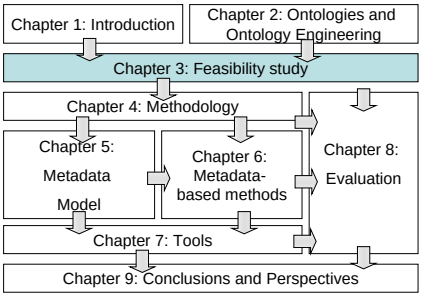
3 Ontology Reuse Feasibility Study



In this chapter we report on the operation and the results of our ontology reuse feasibility study. We start with a motivational discussion on the practicability of ontology reuse in the novel context of the Semantic Web in Section 3.1. In order to identify the factors which have a major impact on the success of a reuse-oriented ontology engineering we conducted in-depth case study research complemented by a detailed literature survey on this topic. These investigations are elaborated in Sections 3.2 to 3.4. The case studies eHealth and eRecruitment are used for exemplification and evaluation purposes in the remaining of this thesis as well. The conclusions of the feasibility study are distilled into a requirements specification for a more efficient and effective Semantic Web ontology reuse from a methodological and technological perspective (Section 3.5). We conclude with a summary of the content of the chapter in Section 3.6.
***References**: This chapter is based on the publications [151, 169, 173, 174, 175, 191, 218].*

3.1 Knowledge Reusability on the Semantic Web

Paraphrasing the general understanding of reuse in adjacent engineering disciplines (cf. [68]) *ontology reuse* can be defined as *the process in which existing ontological knowledge is used as input to generate new ontologies*. The ability of efficiently and effectively performing reuse is commonly acknowledged to play a crucial role in the large scale dissemination of ontologies and ontology-driven technologies, being thus a pre-requisite for the ongoing realization of the Semantic Web.

Firstly, being reusable is an intrinsic property of ontologies, originally defined as means for *"knowledge sharing and reuse"* [154](cf. Chapter 2 for a discussion on this subject). Sharing and reusing existing ontologies increase the quality of the applications using them, as these applications become interoperable and are provided with a deeper, machine-processable and commonly agreed understanding of the underlying domain of interest. Secondly, analogously to other engineering disciplines, reusing existing ontologies, if performed in an efficient way, reduces the costs related to ontology development, because it avoids the re-implementation 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 contributes to an enhancement of the quality of the ontological content, which is by reuse continuously revised by various parties [3, 25, 32, 69].

Still marginally explored in the ontology engineering area, knowledge reusability has a long-standing tradition in a multitude of computer science disciplines, the most prominent of which being probably software engineering and knowledge-based systems [35, 48, 65, 68, 71, 118, 145]. In order to resolve the problematic trade-off between the usefulness in specific application settings and a wide-scale usability, several reusability principles have emerged across all research communities concerned with this subject: *modularity* and *decomposition* along abstraction and functionality levels, the *deployment of standardized tools and technologies*, as well as a careful *documentation* of the development process and up-to-date *component repositories* [29, 77, 117, 118, 145]. While these are all well-accepted means to operationalize reuse activities, building reusable components still remains an art more than a science, as it depends to a similar extent on a set of *non-technical* factors such as the willingness of the engineers to share their knowledge or the level of experience of the engineering team [14]. These are definitely two of the main reasons why, despite its popularity, reuse encountered numerous impediments both in the software and in the knowledge-base area.

While building reusable knowledge bases has mainly failed because of the lack of a practicable maintenance methodology [96], reusing software has proved to be successful in fields such as graphical user interfaces or object-oriented component libraries. Despite of undeniable organizational, economical or psychological factors, which impede *systematic* software reuse [67], significant progresses have been achieved through the emergence of component- and framework-based middleware and the enforcement of object-oriented analysis and design methods [71, 77, 82, 145, 207]. Further on, the work invested by the software engineering community in building reusable software libraries was fundamental for the crystallization of a general understanding of the most reuse-friendly programming languages, programming components and classification systems. Besides specifying such theoretical principles, these experiences contributed to the design and development of methods, techniques and tools to improve the information space of software reuse processes, in this case per default performed in the static context of some component repository [29, 37, 77, 109, 129, 197].

Reusing Semantic Web ontologies has to cope with many of the problems encountered since ages 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 Semantic Web context does, however, present some particularities, which under 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* aimed at being deployed and extended in a variety of contexts to mediate communication between people and applications. Without being reused ontologies are restricted to classical knowledge bases, thus not being able to contribute to the alleviation of the Semantic Web that requires commonly agreed ontologies to mediate between services accessing semantically annotated information spaces. On the other hand, easily finding ontological resources—one of the core requirements for reuse—could be considerably simplified 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 by the enforcement of

Web-suitable knowledge representation languages such as RDF(S), OWL and SWRL. Some elaborated methodologies covering major stages of the ontology life cycle complemented by a plethora of ontology management tools (cf. [162] for an overview) constitute a solid inventory for developing a Semantic Web-suitable ontology reuse platform. Finally, by 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 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 timely recognized 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, both at schema and data level [110, 181, 208].

These factors let us assume that reusing knowledge in Semantic Web context can take benefit from the open, ubiquitous nature of the Web, from the emergence of standards for machine-processable knowledge representation, and from the strong methodological and technological ontology engineering infrastructure in order to alleviate some of the major obstacles impeding reuse in related engineering disciplines so far. However, in order to tap the full potential of these favorable circumstances, we see a clear need to revise and refine current ontology reuse solutions towards a new level of feasibility.

3.2 Case Studies in Ontology Reuse

In this section we give an overview of the most prominent case studies in ontology reuse, which have been published in the knowledge/ontology engineering literature from the early nineties to now. 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* accompanying the ontology engineering team during the case study.

3.2.1 Gómez-Pérez and Rojas-Amaya's Case Study

Gómez-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 [81]. The reuse process clearly focuses on a method for ontology re-engineering, 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 re-engineering methodology proposed by the authors consists of three steps:

1. **Reverse engineering**: on the basis of the code of the source ontology (i.e. its implementation in a particular representation language) one derives a possible conceptual model. This step is performed iteratively, by extracting models with an increasing complexity: the taxonomic structure, followed by relations between concepts and instances and finally more expressive constructs such as axioms or functions.

2. **Re-structuring**: the objective of this step is to evaluate the correctness of the extracted model, correct the detected errors and refine it in conformity with the requirements of the new application setting.

3. **Forward engineering**: the ontology is re-implemented on the basis of the revised conceptual model.

The reuse process was performed along the following stages:

1. **Select reuse candidates**: ontologies stored on the Ontolingua and the Cyc servers were manually selected and evaluated with respect to their relevance to the target domain and with respect to a series of general-purpose modelling guidelines.

2. **Re-engineering**: relevant ontologies were re-engineered as described above

3. **Merging**: the ontologies were merged to a final product.

The focus of the work is to demonstrate the applicability of the re-engineering 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 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 is restricted to taxonomical ontologies containing a manageable number of at most several hundreds of concepts. Nevertheless, the proposed re-engineering workflow was executed manually with impressive efforts relatively to the small size of the ontologies (i.e. the development of the target ontology took 18 months).

3.2.2 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 [224]. 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.

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 to 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 per se related to the production of executable code. However, this step could be assimilated with an pre-integration evaluation of the target ontology.

5. **Integrate into application**: in this step the reused ontology is embedded into the application environment, in that the two versions of the software tool to be built are merged and transformed to executable code. This step typically corresponds to an ontology merging step (i.e. among reused and/or manually built ontologies) in the general reuse process model (see above).

The case study reported in [224], though not investigating the complete reuse cycle at the same level of detail, reveals several important limitations of present reuse technologies. 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 reuse of ontologies at a large scale. 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." [224]

3.2.3 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 modelling 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 the ones stated by Uschold and Healy. While the reuse of the three ontologies was 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 they 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"* [187]. This point of view is reinforced when analyzing 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 filler 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." [187].

3.2.4 Capellades' Case Study

Capellades aimed at building an application ontology by reusing ontologies available at the Ontolingua Server. The reuse process covered two main stages [31]:

1. **Select candidate ontologies**: the selection step does 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, it includes a detailed report on the evaluation procedure which unsuccessfully attempted to apply existing reusability assessment approaches such as [79, 81]. 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 to the application system.

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

3.2.5 Arpírez at al' Case Study

Arpírez and colleagues give an account for a case study in which the $(KA)^2$ ontology [9] was reused in order to build the Reference ontology, a meta-ontology intended to capture information about ontologies and ontology engineering projects [5]. The activities performed in the case study are not representative for a complete reuse life cycle, covering two phases:

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 modelling 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 (including its conceptual structure) and the extensive knowledge of the ontology engineers with respect to the domain of the ontology.

3.2.6 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 [121]. The experiment was not explicitly performed in 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 setting, they are also aware of the limited applicability of their conclusions in arbitrary scenarios and emphasize the fundamental role the original purpose of the reusable knowledge components play for the success of this challenging process. Furthermore, they state the need for a fine-grained reuse methodology in order to allow a widespread dissemination of ontologies beyond the boundaries of the knowledge engineering community.

3.2.7 Bernaras et al' Case Study

Bernaras and colleagues combine a domain ontology of electrical transmission networks with a task ontology for service recovery planning on the same domain for application purposes [10]. 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 to the target application system. It restricts to experiences in merging the aforementioned ontologies at 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 a reuse-oriented knowledge engineering against other development alternatives. These conclusions are based on the experiences made during the case study, as abstraction and standardization, design principles per default considered 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.

The conclusions of the aforementioned 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 3.3 and 3.4 are dedicated to a detailed description of these studies and the lessons learned during their operation.

3.3 Case Study eHealth

3.3.1 A Semantic Web for Pathology

The project "A Semantic Web for Pathology" aims to build a retrieval system for pathology *based on Semantic Web technologies*.[1] Funded by the German Research Foundation (DFG), it is a collaboration between three institutions in the region Berlin-Brandenburg, Germany:

- **The Humboldt Universität zu Berlin**, represented by researchers at the *"Institute for Pathology"* of the Charité Universitätsmedizin Berlin.[2]
- **The Universität Potsdam**, represented by the working group *"Applied Linguistics"* at the Department of Computer Linguistics.[3]
- **The Freie Universität Berlin**, represented by the working group *"Networked Information Systems"* at the Department of Computer Science.[4]

The results reported in this section have been partially achieved on the basis of the work performed within the project and in collaboration with the aforementioned institutions.

The core of the retrieval system is a knowledge base, consisting of an ontology library of 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. The system distinguishes between two kinds of information items:

- **pathology reports** in textual form, containing the observations of the domain experts (pathologists) with respect to medical cases.
- **digital histological slides**, i.e. digital images obtained through high-quality scanning of glass slides with tissue samples.

Every pathology report is de facto a textual representation of a set of histological slides corresponding to a specific medical case. This close relationship can be used to overcome the drawbacks of common retrieval systems for digital pathology and telepathology[5], which concentrate on image-based retrieval algorithms and ignore corresponding medical reports.

[1] `http://swpatho.ag-nbi.de` last visited in May, 2006
[2] `http://www.charite.de` last visited in May, 2006
[3] `http://ling.uni-potsdam.de/cl/` last visited in May, 2006
[4] `http://www.ag-nbi.de` last visited in May, 2006
[5] The main goal of digital pathology and telepathology is the extended usage of digital images for diagnostic support or educational purposes in anatomical or clinical pathology.

Such analysis algorithms have the essential disadvantage that they operate exclusively on structural—or syntactic— image parameters such as color, texture and basic geometrical forms. They ignore the real content and the actual meaning of the pictures. Medical reports, however, contain much more than that since they are textual representations of the *content* of the slides and are easier to analyze than the original image-based data. They capture *implicitly* the concrete semantics of what the picture graphically represent, for example *"a tumor"* in contrast to *"a blob with the length of 15 mm"* or *"a co-located set of 1000 red colored pixels"*. Moreover, medical reports can be treated even as semantic metadata for the images prepared by an expert with high quality, provided an machine-understandable representation of their content [175, 191, 218].

The deployment of Semantic Web technologies is expected to represent a real added value for pathology information systems [218]:

- **Diagnosis**: the system may be used as diagnosis assistant. Since knowledge is made explicit, it supports new query capabilities for diagnosis tasks: similarity or identity of cases based on semantic rules and medical ontologies, differential diagnosis, semantically precise statistical information about occurrences of certain distinguishing criteria in a diagnosis case etc. The information provided will be very valuable in diagnosis work especially for the under-diagnosed cases, since such situations require a deeper investigation of the problem domain and a very strict control mechanism for the diagnosis quality [47].

- **Teaching**: advanced retrieval capabilities may be used for educational purposes by teaching personnel and students. Currently, enormous amounts of knowledge are lost by being stored in data bases, which are behaving as real data sinks. They can and should be used for teaching, e.g., for case-based medical education. The retrieval and reuse of the stored information is limited to string matching techniques and requires technical know-how of the underlying storage system (e.g., query language, relational schema). Besides, the link between pathology reports and the corresponding digital slides is not available at present.

- **Quality controll**: ensuring the quality of diagnostic decisions can be carried out more efficiently because the system uses axioms and rules to automatically check consistency and validity.

- **Inter-organizational communication**: explicit knowledge can be exchanged with external parties such as other healthcare institutions. The representation within the system is already the transfer format for the information exchange. Semantic Web technologies are by design open for the integration of knowledge that is relative to different ontologies and rules.

The realization of the system was performed in two steps: 1) the construction of a *knowledge base*, and 2) the development of the *semantic representation* of medical reports and digital histological images. The knowledge base—developed by a team of two domain experts, one user, one ontology engineer and one programmer—consists of a library of domain and generic ontologies, formalized using the Semantic Web representation languages OWL and RDF(S) and a set of rules, formalized in RuleML and related languages. The domain

ontologies use basically UMLS as information source and adapt this information to the requirements of our concrete application domain *"lung pathology"*. Generic ontologies capture common sense knowledge useful in knowledge intensive tasks like differential diagnosis (i.e. different medical findings with similar symptoms). The necessity of using this second category of ontologies has been emphasized in several similar projects, which analyzed the quality and usage challenges of UMLS in building knowledge bases [73, 193]. Rules are intended to represent decision processes in diagnosis tasks and are acquired in collaboration with domain experts. The role of the rules is also to extend the expressiveness of the ontological knowledge, by formalizing facts, which are not representable using OWL or RDF(S). The semantic representation of the medical reports is realized by a natural language component, which identifies domain specific phrases using the domain ontology. Every pathology report is stored in the system in OWL. The NLP module uses the knowledge base to associate text expressions to ontology concepts and generates for every pathology report an OWL file containing instances of the recognized concepts.

A complete description of the application is out of the scope of this thesis, 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 [191, 218].

3.3.2 Reusing Medical Ontologies

The medical knowledge base was built upon UMLS, as the most complex medical thesaurus available at that time. UMLS as in the release from 2003 contains over 1,5 million concepts from over 100 medical libraries and is permanently growing. New sources and current versions of already integrated sources are mapped into the UMLS knowledge format. Due to the complexity of the thesaurus and the limitations of Semantic Web tools at the present 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 and
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 one, which is tailored for, and can be efficiently used in that application setting. Despite standards and tools for the mainstream technologies, building concrete Semantic Web applications, their potential and acceptance at a larger scale is still a challenging issue for the Semantic Web research community. Besides, building the knowledge base implies also a subsequent adaptation of the content, performed by domain experts. Therefore, they should be able to evaluate and modify the ontology. Apart from technical drawbacks, very large ontologies can not be used efficiently by humans as well.

Evaluation of the UMLS Ontologies

The straight-forward method to address this issue is to use the UMLS Knowledge Server, which provides the MetamorphoSys tool and an additional API to tailor the thesaurus to spe-

cific application needs. However, both allow mainly syntactic filtering methods (e.g., exclude complete UMLS sources, exclude 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 pre-selection 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 pre-selection phase was to identify libraries, which are definitively irrelevant to our application domain. This approach corresponds to the recommendations in [180], which foresees a two-steps selection process, which starts by eliminating ontological sources which are not relevant to the application scope.

As a result of the pre-selection, approximately 50% of the UMLS libraries were selected as possibly relevant for lung pathology, containing more than 500,000 concepts. Managing an ontology of such dimensions with Semantic Web technologies is related to 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 a retrieval engine. The result of this task was 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 few major libraries (e.g., MeSH, SNOMED),[6] which cover important parts of the complete thesaurus, and therefore contain the most of the concepts in our lexicon. To differentiate among the concepts within the resulted 10 libraries, pathology experts selected 4 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[7] to the four initial concepts. The result was a set of approximately 1,000 concepts describing the anatomy of the lung and lung diseases and served as initial input for the domain ontology.

Customization of the Relevant Sources

The linguistic analysis of the patient report corpus evidenced the content-related limitations of UMLS with respect to the concrete vocabulary of the report archive. Additional pathology-specific concepts, like 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 pre-defined naming conventions (e.g., *"RESECTION OF LUNG*

[6] `http://www.nlm.nih.gov/mesh/meshhome.html`, `http://www.snomed.org` last visited in April, 2004

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

WITH RECONSTRUCTION OF CHEST WALL WITHOUT PROSTHESIS"). This situation has direct consequences to the usability of the reused ontologies in our application setting: 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-then-2"*, *"RESECTION OF LUNG WITH RECONSTRUCTION OF CHEST WALL WITHOUT PROSTHESIS"*, *"Unspecified injury of lung with open wound into thorax"* are unlikely to be relevant to the retrieval of pathology reports. Besides, they should be modelled as concepts with corresponding properties and not directly as a single concept, whose names denote their meaning.
- the absence of concept names in 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 to 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. Especially part-whole and spatial relationships are often encountered in medical findings and were therefore included to the ontology library.

Implementation

After identifying the relevant knowledge sources and the list of concepts which can 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 generates the corresponding OWL constructs was implemented for this purpose using Jena2. The resulting ontologies are published server-side and can be accessed by all components in the system. The UMLS consists of two main parts [221]: the UMLS Semantic Network and the UMLS Metathesaurus. The Semantic Network contains generic medical categories and relationships (approximately 150 concepts, i.e. *"semantic types"* and 50 relations i.e. *"semantic relations"* in the actual version). It is used as a *"meta-level"* 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. E.g., 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 (i.e. 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, e.g., 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 in 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, are formalized as `ancestor` and `descendant` in the OWL ontology. These relations could also be ignored, since their meaning can be inferred from the ontology using a reasoner. Due to the fuzzy definition of the rest of the Metathesaurus relations, they were merged to a single `related_to` property. The connection between relations and relation attributes was also considered in the ontology. 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 case where a relation attribute was missing. Further on, they stored for every concept the list of alternative names together with language specifications as `rdfs:label` and connected every concept to the corresponding UMLS libraries it is contained in. 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 the ontologies were checked for consistency. The analysis of the inferred classification hierarchy pointed out few differences compared to the original UMLS hierarchy. The UMLS contains several problematic modelling decisions which have been often 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 had not delivered an optimal solution to cope with this problem. A possible start point could be the Semantic Network, since every Metathesaurus concept is related to it. Besides, the Semantic Network is supposed to be independent of a particular area in medicine. [194] and [28] describe some of the deficiencies of the Semantic Network at ontological level. [182] analyzes the same issue for the UMLS Metathesaurus. However, these issues have proved 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 [34]. Although translating the UMLS data format to OWL was a straight-forward procedure, the expressivity limitations of the language became clear after a detailed analysis of the semantics of the medical knowledge. Reasoning beyond subsumption hierarchies and an 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 in our system by a semi-automatic approach to ontology-based search: for a given search query the user has the possibility to choose 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.

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 interoper-

ability, since the target ontology is, at least partially, aligned to UMLS, which is used by several medical information systems. However, though containing a huge amount of domain information, the reuse of UMLS and integrated libraries in our application setting was not trivial, due to their often ambiguous modelling decisions and an error- prone integration schema [73, 92, 194]. The task specificity of each UMLS library, the complexity of the complete thesaurus and the heterogeneous coverage degree for specific medicine sub-domains such as ours, *lung pathology*, made a high quality customization for concrete application needs difficult. Besides, most of the available medicine ontologies lacked a representation format which supports sharing and reuse. These aspects will become even more important when information sources do not share the same degree of formality (as in the case of the UMLS libraries, which are mapped to a common data format), but are represented as XML- or database schemas or natural language.

The most challenging tasks carried out in this experiment can be easily identified by examining the distribution of the development efforts. The customization of the source ontologies required over 45% of the time necessary to build the overall target ontology. Further 15% of the engineering time were spent on translating the input representation formalisms to OWL. The reuse oriented approach gave rise to considerable efforts to evaluate and extend the outcomes (approximately 40% of the total engineering time). According to our experiences in this project the benefits of reuse were outweighed by their 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, which 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 alternatively building a second application ontology on the basis of the domain corpus of patient records provided by our healthcare partner [174]. 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. [174] 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 of 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 than the second approach (5 person-months for the UMLS-based ontology with 1,200 concepts vs. 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. Orthogonal to 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 rapidly able to evaluate and refine the ontology.

The lessons learned during carrying out this case study were summarized in [169] 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 emerged in the European project OntoWeb, which focus less on technical aspects, but mainly on *"issues that relate to the business en-*

vironment that affects the deployment, integration and acceptance of the ontology-based application" [161]. 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 licence problems etc.), we elaborated the topics which relate directly to the ontology engineering process and adapted them to the medical domain. This list, illustrated in Table 3.1, could be complemented with modeling best practices, which are equally important in a complex domain such as medicine. Such best practices are currently emerging as a result of the W3C Semantic Web Best Practices and Deployment Working Group.[8]

The guidelines contributed to a great extent to the ontology reuse requirements specification in Section 3.5.

3.4 Case Study eRecruitment

3.4.1 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 for particular application scenarios and market sectors.[9]. It is an integral part of *"InterVal"*—Berlin Research Center for the Internet Economy, funded by the German Ministry of Research BMBF and comprises several academic institutions located in the region Berlin-Brandenburg, Germany:

- **The Humboldt Universität zu Berlin**, represented by the working group *"Databases and Information Systems"* at the Department of Computer Science.[10]
- **The Freie Universität Berlin**, represented by the working group *"Networked Information Systems"* at the Department of Computer Science[11] and the working group *"Produktion, Wirtschaftsinformatik und OR"* at the Department of Economics.[12]

The work reported in this section was realized by the author in collaboration with project participants affiliated at the aforementioned institutions.

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. In the last years, the Web became a fundamental technology for a significant number of recruitment applications, be that job portals, personal service agencies or official employment initiatives. While the advantages of using the Web as a dissemination medium are widely recognized by both job applicants and employing companies, current job search engines are far away from offering job seekers high-quality access to job offer resources. Apart from the fact that a significant number of job offers are still published on proprietary, non-publicly accessible

[8] `http://www.w3.org/2001/sw/BestPractices/` last visited in May, 2006
[9] `http://wissensnetze.ag-nbi.de` last visited in January, 2006
[10] `http://dbis.informatik.hu-berlin.de` last visited in May, 2006
[11] `http://www.ag-nbi.de` last visited May, 2006
[12] `http://wiwiss.fu-berlin.de/suhl/index.htm` last visited in May, 2006

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 should be the same as the one used by the users and in the documents to be annotated • concepts should be denominated using naming conventions and in a linguistically predictable form • modelling decisions should be recorded during the conceptualization phase in order to simplify the ontology-driven 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 is related to 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 fulfil the application requirements: • concepts are denominated in an ad-hoc manner even within the same ontology • the semantics of the concepts is sometimes encoded in their names • most of the medical ontologies are stored in proprietary forms, there are no translation tools • most of the ontologies are modelled 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 evaluation of their application relevance is extremely tedious, if not impossible • the same domain is covered to a similar extent by several ontologies.There are no fundamental differences among them w.r.t. their suitability in the Semantic Web context. Eliminating candidate ontologies which are definitely not relevant is sometimes more feasible than an attempt to 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 the user acceptance. • matching and merging ontologies with overlapping domains imposes serious scalability and performance problems to the tools available at the time. Nevertheless, using simple algorithms (e.g., 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. • reasoning over these models requires scalable inference engines
ONTOLOGY MANAGEMENT	The size of the target ontology requires powerful storage mechanisms with adequate reasoning support (e.g., for automatically checking inconsistencies) Elaborate a detailed evaluation framework to control ontology evolution. The maintenance of large size ontologies requires additional effort for documenting modelling decisions.
UPDATES	Medicine is a dynamic domain, most of the ontologies change within 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 non-standard 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 modelled.

Table 3.1: Guidelines for Reusing Medical Ontologies

company sites, the quality of the results of a job search—performed by using either general-purpose or specialized search heuristics—depends on a great extent on various characteristics of the job descriptions available on the Web, such as form, language and purpose. Further on, the free text representation of these sources considerably restricts the precision and recall of the underlying job search engines, which, in absence of an explicit semantic representation of the content, are restricted to flavors of keyword- and statistics-based algorithms.

The technical setting of the case study, sharing many commonalities with what is called a "typical" job search engine, consists of the following components: a crawler component seeking for domain-relevant information across the Web or on pre-defined company sites, an information extraction component aiming at classifying the acquired information items into specific categories and a search front-end to look for, rank and compare them. The system is dealing with two types of information:

- **job postings/offers** consisting of typical metadata information and an extended job description, and
- **job applications/profiles** describing the capabilities and level of experience of individual job seekers.

Job seekers register to the portal and insert their application profile to a repository, thus being automatically taken into consideration in future recruitment tasks. Job descriptions are matched against incoming job candidates on the basis of a pre-defined schema, while the results of this comparison flow into a ranking function used to present the job search results to the users.

The case study analyzed the possibility of extending the existing job search engine with ontology-based technologies. In doing so, domain-relevant ontologies termed as "human resources/HR ontologies" are aimed at being used as semantic indices, by which job descriptions and applications in the selected sector are classified and matched, thus enhancing the system with semantics-aware search functionalities:

- a fine grained, domain narrow classification of the information items increases the precision of user queries and consequently, of the search heuristics. Further on, by means of the domain ontology the system is provided with additional, explicitly represented domain knowledge, thus being able to semantically rewrite user queries: a search request on, for instance, *"programming languages"* could be in this way automatically extended with more specific concepts subsumed by this category, such as *"object-oriented programming"* or *"Java"* in order to improve the recall of the system. The precision value can be improved through the usage of pre-defined search terms, described by the ontology. As an example, consider a job search specified by the name of particular companies: the ontology can be used to extend the user query with standard company identifiers, thus avoiding ambiguities as those emerging through the usage of slightly different spellings.
- besides its primary role as an index structure for system-relevant information, the ontology could be involved in the methods applied to (semi-)automatically classify this

information by (ontologically) pre-defined dimensions. An ontology-driven information extraction procedure has the advantage that the domain specificity of the classification heuristics is stored separately from the system implementation, which can be easily customized to new domains of interest. Given the explicitly represented domain knowledge, the system can automatically decide on new, domain-relevant information types, which are then extracted from the free text job descriptions.

- a third use case for an ontology-driven job portal is the search and ranking functionality. Information items can be compared using ontology-based similarity measures, which take into account domain-specific matching concept labels or taxonomical structures.

Just as in the eHealth scenario, a complete description of the application underlying the case study is out of the scope of this thesis, which is concerned with ontology engineering aspects (cf. for example [13] for a detailed description of the scenario).

3.4.2 Reusing Human Resources Ontologies

The usage of commonly agreed 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 *by contrast to the medical sector*, major governmental and international organizations strove 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 with national impact, ready to be deployed in job portals to simplify the management of electronically available job postings and job seeker profiles and to encourage application interoperability. Standards such as O*NET (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. In the same time they are valuable knowledge resources for the development of application-specific ontologies, which can 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 on potentially reusable ontological sources.

2. **Evaluation of the ontological sources**: the result of the previous step was analyzed with respect to its domain and application relevance, as well as its general quality and availability.

3. **Customization 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.

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 ontology location support technologies available at present:

General-purpose search engines: we used conventional search tools and pre-defined 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. Discussions with domain experts complemented by Internet research led to the identification of several major players in this field: at national level the Federal Agency of Employment/Bundesagentur für Arbeit, at foreign level the American, Canadian, Australian and Swedish correspondents, and at international level institutions like the United Nations/UN or the HR-Consortium. 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 pre-selected 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.

Evaluation of the Reuse Candidates

The engineering team decided to reuse the following resources:

1. **HR-BA-XML**: which is the official German translation of Human Resources XML, the most widely used standard for process documents like 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 5597 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]

[13] `http://www.hr-xml.org` last visited in May, 2006
[14] `http://www.arbeitsamt.de/hst/markt/news/BKZ\_alpha.txt` last visited in May, 2006
[15] `http://www.bls.gov/soc/` last visited in May, 2006

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 competencies required to describe job position requirements and job applicant skills.[18]

The selection of the 6 sources was performed manually without the usage of a pre-defined methodology or evaluation framework. The documentation of the 24 potential reuse candidates was consulted in order to assess the relevance of the modelled 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 ex post modifications in order to adapt them to the requirements of the tasks they were expected to be involved in at application level. The importance of these application-oriented dependencies has been underestimated by the engineering team at that point. In the absence of an appropriate methodology for this purpose they were not taken into account during the evaluation.

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 re-used the SOC and NAICS classifications. As for occupational classifications in the English language, the SOC system was preferred to alternative like NOC or O*NET due to the availability of an official German translation.[19] The same applies for the choice between industry sector classifications: by contrast to ISIC[20] the NAICS system is provided with a German version, while being used in various applications and classifications in the human resources area.

Customization and Integration of Relevant Sources

The main challenge of the eRecruitment scenario was the adaption of the 6 reusable ontologies to the *technical* requirements of the job portal application. From a content oriented perspective, 5 of the sources were included to 100% to 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 form of an extraction of the relevant

[16] `http://www.destatis.de/allg/d/klassif/wz2003.htm` last visited in May, 2006
[17] `http://www.census.gov/epcd/www/naics.html` last visited in May, 2006
[18] `www.kowien.uni-essen.de/publikationen/konstruktion.pdf` last visited in May, 2006
[19] `http://www23.hrdc-drhc.gc.ca/2001/e/generic/matrix.pdf`, `http://www.onetcenter.org/` both last visited in May, 2006
[20] `http://unstats.un.org/unsd/cr/registry/regcst.asp?Cl=17&Lg=1` last visited in May, 2006

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 varied with respect to the represented domain, the degree of formality and the granularity of the conceptualization. They are labeled using different natural languages and are 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 was 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) can 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.

Conclusions and Lessons Learned

A wide range of standards for process modelling 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. Besides, reusing classification schemas like BKZ, WZ2003 and their English variants, which have been completely integrated into the target ontology, implied significant cost reduction. They guaranteed a comprehensive conceptualization of the corresponding subdomains and saved the costs incurred by 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 pre-selected reuse candidates, the ontology evaluation step did not imply major development costs. This is indicated by the distribution of the efforts in each of the enumerated process stages. Solely 15% of the total engineering time were spent on searching and identifying the relevant sources. Approximately 35% of the overall efforts were 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% of the total engineering costs were necessary to translate these sources to the target representation language OWL. Lastly, the refinement and evaluation process required the remaining 10%. 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. Translating various representation formats to OWL proved to be tedious in the current tool landscape. Though non-trivial, 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 the relatively

PROCESS STEP	GUIDELINES
ONTOLOGY DISCOVERY	Finding an appropriate ontology is currently associated to considerable efforts and is dependent on the level of expertise and intuition of the engineering team. In absence of fully-fledged ontology repositories and mature ontology search engines the following strategies could be helpful: • use conventional search engines with queries containing core concepts of the domain of interest and terms like ontology, classification, taxonomy, controlled vocabulary, glossary. E.g., "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 to 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 the 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 EVALUATION	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 overlapping 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 same domain is covered to a similar extent by several ontologies, while there are no fundamental differences among them w.r.t. their suitability in a semantic job portal. Eliminating candidate ontologies which are definitely not relevant is sometimes more feasible than an attempt to 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 at international institutions such as the EU or the UNO 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 upon the vocabulary of the evaluated sources. • these considerations apply for further application scenarios such as eGovernment and eHealth.
ONTOLOGY MERGING AND INTEGRATION	Existing HR ontologies have a considerable size, but a relatively simple structure. Adapt your integration methodology to their particularities: • matching and merging ontologies with overlapping domains imposes serious scalability and performance problems to the tools available at present. Nevertheless, using simple algorithms (e.g., linguistic and taxonomic 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 avoid the complexity of a monolithic evaluation. • dedicated tools extracting lightweight ontological structures from textual documents or Web sites are required. • the integration step requires means to translate between heterogeneous formats (XML to OWL and RDFS, data base schemas to OWL and RDFS etc.). • the customization of these structures w.r.t. particular domains of interest (e.g., a HR ontology for the chemical domain) causes additional efforts as all HR standards are independent of any industrial sector.

Table 3.2: Guidelines for Building HR Ontologies by Reuse

simple, tree-like structure of the sources. However, we see a clear need for tools which assist the ontology engineer during this kind of tasks on real world, large scale ontologies with many thousands of concepts and more complicated structure. Despite the mentioned problems, 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 notably high amount of manual work, even when using common representation languages like 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 side of the ontology engineers.

Just as for the medical case study, the lessons learned in the eRecruitment scenario were summarized in form of a set of guidelines for ontology reuse which might aid ontology developers in similar situations. These are depicted in Table 3.2.

3.5 Requirements for Ontology Reuse

Typically ontology reuse starts with the selection of a set of knowledge sources adequate for the application setting. Once their usability has been positively evaluated, these ontologies are subject of various technology-driven adaptation and integration activities. In an arbitrary setting the reuse candidates differ with respect to diverse content, implementation and provenance aspects. They might model various application-relevant domains from a multitude of viewpoints or optimize the domain representation to particular scopes and purposes. Furthermore, they do not share a common level of formality or the same implementation language, can be used in accordance to specific licence conditions and might still be under development or at least subject to frequent updates and changes. An automatic integration of the source ontologies means not only the translation of their initial representation languages to a common format, but also the matching and merging of the resulting schemas and the associated data. Our findings during the presented case studies clearly showed that none of the mentioned activities can be performed efficiently due to the incapacity of current methodologies, methods and tools to deal with this diversity to a satisfactory extent. At methodological level, ontology engineers are provided with a minimal inventory of methodologies for ontology reuse, which usually restrict to giving a generic, high-level description of the process, while concentrating on the technical integration or merging of the ontologies involved. Selecting appropriate ontologies is a non-trivial task not only because the lack of flexible and fine-grained evaluation frameworks, but because of the difficulties attested by humans when dealing with the extreme heterogeneity of the assessed resources. Furthermore, the technologies employed for ontology management are inherently targeted at particular classes of ontologies, while their user-friendliness and real-world feasibility is still questionable.

In the following we will take a deeper look at the empirical findings gathered during car-

No	Criterion
C1	Inheritance from knowledge engineering
C2	Detail of the methodology
C3	Recommendations for knowledge formalization
C4	Strategy for building ontologies
C5	Strategy for identifying concepts
C6	Recommended life cycle
C7	Differences to IEEE Software Development
C8	Recommended techniques
C9	Application to current projects

Table 3.3: Criteria for Analyzing and Comparing Ontology Engineering Methodologies cf. [63]

No	Criterion
R1	Level of detail of the methodology
R2	Relation to application scenarios
R3	Recommended life cycle
R4	Support methods and tools
R5	Methodology validation

Table 3.4: General Requirements for Ontology Reuse Methodologies

rying out 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 mostly technical requirements for ontology management, particularly for ontology matching, merging and integration.

3.5.1 Requirements for Ontology Reuse Methodologies

As starting point for the specification of the requirements for the planned reuse methodology we revise the analysis benchmark in [63], which addresses the issue of aligning and evaluating counterpart approaches in the more general field of ontology engineering. The proposed framework consisting of 9 criteria is illustrated in Table 3.3.

We reduced this framework to the ontology reuse scenario, as depicted in Table 3.4. The original criteria $C2$, $C4$, $C6$, $C8$ and $C9$ have been revised in accordance to the particularities of the new setting. The remaining ones, $C1$, $C3$, $C5$ and $C7$ were not relevant in our case, as the planned ontology reuse methodology is intended to be only a part of a more complex ontology engineering framework.

(R1) 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 pre-defines the inputs and outputs of each process phase. Further on, the methodology should avoid recommending activities and tasks whose purpose is intuitively understood by the majority of

	SCOPE OF THE CASE STUDY	ONTOLOGY DISCOVERY	ONTOLOGY EVALUATION	ONTOLOGY CUSTOMIZATION, MERGING AND INTEGRATION
Goméz-Pérez & Rojas-Amaya	Ontology reengineering	-	-	restructuring, forward engineering
Uschold & Healy	Ontology reuse	-	understand	translate, refine, integrate
Russ et al	Ontology reuse	-	-	merge, integrate
Cappelades	Ontology reuse	Ontolingua server	understand, select	prune
Pinto & Martins	Ontology integration	-	understand, select	refine, integrate
Laresgoiti et al	Ontology reuse	-	-	translate, integrate
Bernaras et al	Ontology reuse	-	-	merge
Paslaru et al	Ontology reuse	-	understand, assess usability	prune, translate, merge, integrate
Paslaru & Mochol	Ontology reuse	Web	-	prune, translate, integrate

Table 3.5: Reuse Process as Performed in the Analyzed Case Studies

methodology applicants, but whose implementation is not clearly explained or even debatable in particular classes of application scenarios.

While not covering 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, elaborated description of the ontology reuse process. Table 3.5 gives 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 five-staged ontology reuse workflow as follows:

1. **Finding 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, this technical step is to date based on the level of experience and intuition of the process participants, as no established ontology location tools are available.

2. **Selecting the ones to be reused**: some of the case studies argued on the difficulties related to this step, proposing high-level activities like understanding the ontology, checking whether the ontology is application-relevant, proof-reading the ontology etc.

3. **Customization**: once the set of reusable ontologies has been determined, they are translated to a new representation language, extended or simplified/pruned.

4. **Merging**: ontologies covering similar domains are merged to one.

5. **Integration**: ontologies modelling different domains are integrated into a final application ontology and to the application system.

(R2) Relation to Application Scenarios

As repeatedly mentioned in the literature survey and in accordance to our own observations, the prospected reuse methodology should pay particular attention to application-narrow as-

pects and to feasible support methods and tools in order to enhance its large-scale 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, mediation, impose particular constraints on the properties of the ontologies to be reused or constructed.

Further on, the structure and execution of the reuse process allow us differentiate between several *"levels of reuse"*. These describe to which extent which parts of the reused sources are re-utilized in the new setting and which actions are necessary for this purpose.

A third application-narrow dimension having an impact on the way a specific reuse process is performed are the methodology applicants. In order to ensure a wide-scale dissemination of semantic technologies, it is essential that the proposed techniques minimize the amount of expert knowledge required for their operation. Hence it is required that the methodology accounts for this required low barrier of entry and adapts its content and structure to the needs of its users.

(R3) 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 on the way reuse is being executed. In situations in which a particular activity can not 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.

(R4) Support methods and Tools

Helpful for the usability of the proposed methodology are precise tool and method recommendations. In every of the examined case studies we found strong evidence that the lack of appropriate techniques and tools for supporting resource-intensive and error-prone reuse activities was a determinant factor for the feasibility of a reuse-oriented engineering strategy. Therefore, the methodology should be accompanied by (a description of) a fully-fledged methodology-compatible technological environment.

(R5) Methodology validation

At 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 (e.g., [5, 81]) the case studies do not explicitly commit to a pre-defined methodology.

In order to increase its usability, the prospected methodology should be carefully validated in real-world scenarios. A first validation method could rely on the criteria introduced above

No	Criterion
R6.1	Semantically enabled ontology metadata
R6.2	Ontology repositories
R6.2.1	Ontology-based search
R6.2.2	Browsing and navigation
R6.2.3	User rating methods for ontologies
R6.2.4	Attestation methods for ontologies
R6.3	Dedicated crawlers

Table 3.6: Requirements for Ontology Discovery Support Methods and Tools

(cf. Table 3.4) associated with a set of quantified metrics for generating comparable evaluation results. Secondly, engineering methodologies are typically validated using case study research. We will address these issues in more detail in Chapter 8.

3.5.2 Requirements for Ontology Reuse Support Methods and Tools

At method and tool level the conclusions of the case studies focus on the incapacity of the existing technological framework to feasibly deal with scalability and heterogeneity issues. In the remaining of this section we will specify requirements for novel approaches which are meant to provide support in the main phases of the reuse process: the discovery of reuse candidates, the assessment of their application relevance, and their integration into the new application context.

(R6)Requirements for Ontology Discovery

At present the question of how existing ontologies can be found is not trivial. On 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 modelled and attempt to seek if they have made any ontologies/classifications public. If we consider the eRecruitment scenario, the relevant sub-domains 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 precision. On the other hand, a search for taxonomies, classifications and classification systems (i.e. using Google queries such as *"human resources taxonomy"*) has proved to deliver significantly better results, leading to a list of organizations and standards in the aforementioned domains, which, however, necessitated a careful evaluation and customization because of the heterogeneity of their formal and content-related characteristics.

Alternatively, ontologies could be grouped into repositories (cf. Chapter 2). As long as

[21] `http://www.google.com` last visited in May, 2006
[22] `http://swoogle.umbc.edu/` last visited in May, 2006

No	Criterion
R7.1	Semantically enabled ontology metadata
R7.2	User rating methods for ontologies
R7.3	Attestation methods for ontologies
R7.4	View-enabled ontology editors
R7.5	Ontology visualization tools
R7.6	Textual descriptions of ontological content
R7.7	Query engines and reasoners
R7.8	Ontology matching and alignment tools

Table 3.7: Requirements for Ontology Evaluation Support Methods and Tools

the developer knew how to access the repository he or she could seek 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 by different criteria e.g., URI, keyword, submitting organization or express queries in terms of ontology classes and properties.[23] Ontology repositories appear to be a useful means to provide 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 non-semantic search engines such as Google. Queries can not draw on the semantics of ontologies themselves in order to be able to find e.g., generalizations, specializations or equivalences of search terms/concepts . Finally, the repositories link to the complete ontologies from their descriptions, meaning that access is on an "all or nothing" basis, not taking into account the various needs of individual users.

To summarize the analysis of the present state of the art resulted in the formulation of the following requirements for a feasible ontology discovery in Table 3.6.[24]

A fully-fledged ontology repository is expected to provide two types of features, related to general information repositories and to the semantically represented information, respectively (cf. [68]). The first category implies issues like the quality of its content (coverage, actuality, and user-perceived information value) and typical services (classification, search, as well as browsing and navigation). The special nature of the managed information leads to requirements related to semantic aspects of these general-purpose features.

(R7)Requirements for Ontology Evaluation

Assessing the usability of a certain ontology with respect to a set of application-close requirements is one of the most challenging tasks in ontology reuse[31, 180]. While the task is primarily targeted at humans, its efficient operation is still dependant upon technical means to simplify the access of the process participants to the ontologies to be evaluated, which might be complex, large or hardly human-readable. This implies a whole series of computer-aided techniques to interact with an ontology and to align comparable ones to one another, as depicted in Table 3.7.

[23] `www.daml.org/ontologies/` last visited in February, 2006

[24] We build upon the enumeration introduced in the previous section.

Firstly, ontology evaluators can take benefit from the availability of a uniform representation of ontology descriptions. Reliable rating and attestation methods might provide additional control of the sometimes highly subjective selection process. Finally, multi-modal tools for visualizing, editing and queering the content of the ontologies involved are fundamental for dealing with large amounts of information. Tools comparing among the contents and the structure of various resources further aid the ontology evaluators in deciding upon their fitness of use in the new setting.

(R8)Requirements for Ontology Customization, Merging and Integration

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

- translation to a new representation language
- extraction of a sub-ontology
- modification and extension in contents, structure or both.

Multiple ontologies are merged or integrated. This process can be seen as a sequence of two phases, the computation of the similar ontological elements (usually termed to *"matching"*) followed by their aggregation. Every of the analyzed case studies, some of which not included in the present feasibility study, points out the difficulties arisen by the cumbersome technique and tool utilization. Even if appropriate means were eventually available, setting up the technological environment to perform one of the aforementioned 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 under extremely special circumstances.

Despite of 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 often emphasized in recent literature [76, 130, 143]:

- some approaches assume a common or, at least to large extent, overlapping universe of discourse [41].
- they can not be applied across various domains with the same effect (for example, Cupid as stated in [76]).
- they require certain representation (or translation to the suitable format) or natural languages (e.g., the COMA approach [51]).
- 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.

No	Criterion
R8.1	Translation tools
R8.2	Information extraction tools
R8.3	View-enabled ontology editors
R8.4	Ontology matching tools
R8.5	Ontology merging and integration tools

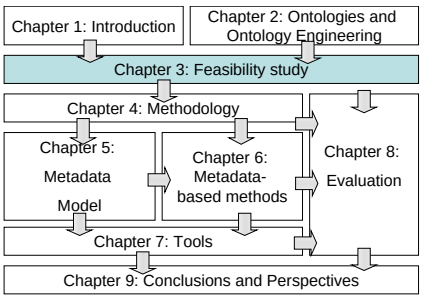
Table 3.8: Requirements for Ontology Customization, Merging and Integration Support Methods and Tools

- they do not perform well on inputs with heterogeneous (graph) structures (e.g., Cupid [76]) or are restricted to tree-based conceptual models (SimilarityFlooding [143], S-Match [76]).
- the results are based on a one-to-one mapping between taxonomies (such as in GLUE [52]), or
- they involve some manual pre-processing (like in GLUE, COMA [50]).

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

Analogously to the previous sections, we conclude with a compact list of the derived requirements in Table 3.8.

3.6 Summary



The main objective of this chapter was to give a detailed overview of the present state of the art in the area of ontology reuse. After reviewing some of the most representative empirical studies aiming at using or reusing existing ontological content, we completed the feasibility study with two extended ontology engineering experiments in the domains of medicine and human resources. The results of our investigations are summarized in terms of an inventory of requirements for methodological and technological support for ontology reuse, which form the basis of our approach in the next chapters.