies we found strong evidence that the lack (or the presence) of appropriate techniques and tools for

supporting resource-intensive and error-prone reuse activities was a determining factor in the feasibility of a reuse-oriented engineering strategy. Therefore, the methodology should be accompanied by (a description of) a full-fledged, methodology-compatible technological environment. Complementarily, the methodology should identify those activities within the reuse process that can be feasibly automated, or, by contrast, require extensive human intervention. This will lead to a redesign of semiautomatic ontology-management tools. To date such tools have typically required human input to revise automatically generated results, instead of using human intelligence optimally to solve knowledge-intensive issues and to support the user by facilitating access to the information she needs.

6.1.5. Methodology validation

At the process level, the analyzed case studies revealed that the application scope of existing reuse-oriented methodologies is limited to the settings they originate from. Except for the work authored by methodology designers (for example, [2,37]) the case studies do not explicitly commit to a predefined methodology.

In order to increase its usability, the prospective methodology should be carefully validated in real-world scenarios. An initial validation method could rely on the criteria introduced above associated with a set of quantified metrics for generating comparable evaluation results. A second option would be case study research [78].

6.2. Requirements for ontology-reuse support methods and tools

At the method and tool level, the conclusions of the case studies focus on the incapacity of the existing technological framework to feasibly deal with the heterogeneity of existing ontological sources in terms of content, level of formality, implementation language, size, etc.

6.2.1. Finding ontologies

At present the question of how existing ontologies can be found is not trivial. On the one hand, the ontology developer can attempt a Web-wide search using a standard search engine such as Google[21] or choose a Semantic Web-specific search tool such as Swoogle.[22] On the other hand, he or she could try to decide which organizations best represent the domain that is to be modeled and attempt to determine whether they have made any ontologies/classifications public. If we consider the eRecruitment scenario, the relevant subdomains would be "*human resources*", "*job classification*", "*occupational classification*", etc. The latter approach could lead to contacting or visiting the sites of employment agencies, while the former would try to find relevant ontologies using queries like "*human resources filetype:owl*", which in Google returns 57 OWL files describing biology and medical informatics. Various queries without file-type restrictions resulted in a broader recall at the cost of an unacceptable lack of precision. On the other hand, a search for taxonomies, classifications, and classification systems (that is, using Google queries such as "*human resources taxonomy*") has proven to deliver significantly better results, leading to a list of organizations and standards in the previously mentioned domains. This, however, necessitated a careful evaluation and customization because of the heterogeneity of the formal and content-related characteristics of the list items.

Alternatively, ontologies could be grouped into repositories. As long as the developer knows how to access the repository, he or she can look there for relevant ontologies. The DAML Ontology Library is one of the most representative examples, offering a simple Web-based interface to the source ontologies. Ontology users can access them using different criteria such as URI, keyword, submitting organization, or express queries in terms of ontology classes and properties.[23] Ontology repositories appear to be a useful means of providing an access point for developers to locate ontologies. However, the present state of the art does not resolve a number of issues. The means of locating ontologies is quite haphazard, and relies on the same type of keyword matching that occurs in nonsemantic search engines such as Google. Queries cannot draw on the semantics of ontologies themselves in order to be able to find generalizations, specializations, or equivalences of search terms/concepts. Finally, the repositories link to the complete ontologies according to their descriptions, meaning that access is on an "all or nothing" basis, not taking into account the various needs of individual users.

To summarize, a full-fledged ontology repository is expected to provide two types of features related to general information repositories and to the semantically represented information, respectively (cf. [33]). The first category implies issues such as content quality (coverage, actuality, and user-perceived information value) and typical services (classification, search, and browsing and navigation). The special nature of the managed information leads to requirements related to the semantic aspects of these general-purpose features.

6.2.2. Selecting ontologies to be reused

Assessing the usability of a certain ontology with respect to a set of application-related requirements is one of the most challenging tasks in ontology reuse [14,55,60]. While the task is primarily targeted at humans, its efficient operation is still dependent upon technical means to simplify the access to the ontologies to be evaluated, which might be complex, large, or hardly human-readable, for the process participants. This implies a whole series of computer-aided techniques to make it possible to interact with an ontology and to align comparable ones. Firstly, ontology evaluators can benefit from the

[21] http://www.google.com.
[22] http://swoogle.umbc.edu/.
[23] www.daml.org/ontologies/.

availability of a uniform representation of ontology descriptions. Reliable rating and attestation methods might provide additional control over the sometimes highly subjective selection process. Finally, multimodal tools for visualizing, editing and querying the content of the ontologies involved are fundamental for dealing with large amounts of information. Tools comparing the contents and the structures of various resources further aid the ontology evaluators in deciding upon the suitability of an ontology for the new setting.

6.2.3. Customization and integration of relevant ontologies

Each of the relevant ontologies might be subject to additional modification and integration operations meant to adapt them to particular technical requirements and to finally embed them in the application system. In the first category, we have identified the following major customization measures:

- Translation to a new representation language
- Extraction of a subontology
- Modification and extension in contents, structure or both

Multiple ontologies are integrated. The integration process can be seen as a sequence of two phases, the computation of the similar ontological elements (usually termed "*matching*") followed by their aggregation. Each of the analyzed case studies, some of which have not been included in the present feasibility study, points out the difficulties arising from the cumbersome technique and tool utilization. Even if appropriate means were eventually available, setting up the technological environment to perform one of the previously mentioned activities was inconceivable in the absence of considerable bodies of expert knowledge. An automatic operation of these tasks – as envisioned by the Semantic Web community – was possible only under extremely special circumstances.

Despite the relatively large number of promising approaches in the fields of matching, merging, and integration, their limitations with respect to certain ontology characteristics have been emphasized often in recent literature [35,49,51]:

- Some approaches assume a common or, at least to large extent, overlapping universe of discourse [18].
- They cannot be applied across various domains with the same effect (for example, Cupid, as stated in [35]).
- They require certain representations (or translation to the suitable format) or natural languages. This is true, for instance, for the COMA approach [27].
- They perform well on relatively small inputs with at most hundreds of concepts and have not been tested or do not scale for real world applications processing complex schemas.
- They do not perform well on inputs with heterogeneous (graph) structures (Cupid [35]) or are restricted to tree-based conceptual models (SimilarityFlooding [51], S-Match [35]).
- The results are based on a one-to-one mapping between taxonomies (as in GLUE [28].
- They involve some manual preprocessing (as in GLUE, COMA [26]).

Ontology-translation approaches deal with similar heterogeneity problems. Furthermore, valuable amounts of ontological knowledge are stored in a semistructured form. Their representation using Semantic Web languages can be achieved only with the help of tools that are able to extract ontological concepts from data bases, XML schemas, or Web-published taxonomies.

7. Conclusions

It is widely acknowledged that the efficient and effective operation of the reuse process is a precondition for the large-scale take-up of semantic technologies. However, the challenges associated with achieving this objective – in the Semantic Web context or beyond – are also well-known, given the inherent limitations of realizing highly reusable, commonly agreed-upon knowledge conceptualizations. The current state of the art in the ontology engineering field highlights the need for additional instruments to increase the reusability of existing ontological sources in new application settings, and to aid humans in carrying out this process. In order to identify the bottlenecks in current ontology-reuse processes, we performed an extended feasibility study comprising both self-conducted case studies and an in-depth review of recent literature in the area. The feasibility study concluded with an analysis and a specification of the requirements for methodologies, methods and tools. It pointed out the fundamental need for mature tools supporting the operation of reuse processes and, more generally, the need for a task- and context-oriented approach to ontology reuse. This includes fine-grained methodologies describing optimal reuse strategies based on additional information about the participants in the reuse process, the ontologies being examined, and the application setting at which the final ontology is targeted. It also includes heuristics for choosing methods and tools for evaluation, customization, and integration which can be feasibly used in these circumstances. The role of humans in the ontology-reuse context, particularly in relation to support tools automatizing specific activities should be rethought. Instead of aiming for full mechanization of knowledge-intensive activities – which are, by their very nature, human-driven – the ontology-engineering community should identify means for optimally combining human and computational intelligence within these tasks.

References

[1] A. Advani, S. Tu, M. Musen, Domain Modeling with Integrated Ontologies: Principles for Reconciliation and Reuse. Technical Report SMI-97-0681, Stanford Medical Informatics, 1998.
[2] J. Arpírez, A. Gómez-Pérez, A. Lozano-Tello, H. Pinto, Reference Ontology and (ONTO)^2 Agent: The Ontology Yellow Pages, Knowledge and Information Systems, vol. 2, 2000, pp. 387–412.
[3] V.R. Benjamins, D. Fensel, S. Decker, A. Gómez-Pérez, (KA)^2: building ontologies for the Internet, International Journal of Human–Computer Studies 51 (1) (1999) 687–712.
[4] A. Bernaras, I. Laresgoiti, J. Corera, Building and reusing ontologies for electrical network applications, in: Proceedings of the 12th European Conference on Artificial Intelligence ECAI, 1996, pp. 298–302.
[5] T. Berners-Lee, J. Hendler, O. Lassila, The semantic Web, Scientific American 284 (5) (2001) 34–43.
[6] C. Bizer, R. Heese, M. Mochol, R. Oldakowski, R. Tolksdorf, R. Eckstein, The impact of semantic web technologies on job recruitment processes, in: Proceedings of the 7th Internationale Tagung Wirtschaftsinformatik WI, 2005, pp. 1367–1382.
[7] M. Bonifacio, P. Bouquet, P. Traverso, Enabling distributed knowledge management: managerial and technological implications, Informatik-Zeitschrift der schweizerischen Informatikorganisationen 1 (2002) 23–29.
[8] D. Brickley, R.V. Guha, RDF Vocabulary Description Language 1.0: RDF Schema, 2004. <http://www.w3.org/TR/rdf-schema/>.
[9] J.d. Bruijn, H. Lausen, A. Polleres, D. Fensel, The web service modeling language: an overview, in: Proceedings of the Thrid European Semantic Web Conference (ESWC2006), 2006, pp. 590–604.
[10] P. Buitelaar, D. Olejnik, M. Sintek, A protégé plug-in for ontology extraction from text based on linguistic analysis, in: ESWS, 2004, pp. 31–44.
[11] A. Burgun, O. Bodenreider, Mapping the UMLS semantic network into general ontologies, in: Proceedings of the AMIA Symposium, 2001.
[12] S.M. Cahn (Ed.), Classics of Western Philosophy, Sixth ed., Hackett Publishing Company, 2002.
[13] I. Cantador, M. Fernandez, P. Castells, Improving ontology recommendation and reuse in WebCORE by collaborative assessments, in: Proceedings of the First International Workshop on Social and Collaborative Construction of Structured Knowledge, Collocated with WWW'07, 2007.
[14] M.A. Capellades, Assessment of reusability of ontologies: a practical example, in: Proceedings of AAAI1999 Workshop on Ontology Management, AAAI Press, 1999, pp. 74–79.
[15] W. Ceusters, B. Smith, J. Flanagan, Ontology and medical terminology: why description logics are not enough, in: Proceedings Towards An Electronic Patient Record TEPR, 2003, CD–ROM.
[16] B. Chandrasekaran, T. Johnson, Generic tasks and task structures: history, critique and new directions, in: Proceedings of the Second Generation Expert Systems, 1993 pp. 232–272.
[17] P. Cimiano, J. Vlker, Text2onto – a framework for ontology learning and data-driven change discovery, 2005. URL: <citeseer.ist.psu.edu/cimiano05textonto.html>.
[18] S. Cohen, L.M. Northrop, Object-oriented technology and domain analysis, in: Proceedings of the Fifth IEEE International Conference on Software Reuse ICSR, 1998, pp. 86–93.
[19] A. Coulet, M. Smanl-Tabbone, A. Napoli, M. Devignes, Suggested ontology for pharmacogenomics (SO-Pharm): modular construction and preliminary testing, in: Proceedings of the OTM Workshops OTM, Springer, 2006, pp. 648–657.
[20] C.V. Damme, M. Hepp, K. Siorpaes, FolksOntology: an integrated approach for turning folksonomies into ontologies, in: Proceedings of the ESWC 2007" Workshop Bridging the Gap between Semantic Web and Web 2.0, 2007.
[21] M. d'Aquin, M. Sabou, M. Dzbor, C. Baldassarre, L. Gridinoc, S. Angeletou, E. Motta, WATSON: a gateway for the semantic web, in: Proceedings of the Fourth European Semantic Web Conference (ESWC), Poster Session, Austria, 2007.
[22] M. d'Aquin, A. Schlicht, H. Stuckenschmidt, M. Sabou, Ontology modularization for knowledge selection: experiments and evaluations, in: Proceedings of the 18th International Conference on Database and Expert Systems Applications DEXA, 2007, pp. 874–883.
[23] L. Ding, T. Finin, Characterizing the semantic web on the web, in: Proceedings of the International Semantic Web Conference ISWC, Springer, 2006, pp. 242–257.
[24] Y. Ding, D. Fensel, Ontology library systems: the key to successful ontology reuse, 2001. URL: <citeseer.ist.psu.edu/ding01ontology.html>.
[25] Y. Ding, D. Lonsdale, D.W. Embley, M. Hepp, L. Xu, Generating ontologies via language components and ontology reuse, in: Proceedings of the 12th International Conference on Applications of Natural Language to Information Systems (NLDB07), Springer, 2007, pp. 131–142.
[26] H. Do, S. Melnik, E. Rahm, Comparison of schema matching evaluations, in: Web, Web-Services, and Database Systems, Springer, 2002, pp. 221–237.
[27] H. Do, E. Rahm, COMA: a system for flexible combination of schema matching approaches, in: Proceedings of the 28th Very Large Data Bases Conference VLDB, 2002, pp. 610–621.
[28] A. Doan, P. Domingos, A. Halevy, Reconciling schemas of disparate data sources: a machine learning approach, in: Proceedings of the ACM SIGMOD Conference, 2001, pp. 509–520.
[29] A. Doan, J. Madhavan, P. Domingos, A. Halevy, Ontology matching: a machine learning approach, Handbook on Ontologies (2004) 385–516.
[30] P. Doran, V. Tamma, L. Iannone, Ontology module extraction for ontology reuse: an ontology engineering perspective, in: Proceedings of the 16th ACM Conference on Information and Knowledge Management CIKM, 2007, pp. 61–70.
[31] D. Fensel, Ontologies: A Silver Bullet for Knowledge Management and Electronic Commerce, Springer, 2001.
[32] M. Fernández-López, A. Gómez-Pérez, Overview and analysis of methodologies for building ontologies, Knowledge Engineering Review 17 (2) (2002) 129–156.
[33] W. Frakes, C. Terry, Software reuse: metrics and models, ACM Computing Surveys 28 (1996) 415–435.
[34] A. Gangemi, D.M. Pisanelli, G. Steve, An overview of the ONIONS project: applying ontologies to the integration of medical terminologies, Data Knowledge Engineering 31 (2) (1999) 183–220.
[35] F. Giunchiglia, P. Shvaiko, Semantic matching, Knowledge Engineering Review 18 (3) (2004) 265–280.
[36] A. Gómez-Pérez, Evaluation of ontologies, International Journal of Intelligent Systems 16 (3) (2001) 391–409.
[37] A. Gómez-Pérez, D. Rojas-Amaya, Ontological reengineering for reuse, in: Proceedings of the 11th European Knowledge Acquisition Workshop EKAW, 1999, pp. 139–157.
[38] T.R. Gruber, Toward principles for the design of ontologies used for knowledge sharing, International Journal of Human–Computer Studies 43 (5/6) (1995) 907–928.
[39] N. Guarino, Formal ontology and information systems, in: Proceedings of the First International Conference on Formal Ontologies in Information Systems FOIS, IOS-Press, 1998, pp. 3–15.
[40] N. Guarino, P. Giaretta, Ontologies and knowledge bases: towards a terminological clarification, Toward Very Large Knowledge Bases, IOS Press, 1995. pp. 25–32.
[41] U. Hahn, M. Romacker, K. Schnattinger, Automatic knowledge acquisition from medical text, in: Proceedings of the 1996 American Medical Informatics Association Annual Symposium AMIA, 1996, pp. 383–387.
[42] M. Heidegger, Ontologie. Hermeneutik der Faktizität, Frnhe Freiburger Vorlesung Sommersemester 1923, Klostermann, 1988.
[43] M. Hepp, Possible ontologies: how reality constrains the development of relevant ontologies, IEEE Internet Computing 11 (1) (2007) 90–96.
[44] M. Jarrar, R. Meersman, Scalability and knowledge reusability in ontology modeling, in: Proceedings of the International Conference on Infrastructure for e-Business, e-Education, e-Science, and e-Medicine SSGRR, 2002.
[45] R. Klamma, M. Spaniol, D. Renzel, Community-aware semantic multimedia tagging from folksonomies to commsonomies, in: Proceedings of I-Media'07, International Conference on New Media Technology and Semantic Systems, Journal of Universal Computer Science, 2007, pp. 163–171.
[46] M. Klein, A. Kiryakov, D. Ognyanov, D. Fensel, Ontology versioning and change detection on the web, in: Proceedings of the 13th International Conference on Knowledge Engineering and Management EKAW, 2002, pp. 197–212.

[47] I. Laresgoiti, A. Anjewierden, A. Bernaras, J. Corera, T.S.A., B.J., Wielinga, Ontologies as vehicles for reuse: a mini-experiment, in: Proceedings of the 10th Banff Knowledge Acquisition for Knowledge-Based Systems Workshop KAW, 1996, pp. 1–21.
[48] A. Lozano-Tello, A. Gómez-Pérez, Ontometric: a method to choose the appropriate ontology, Journal of Database Management 15 (2) (2004) 1–18.
[49] J. Madhavan, P.A. Bernstein, E. Rahm, Generic schema matching with cupid, in: Proceedings of the 27th International Conference on Very Large Data Bases VLDB, 2001, pp. 49–58.
[50] D.L. McGuinness, R. Fikes, J. Rive, S. Wilder, The chimaera ontology environment, in: Proceedings of the 17th International National Conference on Artificial Intelligence AAAI, 2000, pp. 1123–1124.
[51] S. Melnik, H. Garcia-Molina, E. Rahm, Similarity-flooding: a versatile graph matching algorithm, in: Proceedings of the 18th International Conference on Data Engineering ICDE, IEEE Computer Society, 2002, pp. 117–128.
[52] R. Neches, R.E. Fikes, T. Finin, T.R. Gruber, T. Senator, W.R. Swartout, Enabling technology for knowledge sharing, AI Magazine 12 (3) (1991) 35–56.
[53] N.F. Noy, M.A. Musen, ROMPT: algorithm and tool for automated ontology merging and alignment, in: Proceedings of the 17th International National Conference on Artificial Intelligence AAAI, 2000, pp. 450–455.
[54] OntoWeb European Project, Successful scenarios for ontology-based applications, Deliverable D2.2 OntoWeb EU-IST-2001-29243, 2002.
[55] E. Paslaru-Bontas, Practical experiences in building ontology-based retrieval systems, in: Proceedings of the First International ISWC Workshop on Semantic Web Case Studies and Best Practices for eBusiness SWCASE, 2005.
[56] E. Paslaru-Bontas, D. Schlangen, T. Schrader, Creating ontologies for content representation – the ontoseed suite, in: Proceedings of the Fourth International Conference on Ontologies, Databases, and Applications of Semantics ODBASE, 2005, pp. 1296–1313.
[57] P.F. Patel-Schneider, P. Hayes, I. Horrocks, Owl Web Ontology Language Semantics and Abstract Syntax, 2004. <http://www.w3.org/TR/owl-absyn/>.
[58] A. Pease, I. Niles, J. Li, The suggested upper merged ontology: a large ontology for the semantic web and its applications, in: Working Notes of the AAAI-2002 Workshop on Ontologies and the Semantic Web, 2002.
[59] D.N. Peralta, H.S.A.N.P. Pinto, N.J. Mamede, Reusing a time ontology, in: Proceedings of the First International Conference on Enterprise Information Systems ICEIS, 2003, pp. 121–128.
[60] H.S. Pinto, J.P. Martins, A methodology for ontology integration, in: Proceedings of the International Conference on Knowledge Capture K-CAP, ACM Press, 2001, pp. 131–138.
[61] H.S. Pinto, S. Staab, C. Tempich, DILIGENT: towards a fine-grained methodology for distributed, loosely-controlled and evolving engineering of oNTologies, in: Proceedings of the European Conference of Artificial Intelligence ECAI, 2004, pp. 393–397.
[62] D. Pisanelli, A. Gangemi, G. Steve, Ontological analysis of the UMLS metathesaurus, JAMIA 5 (1998) 810–814.
[63] J. Poole, J. Campbell, A novel algorithm for matching conceptual and related graphs, Conceptual Structures: Applications, Implementation and Theory (1995) 293–307.
[64] T. Rattenbury, N. Good, M. Naaman, Towards automatic extraction of event and place semantics from Flickr tags, in: Proceedings of the 30th Annual International ACM SIGIR Conference, SIGIR 07, 2007, pp. 103–110.
[65] T. Russ, A. Valente, R. MacGregor, W. Swartout, Practical experiences in trading off ontology usability and reusability, in: Proceedings of the 12th Workshop on Knowledge Acquisition, Modeling and Management KAW, 1999.
[66] D. Schlangen, M. Stede, E. Paslaru-Bontas, Feeding OWL: extracting and representing the content of pathology reports, in: Proceedings of the NLPXML, 2004.
[67] S. Schulze-Kremer, B. Smith, A. Kumar, Revising the UMLS semantic network, in: Proceedings of the Medinfo, 2004.
[68] J. Seidenberg, A. Rector, Web ontology segmentation: analysis, classification and use, in: Proceedings of the 15th International Conference on World Wide Web (WWW), 2006, pp. 13–22.
[69] J.F. Sowa, Knowledge Representation: Logical, Philosophical, and Computational Foundations, Brooks Cole Publishing Co., 2000.
[70] S. Staab, R. Studer (Eds.), Handbook on Ontologies. International Handbooks on Information Systems, Springer-Verlag, 2004.
[71] L. Stojanovic, A. Maedche, B. Motik, N. Stojanovic, User-driven ontology evolution management, in: Proceedings of the 13th European Conference on Knowledge Engineering and Management EKAW, 2002, pp. 285–300.
[72] G. Stumme, A. Maedche, FCA-merge: bottom-up merging of ontologies, in: Proceedings of the 17th International Joint Conference on Artificial Intelligence IJCAI, 2001, pp. 225–230.
[73] H. Suguri, E. Kodama, M. Miyazaki, H. Nunokawa, S. Noguchi, Implementation of FIPA ontology service, in: Proceedings of the Workshop on Ontologies in Agent Systems at the Fifth International Conference on Autonomous Agents, 2001.
[74] Y. Sure, C. Tempich, D. Vrandecic, Ontology engineering methodologies, in: J. Davies, R. Studer, P. Warren (Eds.), Semantic Web Technologies: Trends and Research in Ontology-based Systems, Wiley, UK, 2006 (Chapter 9).
[75] R. Tolksdorf, E. Paslaru-Bontas, Organizing knowledge in a semantic web for pathology, in: Proceedings of the NetObjectDays, 2004, pp. 39–54.
[76] UMLS, Unified Medical Language System, 2002. <http://www.nlm.nih.gov/research/umls>.
[77] M. Uschold, M. Healy, K. Williamson, P. Clark, S. Woods, Ontology reuse and application, in: Proceedings of the First International Conference on Formal Ontology and Information Systems – FOIS, 1998, pp. 179–192.
[78] R.K. Yin, D.T. Campbell, Case study research: design and methods, Applied Social Research Methods Series, vol. 5, Sage Publications Inc., 2003.
[79] Y. Zhang, W. Vasconcelos, D. Sleeman, Ontosearch: an ontology search engine, in: Proceedings of the 24th SGAI International Conference Innovative Techniques and Applications of AI, 2004.
[80] Y. Zhao, Develop the ontology for internet commerce by reusing existing standards, in: Proceedings of the International Workshop on Semantic Web Foundations and Application Technologies SWFAT, 2003, pp. 51–57.
[81] Y. Zhao, J. Lvdahl, A reuse-based method of developing the ontology for eprocurement, in: Proceedings of the Nordic Conference on Web Services NCWS, 2003, pp. 101–112.

Dr. Elena Simperl is currently working as a senior researcher at the Semantic Technology Institute STI Innsbruck at the University of Innsbruck, Austria. Starting from May, 2007 she was appointed to the position of vice director of the institute. Elena holds a Ph.D. in Computer Science from the Free University of Berlin and a Diploma in Computer Science from the Technical University of Munich. She has held positions as a research assistant at the Technical University of Munich (2002–2003) and the Free University of Berlin (2003–2007) before joining STI Innsbruck early 2007.

Her primary domain of research is Knowledge Engineering. In particular she is interested in user- and business-oriented aspects of ontology building and management, and in methods and paradigms for facilitating and encouraging knowledge sharing and reuse, and approached these topics in several European and national projects. Among the projects she was or is involved in are Knowledge Web (EU FP6 Network of Excellence), TripCom (EU FP6 STREP), Salero (EU FP6 IP), LarKC (EU FP7 IP), Active (EU FP7 IP), SOA4All (EU FP7 IP), Service Web 3.0 (EU FP7 SA), INSEMTIVES (EU FP7 STREP), acting as a scientific coordinator in TripCom, Service Web 3.0, and INSEMTIVES, as activity leader in SOA4All, and as project manager of Knowledge Web.

She published around 60 papers and organized various scientific workshops addressing the aforementioned research topics. She initiated several activities targeted at the supervision and guidance of doctoral students and young researchers such as the PhD Network Berlin Brandenburg and the Knowledge Web PhD Symposium series organized at the European Semantic Web Conference ESWC since 2006. Further educational activities include the lecture of master and bachelor courses at the Free University of Berlin and University of Innsbruck, the organization of the Asian Semantic Web School, as well as the management of the education service within Semantic Technologies Institute International STI International, an association bringing together many of the major players in the fields of semantic technologies.