# A MODEL FOR COMPETENCE GAP ANALYSIS

Keywords: competence, competency, model, gap analisys, lifelong learning

Abstract: Modeling competences is an integral part of many Human Resource (HR) and e-Learning related activities. HR departments use competence descriptions to define requirements needed for performing specific tasks or jobs. The same competences are acquired by employees and applicants by e.g. experience or certifications. Typically, HR departments need to match such required and acquired competences in order to find suitable candidates. In e-Learning a similar situation arises. Curricula or training programmes need to describe prerequisites that must be fulfilled before joining and the competences that will be acquired after successful completion. This paper analyses the limitations and extends existing approaches for modeling competences in order to allow (semi-)automatic competence matching.

## 1 Introduction

Nowadays, people mobility has increased. Learners may study abroad with the benefits of improving their language skills, receiving a better certification, or specializing in a topic not available in their regions. The same applies to the labour market. People do not need to restrict to their city or region while seeking for a job but may consider offers in other countries, too. This situation complicates the already difficult job of managers in learning organizations and Human Resource (HR) departments to decide who may have the right qualifications to join a project or the company itself. For learning organizations, requirements to join the programme must be taken into account. For example, an applicant needs to possess a Bachelor degree to apply for Master studies; in order to attend an expert course on a topic, a certification on a basic level may be required. Furthermore, assuming that an applicant fulfills such requirements, exemptions could be granted for parts of the programme that are similar to earlier followed courses. Imagine a Mathematician starting a Computer Science degree. Most likely, courses like Algebra and Statistics could be exempted. In the case of Human Resource departments, the task is equally complex. HR experts need to match applicant or employee experience and knowledge with the requirements of a job offer or a project, including both mandatory requirements and desired ones (e.g., Business English is required and French would be a plus). Currently, all these competence matches have to be performed manually, with hardly any guidelines or support. One important reason for this is that there are currently no sufficiently expressive common formats for the representation of competences, which is needed for complex competence profiles and requirements. Some initiatives, such as the IEEE Reusable Competency Definition (IEEE RCD, 2005) and HR-XML (HR-XML, 2004), have done initial steps to define common models and schemas for interoperability, but their current work lacks some important information that is required for competence matching, like proficiency levels or context (cf. Section 3), or for increasing reusability. In this paper, we enhance and extend the work that has been developed under the various initiatives and introduce a model for representing competences with their relationships as well as some usage profiles (such as profiles for job requirements description or for learner achievements description). This model provides the basis for allowing advanced (semi-)automatic competence matching and gap analysis.

This paper is organized as follows: section 2 clarifies the terms used throughout the paper and briefly introduces our requirements for modeling competences. Section 3 provides an overview of existing modeling specifications, and section 4 describes our modeling approach for competences in a more detailed manner. Section 5 introduces competence profiles (collections of competences) which represent the most visible aspect of competence modeling in real-life applications. Section 6 gives an example on how a simple profile and related competences can be modeled. Finally, section 7 concludes this paper with a summary and an outlook on future work.

## 2 What is a competence?

In this work we adopt the definition of *competence* as "effective performance within a domain/context at different levels of proficiency", as given in (Cheetam and Chivers, 2005). Note that there exists some confusion on the term *competency*[1] in the literature. (IEEE RCD, 2005; IMS RDCEO, 2002) define the stricter term of competency as "any form of knowledge, skill, attitude, ability, or learning objective that can be described in a context of learning, education or training". This definition is insufficiently expressive for competence gap analysis. For example, it is not clear if "piloting" covers both the ability to pilot a small plane and to pilot a big passenger airplane. Or if the competency "English writing skills" represents a specific level such as intermediate, fluent, native or simply the existence of the competency. In fact, if that information becomes part of the competency definition, its reusability is drastically reduced (with the consequence of, e.g., having different competency definitions for each context in which a competency is applied, and for any proficiency level and proficiency level scale). The definition given in (HR-XML, 2004) tries to extend the previous one: "A specific, identifiable, definable, and measurable knowledge, skill, ability and/or other deployment-related characteristic (e.g., attitude, behavior, physical ability) which a human resource may possess and which is necessary for, or material to, the performance of an activity within a specific business context". In this case, "measurable" indicates a relationship with a specific proficiency level[2] and competency now applies only to the business context. In any case, since context is implicit, the models proposed from these specifications do not include context information.

As stated above, current approaches to modeling competencies do not explicitely address proficiency level and context. On the contrary, we believe that competency, proficiency level and context are three different dimensions that should be modeled separately in order to maximize their reuse. For example, the same competencies may be used in different contexts, or the same proficiency level scales may be reused among different certifications. The same applies to contexts (or "domain models"), which in many situations already exist and therefore may be reused by competences. Therefore, according to what stated above, we model competence (plural:competences) as a three-dimensional variable, made up of a competency (plural:competencies), a proficiency level and a context (see figure 1). For example, "Fluent Business English" would be composed of the competency "English", the proficiency level "Fluent" and the context "Business".

For sake of clarity, and in order to avoid confusion between the terms competence and competency, we may use competency and skill interchangeably hereafter. However the reader should be aware that skill is not a synonym for competency, as it only covers part of its scope.

[1]The reader is alerted for the distinction between the two terms, *competence* and *competency*

[2]Although they later refer to it as "grade", which is different from proficiency level - see section 5.1

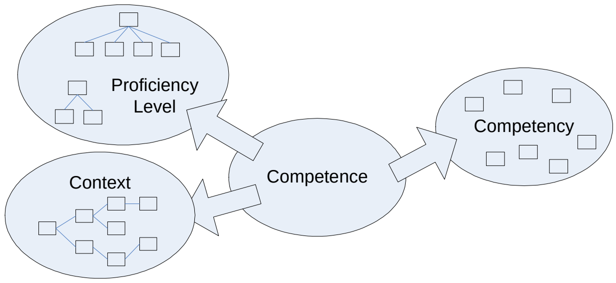


Figure 1: Competence as a combination of competency, proficiency level and context

## 3 Related Work

There exist some standardization efforts on modelling competenc*ies*. These efforts focus on different aspects related to competency: competencies as such, competency profiles and relationships among competencies.

The IMS Reusable Definition of Competencies or Educational Objective (IMS RDCEO, 2002) and the later IEEE Reusable Competency Definition (IEEE RCD, 2005) (based on IMS RDCEO) focus on reusable competency definitions. The primary idea is to build central repositories which define competencies for certain communities. These definitions can be referenced by external data structures, encouraging interoperability and reusability. However, IEEE RCD lacks information on context and proficiency level and does not allow relationships or recursive dependencies among competencies.

HR-XML focuses on the modeling of a wide range of information related to human resource tasks (like contact data or aspects of the curriculum vitae). The work performed in HR-XML Measurable Competencies (HR-XML, 2004) tries to define profiles in order to use such competency definitions. It specifies data sets like job requirement profiles (which describe the competencies that a person is required to have) or personal competency profiles (which describe the competencies a person has). Such profiles are composed of evidences (either required or acquired) referring to competency definitions (e.g., IEEE RCD). Unfortunately, the proposed model does not clearly separate required and acquired profiles. The consequence is that an acquired competency could have mandatory and optional elements according to the model. Furthermore, it is unclear why a competency is composed of several evidences: since a competency is a reusable object, evidences should rather represent a requirement or demonstrate the acquirance of a competency. Hence, the evidences should refer to or contain competency definitions and not vice versa.

The Simple Reusable Competency Map (SRCM, 2006) tries to model relationships between competencies. A map can contain information about dependencies/equivalences among competencies, including the composition of complex competencies from simpler ones. In SRCM, relationships are modeled using a directed acyclic graph. However, the semantics of the model proposed in SRCM is confusing. Relationships among different nodes may have different meanings: composition, equivalence or order dependency. This leads to confusion when modeling tasks as well as when creating algorithms to use such information. Furthermore, combination and weighting of competencies is not clearly defined, and external references to the maps (e.g., from profiles) must point to the root (and not to any node), therefore requiring the traversal of the graph until the appropriate node is found. Moreover, in this paper we argue that it is not possible to model relationships among competencies, because proficiency level and context have to be considered. For example, statistics knowledge may be a requisite for becoming a computer scientist or a sociologist. However, the proficiency level required and the context in which the competency is applied are completely different, hence making impossible to create relationships directly among competencies.

In OntoProPer (Sure et al., 2000), profiles are described by flat vectors containing weighted skills (where weights grow from 0 to 3), which are expressed as labels. Weights represent importance if applied to requirements or skill level if applied to acquired skills. The system itself mainly focuses on profile matching and introduces an automated way of building and maintaining profiles based on ontologies. (Colucci et al., 2003) describes an ontology-based semantic matchmaking (using Description Logics) between skills demand and supply. In (Lefebvre et al., 2005), which also defines a competence ontology for domain knowledge dissemination and retrieval, a competence is related to capabilities, skills and expertise (measured by levels growing from 1 to 5). Although this approach is closer to our definition of competence, still the context is not tackled, the relationships are defined at the skill level and the proficiency levels are not flexible enough.

# 4 Modelling a Competence

In this section we introduce a model for representing a competence with a broader and clearly defined view. We base this model on the three dimensions that a competence is composed of: competency, proficiency level and context. We first describe each dimension separately and finally present how they are combined in order to build a competence and how competences may be composed of sub-competences. Several issues encountered during the modeling process, and possible solutions (eventually with a trade-off between expressiveness and complexity) are described. We also discuss the decisions we have taken as well as their features and the limitations derived from them.

## 4.1 Competency

The IEEE Reusable Competency Definitions (IEEE RCD, 2005) provide a model for the representation of competencies (figure 2). The model does not include proficiency level or context information. In addition, as stated in the specification, IEEE RCD is “intended to meet the simple need of referencing and cataloging a competency, not classifying it”, that is, it does not provide any means to specify relationships between competencies. We agree upon this view and believe that relationships should not be modeled at this level because they also depend on the other two dimensions: proficiency level and context. For example, piloting cannot be related to other competencies without knowing if it refers to helicopters, small planes or passenger planes.

The ideas described in (IEEE RCD, 2005) meet our requirements, with the advantage that this work is already acknowledged from the community as a draft standard. Therefore, we decided to reuse IEEE RCD’s model to represent competencies (see model depicted in figure 2).

The RCD identifier provides the basis for referencing and reusing such RCDs, while title and description provide free text to represent them. We assume the existence of repositories of RCD elements which may be referenced from different competences by their global identifier. In addition, (IEEE RCD, 2005) includes information which is not thought to be machine-processable[3] but for human interpretation. Such information includes structured descriptions for more complete definitions.

## 4.2 Proficiency Level

Different scales (qualitative and quantitative) may be used in order to represent proficiency levels. For instance, a computer science curriculum may simply want to specify whether a student has acquired a competence or not, whereas an English certification institution may want to classify students into intermediate, advanced or proficient. Many different scales may be used but it should be possible to reuse them within and across the borders of the institution. For example, scales are typically the same for most certifications given by one institution and even among them (e.g., all curricula in Spanish universities). Hence, they can be modeled once and referenced many times.

Proficiency levels are not simply a flat set of elements. There are implicit relationships among elements within one scale. For example, a proficiency level may be subsumed by another (“proficient” subsumes “advanced” which subsumes “intermediate” and so on). We need to model such relationships because

[3]Do not confuse with *machine-exchangeable*

they will be needed for competence matching. For instance, a job requiring someone with intermediate English skill typically has implicit the quantifier “with at least”, meaning that anyone with advanced English would also be accepted (and maybe even preferred ). In order to represent this relationships, an ordered list provides a reasonable means to represent a proficiency level scale (see figure 2). In such a list, the minimum value (subsumed by any other in the list) is given by the first element and the maximum is given by the last one. Therefore, the order in the list represents subsumption relationships, that is, the first element is subsumed by the second one which is as well subsumed by the third one and so on.

In order to improve interoperability and matching among scales, an optional field is included for mapping to a universal scale (e.g., [0,1]). The reason why this mapping field is optional is that even though it would be useful to include it, in some contexts it may not be possible to find a suitable mapping or it may not even be necessary (e.g., if a scale is used only within an institution and no interoperability is intended).

Competence descriptions can refer to specific items of these scales in order to represent the proficiency level acquired/required. Algorithms could take relationships among proficiency levels into account in order to find out how much training/learning is required to reach a determined employee/learner proficiency level. For example, if advanced English skills are required, training an employee who already acquired intermediate English skills will cost less time and money than training another employee who has only beginner English skills.

## 4.3 Context

(Webster, ) defines *context* as “the interrelated conditions in which something exists or occurs”, which includes “the circumstances and conditions which surround it” (Wikipedia, ). Regarding to competences, context may refer to different concepts. It might be the specific occupation in which a competence is acquired (e.g., driving as an ambulance driver or a pizza delivery employee), a set of topics within a domain (e.g., telecommunications or tourism, or theoretical vs. applied physics) or even the personal settings related to the learner (e.g., competences are different if acquired in a group-based learning setting than individually). All these (and possibly more) are contexts which may be part of a competence. What actually makes up sufficient context descriptions can not be defined in general, but depends on the scope and purpose of the competence descriptions to which they are attached. As with the skill definitions and proficiency levels, context definitions may be reused.

Modeling contexts may be a complex task, as it may coincide with modeling the whole domain knowledge of an institution. Ontologies can capture such knowledge (Lau and Sure, 2002) and use arbitrary complex structures, from simple sets or tree structures to directed acyclic graphs. Up to date, our investigations of existing relationships between context elements (regarding its use within competences) do not show the need for providing a graph representation or multiple inheritance. For this reason, we decided to first restrict the modeling of context to trees (see model [4] depicted in figure 2). This has multiple benefits:

- it reduces the computation complexity of competences
- it is easier to understand by users
- it avoids the need for cycle-detection mechanisms while modeling is done
- it simplifies the algorithms for competence matching.

We are still investigating the advantages and drawbacks of this decision and we do not discard an extension of the model in case we find some scenarios for which such a structure would be beneficiary. Allowing for more advanced algorithms could also be a reason for choosing a more expressive context model. Furthermore, the relationship among context concepts may also be used by algorithms analyzing competence gaps. For example, assume that a context models all occupations of an airline company within an airport. In case it is needed to train a new pilot for passenger flights, it would be preferred to train some of the pilots of cargo planes instead of a person from the check-in counter. This information could be extracted from e.g. distances between the occupation “pilot” and the rest of occupations in the tree/graph.

## 4.4 Competence

Competences are described as reusable domain knowledge. Any model representing competences describes what a competence is and how it is composed of subcompetences. These competences are general descriptions, independent of specific learners or job descriptions. For example, being a good taxi driver or an expert Oracle database administrator are concepts with fixed meaning (domain knowledge), independent of which person possesses such competences. This is important to be noticed, because competences are to be referenced from certifications or job descriptions, in order to stimulate their reuse. For instance, a company may define required, relevant or desirable competences for their business, which are included in job offers or

[4] The set of attributes in the context structure is the minimum one allowing reference and reuse. This model may of course be extended with more data specific for the areas in which it is used

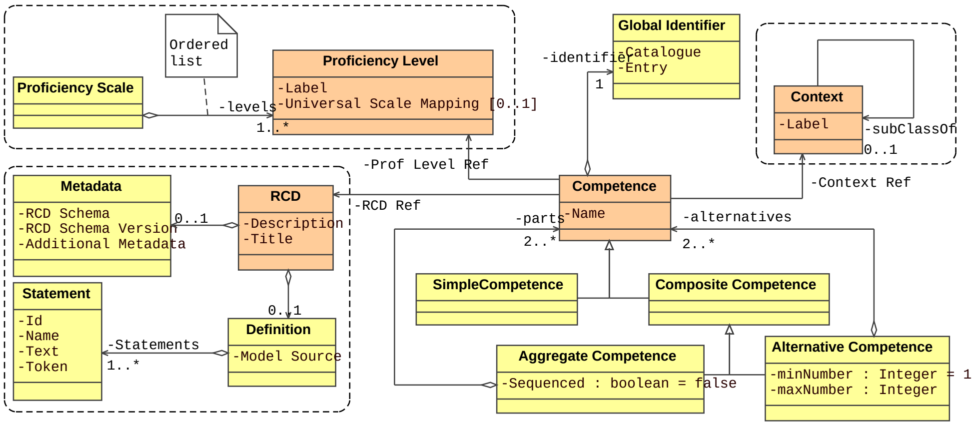


Figure 2: Competence Model

projects descriptions. The exact meaning of these competences is provided by a company-wide competence model. Using this approach, the explanation of a competence needs not to be explicitly included every time it is used[5]. These explanations may cover a broad range of aspects, such as:

- how a competence may be achieved, for example by acquiring some sub-competences;
- to which level each competence should be acquired. As an example, scientific research in a University may require only basic knowledge of mathematics while at NASA, expert knowledge is needed;
- whether sub-competences must be all achieved or simply a subset of them. For instance, it is typical in curricula that in order to get a degree, some topics are mandatory and some other are optional, from which a subset has to be chosen (e.g., pass *k* optional courses out of *n* offers);
- if the sub-competences must be acquired in a specific order. Some companies may require that an applicant acquired a competence on personal task organization *before* becoming a good team leader. Otherwise, they may assume that the performance related to the competence of being a good team leader is reduced.

In order to model all these elements we created an object model derived from the Composite design pattern (Riehle, 1997) (see figure 2). In our model, a competence can be either simple, an aggregation of children, or a selection from children alternatives. `Competence` models a competence, with references to a skill (RCD id), a proficiency level and a context. It can be a `SimpleCompentence` (an atomic description) or a `CompositeCompetence`. The latter can be either be an `AggregateCompetence` or `AlternativeCompetence`. An `AggregateCompetence` can be used to define a competence which consists of several sub-competences, all of them required. The sub-competences can be either an ordered set (meaning that the sub-competences must have been acquired in such an order) or unordered (default). An `AlternativeCompetence` can be used to construct a set of alternative sub-competences. It is possible to specify a minimum and a maximum number of alternatives that must be acquired (e.g., minimum *k* out of *n*). "Exactly" *k* sub-competences might be specified by setting both minimum and maximum to the same number. By default minimum is set to 1 so at least one subcompetence of the set is required.

Such a model allows to represent atomic competences, (un)ordered aggregation (all sub-competences must be acquired), alternative composition (a subset of sub-competences must be acquired) and any combination of all of them.

It is important to notice that if a competence is composed from several sub-competences, the proficiency level referenced in each subcompetence represents the minimum level required. For example, if it is required to have intermediate English skills in the context of science in order to be a good researcher, then anyone with advanced skills fulfills such a requirement. The subsumption relationship modeled within the proficiency levels is used for this purpose, and the proficiency level on the competence itself needs not to include all possible subsumers.

Our model is open to the addition of new relationships, among them an equivalence relationship. This is especially interesting if competence repositories of two communities are joined and mappings between overlapping competences have to be modeled.

[5]As with the use of ontologies, whose classes can be simply referenced without the need of copying the whole ontology every time they are used

# 5 Competence Profiles

Previous sections described how competences and relationships among them can be modeled. In real world applications, competence definitions are to support different tasks like creating job profiles for hiring or selecting people for a particular project, creating personal competence profiles showing the abilities of a person, and modeling the prerequisites and expected results of joining a learning or training programme. These tasks require modeling collections of required or acquired competences. Furthermore, the requirements specified by a job offer must be matched by the acquired competences an applicant provides. It therefore indicates that the model should be similar for all the cases enumerated in order to ease its matching. We refer to this model "competence profile" hereafter.

We can distinguish between two types of competence profiles, depending on their purpose:

**Required Competence Profile:** Specifies the requirements (in terms of competences) to be fulfilled by an applicant. These are typically used for job descriptions or programme prerequisites.

**Acquired Competence Profile:** Specifies the accomplishments (in terms of competences) of employees and learners. These are typically used in order to show (and possibly prove) which competences have been acquired or to represent the expected accomplishment after successful completion of a programme.

Each kind of profile is composed of a set of *ProfileElements*[6]. These profile elements may be *required* or *acquired*, depending on the type of the profile container (see figure 3). A profile element contains data which

- may be part of the criteria a company or a learning programme uses to decide whether an applicant is appropriate
- an institution providing degrees or certifications issues to learners as a prove of the acquired competence
- a learner uses to describe acquired competences in her CV (not necessarily with a proof or certification, e.g.,based on her experience)

Such information includes a type (e.g., driving license or university degree), the competence required or acquired and (possibly) a grade[7], the issuer organization, issue date and expiration date (i.e., from when the driving license is not valid anymore). All these fields are optional since not all are always needed. Typically, requirement profiles do not need to specify all fields of expected profile elements but only part of them. In these cases, some fields may be left empty, ensuring comparison only on those fields which specify constraints. For example, expert computer scientist may be a requirement but it may not be relevant where the competence was acquired (only competence field is filled in) or any applicant with a master degree may be sought but it does not matter in which field (only "type" is filled in and competence is left empty). In contrary, acquired profile elements should typically be filled in to a larger extent, specially if provided by certifications.

Note that the structure of a "ProfileElement" is different for required and acquired profiles. On the one hand, required profiles need to represent mandatory (English and French) and alternative requirements (either English or French) or even desired requirements (English mandatory and French is a plus). For that, we use the same composite model (meta-model) as the one specified for competences in section 4 (with the addition of tagging relationships with e.g. 'desired'), thus easing understanding and simplifying the tools needed to process these models. On the other hand, acquired profiles do not need such complex relationships and will therefore be represented as sets, that is, a flat collection of "SimpleProfileElement" elements.

## 5.1 Competence Proficiency Level vs. Grade in Competence Profile Element

Proficiency levels are part of competences, for example "Fluent English". This is different from grades provided by institutions (e.g., 250 in TOEFL test). While the former represents that "any person who has such a competence is supposed to perform effectively", the latter provides a "way to rate persons having such competence at a specific level of proficiency, by means of some sort of assessment". For example, two people having successfully completed an "Advanced Oracle Database Administrator" programme are able to perform effectively. However, they may have different grades in their final certification, which may be considered by HR departments before accepting any of them. In other words, proficiency levels (which are not bound to specific profiles) represent the scope of the competence acquired (advanced database administration vs. basic database administration) independently of whether a specific learner or employee (bound to a profile) learned the content perfectly or sufficiently to acquire the competence (higher or lower grade). For instance, being a proficient computer scientist requires to have advanced knowledge on databases, be intermediate software engineer and have basic knowledge

[6] For clarity, we did not keep the term *evidence* introduced in (HR-XML, 2004). A *ProfileElement* represents a requirement or a statement of an acquired competence but not necessarily a proof. Therefore, evidence could be misleading since it may be confused with proof or certification

[7] Note that grade and proficiency level represent different concepts (cf. Section 5.1)

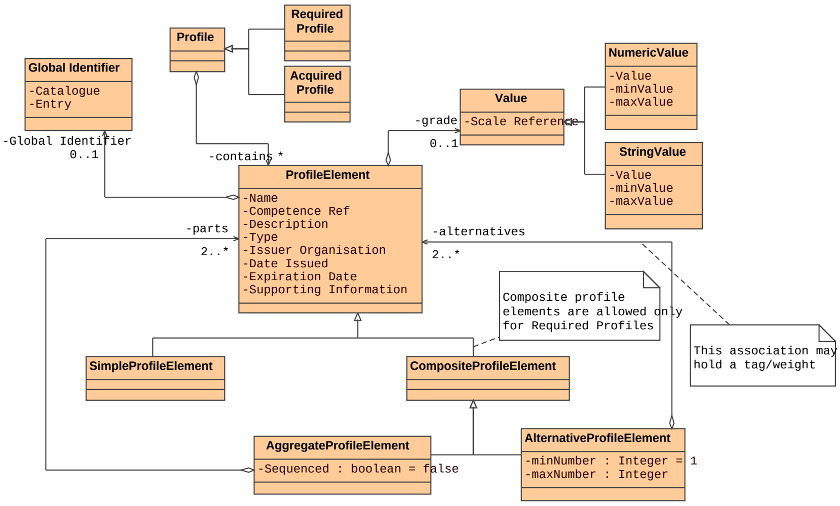


Figure 3: Competence Profile

on economics. Those represent the content (scope) required to acquire the competence, independently of the grade received by learners.

## 6 Example

We assume the existence of repositories with information about skills, proficiency levels, context and competences as depicted in figure 4. In this work we do not deal with the problem of ontology heterogeneity and we therefore assume that there either exist appropriate standards for this information or there are available mappings between different ontologies (see e.g. (Rahm and Bernstein, 2001; de Bruijn and Polleres, 2004)). In addition, how these models are instantiated is also out of the scope of this paper. We assume the existence of appropriate tools to hide the model from end users (e.g., competence management profile or CV creation).

Typically, a recruiter in a HR department would write a job offer [8] like

Wanted: J2EE consultant

- Completed Master's Degree (any faculty)
- Expert Knowledge in Java J2EE, Servlets, JSP
- Very good English and/or French

Among other drawbacks, such an advertisement does not indicate what is mandatory or optional and, more importantly, it is not machine-understandable. Performing a manual matching (as widely performed now from the recruiters), the recruiter will have a hard time matching applications against this offer.
An alternative would be to use the model proposed here, to encode the job advertisement (see figure 4). The model not only enforces a well-structured profiling, it also saves the information in a machine-readable and machine-understandable way. The recruiter can as well reuse information created from previous job advertisements (e.g., reuse the definition of Java Expert for his company, as well as use the well-accepted definition of Master). This 'indexable' representation also has significant advantages compared to the manual approach for the applicants: the applicants can now quickly seek on the advertisements, filter out advertisements for which their profile does not satisfy the requirements. In an even more advanced scenario, the profile representation can enable some ranking of the advertisements for which the applicant satisfies the requirements and some of the *optional* competences. Finally, the cycle is concluded when the applications come back to the recruiter. The recruiter can use a (semi)automatic matching engine to filter the non-satisfactory applicants according to their profiles, and rate the suitable applicants. For example, an applicant profile as depicted in figure 4 would be a perfect match for such an offer. More complex techniques could be used for partial matches and rankings/ratings, as they have been hinted along this paper or in (Colucci et al., 2003). However, elaborating on the matching techniques themselves is out of the scope of this work.

## 7 Conclusions and Further Work

This paper addresses the problem of competence representation and exchange. Current specifications focus on the modeling of competencies (not competences) and they miss important information that should be included, such as proficiency level and context. We provide a machine-processable representation

[8] Excerpt extracted from a newspaper

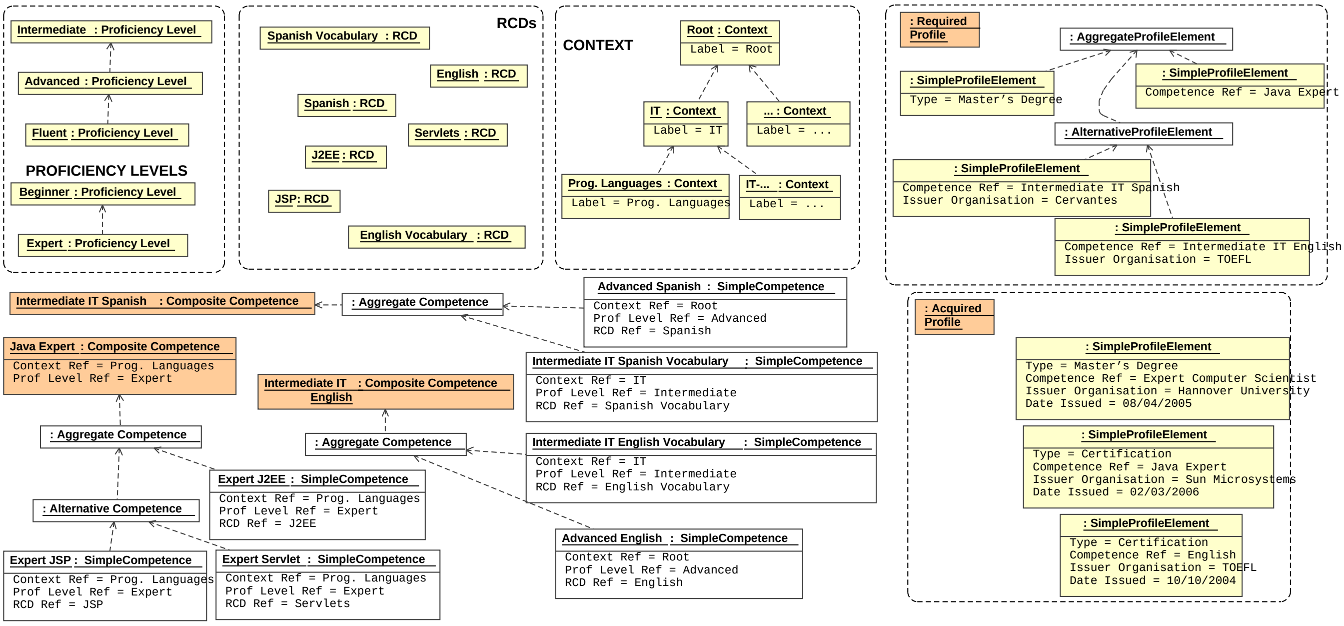


Figure 4: Competence Profile and Personal Profile Example

of competences, relationships among them and competence profiles. Such a model has been specially designed for reusability and allows advanced algorithms for competence and profile matching.

We are currently working on the development of applications in order to help end users to provide such competences and profiles. We will use our model within two different areas. On the one hand, we plan to develop advanced algorithms for competence matching and gap analysis in the business context as part of the EU PROLIX project. On the other hand, we plan to apply to the creation of competence development programmes and advanced assessment and positioning services within the EU TENCompetence project. Furthermore, we are in contact with representatives of the IEEE LTSC WG20 on Competency Definitions and the HR-XML Consortium in order to contribute to the improvements of their specifications according to the ideas presented in this paper.

# REFERENCES


Cheetam, G. and Chivers, G. (2005). *Professions, Competence and Informal Learning*. Edgard Elgar Publishing Limited.

Colucci, S., Noia, T. D., Sciascio, E. D., Donini, F. M., Mongiello, M., and Mottola, M. (2003). A formal approach to ontology-based semantic match of skills descriptions. *J. UCS*, 9(12):1437–1454.

de Bruijn, J. and Polleres, A. (2004). Towards an ontology mapping specification language for the semantic web. DERI Technical Report.

HR-XML (2004). HR-XML Measurable Competencies. `http://www.hr-xml.org`.

IEEE RCD (2005). IEEE 1484.20.1/draft - draft standard for Reusable Competency Definitions (RCD). `http://ieeeltsc.org/wg20Comp/Public/IEEE_1484.20.1.D3.pdf`.

IMS RDCEO (2002). IMS Reusable Definition of Competency or Educational Objective (RDCEO). `http://www.imsglobal.org/competencies/`.

Lau, T. and Sure, Y. (2002). Introducing ontology-based skills management at a large insurance company. In *Modellierung 2002, Modellierung in der Praxis - Modellierung für die Praxis*, pages 123–134, Tutzing, Deutschland.

Lefebvre, B., Gauthier, G., Tadié, S., Duc, T. H., and Achaba, H. (2005). Competence ontology for domain knowledge dissemination and retrieval. *Applied Artificial Intelligence*, 19(9-10):845–859.

Rahm, E. and Bernstein, P. A. (2001). A survey of approaches to automatic schema matching. *VLDB J.*, 10(4):334–350.

Riehle, D. (1997). Composite design patterns. In *ACM SIGPLAN Conference on Object-Oriented Programming Systems, Languages & Applications*, pages 218–228.

SRCM (2006). Simple Reusable Competency Map proposal (SRCM). `http://www.ostyn.com/resources.htm`.

Sure, Y., Maedche, A., and Staab, S. (2000). Leveraging corporate skill knowledge - from proper to ontoproper. In Mahling, D. and Reimer, ., editors, *3rd International Conference on Practical Aspects of Knowledge Management*, Basel, Switzerland.

Webster. Webster online dictionary. `http://www.webster.com/`.

Wikipedia. Wikipedia encyclopedia. `http://en.wikipedia.org/`.