

Automating Competence Management through non-standard Reasoning

S. Colucci^{a,*}, E. Tinelli^b, E. Di Sciascio^c, F.M. Donini^a

^a*Università della Tuscia, Via S. Carlo 32, 01100 Viterbo, Italy*

^b*Dipartimento di Informatica, Università di Bari, Via Orabona 4, 70125 Bari, Italy*

^c*Politecnico di Bari, Via Orabona 4, 70125 Bari, Italy*

Abstract

Competence Management calls for automated knowledge-based services in order to take full advantage from the know-how of a company. This paper presents an integrated semantic-based knowledge management system providing decision support services for several activities typical of competence management, including Core Competence evaluation, human resources allocation, training programs planning. Knowledge resources are represented according to the formalism of Description Logics, which also allows for inference services crucial for the implemented solutions. We present the various features of our approach, which exploit advanced non-standard reasoning services from Description Logics, specifically developed to support knowledge management.

Keywords: Competence Management, Core Competence, Ontologies, Description Logics, Non-standard Reasoning

1. Introduction

It is well-known that knowledge intensive companies have their most promising source of competitive advantage in human resources. Such an assumption is at the basis of the widely investigated research area well known in the literature by the name *Knowledge Management* (KM).

A variety of methodologies have been proposed for an efficient use of the so-called intellectual capital, whose impact on return on investments is nowadays considered as valuable as the one deriving from any other material asset.

Emphasis has been given, in particular, to the identification of capabilities leading companies to business success: several approaches to strategic management have been proposed and classified according to the perspective they take on the problem [1].

*Principal Corresponding Author

Email addresses: simona.colucci@unitus.it (S. Colucci), tinelli@di.uniba.it (E. Tinelli), disciascio@poliba.it (E. Di Sciascio), donini@unitus.it (F.M. Donini)

In particular, many research contributions sustain the *resource-based theory* of the firm ([2], [3]) suggesting to search for competitive advantage in unique company capabilities ([4], [5], [6], [7]).

Other proposals focus on the achievement of competitive advantage through the deployment and exploitation of capabilities embedded in business processes: such a *dynamic capabilities approach* asks for continuous reshaping of firms assets [8].

Alternative approaches ([9], [10], [11]) take the so-called *competence-based perspective*, which identifies in the *Core Competence* of the firm as a whole a source of competitive advantage more crucial than the discrete assets of the firm itself. In [12] the notion of Core Competence was firstly defined as a sort of capability providing customer benefits, hard to be imitated from competitors and possessing leverage potential. Further definitions of Core Competence have been proposed in the literature in the attempt of finding methods for detecting such a collective knowledge ([13], [14]).

Our proposal takes the competence-based perspective and in particular shares with it the interpretation of company strategic competence as a collective asset, resulting from the synergy of human resources. In particular, we adopt a recently proposed semantic-based approach [15] supporting long term organizational strategy in the identification of such knowledge, mentioned above as Core Competence.

The main focus of the paper is to present and evaluate the implementation of such an approach in a Knowledge Management System (KMS) [16] exposing also services for human resources assignment and training programs composition.

The system treats company personnel as a “competence warehouse” available for the production of goods and services to be placed on the market. As with any other asset, knowledge needs to be scheduled with respect to production plans, and acquired from external sources when the figurative warehouse runs out of stock.

When dealing with automatic management of knowledge – rather than material assets – the peculiarities of the involved resource may cause several difficulties. The most significant drawback is due to the subjectivity and intangibility typical of knowledge: the description of personnel competence needs to take implicit and negative information into account and to unambiguously interpret the domain vocabulary.

Using ontologies and formal languages endowed with semantics allows automatic systems to overcome limits peculiar to less expressive sorts of knowledge representation. Moreover, the choice of suitable formal languages allows to define and adopt reasoning services exploiting explicit information description to infer new knowledge. Though standard deductive services over logic-based descriptions, such as subsumption and satisfiability, can already infer informative content crucial for solutions to human resources management, specific non-standard inferences can be devised that are characterized by a higher potential impact, as we show in this paper.

Among available languages for knowledge representation, we adopt Description Logics (DLs) [17], which offer a formalism to create representations whose

expressiveness grows with the cardinality of adopted formalisms set. A trade-off is of course required between expressiveness of descriptions and computational complexity of provided reasoning services. Thus, we give up to the use of formal languages providing full expressiveness of descriptions, because the proposed KMS implements specifically developed non-standard inferences in DLs to provide sophisticated knowledge management services.

The remaining of this article is organized as follows: the logical architecture of the integrated KMS is introduced in Section 2. Then the logical framework supporting Core Competence identification is presented. KMS usage is then shown with reference to all possible use cases in Section 4. A performance evaluation focused on Core Competence computations is reported in Section 5. Related work and conclusions close the paper. The interested reader can find in Appendix A an overview of basic formalism and reasoning services we exploit here.

2. An integrated knowledge management system

The integrated KMS we designed and implemented basically aims at helping management in identifying strategies for an efficient human resources allocation and management.

Figure 1 depicts the system logical architecture, made up by three categories of decision support services, which are detailed in the following sections.

2.1. *Strategical Choices*

Imagine a company in which personnel profiles have been stored as formal descriptions in an ontology: management has the whole “knowledge inventory” at hand for possible strategical choices. In particular, the knowledge representation effort may be exploited for the identification of the collective competence characterizing a company, the fields of excellence to invest on in long term strategy: Core Competence. The KMS allows for automatically extracting possible sources of collective knowledge from personnel profile descriptions or, given some target Core Competence, for evaluating whether the company possesses it (*Core Competence Identification*). The resulting fields of excellence may be exploited to characterize companies in term of skill excellence cluster, so that a Core Competence-based classification process for companies may be performed (*Cluster Identification*) [18]. Different clusters include companies sharing formalized classes of knowledge. Through the proposed process a company may discover its profile in terms of Core Competence and check whether it can join a given knowledge class or, in case it cannot, the reasons why it happens. In both cases, investigations aimed at strategical choices may lead to the determination of a knowledge gap, which may be source of suggestions for possible training programs.

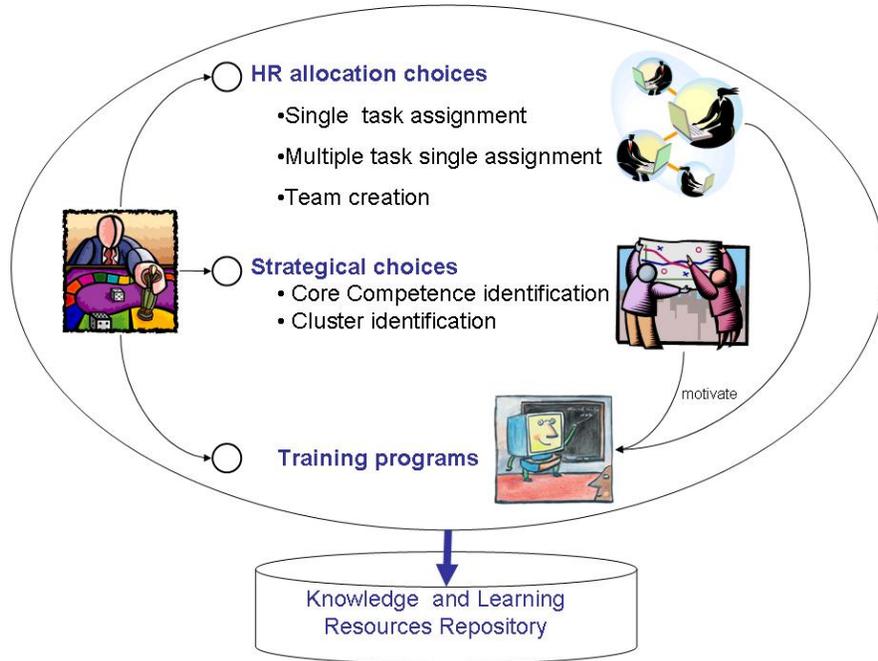


Figure 1: Functional architecture of the KMS

2.2. HR Allocation Choices

Organizational life is characterized by several tasks and activities to be performed by employees enabled with the needed competence. The assignment of the right individual to the right job is crucial for business success, even though very hard to achieve in an efficient fashion. An automated support to such a critical activity may help management in overcoming the danger of wrong assignments due to incomplete information or subjective choice criteria. If personnel profiles are stored as formal descriptions in an ontological repository, it is possible to provide automatic allocation proposals based on the semantic similarity between the profile and the needed task, which requires to be represented as a formal description according to the same ontology. Of course a perfect match between individual and task description, although wished, is quite rare to happen: also personnel lacking of some competence to completely fulfill the task may represent good assignment candidates if an explanation about missing skills is provided and is judged acceptable. In some cases, instead, the request is such that no profile can potentially match it: only personnel with features conflicting with the required ones are available. Such an unlucky situation may be supported by suggesting conflicting characteristics to be retracted in the request, in a possible negotiation process. An extended matchmaking process allows for evaluating also missing and conflicting information in profile descriptions to determine the similarity degree of task and profile description.

The approach exploits Concept Abduction and Concept Contraction reasoning services [16]. In particular, missing information may be interpreted as possible suggestions for training programs, enabling the selected individual with the missing knowledge. Furthermore, not all tasks are solvable by only one individual and sometimes allocations affect each other: if more than one task at a time needs to be assigned, criteria for overall optimization need to be adopted. Three different allocation services, implementing the approach originally proposed in [16], are therefore provided: the assignment of one individual to one task (*Single Task Assignment*), the contemporary choice of more than one single assignments (*Multiple Task Single Assignment*), the creation of an ad-hoc team (*Team Creation*). Most of HR Allocation services have been implemented in the framework of I.M.P.A.K.T., a novel and optimized commercial system for competences and skills management [19].

2.3. Training Programs

The shared repository for knowledge resources may be enriched with learning resources descriptions referred to the same ontology as profile descriptions: the provided vocabulary details in fact competence domain and gives terms useful for describing both knowledge to acquire (by learning resources) and possessed skills (in personnel profiles). The inventory of learning resources descriptions may be then addressed each time management needs to plan training programs involving personnel selected for acquiring new competence. As hinted before, the decision to plan training programs derives in most cases from the identification of a knowledge gap, either in the profile of one or more selected employees (after HR allocation choices) or in the whole company excellence profile (after strategical choices). Regardless of the reasons for a learning need, if required, support is provided in composing available learning resources to create personalized training paths. The composition process exploits non-standard reasoning services, namely Concept Covering and Concept Abduction [16].

3. Core Competence Identification

The process of identifying Core Competence is usually characterized by high complexity and low objectivity because of the intangibility of knowledge itself and difficulties inherent in formalizing them. We therefore propose an automated logic-based solution performing non-standard inferences over concept descriptions representing personnel profiles formalized in \mathcal{ALN} .

We observe that, despite the collective nature of company Core Competence, it has not necessarily to be shared by the whole personnel: a significant portion of it could be enough. In order to understand the rationale of this observation, consider the following simple example. In a small consulting company only three people are employed: *Alex*, an engineer with knowledge about C++; *Frank*, a TCP/IP expert, skilled in Java; *Mike*, a C programmer.

The implication relationships between concepts involved in these individual profile descriptions, quite intuitive, may be found in the ontology in Figure 3 for the reader interested in their logical formalization.

If we give only to competence held by all of the employees the name Core Competence, we may simply assert that the company in our tiny example has programming knowledge in its Core Competence. If we instead give up to such a full skill coverage and assume that Core Competence is held by the majority of employees, we may assert that object oriented programming is a Core Competence for the company, which is a more specific and then a much more significant result than the previous one.

We assume hereafter for the sake of simplicity that the only source of knowledge lies in company personnel and we therefore focus on employees profile descriptions, although it is obvious that real-world applications may have to exploit all available information.

We implement two alternative processes originally proposed in [20] for Core Competence identification: the first one, Core Competence Extraction, exploits non-standard inference services to discover unknown fields of excellence of the company; the second one, Target Core Competence Reaching Evaluation, checks for the possession of a list of known target competencies by a significant portion of company personnel and explains how to reach the target in case the check fails.

Core Competence extraction. The objective of the implementation of services for automatic Core Competence extraction is identifying a common know-how in a significant portion of company personnel, with a degree of coverage to be set by the management.

Least Common Subsumers (LCS) have been originally proposed by Cohen, Borgida and Hirsh [21] as novel reasoning service for finding commonalities in a collection of concepts. By definition, for a collection of concept descriptions, their LCS represents the most specific concept description subsuming all of the elements of the collection:

Definition 1 (LCS, [21]). *Let C_1, \dots, C_n be n concepts in a DL \mathcal{L} . An LCS of C_1, \dots, C_n , denoted by $LCS(C_1, \dots, C_n)$, is a concept E in \mathcal{L} such that the following conditions hold:*

- (i) $C_i \sqsubseteq E$ for $i = 1, \dots, n$
- (ii) E is the least \mathcal{L} -concept satisfying (i), i.e., if E' is an \mathcal{L} -concept satisfying $C_i \sqsubseteq E'$ for all $i = 1, \dots, n$, then $E \sqsubseteq E'$.

Intuitively, the LCS represents properties shared by *all* the elements of a given collection. In some other applications, like Core Competence evaluation, such a sharing is not required to be full: in other words we are interested in finding a concept description subsuming *a portion of* the elements in a collection, rather than the collection as a whole. We name such concept description *k-Common Subsumer (k-CS)*:

Definition 2 (k-CS([20])). *Let C_1, \dots, C_n be n concepts in a DL \mathcal{L} , and let be $k < n$. A k -Common Subsumer (k -CS) of C_1, \dots, C_n is a concept D such that D is an LCS of k concepts among C_1, \dots, C_n .*

In particular we are interested in finding k -Common Susumers adding informative content to the LCS, defined in the following as Informative k -common subsumers:

Definition 3 (IkCS([20])). Let C_1, \dots, C_n be n concepts in a DL \mathcal{L} , and let $k < n$. An Informative k -Common Subsumer (IkCS) of C_1, \dots, C_n is a k -CS E such that $LCS(C_1, \dots, C_n)$ strictly subsumes E .

Among possible IkCSs, some are characterized by maximum cardinality of the set of subsumed concepts: we define in the following such concepts as Best Informative common subsumers.

Definition 4 (BICS([20])). Let C_1, \dots, C_n be n concepts in a DL \mathcal{L} . A Best Informative Common Subsumer (BICS) of C_1, \dots, C_n is a concept B such that B is an Informative k -CS for C_1, \dots, C_n , and for every $k < j \leq n$ every j -CS is not informative.

If a collection admits an LCS not equivalent to the universal concept, such LCS is the best common subsumer the collection may have. Alternatively, only for collections whose LCS is equivalent to the universal concept the following definition makes also sense:

Definition 5 (BCS[20]). Let C_1, \dots, C_n be n concepts in a DL \mathcal{L} . A Best Common Subsumer (BCS) of C_1, \dots, C_n is a concept S such that S is a k -CS for C_1, \dots, C_n , and for every $k < j \leq n$ every j -CS $\equiv \top$.

Even though the services defined above may appear similar at first sight, it has to be underlined that they deal with different issues:

- k -CS: can be computed for every collection of elements and finds least common subsumers of k elements among the n belonging to the collection;
- IkCS: describes those k -CSs adding an informative content to the one provided by LCS, *i.e.*, more specific than LCS. Observe that IkCS does not exist when every subset of k concepts has the same LCS as the one of all C_1, \dots, C_n ;
- BICS: describes IkCSs subsuming h concepts, such that h is the maximum cardinality of subsets of the collection for which an IkCS exists. A BICS does not exist if and only if $C_i \equiv C_j$ for all $i, j = 1, \dots, n$;
- BCS: may be computed only for collections admitting only LCS equivalent to the universal concept; it finds k -CSs such that k is the maximum cardinality of subsets of the collection for which an LCS not equivalent to \top exists.

In the following we define, with respect to a collection of concept descriptions, \underline{BCS} the set of BCSs, \underline{BICS} the set of BICSs, \underline{ICS}_k the set of IkCSs, given $k < n$ and \underline{CS}_k the set of k -CSs, given $k < n$.

In [15] we proposed the algorithm *Common Subsumer Enumeration* determining the sets \underline{BICS} , \underline{CS}_k , \underline{ICS}_k , \underline{BCS} of a collection $\{C_1, \dots, C_n\}$ of concepts in \mathcal{ALN} . The algorithm takes as input the collection in the form of a *Collection Subsumers Matrix*, introduced in the following (see Definition 8), after two preliminary definitions, 6 and 7.

Definition 6 (Concept Components). *Let C be a concept description in \mathcal{ALN} , with C written in a conjunction $C^1 \sqcap \dots \sqcap C^m$. The Concept Components of C are defined as follows:*

1. *if C^j , with $j = 1 \dots, m$ is a concept name, a negated concept name or a number restriction, then C^j is a Concept Component of C*
2. *if $C^j = \forall R.E$, with $j = 1 \dots, m$, then $\forall R.E^k$ is a Concept Component of C , for each E^k Concept Component of E .*

Definition 7 (Subsumers Matrix). *Let C_1, \dots, C_n be a collection of concept descriptions C_i in a Description Logics \mathcal{L} and let $D_j \in \{D_1, \dots, D_m\}$ be the Concept Components deriving from a set of concepts. We define the **Subsumers Matrix** $S = (s_{ij})$, with $i = 1 \dots n$ and $j = 1 \dots m$, such that:*

- $s_{ij} = 1$ *if the component D_j subsumes C_i*
- $s_{ij} = 0$ *if the component D_j does not subsume C_i .*

Definition 8 (Collection Subsumers Matrix). *Let C_1, \dots, C_n be a collection of concept descriptions C_i in a Description Logic \mathcal{L} . We define the **Collection Subsumers Matrix** as a Subsumers Matrix in which $D_j \in \{D_1, \dots, D_m\}$ are the concept components deriving from all concepts in the collection.*

With reference to complexity issues for the computation of solution sets, the following Theorem provides some useful bounds [20]¹.

Theorem 1. *Let C_1, \dots, C_n, T be n concepts and a simple Tbox in \mathcal{ALN} , let m be the sum of the sizes of C_1, \dots, C_n , and let $S(s)$ be a monotone function bounding the cost of deciding $C \sqsubseteq_{\mathcal{T}} D$ in \mathcal{ALN} , whose argument s is $|C| + |D| + |T|$. The computation of the solution sets \underline{BCS} , \underline{BICS} , \underline{CS}_k and \underline{ICS}_k for a collection of concept descriptions in \mathcal{ALN} is then a problem in $O(m^2 + (S(m))^2)$.*

Target Core Competence Reaching Evaluation. An approach was proposed in [18] to evaluate whether a company possesses a given Core Competence, taken as target. The approach implemented an algorithm performing a subsumption check for each profile description in the repository, in order to determine the number of employees holding the target knowledge. We added to this approach

¹In the complexity analysis we decouple the contribution of the subsumption tests in the subsumers matrix computation from the computation of different introduced common subsumers

an explanation feature, hypothesizing, in case the target is not reached, the reasons why this happens: in [15] we provided an algorithm for evaluation of target Core Competence reaching, with explanation features. The algorithm takes as input the *Concept Subsumers Matrix* (see Definition 9) deriving from a collection of employees knowledge profiles C_1, \dots, C_n and a target Core Competence description R , both in \mathcal{ALN} .

Definition 9 (Concept Subsumers Matrix). *Let C_1, \dots, C_n be a collection of concept descriptions C_i and R a concept description, both in a Description Logic \mathcal{L} .*

*We define the **Concept Subsumers Matrix** as a Subsumers Matrix in which $D_j \in \{D_1, \dots, D_m\}$ are the concept components of R , denoted by R_j .*

The complexity results for the computation of an explanation set through the above mentioned algorithm are given in Theorem 2 (we omit the proof because it is similar to the one for Theorem 1 and needs some definitions omitted in the paper for the sake of brevity).

Theorem 2. *Let $C_1, \dots, C_n, \mathcal{T}$ be n concepts and a simple Tbox in \mathcal{ALN} , let m be the sum of the sizes of C_1, \dots, C_n , and let $S(s)$ be a monotone function bounding the cost of deciding $C \sqsubseteq_{\mathcal{T}} D$ in \mathcal{ALN} , whose argument s is $|C| + |D| + |\mathcal{T}|$. The computation of the explanation set E for a collection of concept descriptions and a concept in \mathcal{ALN} is then a problem in $O(m^2 + (S(m))^2)$.*

4. The KMS in use

We here introduce the architecture of the proposed KMS with respect to the use cases outlined in Section 2. A sketch of the functional architecture is represented in Figure 2.

It is noteworthy that the proposed KMS relies on a repository of descriptions related to different sources of knowledge. All such descriptions need to adopt the same vocabulary for knowledge representation.

In Figure 3 we provide the \mathcal{ALN} TBox corresponding to an excerpt of such a vocabulary in order to make the example terminology available to the reader.

The repository exploited by the KMS contains the knowledge sources descriptions making up the ABox of the employed DL System. The available individuals describe two different kinds of knowledge sources:

- **Personnel Profiles:** descriptions according to the TBox vocabulary of the persons available for the solution of different tasks;
- **Learning Objects:** descriptions according to the TBox vocabulary of the learning resources to be composed for the creation of personalized training programs.

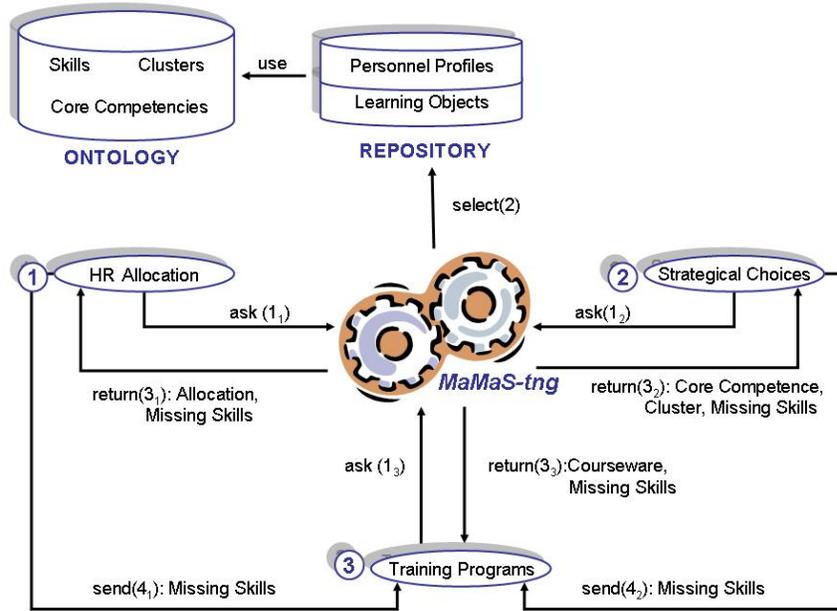


Figure 2: KMS architecture

Considering personnel profile descriptions it is noteworthy that, coherently with the aim of providing knowledge management support services, only personnel competence is conveyed in them: features different from skills are not taken into account in the semantic base processes and then there is no need to model them as DL descriptions.

Learning Object descriptions are modeled in the ontology as concept descriptions made up by a background knowledge and a description component, coherently with the LOM standard [22] for learning content modeling.

The KMS operations can be triggered by any of the use cases in Figure 2, in accordance with the contextual need of the management. The system obviously allows to permanently store knowledge profiles in the repository for reuse. In order to perform each of the available services, the KMS asks the Reasoner for the solution of the inference problems at the basis of the implemented algorithms.

In particular, the solution of Concept Abduction and Concept Contraction problems require using **MaMaS-tng**², that implements solutions to those inference services. MaMAS-tng extends DIG standard [23] for communication with DL reasoners, in order to implement also the needed non-standard reasoning services.

²MAch MAker Service is available at <http://sisinflab.poliba.it/MAMAS-tng/>

ActiveX \sqsubseteq Skill
 ITConsulting \sqsubseteq Skill
 Flash4 \sqsubseteq Skill
 Psychology \sqsubseteq Skill
 QualityTechniques \sqsubseteq Skill
 StatisticalMethods \sqsubseteq Skill
 SupplyChain \sqsubseteq Skill
 AssetAllocation \sqsubseteq Skill
 Design \sqsubseteq Skill
 ClientServer \sqsubseteq Skill
 ComputerGraphics \sqsubseteq Skill
 InformationSystems \sqsubseteq Skill
 DBMS \sqsubseteq InformationSystems
 ERPsystem \sqsubseteq InformationSystems
 TCP/IP \sqsubseteq InformationSystems
 InternetDevelopment \sqsubseteq Skill
 WebDesign \sqsubseteq InternetDevelopment
 InternetTechnology \sqsubseteq Skill
 InternetUse \sqsubseteq Skill
 MeetingsCoordination \sqsubseteq Skill
 TeamCoordination \sqsubseteq Skill
 MarkupLanguages \sqsubseteq Skill
 HTML \sqsubseteq MarkupLanguages
 XML \sqsubseteq MarkupLanguages
 SoftwareEngineering \sqsubseteq Skill
 UML \sqsubseteq SoftwareEngineering
 Programming \sqsubseteq Skill
 OOP \sqsubseteq Programming
 Java \sqsubseteq OOP
 C++ \sqsubseteq OOP
 StrucProgramming \sqsubseteq Programming
 C \sqsubseteq StrucProgramming
 ScriptLanguages \sqsubseteq Programming
 Javascript \sqsubseteq ScriptLanguages
 VBscript \sqsubseteq ScriptLanguages
 OperationsOptimization \sqsubseteq Skill
 DistributionManagement \sqsubseteq Skill
 ProductionManagement \sqsubseteq Skill
 Management \sqsubseteq Skill
 ProcPerformMonitoring \sqsubseteq Skill
 ComputerArchitecture \sqsubseteq Skill
 ProcessPlanning \sqsubseteq Skill
 English \sqsubseteq Language
 German \sqsubseteq Language
 Engineering \sqsubseteq Degree
 ComputerScience \sqsubseteq Degree
 MechanicalEngineering \sqsubseteq Engineering
 ElectronicsEngineering \sqsubseteq Engineering
 ComputerScienceEngineering \sqsubseteq Engineering
 learningObject $\equiv \exists$ backgroundKnowledge $\sqcap \exists$ hasDescription

Figure 3: Skills Ontology excerpt

The execution flow of each of the three main use cases in Section 2 (Strategical choices, Human Resources Allocation Choices, Training Programs) is indicated by the numbers on the arrows in Figure 2: the ask phase (1) initiates the use case indicated in the subscript (*e.g.*, 1_1 is the ask phase related to use case 1) and is followed by the select (2) and the return phase (3). We notice that Figure 2 explains the execution flow of use cases 1 and 2 without delving into details about their sub-cases, *i.e.*, by generalizing them according to a uniform view.

The system provides a GUI based on AJAX technology [24] and exploiting Google Web toolkit libraries³. The GUI is consistent with the standard for personnel profile descriptions for recruitment process proposed by HR-XML consortium⁴: the Staffing Exchange Protocol (SEP). All of the features identified by SEP are embedded in our system. In order to provide the semantic description of the profile, either to store or to search for, the GUI makes one more panel available for ontology navigation aimed at profile composition.

According to the approach outlined in Section 2, the KMS exploits MaMaS-tng services to select in the repository the resources providing answers to the contextual need. In particular the HR Allocation use case, regardless of the chosen multiplicity, returns an allocation proposal and the explanation about missing skills in the selected individuals. Such unavailable knowledge can be used as learning request in a training program definition process. The first use case may then call the services provided by the third one, which exploits the reasoner services to compose learning objects in the repository, according to the implemented courseware composition algorithm. The remaining missing knowledge after the possible courseware fruition is also returned to the system user.

The second use case asks the reasoner for the inferences needed in the strategical support services detailed in Section 2.1. By investigating on the individual profiles stored in the repository, the KMS returns the possible Core Competence of the Company or the knowledge clusters the company belongs to, together with the explanation about competence missing to fit the cluster. Such missing skills may motivate the need for training programs, as hinted in Section 2.1, to perform through the third use case.

The operating mode of our system is hereafter detailed with the help of an extended example, covering all possible use cases of the KMS.

Let us suppose the system profiles repository is made up by the four individual descriptions in Table 1, for the sake of simplicity.

Use Case 1: HR Allocation Choices

We assume the management of the company, the individuals in Table 1 are employed in, has an assignment need. According to the multiplicity of the

³<http://code.google.com/webtoolkit>

⁴<http://hr-xml.org>

Profile	Formal Definition
1. Raffaella Guerrini	$\exists \text{hasMasterDegree} \sqcap$ $\forall \text{hasMasterDegree. ComputerScience} \sqcap$ $\exists \text{advKnowledge} \sqcap \forall \text{advKnowledge. (ActiveX} \sqcap$ $\text{Flash4} \sqcap \text{MarkUpLanguages} \sqcap \text{AssetAllocation}) \sqcap$ $\exists \text{basKnowledge} \sqcap \forall \text{basKnowledge. Psychology} \sqcap$ $(\leq 4 \text{ hasExperience})$
2. Allison Hence	$\exists \text{hasMasterDegree} \sqcap$ $\forall \text{hasMasterDegree. MechanicalEngineering} \sqcap$ $\exists \text{advKnowledge} \sqcap \forall \text{advKnowledge. (ActiveX} \sqcap \text{HTML} \sqcap$ $\text{ITConsulting}) \sqcap \exists \text{hasLanguageKnowledge} \sqcap$ $\forall \text{hasLanguageKnowledge. (English} \sqcap \text{German})$
3. Tom Barclay	$\exists \text{hasMasterDegree} \sqcap$ $\forall \text{hasMasterDegree. ElectronicsEngineering} \sqcap$ $\exists \text{advKnowledge} \sqcap \forall \text{advKnowledge. (AssetAllocation} \sqcap$ $\text{TCP/IP} \sqcap \text{ClientServer} \sqcap \text{InternetTechnology} \sqcap$ $\text{InternetUse} \sqcap \text{Javascript}) \sqcap \exists \text{toolsKnowledge} \sqcap$ $\forall \text{toolsKnowledge. InternetDevelopment} \sqcap$ $\exists \text{basKnowledge} \sqcap \forall \text{basKnowledge. Psychology} \sqcap$ $(\geq 5 \text{ hasExperience})$
4. Miriam Lebez	$\exists \text{hasMasterDegree} \sqcap$ $\forall \text{hasMasterDegree. ComputerScienceEngineering} \sqcap$ $\exists \text{advKnowledge} \sqcap \forall \text{advKnowledge. (ClientServer} \sqcap$ $\text{InternetTechnology} \sqcap \text{VBscript} \sqcap \text{OOP} \sqcap \text{TCP/IP}) \sqcap$ $\exists \text{toolsKnowledge} \sqcap$ $\forall \text{toolsKnowledge. InternetDevelopment}$

Table 1: Employees profiles definitions

specific assignment problem, the management can choose among one of the three use cases detailed hereafter.

Use Case 1.1: Single Task Assignment

Suppose in our example company there is a job vacancy asking for *a computer science degree, an advanced knowledge about Active X, Flash, Markup Languages and at least five years of working experience*. The task is formalized in the following w.r.t. the ontology in Figure 3:

- $T_1 = \exists \text{hasMasterDegree} \sqcap \forall \text{hasMasterDegree.ComputerScience} \sqcap \forall \text{advKnowledge} . (\text{ActiveX} \sqcap \text{Flash4} \sqcap \text{MarkUpLanguages}) \sqcap (\geq 5 \text{ hasExperience})$.

Figure 4 shows the panel of the GUI supporting ontology navigation for the composition of the query. Profiles are ranked according to a function [16] which measures the utility of assigning a task T_i to an assignee A_j by taking into account both missing and conflicting features. The higher utility values are, the less the related assignment is preferred.

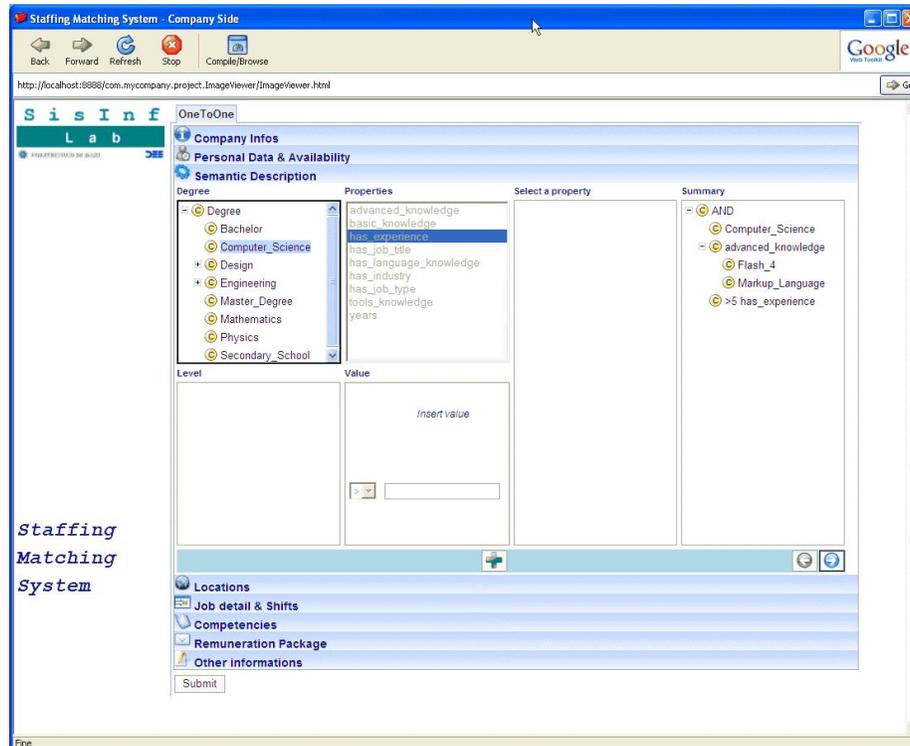


Figure 4: Query Composition Panel

As a consequence, the employee Raffaella Guerrini is selected: her profile almost perfectly matches the required task, except for the experience require-

ment, which is in conflict and then returned in a specific panel for features to be possibly given up in the request.

Use Case 1.2: Multiple Task Single Assignment

Let T be the set of knowledge requests composed by the tasks given in the following together with their \mathcal{ALN} descriptions:

- *Computer Scientist speaking English and endowed with advanced knowledge of C++, DBMS and Markup Languages:*
 $T_2 = \exists \text{hasMasterDegree.} \sqcap \forall \text{hasMasterDegree.ComputerScience} \sqcap$
 $\forall \text{advKnowledge.}(C++ \sqcap \text{DBMS} \sqcap \text{MarkUpLanguages}) \sqcap$
 $\forall \text{hasLanguageKnowledge.English};$
- *Engineer with advanced knowledge of client server protocol, asset allocation and quality assurance techniques:*
 $T_3 = \exists \text{hasMasterDegree.} \sqcap$
 $\forall \text{hasMasterDegree.Engineering} \sqcap \forall \text{advKnowledge.}$
 $(\text{ClientServer} \sqcap \text{AssetAllocation} \sqcap \text{QualityTechniques});$
- *Computer Science Engineer with advanced knowledge of client server protocol, object oriented programming and ability in meeting and team coordination:*
 $T_4 = \exists \text{hasMasterDegree.}$
 $\sqcap \forall \text{hasMasterDegree.ComputerScienceEngineering}$
 $\sqcap \forall \text{advKnowledge.}(\text{ClientServer} \sqcap \text{OOP}$
 $\sqcap \text{MeetingsCoordination} \sqcap \text{TeamCoordination}).$

The matching process detailed in [16] is performed and an assignment problem [26] is solved according to the values, provided by the same utility function in use case 1.1, shown in the following table:

	T_2	T_3	T_4
<i>Raffaella Guerrini</i>	0.64	0.71	0.83
<i>Allison Hence</i>	0.55	0.57	0.75
<i>Tom Barclay</i>	0.82	0.14	0.67
<i>Miriam Lebez</i>	0.64	0.43	0.42

The assignment solutions consist in a set of 4-tuples containing a task, an assignee, the knowledge lacking in the assignee profile, the requirements to be possibly given up in the task description.

In our case example the following 4-tuples are therefore returned:

1. $x_{21} = 1$
 $(T_2, \text{Allison Hence},$
 $H_{21} = \forall \text{hasMasterDegree.ComputerScience} \sqcap \forall \text{advKnowledge.}(C++$
 $\sqcap \text{DBMS}),$
 $G_{21} = \top);$

2. $x_{32} = 1$
 $(T_3, Tom\ Barclay,$
 $H_{32} = \forall \text{advKnowledge.QualityTechniques},$
 $G_{32} = \top);$
3. $x_{43} = 1$
 $(T_4, Miriam\ Lebez,$
 $H_{43} = \forall \text{advKnowledge}.\text{(MeetingsCoordination} \sqcap \text{TeamCoordination)),}$
 $G_{43} = \top).$

Use Case 1.3: Team Composition

If the management of our company is searching for a team endowed with *advanced knowledge about Active X, HTML, software engineering and client server protocol and with at least one member with a computer science degree*, the required task is formalized as:

$$T_5 = \exists \text{hasMasterDegree}.\sqcap \forall \text{hasMasterDegree.ComputerScience} \sqcap \forall \text{advKnowledge}.\text{(ActiveX} \sqcap \text{HTML} \sqcap \text{SoftwareEngineering} \sqcap \text{ClientServer}).$$

The system solves a Concept Covering Problem by implementing a greedy approach [16] adding profiles maximizing the utility function recalled in use case 1.1 to the selected set, trying to cover the whole task. The solution algorithm returns the assignment set, together with missing and conflicting information. The team composed by *Raffaella Guerrini, Allison Hence* and *Tom Barclay* is selected, as shown in Figure 5. Unfortunately, no one in the company has knowledge about software engineering, which remains uncovered after the team composition.

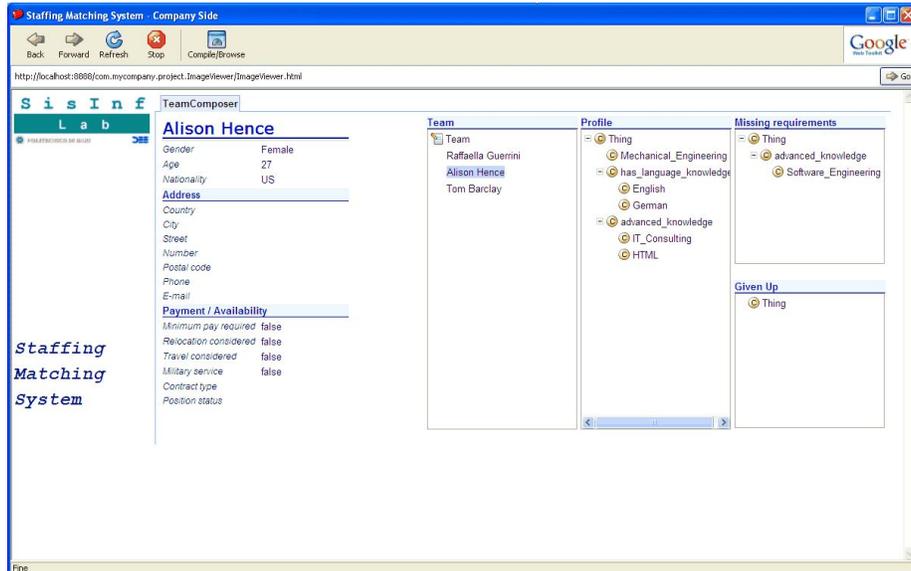


Figure 5: Many to One Match Results Panel

Use Case 2: Statistical Choices

The management of the small company employing the personnel described in Table 1 is involved in the decisional problem of identifying collective Core Competence in the company. According to the approach detailed in Section 3, the degree of competence sharing needs to be set: imagine the required coverage is half of the employees ($k = 2$).

The system makes available a GUI for choosing between the two evaluation processes introduced in Section 2.1.

Profiles in Table 1 constitute the input collection.

Use Case 2.1: Core Competence Extraction

If the user asks for automated Core Competence extraction, the system computes the set of partial common subsumers as recalled in Section 3.

The concept components deriving from the input collection are listed hereafter:

- $D_1 = \exists \text{hasMasterDegree}$
- $D_2 = \forall \text{hasMasterDegree.ComputerScience}$
- $D_3 = \exists \text{advKnowledge}$
- $D_4 = \forall \text{advKnowledge.ActiveX}$
- $D_5 = \forall \text{advKnowledge.Flash4}$
- $D_6 = \forall \text{advKnowledge.MarkUpLanguages}$
- $D_7 = \forall \text{advKnowledge.AssetAllocation}$
- $D_8 = \exists \text{basKnowledge}$
- $D_9 = \forall \text{basKnowledge.Psychology}$
- $D_{10} = (\leq 4 \text{ hasExperience})$
- $D_{11} = \forall \text{hasMasterDegree.MechanicalEngineering}$
- $D_{12} = \forall \text{advKnowledge.HTML}$
- $D_{13} = \forall \text{advKnowledge.ITConsulting}$
- $D_{14} = \exists \text{hasLanguageKnowledge}$
- $D_{15} = \forall \text{hasLanguageKnowledge.English}$
- $D_{16} = \forall \text{hasLanguageKnowledge.German}$
- $D_{17} = \forall \text{hasMasterDegree.ElectronicsEngineering}$
- $D_{18} = \forall \text{advKnowledge.TCP/IP}$
- $D_{19} = \forall \text{advKnowledge.ClientServer}$
- $D_{20} = \forall \text{advKnowledge.InternetTechnology}$
- $D_{21} = \forall \text{advKnowledge.InternetUse}$
- $D_{22} = \forall \text{advKnowledge.Javascript}$
- $D_{23} = \exists \text{toolsKnowledge}$
- $D_{24} = \forall \text{toolsKnowledge.InternetDevelopment}$
- $D_{25} = (\geq 5 \text{ hasExperience})$
- $D_{26} = \forall \text{hasMasterDegree.ComputerScienceEngineering}$
- $D_{27} = \forall \text{advKnowledge.VBscript}$
- $D_{28} = \forall \text{advKnowledge.OOP}$

The LCS of the whole collection is
 $LCS = \exists \text{hasMasterDegree} \sqcap \exists \text{advKnowledge} \sqcap \exists \text{basKnowledge}$
which is in fact quite a useless information for strategic choices.

By analyzing the subsumers matrix of the collection, we obtain that both
Tom Barclay and *Miriam Lebez* know about

$$CC_1 = \exists \text{advKnowledge} \sqcap \forall \text{advKnowledge} . (\text{TCP/IP} \sqcap \text{ClientServer} \\ \sqcap \text{InternetTechnology} \sqcap \text{ScriptLanguages}) \sqcap \exists \text{toolsKnowledge} \\ \sqcap \forall \text{toolsKnowledge} . \text{InternetDevelopment} \sqcap \\ \exists \text{hasMasterDegree} \sqcap \forall \text{hasMasterDegree} . \text{Engineering}.$$

Analogously, the concept
 $CC_2 = \exists \text{advKnowledge} \sqcap \forall \text{advKnowledge} . (\text{AssetAllocation}) \sqcap \exists \text{basKnowledge} \\ \sqcap \forall \text{basKnowledge} . (\text{Psychology})$
subsumes both *Raffaella Guerrini* and *Tom Barclay*.

Moreover both *Raffaella Guerrini* and *Allison Hence* share knowledge $CC_3 = \\ \exists \text{advKnowledge} \sqcap \forall \text{advKnowledge} . (\text{ActiveX} \sqcap \text{MarkUpLanguages})$

The concept $CC_4 = \exists \text{hasMasterDegree} \sqcap \forall \text{hasMasterDegree} . \text{Engineering}$
subsumes instead *Allison Hence*, *Tom Barclay* and *Miriam Lebez*.

Given the input threshold $k = 2$, we can assert that CC_1 , CC_2 , CC_3 and
 CC_4 are k -CSs of the collection.; in particular they are l_k CSs, because they add
informative content to the LCS.

CC_4 subsumes the largest number of individual profiles less than $n (= 4)$:
hence it is also a BICS.

As a result, the company knowledge profile is defined as:
 $C = CC_1 \sqcap CC_2 \sqcap CC_3 \sqcap CC_4.$

Now, imagine the knowledge classes in which the network of companies may
be classified are the ones formally defined in Table 2.

The system solves a Concept Abduction problem (CAP) to find skills re-
quired in the knowledge classes, but unavailable in the company knowledge
profile [18]. Therefore, we get to the following results, which show the compe-
tence missing in our example company to fit each of the clusters presented in
Table 2.

- **Knowledge Class:** ICT Solutions For Human Resources Management
Explanation Results:
 $H_1 = \forall \text{basKnowledge} . \text{WorkflowManagement};$
- **Knowledge Class:** Industrial Software Production For Operations
Management
Explanation Results:
 $H_2 = \exists \text{advKnowledge} \sqcap \forall \text{advKnowledge} . (\text{ComputerGraphics} \sqcap \text{OOP} \\ \sqcap \text{StrucProgramming} \sqcap \text{SoftwareEngineering} \sqcap \text{OperationsOptimization} \\ \sqcap \text{DistributionManagement} \sqcap \text{ProductionManagement}) \sqcap \exists \text{basKnowledge} \sqcap \\ \forall \text{basKnowledge} . (\text{Design} \sqcap \text{InformationSystems} \sqcap \text{SupplyChain} \\ \sqcap \text{ProcPerformMonitoring}) \\ \equiv \text{Industrial Software Production For Operations Management};$

Knowledge Class	Formal Definition
ICT Solutions For Human Resources Management	$\exists \text{advKnowledge} \sqcap \forall \text{advKnowledge} . (\text{TCP/IP} \sqcap \text{ClientServer} \sqcap \text{InternetTechnology} \sqcap \text{ScriptLanguages} \sqcap \text{AssetAllocation}) \sqcap$ $\exists \text{toolsKnowledge} \sqcap$ $\forall \text{toolsKnowledge} . \text{InternetDevelopment} \sqcap$ $\exists \text{basKnowledge} \sqcap$ $\forall \text{basKnowledge} . (\text{Psychology} \sqcap \text{WorkflowManagement})$
Industrial Software Production For Operations Management	$\exists \text{advKnowledge} \sqcap$ $\forall \text{advKnowledge} . (\text{ComputerGraphics} \sqcap \text{OOP} \sqcap \text{StrucProgramming} \sqcap \text{SoftwareEngineering} \sqcap \text{OperationsOptimization} \sqcap \text{DistributionManagement} \sqcap \text{ProductionManagement}) \sqcap \exists \text{basKnowledge} \sqcap$ $\forall \text{basKnowledge} . (\text{Design} \sqcap \text{InformationSystems} \sqcap \text{SupplyChain} \sqcap \text{ProcPerformMonitoring})$
Total Quality Management Solutions	$\exists \text{advKnowledge} \sqcap$ $\forall \text{advKnowledge} . (\text{QualityTechniques} \sqcap \text{StatisticalMethods}) \sqcap \exists \text{basKnowledge} \sqcap$ $\forall \text{basKnowledge} . (\text{ProcPerformMonitoring} \sqcap \text{InformationSystems})$

Table 2: Knowledge Classes Definitions

- **Knowledge Class:** Total Quality Management Solutions

Explanation Results:

$H_3 = \exists \text{advKnowledge} \sqcap \forall \text{advKnowledge} . (\text{QualityTechniques} \sqcap \text{StatisticalMethods}) \sqcap \exists \text{basKnowledge} \sqcap \forall \text{basKnowledge} . \text{ProcPerformMonitoring} .$

The company does not belong to any knowledge cluster, but the first result suggests that only a basic knowledge about workflow management is needed to fit **ICT Solutions For Human Resources Management** cluster. Such a result may be interpreted as a learning need to be possibly covered with a learning path ad-hoc composed by the system (use case 3).

The second result shows instead that the company has nothing in common with the **Industrial Software Production For Operations Management** cluster, given that it lacks of the whole knowledge required to fit it. Finally, the company lacks almost all knowledge required to fit **Total Quality Management Solutions** cluster, except basic knowledge of **InformationSystems** deriving from CC_1 Core Competence.

Use Case 2.2: Target Core Competence Reaching Evaluation

The user chooses to investigate on possible fields of excellence by evaluating whether a given Core Competence, taken as target, is hold by an established number of employees. Again such a threshold value is set to fifty percent of company personnel (two in our example). Our example target Core Competence is **Human Resource Management** described as:

Human Resources Management =

$\exists \text{advKnowledge} \sqcap \forall \text{advKnowledge} . (\text{AssetAllocation} \sqcap \text{Management}) \sqcap \exists \text{basKnowledge} \sqcap \forall \text{basKnowledge} . (\text{Psychology}) .$

The concept components deriving from **Human Resources Management** are listed below:

$R_1 = \exists \text{advKnowledge}$

$R_2 = \forall \text{advKnowledge} . \text{AssetAllocation}$

$R_3 = \forall \text{advKnowledge} . \text{Management}$

$R_4 = \exists \text{basKnowledge}$

$R_5 = \forall \text{basKnowledge} . (\text{Psychology})$

By performing the process recalled in Section 3, the systems returns the following explanation set: either *Raffaella Guerrini* or *Tom Barclay* need to learn about management in order for the company to hold **Human Resources Management** Core Competence.

Use Case 3: Training Programs

The example provided for use case 1.3 is solved by the composition of a team, composed by *Raffaella Guerrini*, *Allison Hence* and *Tom Barclay*. The three employees cover almost the whole task request, except for software engineering competence, which is not available in the company.

Such an explanation about unavailable knowledge can be filled by automatically composed personalized coursewares [16]. The approach solves a Concept

Covering Problem analogous to the one recalled in use case 1.3. Imagine for example that the learning objects in Table 3 are available for fruition. Observe that learning objects (LOs) are defined as composed in a description, *i.e.*, the knowledge taught by the LO, and a background knowledge, *i.e.*, the competence profile needed to be able to take the course.

Name	Formal Description
λ^1 : <i>Software Engineering and UML course</i>	learningObject \sqcap \forall backgroundKnowledge. (\forall advKnowledge.Design) \sqcap \forall hasDescription. (\forall advKnowledge.UML)
λ^2 : <i>Design course</i>	learningObject \sqcap \forall backgroundKnowledge.English \sqcap \forall hasDescription. (\forall advKnowledge.Design)
λ^3 : <i>Software Engineering Course</i>	learningObject \sqcap \forall backgroundKnowledge. (\forall advKnowledge.Programming) \sqcap \forall hasDescription. (\forall advKnowledge.SoftwareEngineering)
λ^4 : <i>English Course</i>	learningObject \sqcap \forall backgroundKnowledge. \top \sqcap \forall hasDescription.(\forall advKnowledge.English)

Table 3: Learning Objects Descriptions

The learning need description originating the courseware composition process is formalized as follows:

$$T_6 = \rho_D = \forall \text{advKnowledge.SoftwareEngineering}$$

Given that the three individuals selected for the team do not share the background knowledge (*i.e.*, the knowledge profile), three different personalized coursewares are provided:

1. Raffaella Guerrini

Learning Need

$$\begin{aligned} \rho_1 = & \exists \text{backgroundKnowledge} \sqcap \forall \text{backgroundKnowledge}. \\ & (\exists \text{hasMasterDegree} \sqcap \forall \text{hasMasterDegree.ComputerScience} \\ & \sqcap \exists \text{advKnowledge} \sqcap \forall \text{advKnowledge} . (\text{ActiveX} \sqcap \text{Flash4} \\ & \sqcap \text{MarkUpLanguages} \sqcap \text{AssetAllocation}) \sqcap \exists \text{basKnowledge} \\ & \sqcap \forall \text{basKnowledge.Psychology} \sqcap (\leq 4 \text{ hasExperience})) \\ & \sqcap \exists \text{hasDescription} \\ & \sqcap \forall \text{hasDescription} . (\forall \text{advKnowledge.SoftwareEngineering}) \end{aligned}$$

$$\text{Courseware } \Lambda = \{\lambda^4, \lambda^2, \lambda^1\}$$

Missing knowledge : \top

Additional Knowledge : UML

2. Allison Hence

Learning Need

$\rho_2 = \exists \text{backgroundKnowledge}$
 $\sqcap \forall \text{backgroundKnowledge} . (\exists \text{hasMasterDegree}$
 $\sqcap \forall \text{hasMasterDegree} . \text{MechanicalEngineering} \sqcap \exists \text{advKnowledge}$
 $\sqcap \forall \text{advKnowledge} . (\text{ActiveX} \sqcap \text{HTML} \sqcap \text{ITConsulting}))$
 $\sqcap \exists \text{hasLanguageKnowledge} \sqcap \forall \text{hasLanguageKnowledge} . (\text{English}$
 $\sqcap \text{German})) \sqcap \exists \text{hasDescription} \sqcap$
 $\forall \text{hasDescription} . (\forall \text{advKnowledge} . \text{SoftwareEngineering}))$

Courseware $\Lambda = \{\lambda^2, \lambda^1\}$

Missing knowledge : \top

Additional Knowledge : UML

3. Tom Barclay

Learning Need

$\rho_3 = \exists \text{backgroundKnowledge}$
 $\sqcap \forall \text{backgroundKnowledge} . (\exists \text{hasMasterDegree}$
 $\sqcap \forall \text{hasMasterDegree} . \text{ElectronicsEngineering}$
 $\sqcap \exists \text{advKnowledge} \sqcap \forall \text{advKnowledge} . (\text{AssetAllocation}$
 $\sqcap \text{TCP/IP} \sqcap \text{ClientServer} \sqcap \text{InternetTechnology} \sqcap \text{InternetUse}$
 $\sqcap \text{Javascript}))$
 $\sqcap \exists \text{toolsKnowledge} \sqcap \forall \text{toolsKnowledge} . \text{InternetDevelopment}$
 $\sqcap \exists \text{basKnowledge} \sqcap \forall \text{basKnowledge} . \text{Psychology}$
 $\sqcap (\geq 5 \text{ hasExperience})) \sqcap \exists \text{hasDescription}$
 $\sqcap \forall \text{hasDescription} . (\forall \text{advKnowledge} . \text{SoftwareEngineering}))$

Courseware $\Lambda = \{\lambda^3\}$

Missing knowledge : \top

Additional Knowledge : \top

All of the three proposed coursewares completely fulfill the learning need (the unavailable knowledge is equivalent to the universal concept). The courseware most immediate to learn is the one proposed to Tom Barclay, which nevertheless does not provide any additional knowledge. On the contrary, the coursewares proposed to Raffaella Guerrini and Allison Hence give UML knowledge as bonus, even though their fruition is less immediate. Moreover the second solution should be preferred to the first one because of the courseware length. Still the choice of the courseware to start is up to the management, which could decide between the second and the third solution on the basis of a trade-off between gained additional knowledge and fruition simplicity.

5. System Evaluation

In the following we present system evaluation from a two fold perspective: first we present results in terms of accuracy in retrieving best fulfilling profiles, then we report on computational performances of the Core Competence Identification component.

5.1. Human Resource Allocation

In order to show the significance of results provided by the KMS in business scenarios, we report here on one to one skill matching as test functionality, given the high number of available alternative approaches and systems performing the above mentioned process. In particular, our experiments aimed at evaluating our KMS solutions both vs. the ones provided by an alternative system and vs. human users judgment. In order to compare our KMS to an alternative automated system, we involved a recruitment agency, that currently uses a custom RDBMS-based system to automatically perform human resources assignment. The agency stores several hundred CVs, manually translated into a structured template, as database records. Such structured information constitute an important asset of the agency. For the ICT domain, their CV template is composed of four sections: *Education&Training*, *Languages*, *Knowledge&Technical Skill* (it represents both candidate abilities and competences with the respective expertise level) and *Projects* (it represents both candidate working experience and held job titles). The candidates database can be queried through a GUI allowing for the insertion/selection of different sets of keywords, where each set is related to a CV template section mapped to the database. The search process relies on a string matching over database fields and on *OR operator* employment(requirements expressed by keywords are considered not strict). In accordance to such a GUI, we proposed the task T_1 in Use Case 1.1. From the most recently inserted CVs, the agency extracted 96 ICT professionals individuals, where only 55 out of 96 had at least one feature shared with requirements of our task. From the 96 profiles, we selected 16 ones (13 belonging to the set of 55 selected outcome and 3 profiles not so) to model in accordance with our skills ontology, sketched in Figure 3. The four individuals in Table 1, chosen as knowledge source for modeling the use cases described in Section 4, were extracted from such a set. We then asked the company to search in its current system for CVs solving our task, *i.e.*, *Professionals endowed with a computer science degree, an advanced knowledge about Active X, Flash, Markup Languages and at least five years of working experience*. Their system returned an *unordered* list of 82 CVs: 40 of the 55 ICT professional profiles and 42 more CVs, belonging to people with a degree in Computer Science, but not working as ICT professionals. Only 11 profiles among the 16 modeled in our system were returned. The system returned such candidates because they matched some of the keywords included in the query. We noted that the employment agency personnel manually extracted 55 ICT professional profiles by considering them as possible matches for the task T_1 in Use Case 1.1, but the RDBMS-based system detected only 40 matches. Our semantic-based KMS, ranking candidates according to the

Recruiter	Ranking				
	1st	2nd	3rd	4th	5th
1	1	2,6,11	7,8		
2	1,6	11,8	2,5	3	
3	1,6	11,8	2	7	3
final	1	11,8	2	7	3

Table 4: T_1 -Human Ranking

process outlined in Use Case 1.1 description, allowed for significant automation features:

- the KMS returns the highest ranked candidate: *Raffaella Guerrini*, together with explanation features crucial for the recruiter to understand the proposed assignment;
- the KMS candidates set includes 13 of the 16 ICT Professionals modeled according to the ontology: two profiles not returned by database search potentially match the task request if implicit information is taken into account; in particular, the fifth candidate ranked by the KMS is an ICT professional excluded by the company recruiting system, *7. Michele Natali*, a Computer Science Engineer, with an advanced knowledge of XML, German Language, TCP/IP and three years of working experience.

Performance of the proposed KMS vs. the judgment of human experts is hereafter presented in terms of R_{norm} [25] as quality measure for the retrieval effectiveness. R_{norm} values range between 0 and 1, where a value of 1 corresponds to full agreement between system ranking and human ranking, and values less than 1 measure different levels of disagreement. To evaluate human judgment for the similarity of different pairs Task-Profile, we asked 3 recruiters in the agency to independently rank in decreasing order the 13 profiles returned as candidates set by the system with respect to the queries $T_1, T_2, T_3, T_4, T_5, T_6$, presented in Section 4. The first 5 queries ask for features corresponding to multiple CV sections (and they then allow for evaluating recruiters priorities for different features), while the sixth one asks for a single feature. As an example, consider the classification related to the query T_1 in Table 4. Given that the three recruiters provided not univocal classifications, the final ranking of each profile was determined as the minimum one among the 3 available. Table 5 shows R_{norm} values for each of the six performed queries.

The evaluation shows that our KMS is at least as effective in performing human resources allocation as a traditional DBMS-based system and as a human recruiter: the KMS is able to rank all promising candidates – even when the traditional DBMS-based systems fail; it excludes all those not relevant with respect to the required task – as opposite to the unordered lists of results returned by traditional DBMS-based systems; it behaves almost exactly like a recruiter. We notice that KMS disagreement with human judgment, revealed by R_{norm} values less than 1, may depend on ontology modeling issues: the representation

Query	Profile	R_{norm}
1	1	1
2	2	0.99
3	11	1
4	4	0.98
5	1	1
6	8	0.95

Table 5: R_{norm} Values

of some concepts in the ontology does not correspond to the way human beings conceive them; R_{norm} can thus be also a significant measure to support ontology engineering and maintenance efforts.

5.2. Core Competence Identification

For the evaluation, 20 CVs focused on the ICT knowledge domain have been randomly selected. Such competence profiles have been modeled according to the ontology used in our KMS.

The test procedure has been designed to randomly extract a number of profiles among the twenty and perform Common Subsumer Enumeration and Target Core Competence Reaching Evaluation algorithms detailed above.

The performed tests can be classified in two main categories:

- variable number of profiles and concept components(200 tests);
- constant number of profiles and variable number of concept components(50 tests).

In the Common Subsumer Enumeration process the number of profiles is randomly selected, while the number of concept components depends on the composition of the selected profiles.

Instead, in the Target Core Competence reaching Evaluation process, the number of profiles is randomly selected, while the number of concept components depends on the description of the input Core Competence, which has been automatically generated by conjoining concept components in the ontology. Moreover, in each test the threshold values k have been set to a number ranging from 1(included) to the number of profiles (excluded).

To numerically evaluate performances, the following parameters have been taken into account:

- n_p = number of profiles;
- n_{cc} = number of concept components deriving from the n_p profiles;
- k = threshold;
- t_{sscsc} = Collection Subsumers Matrix computation time;

- t_{cse} = Common Subsumer Enumeration algorithm execution time;
- n_{tcc} = number of concept components deriving from the the input Core Competence of Target Core Competence Evaluation algorithm;
- $t_{sstccre}$ = Concept Subsumers Matrix computation time;
- t_{tccre} = Target Core Competence Evaluation algorithm execution time.

The collected data show that time for performing Core Competence Extraction and Target Core Competence evaluation algorithms are negligible w.r.t. subsumers matrix computation times: in the following we therefore represent the time complexity results of the presented approach in terms of subsumers matrix computation time.

Figure 6 and 7 show, respectively, the time for collection and concept subsumers matrix computation (in seconds) vs. the number of profiles (n_p).

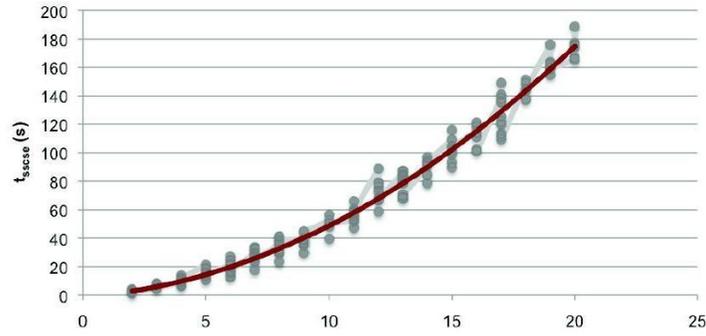


Figure 6: Computation time of Collection Subsumers Matrix vs. number of profiles

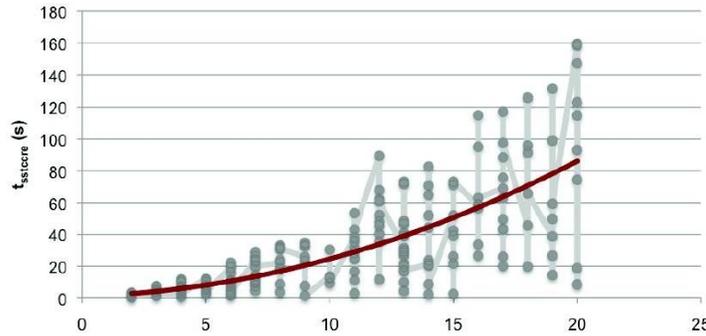


Figure 7: Computation time of Concept Subsumers Matrix vs. number of profiles

The functions are polynomial (second order) w.r.t. the number of profiles. The offsets from the function are due to the different number of concepts components generated by the profiles selected (Figure 6) or the competence in input (Figure 7) in the specific test. Figure 8 and 9 show, respectively, the time for computing collection and concept subsumers matrix in function of the number of concept components (constant number of profiles). In this second test category the resulting functions are almost linear. The offset from average times for tests with the same number of profiles and concept components is due to connection delays in the reasoner querying process.

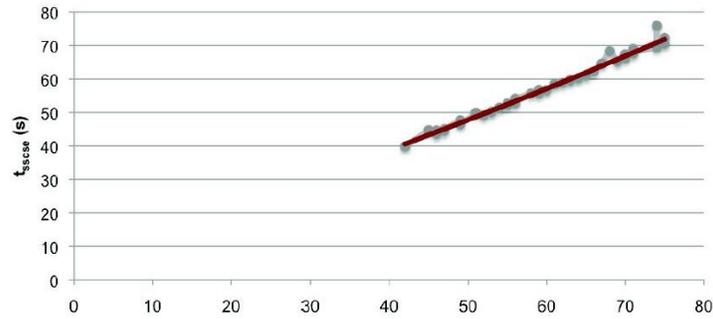


Figure 8: Computation time of Collection Subsumers Matrix vs. number of components($n_p = 11$)

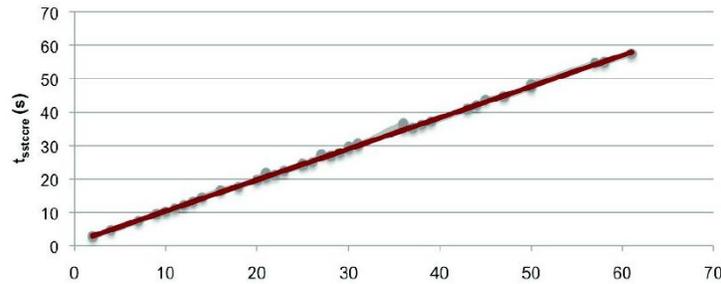


Figure 9: Computation time of Concept Subsumers Matrix vs. number of components($n_p = 11$)

6. Related Work

Recent research in human resources management is populated by several attempts to overcome limits of classical keyword-based retrieval by enriching resource descriptions through a semantic annotation. Non-logic approaches to retrieval show their limits in efficiently managing the informative content of

knowledge resources. As an example, database querying or search based on similarity between weighted vectors of stemmed terms –a typical approach of text-based Information Retrieval– have been used to evaluate possible matches between job offers and candidates [27]. Even if recent systems for ranking candidate answers w.r.t. a job offer have been proposed [28], forcing profiles in being expressed by data structures or terms vectors does not allow to deal with incomplete information, always present in matchmaking contexts as either unavailable or irrelevant data.

Skill matching has been also modeled as a graph containing two sets of nodes. The first one includes employees and the second one tasks to be performed [29]. Edges link people to tasks. A cost function associates each edge with a real value, hence a weighted graph ensues, where the association task-people results in a well known problem in Operational Research area: the “Assignment” one [30, 31, 32]. The assignment is only based on the evaluation of matching cost, which neither takes into account semantics nor is automatically computed. As a consequence such approaches show their limits both in effectiveness and in expressiveness.

In [33], two skill matching systems –*ProPer* and *OntoProper*– were presented, both storing in a database skill profiles represented as vectors. Approaches derived from decision theory are exploited to allow approximate matches, not allowed by plain database queries. *OntoProper* also embeds an ontology so reducing the effort for database maintenance. In fact, database information is updated/enriched with facts derived from secondary information sources, such as project documents. Unfortunately, both systems lack of an ontology modeling skills and a KB storing job positions, hence inferences on introduced profiles are made impractical.

In [34] two *People Finder Knowledge Management Systems* are proposed: the *Searchable Answer Generating Environment (SAGE)* and the *Expert Seeker*. Both use databases as skill repositories and a query engine for retrieving expertise (keyword-based as well as other search options are allowed). There is no possibility of both eliciting knowledge from skill matching operations and to keep automatically up to date profiles.

Also agent technologies have been employed to support the search of the right expert: in [35] it has been proposed an XML multi-agent system providing, among many other facilities, support in searching the most suitable employee for a specific job. Although agent-technologies allow for a high degree of automation in skill search, the proposed framework needs a significant formalization effort to achieve the expressiveness required for the knowledge representation problem.

The issue of performing the right modeling of competences to facilitate the further matching phases has been addressed by [36], where a web-based system for validating relationships among skills modeled in a graph structure is set. The proposed modeling does not handle semantics, causing the matching process not to be semantic-based.

In spite of the variety of non logic approaches proposed so far, the use of ontologies as knowledge repositories has been largely recognized as useful to provide a common vocabulary, to use inference services ([37], [38]) and to model

general Knowledge Management procedures [39].

Competence definition initiatives like IEEE Reusable Competency Definition (IEEE RCD)⁵ and HR-XML Measurable Competencies⁶ evidence the need for interoperable and machine understandable data structures for managing knowledge description.

In [40] an ontology-based skill management system is proposed, allowing employees to elicit their skills and contributing to an advanced expert search through the intranet. Such an approach only handles explicit information in the matching process and is mainly focused on competence gap analysis.

In [41] a system integrating the accuracy of a semantic-based search with the flexibility of a keyword-based search is proposed to find expertise within academia. The system is based on the use of Semantic Web technologies – particularly RDF and XML – in order to extract expertise profiles from heterogeneous information sources. On the one hand, a problem as complex as expert search would need a higher expressiveness for representation; on the other hand, the keyword-based search process apparently does not take semantics into account.

In [42] a novel, machine-understandable definition of competences is proposed able to take into account both proficiency and context awareness in definition of job profiles. Although the paper does not propose a modern, semantic-based, retrieval infrastructure, it provides a useful distinction of competences in *required* and *acquired* ones. Particularly, recruiter’s need are modeled as “strict” or “optional” in required competences and “quantitative” information helpful in ranking results (accomplishments, dates, marks, experiences), are added in acquired ones.

An OWL-DL competence modeling has been also proposed in [43] to support and integrate most of the aspects of knowledge management, from business process management to competence enhancement, through learning. The effort spent in ontology modeling is valuable, but the paper does not present services exploiting the proposed ontology. In [44] a query reformulation approach through rule rewriting is proposed with the aim of relaxing job requests, typically over-constrained and therefore leading to an unacceptably low number of retrieved candidates. The proposed process makes more flexible query formulation, but forces constraint relaxing of the whole query. No logic-based explanations are provided in case of approximate matches. The approach is implemented in a semantic-enabled job portal [45] based on a skill ontology which is part of the HR ontology developed for the *Knowledge Nets* project⁷, according to the guidelines proposed in [46]. The semantic job portal embeds a match engine relying on the framework proposed in [47], based on conceptual graphs matching [48]. The retrieval process ranks selected candidates by computing the similarity of thematic clusters identified in both job postings and workers

⁵<http://ieeeltsc.org/wg20Comp/wg20rcdfolder/>

⁶<http://www.hr-xml.org/>

⁷<http://wissensnetze.ag-nbi.de>

profiles. The matching process is therefore less accurate than ones based on single features comparison.

A relevant issue arises in *using* ontologies once they have been built, *i.e.*, reasoners and reasoning services must be designed and implemented to take full advantage from the effort placed in structuring an ontology. Although several semantic facilitators have been proposed in literature for several scenarios ([49], [33], [50]) often they do not fully leverage the ontological structure limiting their inferences to simple subsumption matching. The work in [16] gathers several semantic-based approaches to retrieval, based on specifically devised non-standard services in Description Logics. Thanks to them, the matchmaking process may be extended w.r.t. exact match. Also in [51] an approach seemingly similar in overcoming exact matches is proposed. It extends the one for measuring similarities in ontologies by Ehrig et al. [52] for combining the advantages of similarity-based search with those of ontology-based systems. Nevertheless, such an approach does not allow to provide explanation of results in case of non-exact matches.

In [53] an approach to skill matching, which may appear close to the one proposed in [16], is presented. It is based on the technique presented in [54] for ranking query results. The relevance of query outcomes is computed taking into account the structure of the underlying domain (using the Knowledge Base content) and the inference process characteristics. The ranking, though providing a useful support to the choice between the returned profiles, only classifies answers w.r.t. queries formalized with a well defined structure. Such an approach lacks then of expressiveness in the querying process. Moreover, the open world assumption is not made, because only answers explicitly providing characteristics required by the query are considered. Finally, no explanations are provided about the rationale in case of absence of match.

Nowadays, most of commonly employed recruitment systems are on-line job search sites that, therefore, need to handle a large amount of unstructured information describing job offers and profiles. In [55] a system supporting analysis and categorization of job offers and candidates responses is proposed. The system aims at automating the recruitment process by implementing strategies based on vectorial and probabilistic models to solve the problem of profiling applications according to a specific job offer. A noteworthy effort is placed in the information extraction, but the paper lacks of a semantic-based matching mechanism.

All the systems and approaches so far outlined deal with the search for skills among the available assignees. Creating new competencies when the available ones are not enough to perform all the needed tasks may represent a competitive advantage opportunity. In order to achieve such knowledge creation, KMSs may integrate components supporting the training process of employees, exploiting e-learning technologies. The term e-learning has become common, describing several concepts, from complete web-based courses to distance learning and tutoring. Recently, also thanks to various standardization efforts [56], emphasis has been placed on the concept of learning object *i.e.*, small and easily reusable educational resources to be composed to allow personalized instruction

and courseware creation [57, 58, 59, 60].

Obviously, discovery and composition of such learning objects in an automated way requires the association of unambiguous and semantically rich metadata, defined in accordance with shared ontologies ([61],[62]). The LOM –Learning Object Metadata [22]– standard, though limited in the basic annotation items, allows to freely define annotated metadata describing a learning resource.

The semantic-based annotation of educational resources is hence fully in the stream of the Semantic Web initiative [63], and it can share with it both techniques and approaches [64, 65, 61].

In particular, as more and more learning objects become available on the Web as services with well-defined machine interpretable interfaces as described *e.g.*, in OWL-S [66, 67], personalized learning units can be built by scratch, by retrieving learning resources. Automated composition of learning resources, exposed as web services for example, can then match a personalized learning need.

Recent studies [68] underline the rarity of approaches integrating knowledge and learning management: in this paper an approach of such kind has been proposed. The semantic-based integration of competences with learning needs is also tackled *e.g.*, in [69]. Nevertheless, relying on standard services and on RDQL, such systems cannot deal with approximation nor provide explanation services. An alternative proposal of integration may be found in [70], in which the composition of learning objects is based on the solution of a Concept Covering problem defined in terms of LCS computation. Such an approach, although dealing with approximation, does not provide explanation facilities.

One attempt to integration mostly focused on knowledge modeling is proposed in [71], which present a case study for modeling e-learning procedures in the general purpose ontology for knowledge management proposed in [39]. The paper nevertheless does not cope with automated search of knowledge resources.

6.1. Commercial Tools

Currently, several commercial tools for talent management⁸ and e-recruitment are available. Most of them are enterprise suites supporting human resource management, including solutions that, even though improving the recruitment process by means of innovative media and tools, do not introduce a significant novelty charge. Available solutions in fact exploit databases to store candidate personal and employment information, and do not ground on a logic-based structure.

To the best of our knowledge, one of the first logic-based solutions for recruitment and referral process is STAIRS⁹, a system in use at US Navy Department allowing to retrieve referral lists of best qualified candidates w.r.t. a specific

⁸<http://www.attract-hr.com/cm/about>,
http://www.oracle.com/applications/human_resources/irecruit.html
⁹<http://www.hrojax.navy.mil/forms/selectguide.doc>

mansion, according to the number of required skills they match. The commercial software supporting STAIRS is RESUMIX¹⁰ an automated staffing tool making use of artificial intelligence techniques and adopted only as an internal tool. The system allows also to distinguish skills in *required* and *desired* ones in the query formulation: all required skills must be matched by the retrieved candidate, differently from *desired* ones. No explanation is returned for the proposed solutions.

Among semantic-based tools for recruitment, noteworthy are products offered by Sovren¹¹, which provide solutions for both CV and job requests parsing starting from several text formats to HR-XML schema. A semantic-based matchmaking engine automatically returns a ranked list of best candidates given a job vacation, posted in any textual format. The matching process is by the way completely hidden to the recruiter, which does not receive any explanation about retrieved results. Sovren tools also manage the distinction between required and preferred skills in job offers posting.

Last year *Monster.com*^(R), the leading web job matching engine, introduced *MonsterPowerResumeSearch*TM¹² to improve the relevance of candidates recruitment. The product relies on the semantic *6Sense*^(TM) search technology, patented by Monster Worldwide, Inc. itself. Thanks to such a semantic search technology, the novel engine retrieves candidates according to most of their qualifications including how closely they match job requirements, the recency of their experience, how they compare to similar candidates. The query interface allows for distinguishing between essential and nice-to-have skills and allows the user to describe her request in natural language keywords. The search process exploits the semantics of queries and available candidate profiles to rate them w.r.t. to the entered search criteria and rank them according to the comparison between each other.

Summarizing, the proposed KMS presents several distinguishing features w.r.t. approaches existing both in literature and in commerce, above reported. It is a semantic-based system, which incorporates in a unique modeling framework knowledge management solutions related to different organizational needs, rarely –if ever– faced from an integrated perspective. To the best of our knowledge, it is among the few systems fully exploiting the knowledge representation effort spent in modeling informative resources. The KMS proposes, in fact, significant explained solutions double-tied with Description Logics inferences. Moreover, such inferences have been specifically developed for explanation purpose, causing functionalities provided by the system to be original.

¹⁰<http://www.cpol.army.mil>

¹¹<http://www.sovren.com/default.aspx>

¹²http://info.monster.com/products/power_resumearch.asp

7. Final Remarks

A growing interest in solutions extracting more informative and meaningful results from knowledge resources is witnessed both in literature and in actual commercial systems. Fully in this trend, we propose the integration in a single knowledge management system of a variety of advanced semantic-based services supporting competence management. The KMS treats competence as any other material asset, and exploits the knowledge sources descriptions stored in a shared repository, working as a competence warehouse. The employed formalism for knowledge representation, Description Logics, on the one hand ensures interoperability of our services with the languages developed for the Semantic Web Initiative and on the other hand allows all implemented solutions to exploit non-standard DL inferences devised to overcome some limits of available standard ones. The implementation of such non-standard inferences ensures the novelty of our system by distinguishing it from any other available solution.

The main contribution of the paper is the seamless integration of recently proposed services supporting long term strategical choices in a KMS devoted to human resources allocation and training programs planning. Our KMS allows in fact for searching in the warehouse for the knowledge resources needed to solve a task in three different multiplicity cases. Furthermore, our system supports the identification of flexible learning resources present in the repository to plan training programs aimed at covering skills missing in the figurative warehouse. The introduced services supporting long term strategical choices add a third class of functionalities for the evaluation of company Core Competence and the determination of knowledge clusters the company possibly belongs to. In particular, new non-standard inferences have been defined for the identification of company Core Competence from the knowledge warehouse and related computation algorithm have been implemented and evaluated.

Computation algorithms for such inferences in DLs more expressive than \mathcal{ALN} have been object of our investigation [20] and led to the introduction of a tableau-based framework for Least Common Subsumer computation [72], recently generalized for the solution of several non-standard inferences [73]. The implementation of such a framework is object of our current research work, aimed at the release of a novel KMS version able to handle more expressive descriptions in the knowledge warehouse while still solving introduced non-standard inferences crucial for competence management.

References

- [1] K. Hafeez, Y. Zhang, N. Malak, Core competence for sustainable competitive advantage: a structured methodology for identifying core competence, *Engineering Management, IEEE Transactions on* 49 (1) (2002) 28–35.
- [2] B. Wernerfelt, The Resource-Based View of the Firm: Ten Years After, *Strategic Management Journal* 16 (3) (1995) 171–174.

- [3] B. Wernerfelt, A Resource-Based View of the Firm, *Strategic Management Journal* 5 (2) (1984) 171–180.
- [4] J. A. L. Halawi, R. McCarthy, Resource-Based View of Knowledge Management for Competitive Advantage, *The Electronic Journal of Knowledge Management* 3 (2) (2005) 75–86.
- [5] P. Meso, R. Smith, A Resource-Based View of Organizational Knowledge Management Systems, *Journal of Knowledge Management* 4 (3) (2000) 224–231.
- [6] J. B. Barney, Strategic Factor Markets: Expectations, Luck and Business Strategy, *Management Science* 32 (1986) 1231–1241.
- [7] J. B. Barney, Firm Resources and Sustained Competitive Advantage, *Journal of Management* 17 (1) (1991) 99–120.
- [8] D. J. Teece, G. Pisano, A. Shuen, Dynamic capabilities and strategic management, *Strategic Management Journal* 18 (7) (1997) 509–533.
- [9] M. Tampoe, Exploiting the core competences of your organization, *Long Range Planning* 27 (4) (1994) 66 – 77, ISSN 0024-6301, doi:\bibinfo{doi}{DOI:10.1016/0024-6301(94)90057-4}, URL <http://www.sciencedirect.com/science/article/B6V6K-45K4F3J-7/2/bbb97f402644fd446a6dfb0cb7a3db61>.
- [10] R. Sanchez, A. Heene, Reinventing strategic management: New theory and practice for competence-based competition, *European Management Journal* 15 (3) (1997) 303 – 317, ISSN 0263-2373, doi:\bibinfo{doi}{DOI:10.1016/S0263-2373(97)00010-8}, URL <http://www.sciencedirect.com/science/article/B6V9T-3SWXWWW-9/2/e4ae2d056e3ef23222e396122a08559b>.
- [11] W. Bogner, H. Thomas, Core competencies and competitive advantage: a model and illustrative evidence from the pharmaceutical industry, in: G. Hamel, A. Heene (Eds.), *Competencies-based Competition*, Wiley, New York, 111–144, 1994.
- [12] G. Hamel, C. K. Prahalad, The core competence of the corporation, *Harvard Business Review* May-June (1990) 79–91.
- [13] C. C. Markides, P. J. Williamson, Related Diversification, Core Competences and Corporate Performance, *Strategic Management Journal* 15 (1994) 49–65.
- [14] R. R. Nelson, Why do firms differ, and how does it matter?, *Strategic Management Journal* 12 (1991) 61–74.

- [15] S. Colucci, E. Di Sciascio, F. M. Donini, A Knowledge-based Solution for Core Competence Evaluation in Human-Capital Intensive Companies, in: I-KNOW-08, Proceedings of 8th International Conference on Knowledge Management, 259–266, 2008.
- [16] S. Colucci, T. Di Noia, E. Di Sciascio, F. M. Donini, A. Ragone, Semantic-based Skill Management for Automated Task Assignment and Courseware Composition, *Journal of Universal Computer Science* 13 (9) (2007) 1184–1212.
- [17] F. Baader, D. Calvanese, D. Mc Guinness, D. Nardi, P. Patel-Schneider (Eds.), *The Description Logic Handbook*, Cambridge University Press, 2002.
- [18] S. Colucci, T. Di Noia, E. Di Sciascio, F. M. Donini, A. Ragone, Measuring Core Competencies in a Clustered Network of Knowledge, in: C. Sary, F. Barachini, S. Hawamdeh (Eds.), *Knowledge management: Innovation, technology and cultures*, vol. 6 of *Innovation and Knowledge Management*, World Scientific, 279–291, 2007.
- [19] E. Tinelli, A. Cascone, M. Ruta, T. D. Noia, E. D. Sciascio, F. M. Donini, I.M.P.A.K.T.: An Innovative Semantic-based Skill Management System Exploiting Standard SQL, in: J. Cordeiro, J. Filipe (Eds.), *ICEIS* (2), ISBN 978-989-8111-85-2, 224–229, 2009.
- [20] S. Colucci, E. D. Sciascio, F. M. Donini, E. Tinelli, Finding informative commonalities in concept collections, in: *CIKM 2008, Proceedings of the 17th ACM Conference on Information and Knowledge Management*, Napa Valley, California, USA, October 26-30, ACM, 807–817, 2008.
- [21] W. W. Cohen, A. Borgida, H. Hirsh, Computing Least Common Subsumers in Description Logics, in: P. Rosenbloom, P. Szolovits (Eds.), *Proceedings of the Tenth National Conference on Artificial Intelligence*, AAAI Press, Menlo Park, California, 754–761, 1993.
- [22] IEEE, *IEEE Standard for Learning Object Metadata*, std 1484.12.1-2002 edn., 2002.
- [23] S. Bechhofer, R. Mller, P. Crowther, The DIG Description Logic Interface, in: *Proceedings of the Sixteenth International Workshop of Description Logics*, vol. 81 of *CEUR Workshop Proceedings*, 2003.
- [24] T. O’Reilly, *O’Reilly – What Is Web 2.0*, <http://www.oreillynet.com/pub/a/oreilly/tim/news/2005/09/30/what-is-web-20.html>, 2005.
- [25] P. Bollmann, F. Jochum, U. Reiner, V. Weissmann, H. Zuse, The LIVE-project: retrieval experiments based on evaluation viewpoints, in: *Research and Development in Information Retrieval*, 213–214, 1985.

- [26] H. Kuhn, The Hungarian method for the Assignment Problem, *Naval Research Logistics Quarterly* 2 (1955) 83–97.
- [27] D. Veit, J. Muller, M. Schneider, B. Fiehn, Matchmaking for Autonomous Agents in Electronic Marketplaces, in: *Proceedings of International Conference on Autonomous Agents*, ACM, 65–66, 2001.
- [28] R. Kessler, N. Béchet, M. Roche, M. El-Bèze, J. M. Torres-Moreno, Automatic Profiling System for Ranking Candidates Answers in Human Resources, in: *OTM '08: Proceedings of the OTM Confederated International Workshops and Posters on On the Move to Meaningful Internet Systems*, Springer-Verlag, Berlin, Heidelberg, ISBN 978-3-540-88874-1, 625–634, 2008.
- [29] H. B. Saip, C. Lucchesi, Matching algorithms for bipartite graph. Technical Report DCC -03/93. Departamento de Cincia da Computao, Universidade Estadual de Campinas, 1993.
- [30] Z. Galil, Efficient algorithms for finding maximum matching in graphs, *ACM Computer Survey* 18 (1) (1986) 23–38.
- [31] F. S. Hillier, G. J. Lieberman, *Introduction to Operations Research*, McGraw-Hill, New York, 1995.
- [32] J. Kennington, Z. Wang, An empirical analysis of the dense assignment problem: Sequential and parallel implementations, *ORSA Journal on Computing* 3 (4) (1991) 299–306.
- [33] Y. Sure, A. Maedche, S. Staab, Leveraging Corporate Skill Knowledge - From ProPer to OntoProPer, in: *Proceedings of the Third International Conference on Practical Aspects of Knowledge Management*, 2000.
- [34] I. Becerra-Fernandez, The Role of Artificial Intelligence Technologies in the Implementation of People-Finder Knowledge Management Systems, *Knowledge-Based Systems* 13 (5) (2000) 315–320.
- [35] A. Garro, L. Palopoli, An XML Multi-agent System for E-learning and Skill Management, in: *Agent Technologies, Infrastructures, Tools, and Applications for E-Services*, vol. 2592 of *LCNS*, Springer, 283–294, 2003.
- [36] P. Keenan, Human Resource Management DSS, in: *Decision Support in an Uncertain and Complex World: The IFIP TC8/WG8.3 International Conference 2004*, 2004.
- [37] D. OLeary, Enterprise knowledge management, *IEEE Computer* 31 (3) (1998) 54–61.
- [38] D. OLeary, Using AI in knowledge management: Knowledge bases and ontologies, *IEEE Intelligent Systems* 13 (3) (1998) 34–39.

- [39] C. W. Holsapple, K. D. Joshi, A formal knowledge management ontology: Conduct, activities, resources, and influences: Research Articles, *Journal of the American Society for Information Science and Technology* 55 (7) (2004) 593–612, ISSN 1532-2882.
- [40] T. Lau, Y. Sure, Introducing ontology-based skills management at a large insurance company, in: *Proceedings of the Modellierung, 2002*.
- [41] P. Liu, P. Dew, Using Semantic Web Technologies to Improve Expertise Matching within Academia, in: *I-KNOW-04, Proceedings of 4th International Conference on Knowledge Management, J.UCS, 370–378, 2004*.
- [42] J. L. D. Coi, E. Herder, A. Koesling, C. Lofi, D. Olmedilla, O. Papapetrou, W. Siberski, A Model for Competence Gap Analysis, in: *Proceedings of the Third International Conference on Web Information Systems and Technologies: Internet Technology/ Web Interface and Applications, INSTICC Press, Barcelona, Spain, 2007*.
- [43] A. Schmidt, C. Kunzmann, Towards a Human Resource Development Ontology for Combining Competence Management and Technology-Enhanced Workplace Learning, in: R. Meersman, Z. Tahiri, P. Herero (Eds.), *On The Move to Meaningful Internet Systems 2006: OTM 2006 Workshops. Part I. 1st Workshop on Ontology Content and Evaluation in Enterprise (Onto-Content 2006)*, vol. 4278 of *Lecture Notes in Computer Science*, Springer, 1078–1087, 2006.
- [44] M. Mochol, A. Jentzsch, H. Wache, Suitable employees wanted? Find them with semantic techniques, in: *European Semantic Technology Conference 2007, Vienna, Austria, making Semantics Work For Business, 2007*.
- [45] M. Mochol, H. Wache, L. J. B. Nixon, Improving the Accuracy of Job Search with Semantic Techniques, in: W. Abramowicz (Ed.), *BIS*, vol. 4439 of *Lecture Notes in Computer Science*, Springer, ISBN 978-3-540-72034-8, 301–313, 2007.
- [46] M. Mochol, E. Paslaru, B. Simperl, Practical Guidelines for Building Semantic eRecruitment Applications, in: *I-KNOW-06, Proceedings of 6th International Conference on Knowledge Management, special Track: Advanced Semantic Technologies (AST), 2006*.
- [47] R. Oldakowski, C. Bizer, SemMF: A Framework for Calculating Semantic Similarity of Objects Represented as RDF Graphs, Poster at the 4th International Semantic Web Conference (ISWC 2005) .
- [48] J. Zhong, H. Zhu, J. Li, Y. Yu, Conceptual Graph Matching for Semantic Search, in: *ICCS '02: Proceedings of the 10th International Conference on Conceptual Structures*, Springer-Verlag, London, UK, 92–196, 2002.
- [49] S. Staab, H. Schnurr, R. Studer, Y., Knowledge Processes and Ontologies, *IEEE Intelligent Systems* 16 (1).

- [50] D. Trastour, C. Bartolini, C. Priest, Semantic Web Support for the Business-to-Business E-Commerce Lifecycle, in: Proceedings of the Eleventh International World Wide Web Conference, ACM, 89–98, 2002.
- [51] E. Biesalski, A. Abecker, Similarity Measures for Skill-Profile Matching in Enterprise Knowledge Management, in: Y. Manolopoulos, J. Filipe, P. Constantopoulos, J. Cordeiro (Eds.), ICEIS (2), ISBN 972-8865-41-4, 11–16, 2006.
- [52] M. Ehrig, P. Haase, N. Stojanovic, M. Hefke, Similarity for Ontologies — A Comprehensive Framework, in: D. Bartman, F. Rajola, J. Kallinikos, D. Avison, R. Winter, P. Ein-Dor, J. Becker, F. Bodendorf, C. Weinhardt (Eds.), Information Systems in a Rapidly Changing Economy: Proceedings of the 13th European Conference on Information Systems (ECIS 2005), May 26-28, 2005, Regensburg, Germany, 2005.
- [53] M. Hefke, L. Stojanovic, An Ontology-Based Approach for Competence Bundling And Composition of ad-hoc Teams in an Organization, in: I-KNOW-04, Proceedings of 4th International Conference on Knowledge Management, 2004.
- [54] R. S. N. Stojanovic, L. Stojanovic, An Approach for the Ranking of Query Results in the Semantic Web, in: Proceedings of the Second International Semantic Web Conference, 2003.
- [55] R. Kessler, N. Béchet, J. M. T. Moreno, M. Roche, M. El-Bèze, Job Offer Management: How Improve the Ranking of Candidates, in: J. Rauch, Z. W. Ras, P. Berka, T. Elomaa (Eds.), ISMIS, vol. 5722 of *Lecture Notes in Computer Science*, Springer, ISBN 978-3-642-04124-2, 431–441, 2009.
- [56] IEEE, IEEE standard for learning technology-learning technology systems architecture (LTSA), std 1484.1-2003 edn., 2003.
- [57] G. Vossen, P. Jaeschke, Learning objects as a uniform foundation for e-learning platforms, in: Proceedings of the International Symposium on Database Engineering and Applications, 278–287, 2003.
- [58] K. Ajami, Specifying and implementing interoperable and reusable learning objects: one step beyond, in: Proceedins of International Conference on Information and Communication Technologies: From Theory to Applications, 111–112, 2004.
- [59] A. Cabezuelo, J. Beardo, Towards a model of quality for learning objects, in: Proceedings of International Conference on Advanced Learning Technologies, IEEE, 822–825, 2004.
- [60] A. Ip, A. Young, I. Morrison, Learning objects - Whose are they?, in: Proceedings of 15th Conference of the National Advisory Committee on Computing Qualifications, 315–320, 2002.

- [61] S. Sanchez, M. Sicilia, On the semantics of aggregation and generalization in learning object contracts, in: Proc. of Intl. Conf. on Advanced Learning Technologies, IEEE, 425–429, 2004.
- [62] M. Sicilia, M. Lytras, The semantic learning organization, The learning organization: An international journal 12 (5) (2005) 402–410.
- [63] T. Berners-Lee, J. Hendler, O. Lassila, The Semantic Web, Scientific American 248 (4) (2001) 34–43.
- [64] N. Bennacer, Y. Bourda, B. Doan, Formalizing for querying learning objects using OWL, in: Proceedings of International Conference on Advanced Learning Technologies, IEEE, 321–325, 2004.
- [65] D. Gasevic, J. Jovanovic, V. Devedzic, Enhancing learning object content on the semantic web, in: Proceedings of International Conference on Advanced Learning Technologies, IEEE, 714–716, 2004.
- [66] OWL-S, The OWL Services Coalition, www.daml.org/services/owl-s/1.0/owl-s.html, 2004.
- [67] K. P. Sycara, M. Paolucci, A. Ankolekar, N. Srinivasan, Automated discovery, interaction and composition of Semantic Web services, J. Web Sem. 1 (1) (2003) 27–46.
- [68] F. Draganidis, G. Mentzas, Competency Based Management: A Review of Systems and Approaches, Information Management and Computer Security 14 (1) (2006) 51–64.
- [69] F. Draganidis, P. Chamopoulou, G. Mentzas, An Ontology Based Tool for Competency Management and Learning Paths, in: I-KNOW-06, Proceedings of 6th International Conference on Knowledge Management, 2006.
- [70] N. Karam, S. Linckels, C. Meinel, Semantic Composition of Lecture Subparts for a Personalized e-Learning, in: Proceedings of the Fourth European Semantic Web Conference, vol. 4519 of *Lecture Notes in Computer Science*, Springer, 716–728, 2007.
- [71] M. A. Sicilia, M. Lytras, E. Rodriguez, E. Garcia-Barriocanal, Integrating descriptions of knowledge management learning activities into large ontological structures: a case study, Data Knowledge Engineering 57 (2) (2006) 111–121, ISSN 0169-023X.
- [72] F. M. Donini, S. Colucci, T. D. Noia, E. D. Sciascio, A Tableaux-Based Method for Computing Least Common Subsumers for Expressive Description Logics, in: IJCAI 2009, Proceedings of the 21st International Joint Conference on Artificial Intelligence, Pasadena, California, USA, July 11–17, 739–745, 2009.

- [73] S. Colucci, T. D. Noia, E. D. Sciascio, F. M. Donini, A. Ragone, A Unified Framework for Non-standard Reasoning Services in Description Logics, in: ECAI 2010, Proceedings of the 19th European Conference on Artificial Intelligence, Lisbon, Portugal, August 16-20, 479–484, 2010.
- [74] DAML+OIL, DAML+OIL Specifications, www.daml.org/2001/03/daml+oil-index.html, 2001.
- [75] D. McGuinness, R. Fikes, J. Hendler, L. Stein, DAML+OIL: An Ontology Language for the Semantic Web, IEEE Intelligent Systems 17 (5) (2002) 72–80.
- [76] OWL, www.w3.org/TR/owl-features/, 2004.
- [77] OWL 2 Web Ontology Language Structural Specification and Functional-Style Syntax, www.w3.org/TR/2009/REC-owl2-syntax-20091027/, 2009.
- [78] F. Baader, I. Horrocks, U. Sattler, Description Logics as Ontology Languages for the Semantic Web, in: D. Hutter, W. Stephan (Eds.), Festschrift in honor of Jörg Siekmann, Lecture Notes in Artificial Intelligence, SV, 2003.
- [79] I. Horrocks, U. Sattler, S. Tobies, Practical Reasoning for Expressive Description Logics, in: H. Ganzinger, D. McAllester, A. Voronkov (Eds.), Proceedings of the 6th International Conference on Logic for Programming and Automated Reasoning (LPAR’99), no. 1705 in Lecture Notes in Artificial Intelligence, Springer-Verlag, 161–180, 1999.
- [80] I. Horrocks, O. Kutz, U. Sattler, The Even More Irresistible *SR_QIQ*, in: Proc. of the 10th Int. Conf. on Principles of Knowledge Representation and Reasoning (KR 2006), AAAI Press, 57–67, 2006.
- [81] A. Borgida, P. F. Patel-Schneider, A Semantics and Complete Algorithm for Subsumption in the CLASSIC Description Logic, Journal of Artificial Intelligence Research 1 (1994) 277–308.

Appendix A. Basic Description Logics

DLs are a family of formalisms and reasoning services widely employed for knowledge representation in a decidable fragment of First Order Logic.

The alphabet of each DL is therefore made up by unary and binary predicates, denoted as **Concepts Names** and **Role Names**, respectively. The domain of interest is represented through more expressive and complex **Concept Descriptions**, involving *constructors* over concept and role names. The set of constructors allowed by a DL characterizes it in terms of expressiveness and reasoning complexity: of course the more a DL is expressive, the harder is inferring new knowledge on its descriptions.

Concept Definitions allow to assign an unique concept name to complex concept descriptions: the so called **Unique Name Assumption** (UNA) holds

in every DL. Names associated to concept descriptions constitute the set of *Defined Concepts*, distinguished by *Primitive Concepts* not appearing on the left hand side of any concept definition and corresponding to the concept names. Defined and primitive concepts constitute the set of concept names, which will be referred to by the letter A in the following; C and D will denote instead arbitrary concept descriptions.

Concept Inclusions detail instead specificity hierarchies among concepts, either defined or primitives.

The set of concepts inclusions and definitions represents the formal representation of the domain of interest, the intensional knowledge which takes the name **TBox** in DL systems and **Ontology** in the generic knowledge representation framework. TBoxes containing recursive concept definitions are called *cyclic* (*acyclic* otherwise).

The semantic of concept descriptions is conveyed through an **Interpretation** $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$, where $\Delta^{\mathcal{I}}$ is a non empty set denoting the domain of \mathcal{I} and $\cdot^{\mathcal{I}}$ is an interpretation function such that:

- $\cdot^{\mathcal{I}}$ maps each concept name A in a set $A^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}}$
- $\cdot^{\mathcal{I}}$ maps each role name R in a binary relation $r^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$

Possible DL constructors and the related semantics are shown in Table A.6¹³.

Constructor Name	Syntax	Semantics
top-concept	\top	$\Delta^{\mathcal{I}}$
bottom-concept	\perp	\emptyset
full negation	$\neg C$	$\Delta^{\mathcal{I}} \setminus C^{\mathcal{I}}$
atomic negation	$\neg A$	$\Delta^{\mathcal{I}} \setminus A^{\mathcal{I}}$
conjunction	$C \sqcap D$	$C^{\mathcal{I}} \cap D^{\mathcal{I}}$
disjunction	$C \sqcup D$	$C^{\mathcal{I}} \cup D^{\mathcal{I}}$
value restriction	$\forall r.C$	$\{x \in \Delta^{\mathcal{I}} \mid \forall y : (x, y) \in r^{\mathcal{I}} \rightarrow y \in C^{\mathcal{I}}\}$
existential restriction	$\exists r.C$	$\{x \in \Delta^{\mathcal{I}} \mid \exists y : (x, y) \in r^{\mathcal{I}} \wedge y \in C^{\mathcal{I}}\}$
at-least restriction	$(\geq n r)$	$\{x \in \Delta^{\mathcal{I}} \mid \#\{y \in \Delta^{\mathcal{I}} \mid (x, y) \in r^{\mathcal{I}}\} \geq n\}$
at-most restriction	$(\leq m r)$	$\{x \in \Delta^{\mathcal{I}} \mid \#\{y \in \Delta^{\mathcal{I}} \mid (x, y) \in r^{\mathcal{I}}\} \leq m\}$

Table A.6: DLs Set of Constructors

The behavior of the interpretation function in presence of concept definitions and inclusions is detailed in Table A.7 and can be inductively explained by taking the semantics of allowed constructors into account.

An interpretation \mathcal{I} is a **model** for a TBox \mathcal{T} if it satisfies the whole set of assertions in \mathcal{T} .

A DL system embeds also an assertional component, which takes the name **ABox** and includes the following two possible elements:

¹³We notice that, given a set S , we denote by $\#S$ its cardinality

	Syntax	Semantics
concept definition	$A \equiv C$	$A^{\mathcal{I}} = C^{\mathcal{I}}$
concept inclusion	$A \sqsubseteq D$	$A^{\mathcal{I}} \subseteq D^{\mathcal{I}}$

Table A.7: Syntax and Semantics of TBox assertions

- **Concept assertions:** $C(a)$ states that individual a belongs to the concept description C
- **Role assertions:** $R(a, b)$ states that individual a participates to role R with individual b

An interpretation \mathcal{I} assigns an element $a^{\mathcal{I}}$ of the domain Δ to each individual a and is a **model** for a ABox \mathcal{A} if it satisfies $(a^{\mathcal{I}}, b^{\mathcal{I}}) \in R^{\mathcal{I}}$ for all role assertions $R(a, b) \in \mathcal{A}$ and $a^{\mathcal{I}} \in C^{\mathcal{I}}$ for all concept assertions $C(a) \in \mathcal{A}$.

The proposed KMS adopts \mathcal{ALN} sublanguage of DLs for knowledge representation. The full expressiveness of \mathcal{ALN} is explained, with the aid of constructors usage examples, in Table A.8.

Ontologies using DL can be easily modeled using languages for the Semantic Web [74, 75, 76, 77]. These languages have been conceived to allow for representation of machine understandable, unambiguous, description of web content through the creation of domain ontologies, and aim at increasing openness and interoperability in the web environment. The strong relations between DLs and the above introduced languages for the Semantic Web [78] is also evident in the definition of the OWL sub-languages (OWL-Lite, OWL-DL, OWL-Full), which reach the same expressiveness as *SHOIN*[79]. One more evidence may be found in the introduction of OWL2, which extends the semantics of the description logic *SROIQ*[80].

The choice of \mathcal{ALN} may appear limiting in terms of expressiveness, but such a languages allows for keeping bounded the complexity of reasoning. In particular standard inferences may be computed in polynomial time in \mathcal{ALN} [81] and non-standard reasoning may be exploited to provide crucial advanced services, also solvable in polynomial time.

The most important service characterizing reasoning in DL checks for specificity hierarchies, by determining whether a concept description is more specific than another one or, formally, if there is a *subsumption* relation between them.

Definition 10 (Subsumption). *Given two concept descriptions C and D and a TBox T in a DL \mathcal{L} , we say that D subsumes C w.r.t. T if for every model of T , $C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$. We write $C \sqsubseteq_T D$, or simply $C \sqsubseteq D$ if we assume an empty TBox.*

For example, consider the following concept descriptions, referred to a required task and a personnel profile, respectively:

- $T_1 = \forall \text{basKnowledge.programming}$

Expression	Example	Explanation
top-concept		whole domain
bottom-concept		empty set
atomic negation	$\neg\text{Skill}$	concepts which are not skills
conjunction	$\text{C++} \sqcap \text{Java}$	knowledge about both Java and C++
value restriction	$\forall\text{knowsLanguage.English}$	elements of the domain knowing only English as foreign language
at-least restriction	$(\geq 3 \text{ years})$	elements of the domain endowed with more than 3 years of working experience
at-most restriction	$(\leq 2 \text{ knowsLanguage})$	elements of the domain knowing not more than two foreign language
concept definition	$\text{Engineer} (\geq 1 \text{ hasMasterDegree}) \sqcap \forall\text{hasMasterDegree.Engineering} \equiv$	An engineer is an element of the domain endowed with at least one degree, which needs to be in engineering
concept inclusion	$\text{Consultant} \sqsubseteq \neg\text{Employee}$	Elements of the domain working as consultants may not be included in employees set

Table A.8: \mathcal{ALN} expressiveness explained

- $P_1 = \forall \text{basKnowledge.Java} \sqcap \forall \text{hasMasterDegree.ComputerScience} \sqcap \text{Consultant}$

Considering a TBox with the two following concept inclusions $\text{Java} \sqsubseteq \text{OOP}$ and $\text{OOP} \sqsubseteq \text{Programming}$, knowledge expressed by P_1 is more specific than the one required by T_1 : according to the previous definition T_1 subsumes P_1 .

Several widely used services may be reduced to subsumption, as shown in the following definitions.

Definition 11 (Concept Equivalence). *Given two concept descriptions C and D and a TBox \mathcal{T} in a DL \mathcal{L} , we say that D is equivalent to C ($C \equiv_{\mathcal{T}} D$) iff $C \sqsubseteq_{\mathcal{T}} D$ and $D \sqsubseteq_{\mathcal{T}} C$.*

Intuitively, two concept descriptions are said equivalent if they, regardless of syntactic differences, subsume each other. For example, the descriptions P_1 and T_1 introduced before are not equivalent, because P_1 does not subsume T_1 .

Definition 12 (Concept Satisfiability). *Given a concept descriptions C and a TBox \mathcal{T} in a DL \mathcal{L} , we say that C is satisfiable w.r.t. \mathcal{T} if \perp does not subsume C ($C \not\sqsubseteq_{\mathcal{T}} \perp$), unsatisfiable otherwise.*

Intuitively, a concept description is satisfiable w.r.t. \mathcal{T} if there exists any model in \mathcal{T} assigning a non empty subset of the domain to it, if it can be somehow interpreted in the knowledge domain.

For example, consider the following profile, adding new features to P_1 :

- $P_2 = \forall \text{basKnowledge.Java} \sqcap \forall \text{hasMasterDegree.ComputerScience} \sqcap \text{Consultant} \sqcap \text{Employee}$

If \mathcal{T} contains the concept inclusion in Table A.8, P_2 is unsatisfiable w.r.t. \mathcal{T} , because no interpretation can be given in \mathcal{T} to a concept describing an element working both as a consultant and as an employee.