

An Ontology and a Software Framework for Competency Modeling and Management

Gilbert Paquette

LORNET and CICE Director
LICEF Research Center, Télé-université
gpaquett@teluq.uqam.ca
www.licef.teluq.quebec.ca/gp

100 Sherbrooke Street West, Montreal H2X 3P2

Phone: (514) 840-2747 extension 2818

Fax: (514) 843-2158

Abstract

The importance given to competency management is well justified. Acquiring new competencies is the central goal of any education or knowledge management process. Thus, it must be embedded in any software framework as an instructional engineering tool, to inform the runtime environment of the knowledge that is processed by actors, and their situation towards achieving competency acquisition objectives. We present here some of our results in the last ten years that have led to an ontology for designing competency-based learning and knowledge management applications. Based on this ontology, we present a software framework for eLearning systems that are ontology driven.

A search on the Internet is sufficient to show the importance given to competency profiles in human resource management and education. Ministries of education, school boards and teacher training institutes use competency profiles to define school programs or required qualities from the teachers, especially in the use of technologies in education. Consulting companies present their expertise by enumerating competencies, marketing their services in this way. Other companies offer services or computerized tools to help their prospective customers define or manage the competence of their staff, looked upon as the main asset of an organization in a knowledge management perspective. Governmental agencies or professional associations use competency-based approaches to define conditions to the exercise of a profession and to orient their vocational training programs.

To address the challenges of the Knowledge society, we need to support better the process of competency acquisition in the context of lifelong learning, which is more and more required from every citizen. We need more flexible, adaptive learning systems, inside and outside public education system, before, after and during work. We need to respond to the huge demand for Web-based resources for work and learning to cope with the exponential growth of information, by making the semantic Web more and more a reality. We need to make educational modeling more widely used through powerful yet user-friendly tools and methods. Finally, we need to provide citizens with tool to personalize their learning processes based on evaluation of their competencies.

1- Explorations in Competency Management

In this introductory section we will survey a set of applications related to competency management. These applications have been built in various projects we have achieved since 1992. It is important to note that

we have started these projects with a clear definition of a competency that has been essentially confirmed and refined by many projects some of which will be presented in this section.

Our definition is founded on the relation between specific knowledge in an application domain and generic skills. *Competencies are statements that someone, and more generally some resource, can demonstrate the application of a generic skill to some knowledge, at a certain degree of performance.*

For example, suppose we say that a technician can *diagnose the faults in a car engine, with precision for all kinds of car*. This is a competency where a technician applies the “diagnose” generic skill to his knowledge of “faults in a car engine”, with a degree of performance that involves “precision for all kinds of cars”. A lower level of competency would be to “sometimes identify that there are some faults in a car engine” because “identify” is a simpler generic skill than “diagnose” and “sometimes” is less demanding as a performance than “all the time”. These kinds of relationships between competencies are very important for instructional engineering as we will show later on.

In this section, we will present applications where we have used this definition to help develop and validate the ontology that will be presented in section 2, which gives a more precise meaning to the concept of competency. Then, based on this ontology, we shall define tools and services for a software framework for competency management, which is the subject of section 3.

1.1 Competency in an Instructional Engineering Method (MISA)

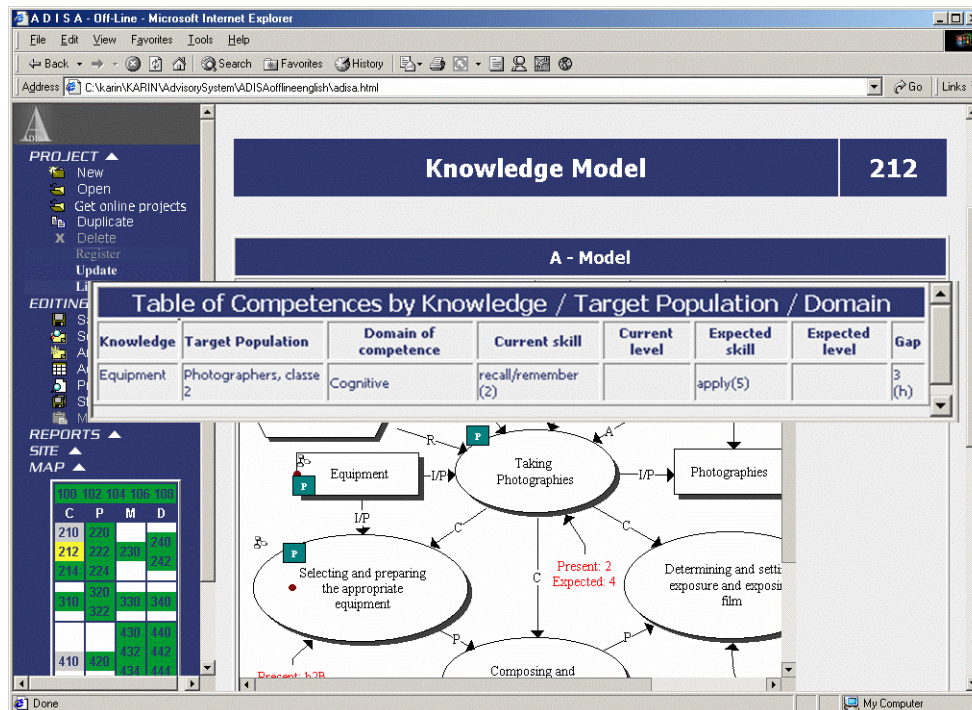
As early as 1992, we have started to integrate knowledge and competency modeling within the MISA Instructional Engineering method (Paquette, Crevier, Aubin 1994; Paquette 2002a, 2004). Competency ontologies are central for any instructional engineering methodology. It is important before, during delivery and after delivery. Before delivery, it serves to guide the design or adaptation of a learning environment. During delivery, it guides the action of facilitators for learners’ assistance. After delivery, it helps assess the learning results and the evaluation of the quality of a learning environment. We give here two examples of the use of competency modeling for instructional engineering that have served in many of our applications projects.

Guiding the definition of knowledge and activities using competency gaps. Figure 1 presents two of the tools in ADISA, a Web-based workbench to support the 35 main tasks of the MISA instructional design method. These tools are used to define a set of competencies associated to a knowledge model that defines the content for a course. In task/tool 212, the designer builds a graph of the knowledge that will be processed by learners and facilitators in an eLearning application, adding “P” labels to the knowledge elements that are important priorities for learning.

In task/tool 214, a list of these priority knowledge units is automatically transferred from the model and displayed to the designer in order for him to assign prerequisite and target competency statements for these knowledge elements. An example for the knowledge of photography “Equipment” is shown on the figure. For each priority knowledge element, and each target population of learners, the skill name and level, and the performance name and level are entered for prerequisite (“current skill” on the figure) and target or expected skill. Here, no performance level is shown: current and expected levels are empty.

This data enables the system to calculate the gap between entry and target competency, which is defined here as the absolute value of the difference between the two skill level. The notion of skill level is related to generic skill’s taxonomies that will be explained later on table 4. Here the current or entry skill is “recall/remember” which is the second on a scale of generic skills of increasing complexity. The expected skill is “apply” which is the fifth on the same scale. So the gap is simply $|5 - 2| = 3$. If performance level would have been defined, we obtain a two-dimensional scale such as the one on figure 8 to evaluate the gap more precisely.

Figure 1 – Associating competencies to knowledge element

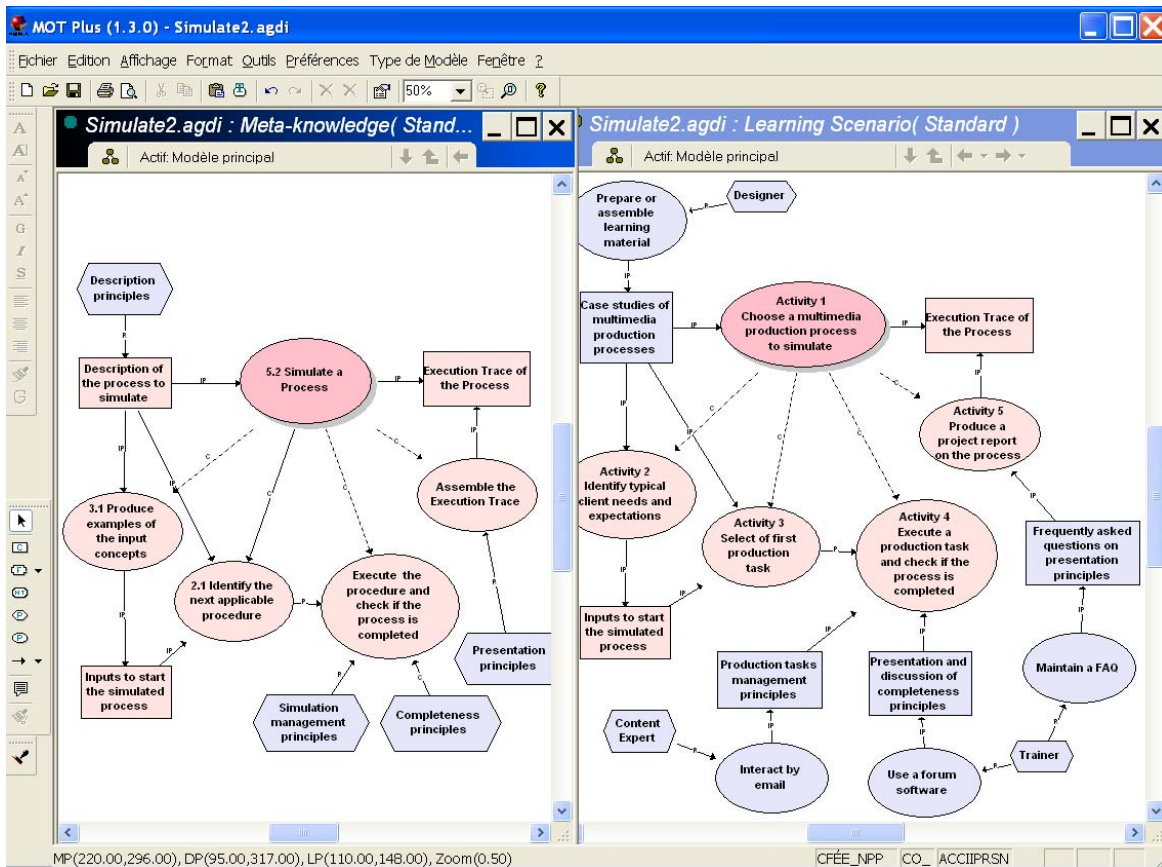


The system can then display a table of the priority knowledge element versus their competency gap (shown on figure 1 for the knowledge entity “Equipment”). Using this table, it is possible to guide the further development of the knowledge model, going back to the modeling tool in task 212. For example, for a knowledge element with gap of 0, it is not necessary to develop the model further; in fact it shouldn’t be a learning priority at all. On the contrary, for a knowledge element with a gap of 3 (prerequisite: 2-recall/remember; target:: 5-apply), we should add more related knowledge in the model, and thus more activities in the pedagogical model to acquire that knowledge.

Constructing an activity structure based on a generic skill. The generic skills on which a competency is based are processes acting on knowledge in an application domain. They can also be represented as process models (Paquette 1999). On the left side of figure 2, there is a graph of a generic skill, simulate a process, with its main operations (ovals) such as “produce examples of the input to the process” to be simulated, “identify the next applicable procedure” in the process, “apply this procedure”, and finally “assemble the simulation trace” by collecting the products (rectangles) of these operations. On the graph, four groups of principles (hexagons) are added to constrain the products and/or control the operations of the generic simulation process. Note that this model is totally generic, applicable to any specific knowledge domain, such as Internet processes, manufacturing processes, or others.

The graph on the right side of figure 2 presents a corresponding learning scenario based on a generic skill model where learners simulate a multimedia production process by performing learning activities that correspond to the main operations in the “simulate a process” skill. Such an activity structure is based on a graph almost isomorphic to the generic process, however, taking a “learning activity” viewpoint. The specific domain vocabulary is used, and the activities are formulated in an “assignment style” format.

Figure 2 – From generic skill to learning scenario definition



This provides the skeleton of a learning design. To complete the process, we need to add resources to help learners achieve their tasks. The important thing here is that the generic process in a target competency provides the structure the learner’s assignments. In that way, it is possible to make sure that the learner works at the right skill level, in this case “simulating a process”, while at the same time processing specific knowledge domain.

1.2 Constructing a Professional Training Program

We have used the instructional engineering tools and the principles presented above to develop a complete professional training program. In 2003, we have been contracted by a Montreal-based company to apply our competency engineering approach and our taxonomy of generic skills, in order to re-engineer a training program offered by the School of the Quebec Bar. This one-year program serves to habilitate every new layer in Quebec to professional practice. Working meetings of our team members with an Expert Committee, composed of 12 experienced lawyers, allowed us to build a relevant knowledge model for the domain of law practice. In a second step, we have identified cognitive and socio-affective skills associated to this knowledge, as well as the conditions of performance that are required from novice lawyers to be able to start professional practice.

Data on these elements were collected during group sessions with the Expert Committee, and individual face-to-face sessions with some of its members. Questionnaires were filled in by all members. The consultation of different content documents used in the program served to enrich the information. Systematically, the analysis of obtained data led to a document synthesizing a competency profile which was validated by the Committee and which brought new elements to advance the work and complete the

models. The iterative revision of the different versions of the competency profile led to a list of thirty-five main knowledge element with their associated target competencies.

Table 1 presents a sample of some of these competencies Each competency is expressed by a statement specifying the generic skill (in bold italic) which the novice lawyer has to apply to a knowledge element (in italic) according to particular performance conditions expressed in the rest of the competency statement.

Table 1 – A sample of the 35 competencies for the law training program

Id	Group A- Law concept, regulations and standards of the profession	Priority	Entry	Gap
A1	(6) Analyze the <i>applicable texts of law</i> to a situation, without help for simple and average complexity situations, with help in complex ones	1	(2)	4
A3	(3) Specify the applicables law regulation, autonomously in any case	2	(1)	2
A8	(5) Apply <i>pertinent proofs and procedures</i> , without help for simple and average complexity situations.	1	(2)	3
Id	Group B- Communication with the client	Priority	Entry	Gap
B1	(6) Analyze <i>interactions with the client</i> , without help in any communication situation.	2	(2)	4
B2	(9) Evaluate the quality of one's capacity <i>to listen to the client</i> , without help in any communication situation	2	(1)	8
B4	(4) Transpose in one's social and affectives interactions with the client, <i>principles of communication and human behavior</i> , sans aide, without help for average complexity situations.	2	(1)	3

Competencies were grouped in five domains based on the knowledge model: A- Law concepts, regulations and standards of the profession, B- Communication with the client, C-Establishment of a diagnosis, D- Elaboration, assessment and application of a solution, E- Management of case data and quality control. The four last categories show that the knowledge model was decided to be mainly procedural describing the main element of law practice, a sound decision for a professional program. The Committee attributed a priority to each of the competencies shown on the priority column of table 1.

To plan the new program, it was important to identify, for every competency, the distance between the prerequisite competency that the students should possess before entering the program, and the target competency to be acquired at the end of the program. The levels of the target generic skill were first identified by the expert committee and entry levels were set in a second phase by trainers in the program. The difference between the two is the gap shown in the last column of table 1.

Once stabilized, the competencies, their groupings and the estimated competency gap contributed to define the structure of the new program. As shown on table 2, competencies were distributed in learning units (called courses), taking in account groupings, priorities, and the competency gap between entry and target levels.

The gap between the entry and target competency levels proved to be very important for the construction of the program. There were 9 competencies with a gap from 1 to 3, 18 with a gap 4 or 5; 5 with a gap 6 or 7; and 5 with a gap of 8. The competencies were distributed in a spiral approach into four sequential learning units (courses 1 to 4) according to the gap. For example, the B2, E4 to E7 competencies (with a gap of 8) were integrated in all four courses to increase progressively the generic skill and performance

levels of the learners. Competencies A1 and others were distributed only in the first two courses because they are easier to acquire. Competencies A3 and others were included only in the first learning unit which seems to be sufficient for their acquisition. The target competencies, in of the courses serve as learning objectives to be measured by exams and other means of evaluation of the competence of the students. The following phases of the project have focused on building learning scenarios for each learning unit or sub-unit based on the generic skills in the associated competencies.

Table 2 – Distribution of competencies into courses of the program

Gap	Competencies	Course 1	Course 2	Course 3	Course 4
1-3	A3, A4, A5, A6, A7, A8, B4, C2, C6	x			
4-5	A1, A2, A9, A10, B1, B3, C1, C3, C5, D2, D5, D6, E1, E2, E3, E8	x	x		
6-7	C4, D1, D3, D4, E9	x	x	x	
8	B2, E4, E5, E6, E7	x	x	x	x

1.3 Integration in an LCMS (Explor@-2)

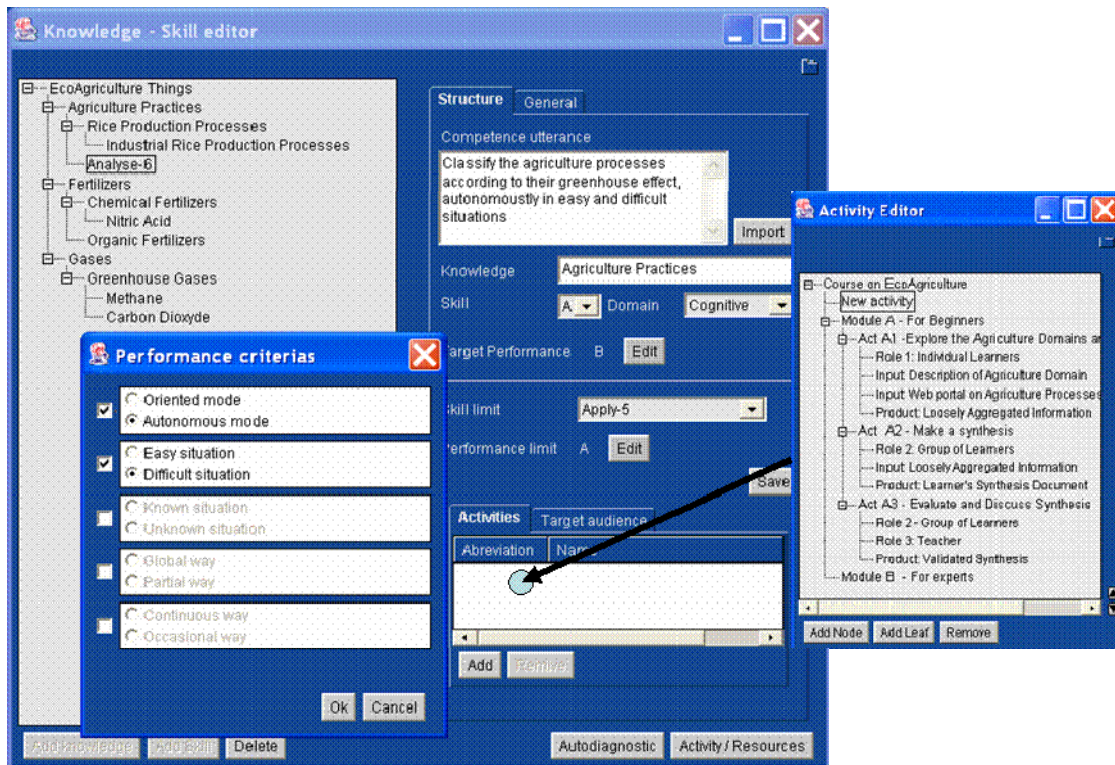
The same general approach to competency modeling has been integrated in an eLearning delivery system called Explor@ (Paquette et al 2005). Unlike most LCMS, Explor@ is build around two structures: the activity structure (or learning design) that decomposes a program or a course into smaller activity structures, activities and resources, and the knowledge/competency structure that presents a hierarchy of concept in a application domain with their associated entry and target competencies. Figure 3 present the tools that help create and manage these two structures.

The large window on the left presents a tree of concepts from a lightweigh ontology in a subject-matter domain, here eco-agriculture. The nodes of this tree are concepts from the domain ontology and the leaves serve to select a generic skill associated to its parent node. When selecting a leaf like “Analyze-6” with parent node “Agriculture practice”, the right-part of this window enables a designer to document corresponding competencies, writing the target competency statement. The knowledge and skill elements are copied automatically from the tree structure. The designer then adds the meta-domain (here “cognitive”, could also be “affective” or “psycho-motor”). The performance level is set by selecting some or all of the “performance criteria” in an auxiliary window, which are combined by the system into A, B, C, or D level (here B-Familiarity). The rest of the right side of the window is to fix similar elements for the entry competency on the same knowledge, here based on the “Apply-5” generic skill.

The lower right side of the window serves to associate these competencies to activities and resources in the activity structure shown on the other window. This is done by simply selecting activities or resources from this window; resources are on the leaves of this tree below the activity or activity structure where they are used or produced.

Another auxiliary tool (not shown on figure 3) displays a summary table of competencies at different levels of the activity structure, enabling designers to add weights to compose evaluation notation. From this data, it is possible to generate automatically a self diagnosis questionnaire like the one on figure 4 below, where learners can self assess their competencies and select proper activities and resources to help improve them. Similar interfaces can be generated for trainers to help them evaluate the student’s progress.

Figure3 – Competency management in Explor@



1.4 Emergent self-composed training programs for life-long learning

Figure 4 shows screens of a user-friendly self-diagnosis Web tool that can help students diagnose their competency and compose their own training programs with or without the help of a trainer or facilitator (Ruelland, Brisebois, Paquette 2005). The tool comprises three steps displayed in the three windows of the figure.

In the first page, a list of competencies imported from a competency editor is presented to the user (learner or trainer). For each competency, the user selects his/her actual performance level between four levels.

In the second page, a global summary of these combined levels is displayed in the form a bar graph of the gaps between actual and target competencies, to identify strengths and weaknesses.

In the third page, recommendations for a plan of action are provided in the form of resources associated to each competency or competency group. Access is given to these resources through an hyperlink to enable user navigation between the resources. The resources can be any document, Web site, on-line or blended learning course or address of a resource person that have been previously associated to one or more competencies. The association tools between competencies and resources can be made available to students as well as trainers to enable the composition of a training program for life long learning. In that case, federated search tools in learning object repositories can be used to find appropriate resources and associate them to competencies.

Figure 4 – A competency self-assessment tool

1. Self-assessment

3. Associated resources

2. Summary results

2- An Ontology for Competency Modeling

Drawing on the previous work and the experience gained in numerous projects, we now present an ontology for competency modeling that combines the concepts of knowledge, skill, attitudes and performance. It is rooted in different fields of research such as instructional design, software engineering and artificial intelligence. It provides ways to annotate semantically resources in eLearning and knowledge management environments, in particular to define competencies of individual actors, prerequisites and goals for activities and resource content, evaluation criteria and personalization capabilities for eLearning and Knowledge Management applications.

2.1 Beyond textual competency statement

Most often, competencies are expressed as simple natural language sentences, stating informally that a group of person has the “capacity” or the “knowledge” to do certain things. Competency profiles are in general loosely structured collections of such texts that are not always easy to interpret, communicate or use, especially if the goal is to plan learning events to support the acquisition of new competencies.

Efforts are being made to facilitate the use of competencies in Education and Training. For example, the IMS organization, involved in defining eLearning standards, has produced in 2002 a specification for Reusable Definition of Competency or Educational Objective (IMS-RDCEO 2002). It defines an information model for describing, referencing, and exchanging definitions of competencies, primarily in the context of online and distributed learning. Its goal is to enable the interoperability among learning systems that deal with competency information by providing a means for them to refer to common definitions with common meanings.

As stated in this RDCEO documents, “the word competency is used in a very general sense that includes skills, knowledge, tasks, and learning outcomes”. Furthermore, “the core information in a RDCEO is an unstructured textual definition of the competency that can be referenced through a globally unique identifier.” The RDCEO does not provide any structured model for a competency, but it mentions that “this information may be refined using a user-defined model of the structure of a competency”.

As we have pointed earlier (Paquette and Rosca, 2004; Paquette and Marino, 2005), a crucial area where a learning design specification like IMS-LD needs to be improved is knowledge representation. Actually, the only way to describe the knowledge involved in a learning design is to assign *optional* educational objectives and prerequisites to the Unit of learning as a whole and/or some of the learning activities.

Without a structural model for a competency, these entry and target competency will be unstructured pieces of text which are difficult to use by eLearning systems, forbidding for example consistency checking between different levels of the LD structure, and even, at the same level, between the content of learning activities, their related resources, and the learners’ competency. In fact, in IMS-LD the knowledge in learning resources is not described at all, and the actor’s knowledge and competencies are only indirectly defined by their participation in learning units or activities, only if educational objectives are associated.

Without a good representation of the knowledge and competency to be processed, a delivery system will be unable to help its users according to their present and expected competency state. In other words, it will be unable to adapt to its users, to personalize learning activities. What we need is both a qualitative structural representation of knowledge in activities and resources, but also a quantitative one providing a metric for evaluating competency gaps during the learning process. The association between learning objects (documents, tools, actors, activities) within a unit-of-learning, and the knowledge and competencies they possess contain or process is a key concept for semantic Web application (Berner-Lee and al, 2001).

The knowledge domain in an eLearning or Knowledge Management application can be structured in many ways: dictionaries, thesaurus, book summary, library catalog, indexes and metadata, knowledge graphs, ontologies, etc. The tree organization of a knowledge domain is an important property that can reduce significantly the processing but it is insufficient to describe the rich network of relations that ties the concept structures. It needs to be complemented with relations between concepts and axioms in order to sustain more refined mechanism of conceptual matching and inference. In other works, we need to use some form of domain ontology (Davies et al 2003, Breuker et al 1999).

But if we use only domain ontologies without defining mastery levels for the knowledge, we limit ourselves to weak semantic management capabilities, both by human facilitators or computer support systems. We can use different mastering scales: simple quantitative percentage, levels in Bloom taxonomy, combinations between generic skills taxonomies and performance levels, which is our proposal. The description of knowledge mastery must be reasonably simple, to be manageable. Still, the levels must correspond to clearly identify generic processes such as applying, synthesizing or evaluating knowledge.

Combining the preceding requirements suggests that a good candidate for the semantic indexing of educational resources, actors and activities will be a combination between domain ontologies and a simple and expressive generic skills ontology.

2.2 A Competency Ontology

The basis for the competency ontology we will now introduce can be found in many related fields such as mathematical logic (Thayse 1988), science methodology (Popper 1961), problem solving and its teaching (Polya 1967), educational technology (Romisowski 1987, Merrill 1994), software and cognitive engineering (Breuker and Van de Velde 1994; Steel 1990) and artificial intelligence (Pitrat 1991).

Our definition is founded on the relation between specific knowledge and generic skills. In his work on Artificial Intelligence, Jacques Pitrat (Pitrat 1991) has produced an important synthesis in which he distinguishes several meta-knowledge categories and proposes the following definition: « meta-knowledge is knowledge about knowledge, rather than knowledge from a specific domain such as mathematics, medicine or geology».

In the Education Sciences, Romiszowki (1983) expresses very well the simultaneous phenomenon of knowledge acquisition in a particular domain, and the meta-knowledge building of generic skills : « The learner follows two kinds of objectives at the same time - learning specific new knowledge and learning to better analyze what he already knows, to restructure knowledge, to validate new ideas and formulate new knowledge ». The same idea is expressed by Pitrat : « meta-knowledge (generic skills) is being created at the same time as knowledge ». In other words, meta-knowledge or generic skills develop while they are applied on knowledge in a particular field. Anybody learning new knowledge uses generic skills (at least minimally) without necessarily being aware of it. However, using generic skills should really be a learner's conscious act.

On a more practical ground, we have studied (Paquette 2002b) a sample of competency profiles in diverse sectors such a large company like (Hydro-Quebec), the Canada Public Service, Profession Requirements, High School Curricula, Library and Information Processing. We find a large diversity of goals, uses, and even explicit or implicit conceptual framework differences. Our goal is to find unity behind that diversity to be able to extract models of competency that can be used at an operational eLearning framework.

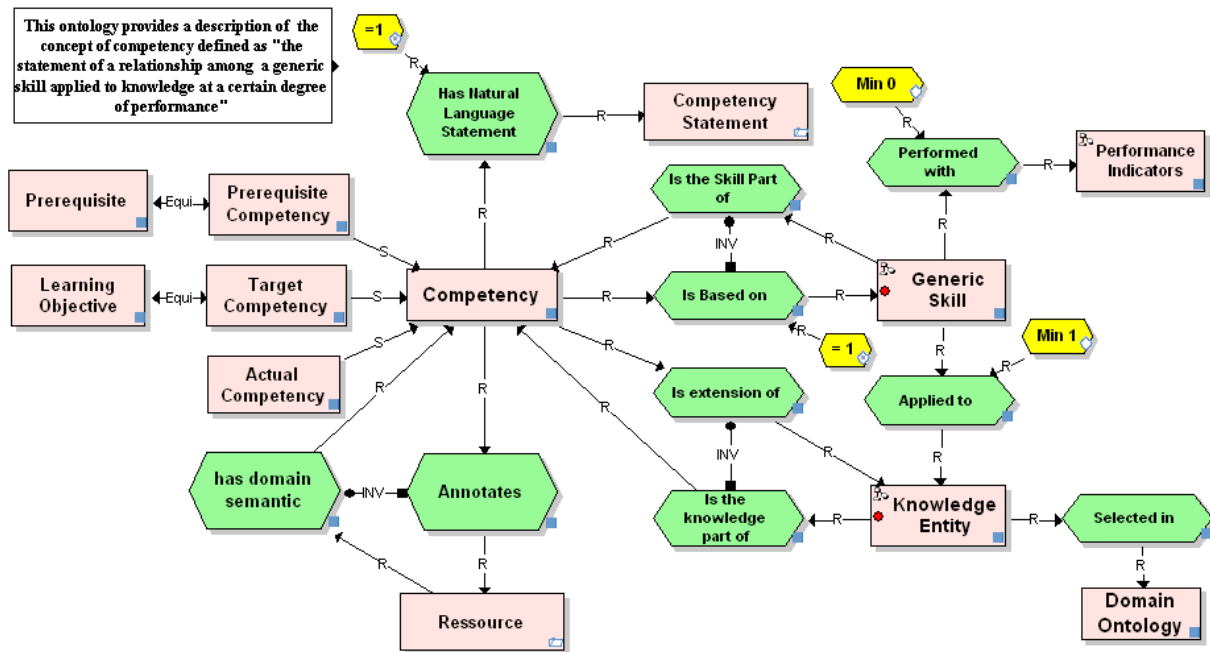
Competencies are statements that link together skills and attitudes to knowledge required from a group of persons and more generally from resources. Some examples integrate other elements than these or ignore some of them. For example, the competency profile of the Canada Public service presents a model that lies partly outside the realm of competency definition, taking in account interest and belief of the public servants. At the other end, statements like “ability to plan work”, “openness to critique” and “some knowledge of creation tools” in a competency profile for multimedia producers seem insufficiently precise, describing respectively a generic skill, an attitude and a knowledge element that are part of a competence, but insufficient to describe it.

In most competency profile definitions however, a competency is defined as a combination of skills, attitudes and knowledge to enable a group of person to fulfill a role in an organization or in society, for example to act as a lawyer, a nurse, a technician, a teacher, an information seeker, a media producer. Such roles require general competencies that might be shared with other roles, as well as more specific competencies required by their task context and the problems they will have to solve. For example, the role of a nurse will not be the same in a developing country as in a society where there exists a highly organized, sometimes bureaucratic, health system.

Figure 5 presents the top level of the ontology that we propose, defining the main competency concepts that will be explained and expanded later on.

Competencies serve to annotate resources, human as well as media resources, given them a semantic, a meaning, as to the knowledge and skills they own or contain. These annotations can represent prerequisite to achieve a task, or to be attained as a result of a task. They can also be declared as actually owned by the resource they annotate. Each competency is composed of a single *competency statement*, exactly one *generic skill* that may require precisions using performance indicators, and at least one *knowledge entity*. The competency statement is a natural language phrase that refers to the other components, stating that in the generic skill (with optional performance indicators) can be applied to the knowledge.

Figure 5 – Top level Ontology for Competency Definition



This graph uses MOT+OWL graphic syntax that covers all OWL-DL primitives (W3C 2003). Rectangles represent classes, hexagons represent properties linked to their domain and co-domain by incoming or outgoing *R* links. *S* links from one class to another means that the first is a sub-class of the second.

The *knowledge part* of the competency can be a concept, an action or a process, a principle or a fact that is selected in a *domain ontology*. In a competency profile for a profession like nursing, this knowledge part will be selected in a health care structured description of facts, concepts, procedures or principles. In a competency profile for media producers, the knowledge entity will be one of the techniques, methods, objects or products from the multimedia domain. In general, we will consider that the competency ontology is extended by an application domain ontology from which the knowledge part has been selected as a class (concept) or an individual of the ontology.

A *generic skill* is a process that can be applied to knowledge in more than one application domain, for example to perceive, memorize, assimilate, analyze, synthesize, or evaluate knowledge items. A generic skill is described by an action verb, sometimes with performance indicators like “in new situations” or “without help” that serve to make the skill more specific, while remaining independent from any application domain. For example, a generic skill like “establish a diagnosis”, or “establish a diagnosis in new situations without help” can be applied in diverse application domains to knowledge items like “skull fracture”, “car motor failure” or “exam failure risk”. A generic skill is also selected from a generic skill’s ontology that will be considered as an extension of the competency ontology.

Table 3 gives a sample of competencies from different sources showing how this competency ontology can be applied. Note that the fourth and fifth examples are merely expressions of a generic skill without a knowledge part, so we have concluded that implicitly it was meant to cover all knowledge in the application domain ontology. The fifth one is in fact two competencies that should be separated because a competency contains only one generic skill. Note also that the second and the fifth competencies have a skill's performance indicator, respectively "accurately" and "in everyday life".

Table 3 - A sample of competency statements in different fields and their breakdown

Source	Competency Statement	Generic Skill	Knowledge Entity
ANCI profile for nurses ⁽¹⁾	Demonstrates knowledge of legislation and common law pertinent to nursing practice	Apply	Australian law related to nursing practice
ANCI profile for nurses ⁽¹⁾	Analyzes and interprets data accurately	Analyze without error	Patient health care data
Multimedia Producer ⁽²⁾	Ability to evaluate of project's feasibility	Evaluate	Project description
Multimedia Producer ⁽²⁾	Hability to convince others	Influence	(Team members and clients)
MEQ – Student Competencies ⁽³⁾	Analysis and Synthesis Capability	-Analyze -Synthesize	(All subject matter in the curricula ?)
MEQ – Student Competencies ⁽³⁾	Apply in everyday life, rules of life in society	Apply in everyday life	Rules of life in society
Teaching Competencies ⁽⁴⁾	Operates within the framework of law and regulation	Apply	Laws regulating the teaching profession
Teaching Competencies ⁽⁴⁾	Plans purposeful programs to achieve specific learning outcomes	Synthesize	Programs adressing specific outcomes
Information Literacy Profile ⁽⁵⁾	Identifies a variety of types and formats of potential sources for information	Identify	Types and formats of information sources
Information Literacy Profile ⁽⁵⁾	Determines whether the initial query should be revised	Évaluer	Query for information

Extended examples of these competency profiles can be found in (Paquette 2002 b)

(1) Australian Nursing Coucil (ANCI): <http://www.anci.org.au/competencystandards.htm>

(2) Technocompetences' multimedia profiles: http://www.technocompetences.qc.ca/html/frame_rech_etud.html

(3) Quebec Education Department : Rapport sur les compétences transversales à l'école

(4) Australia North Territory Interactive Teaching Competencies Profile

(5) Association of College and Research Libraries (ACRL) <http://www.ala.org/acrl/assoinfo.html>

2.3 Generic Skills Sub-ontology

We will now expand the competency ontology for the generic skill component of a competency requirement. The backbone of this sub-ontology is a generic skill taxonomy that presents more and more specialized classes of intellectual processes (such as to memorize, transpose, analyze, or, evaluate) that can be applied in different knowledge domains. Each generic skill class groups individual knowledge processing activities.

Possessing a generic skill means that a learner can solve a corresponding class of problems (Chandrasekaran 1987, McDermott 1988, Steel 1990). For example, if a learner possesses a diagnostic or classification skill, it implies that this learner is able to solve some diagnostic or classification problems with a certain degree of a performance. Another view is to see cognitive skills as active procedural meta-knowledge (generic processes) that can be applied to knowledge (Pitrat 1991, 1993). A third view considers the association between cognitive skills and application knowledge as objects to be learned together, such as educational objectives (Bloom 1975, Krathwohl et al 1964, Reigeluth 1983, Martin and Briggs 1986). Integrating all three viewpoints gives us a solid foundation for a generic skills' taxonomy.

Basic Hierarchy of Generic Skills. Table 4 presents an overview of the proposed skills taxonomy. This table displays a loose correspondence between an artificial intelligence taxonomy (Pitrat 1991), a software engineering taxonomy (Breuker and Van de Velde, 1994; Scheiber et al. 1993) and two educational taxonomies (Bloom 1975 ; Romiszowski 1981). Although the terms are not in direct correspondence, table 2 distributes them onto ten levels that lay the foundations for our taxonomy (Paquette 1999, 2004) shown in the left part of table 2. It portrays three layers, from left to right, from generic to specific. It could be expanded to more levels for additional precision.

Table 4 – Generic Skills Processes compared to processes in other taxonomies

Generic Skills Classes			Active meta-knowledge (Pitrat)	Generic problems (KADS)	Cognitive objectives (Bloom)	Skills cycle (Romiszowski)
1	2	3				
Receive	1. Acknowledge					Attention
	2. Integrate	2.1 Identify 2.2 Memorize			Memorize	Perceptual acuteness and discrimination
Reproduce	3. Instantiate/ Specify	3.1 Illustrate 3.2 Discriminate 3.3 Explain	Knowledge Search and Storage		Understand	Interpretation
	4. Transpose/ Translate					Procedure Recall Schema Recall
	5. Apply	5.1 Use 5.2 Simulate	Knowledge Use, Expression		Apply	
Produce/Create	6. Analyze	6.1 Deduce 6.2 Classify 6.3 Predict 6.4 Diagnose	Knowledge Discovery	Prediction, Supervision, Classification, Diagnosis	Analyze	Analysis
	7. Repair			Repair		Synthesis
	8. Synthesize	8.1 Induce 8.2 Plan 8.3 Model/ Construct		Planning, Design, Modeling	Synthesize	
Self-manage	9. Evaluate		Knowledge Acquisition		Evaluate	Evaluation
	10. Self-control	10.1 Initiate/ Influence 10.2 Adapt/ control				Initiation, Continuation, Control

We will now discuss some of the properties of this taxonomy. Contrary to the behaviourist view on learning objectives, intellectual skills are here viewed from a cognitivist viewpoint as processes that can be described, analyzed and evaluated, by themselves or in relation to various knowledge domains.

Table 5 presents process definitions to compare generic skills in the first layer. For each class of generic skills, it shows input and products, as well as examples in the class. These definitions support the hypothesis that skills in the first layer are ordered from simple to complex. Self-management skills

involve explicit meta-cognitive operations for evaluation and decisions; these entails the need for produce/create operations to be performed. Creation or production of new knowledge from more specialized ones entails the use of reproduction skills. Reproduction skills are essentially instantiation or translation from more general knowledge that require reception skills. Finally, reception skills involve only attention and memory operations that are needed in reproductive processes.

Table 5 – Comparison of generic skills, from simple to complex (first layer)

Name of skill	Definition	Examples
Receive	Input = internal or external stimulus; Product = facts or knowledge found or stored in memory	Pay attention to an event, to a movement, to an emotion, to a social context; Identify knowledge, associated impressions; Memorize knowledge, impressions.
Reproduce	Input = knowledge and models; Products = facts obtained through instancing or knowledge obtained through reformulation	Use examples to explain or illustrate a concept, a procedure or a principle; Use a model to explain facts; Simulate a process.
Produce/ Create	Input = knowledge and models; Products = new knowledge or models resulting from analysis and synthesis	Classify objects according to a taxonomy; Repair defective system components ; Plan a project; Model and build a system.
Self-manage	Input = knowledge, models, values; Product = knowledge, models, meta- knowledge (values or generic skills) linked to domain model	Assess knowledge validity or self competence; Initiate a change process after assessing the situation; Apply a generic strategy to improve learning and performance.

Generic Skill’s Complexity. We will give here a definition of a generic skill’s complexity: A skill A is more complex than skill B if for any generic process P in class A, there exist a generic process in class B acting as a sub-process of P.

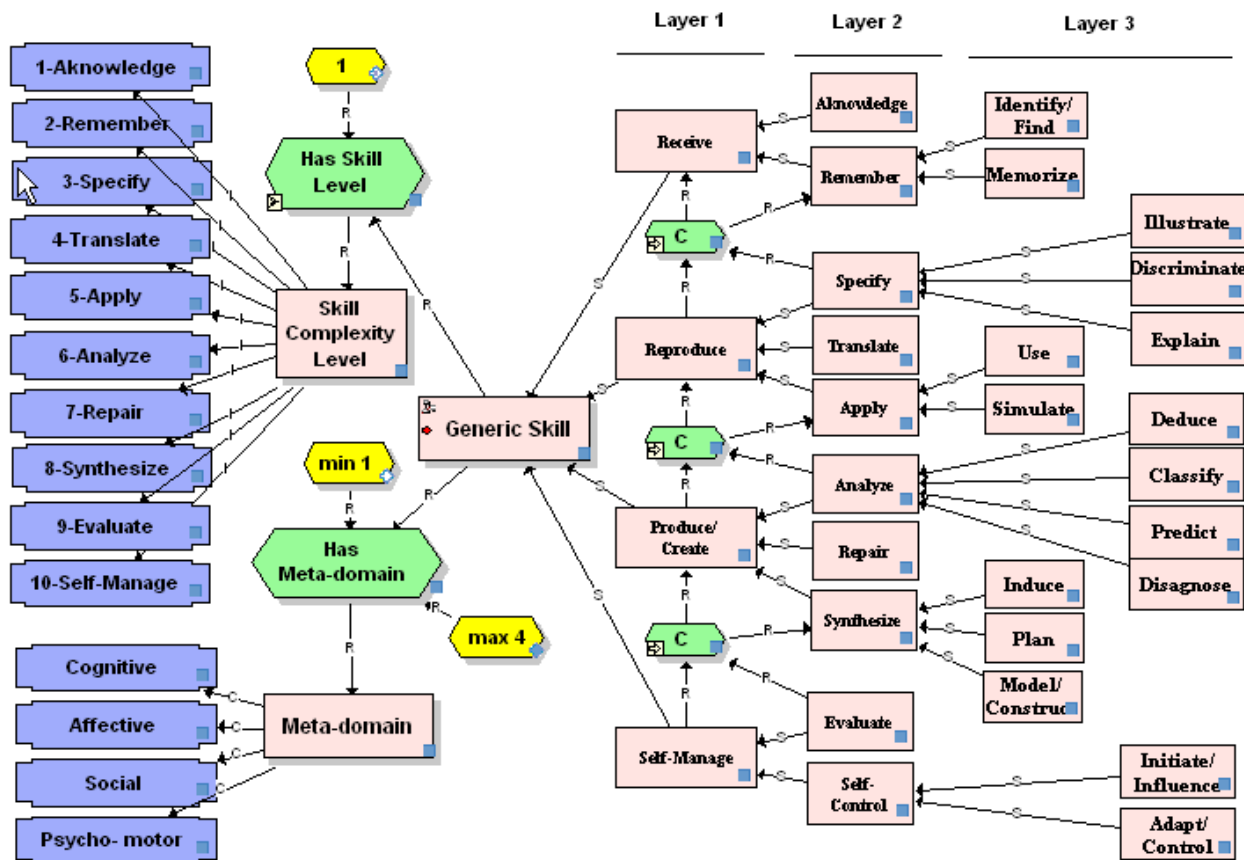
We have provided elsewhere (Paquette 2002 b) evidence of a complexity ordering for the 10 classes in the second layer of our taxonomy by constructing process graphs of these generic skills. For example, the “Simulate” skill (level of complexity 5.2) can be decomposed into four sub-processes such as: produce examples of the input concept (instantiate: level 3), identify the next applicable procedure (identify: level 2), execute the procedure applying its control principles (apply: level 5, in a simpler way), and finally assemble the simulation trace (transpose: level 4). This shows, according to our definition of complexity, that “Simulate” is more complex than skills at levels 2, 3 and 4.

Meta-domain. Generic skills, being processes acting on knowledge, can be classified according to the kind of input they process or produce. For example, “To persevere and to adapt a course of action in a project” is a process where affective and social input and output are deeply required. In (Paquette 2002b)

we have built a complete table showing examples in the cognitive, affective, social and psycho-motor meta-domains for each of the 10 major skills on the second layer of the taxonomy. It shows that this taxonomy can be interpreted in each of the four meta-domains: cognitive, psycho-motor, affective or social. For example, we can repair theories and movements, as well as attitudes or social relations. What differentiate these four meta-domains is essentially the type of input to a skill and its resulting outputs. If the stimuli or the result concerns rational thought, motor capacities, affective attitude or social interactions, we will label the skill respectively to be cognitive, psychomotor, affective or social.

Sub-ontology for Generic Skills. Figure 6 summarizes the preceding discussions on Generic Skill’s properties, presenting the resulting sub-ontology for Generic Skills.

Figure 6 – Extension of the Competency Ontology to Generic Skills



The taxonomy is ordered by layers (using specialization S links) from left to right, from general to more specialized skills. The first layer shows a “C” property between the four classes of skills. This property reads from bottom to top as “is more complex than”. We add here a transitivity axiom for the C property by adding an arrow label on its graphic symbol. This property also exists between skills of the second layer. Only some of C property symbols are shown on figure 2 to ease the graph’s readability, but there is one for example from synthesize to repair and from repair to analyze. On the other hand, there are no direct C properties between skills of the third layer, only through their second layer parent classes.

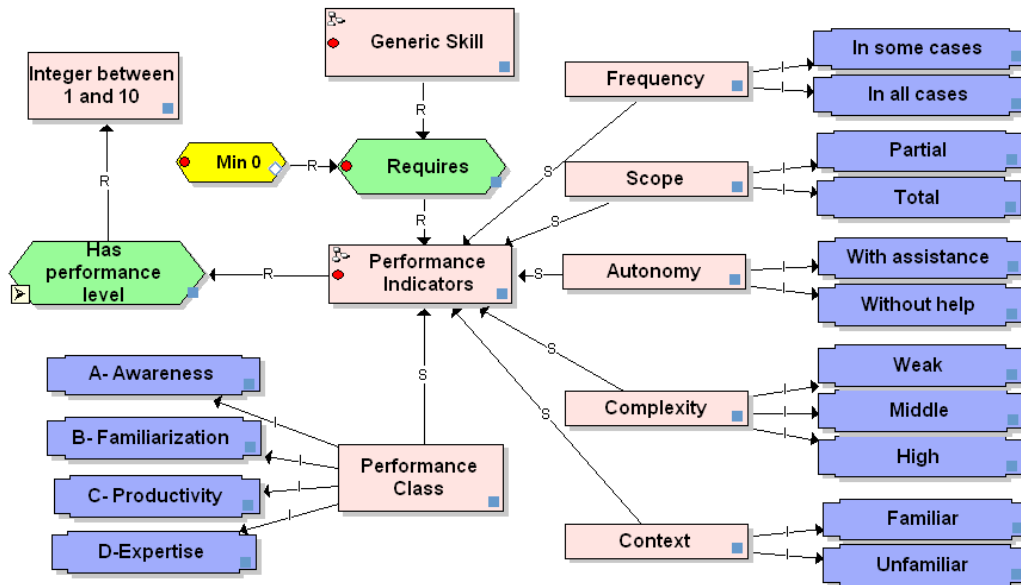
This MOT+OWL graph shows a third type of OWL objects representing individuals. It serves to assert that generic skills have a meta-domain property that can have as value “cognitive”, “affective”, “social” or “psycho-motor”, as well as any combination of these values (the “max 4” cardinality axiom affecting the property). It also shows that, based on the second layer’s total ordering by complexity, we can assign one

and only one number from 1 to 10 representing the complexity level of a generic skill. In OWL terminology, it means that the “has skill level” property is functional. A corresponding axiom is added by a label on the hexagon representing this property.

2.4 Performance Indicators and Levels

We now complete the competency ontology by developing a sub-model for the “performance indicator” class shown on figure 5. There are many possible performance indicators that are used by practitioners and some that we have explored in our own projects in the last ten years. The ones shown on figure 7 have been found most frequently useful.

Figure 7 – Extension of the Competency Ontology to Performance Indicators



For any generic skill, it is possible to add performance indicators such as frequency, scope, autonomy, complexity and/or context of the use. For example, a competency like “diagnose the source of malfunction of a car engine” could be made more precise by adding at the end performance indicators like “in all cases” or “in the majority of cases” (frequency), “for part of the causes” or “for all causes” (scope), “without help” or “with little assistance” (autonomy), “for high complexity engines” (complexity), or “in unfamiliar cases” (context of use). Some of these values are shown on figure 7 as instances of the ontology. Other individuals and other values could be added to extend the ontology. The usefulness of such indicators is to help built ways to assess the competency, for example to design exam questions or to register student actions in some model of his progress.

Alternative and more simple performance indicators classify performance for a generic skill in four broad categories such as “awareness”, “familiarization”, “productivity” or “expertise”, or simply by a number on a 1-10 scale for the performance level. These categories or levels can be direct evaluation results, or they can be calculate from the other indicators. One way to combine indicators or criteria to define performance classes or levels is shown on table 6.

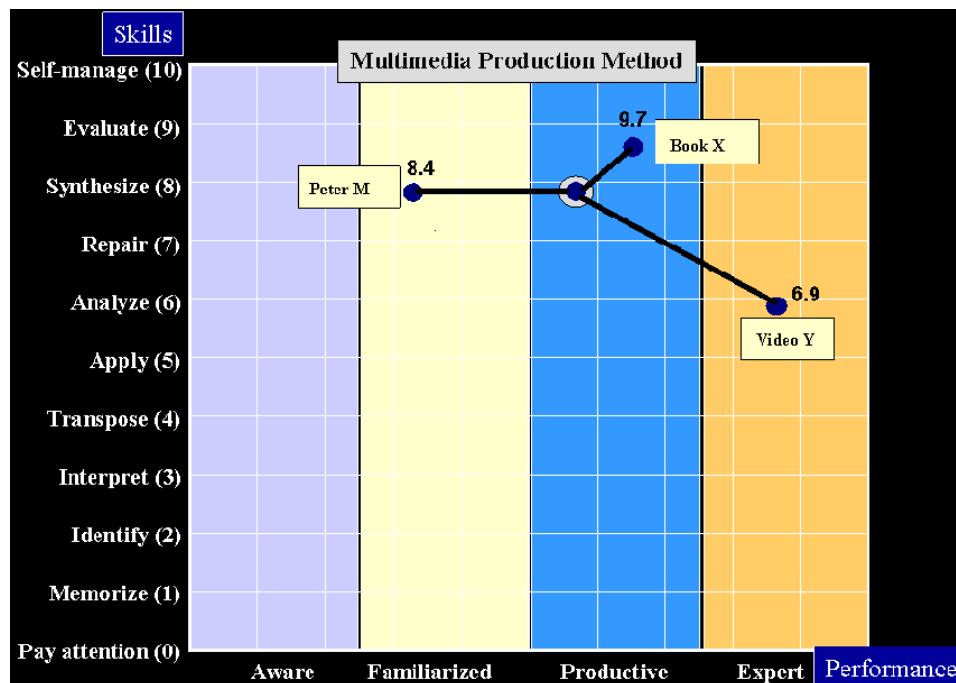
Table 6 – Performance Categories or Levels vs Other Indicators

CRITERIA	PERFORMANCE LEVELS			
	Awareness 0.0 – 2.5	Familiarity 2.5 – 5.0	Mastery 5.0 – 7.5	Expertise 7.5 – 10
Frequency	Sometimes	Always	Always	Always
Scope	Partial	Partial	Partial	Total
Autonomy	With assistance	With assistance	Without help	Without help
Complexity	Weak	Weak	Middle	High
Context	Familiar	Familiar	Familiar	Unfamiliar

2.8 Competency Scale

By combining the generic skills’ levels with performance levels, we can design a 2-dimentional competency-based scale that will help situate resources according to their competency for a certain knowledge item. For example, figure 8 shows such a competency scale for the knowledge of a “multimedia production method”. It shows a course having a target competency of 8.6, which means it aims at “Synthesize productively a multimedia production method”. For that course, Peter M has an actual competency of 8.4, which means he is “familiar with synthesizing a multimedia production method”. Video Y, at a level of 6.9, should not be very useful for that course, except maybe as a refresher, because it focuses on “Analysing at expert level a multimedia production method”, which is a lower generic skill level.

Figure 8 – Situating resources on a Skills/Performance Scale



3- A Software Framework for Competency Management in TELOS

This concluding section will summarize part of the actual research we are conducting within the LORNET research network to generalize and integrate the concepts presented above into the architecture of the TeleLearning Operating System (TELOS).

3.1 TELOS – an Ontology-driven Architecture

TELOS is basically an assembly and coordination system. The term “Tele-Learning Operating System” means that TELOS is planned essentially as a set of coordination and synchronization functionalities supporting the interactions of persons and computerized resources that compose a learning or knowledge management system. It integrates together human and computer agents using two basic processes, semantic representation of resources and resource aggregation.

An important goal is to help the system survive the rapid evolution of technologies by embedding in the system technology-independent models expressed as ontologies. The TELOS system is able to reuse ontologies as “conceptual programs”. In this vision, the conceptual models are not just prerequisite to building the TELOS system; they are part of the system, as one of its most fundamental layer.

3.2 Global View of the TELOS Competency Management Framework

The TELOS Competency management framework is composed of ten major tools shown on figure 9.

The tools in the first column serve to design fundamental elements of a TELOS Learning and Knowledge Management Application (LKMA).

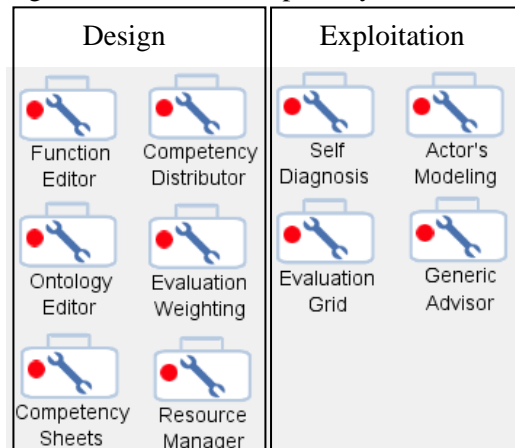
The *Scenario Editor* defines the flow of control and data between actors, activities, and the resources used and produced by the actors in the activities of a learning design or a workflow. In particular, the graphic design can be translated in an XML file compliant with the IMS-LD specification (2003).

The *Ontology Editor* defines a domain semantic structuring the main concepts, properties and individuals to be studied in an application domain. It is complemented by a *Competency Sheets* tool that adds competency definitions to some of the knowledge entities in the domain ontology. The form-based interface of the competency sheets is configured by a competency ontology such as the one presented in section 2.

These editors define two structures, the first one related to actor’s tasks and productions and the second one to the domain knowledge and competencies being processed by these actors. The next two tools, *Competency Distributor* and *Evaluation Weighting* serve to define associations between the two structures. Also, a *Resource Manager* can be used to find and associate complementary resources not in the scenario, with knowledge and competencies.

These associations will be used to parameterize the evaluation tools. The *Self Diagnosis* tool can be used by any LKMA actor (learners, trainers, content experts, technical support, evaluators, etc.) to assess their own competencies linked to appropriate resources that they can use for support or learning. The *Evaluation Grid* is a tool for learners’ evaluation by an assessment actor.

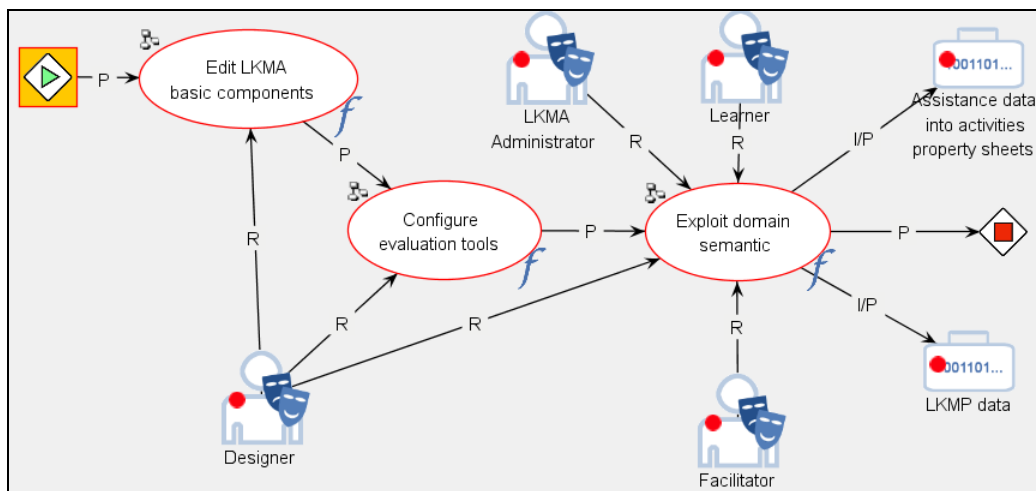
Figure 9 - TELOS Competency Framework



Finally some of these evaluation results produced by the runtime actors can be integrated into an actor's model and/or into an ePortfolio (also termed a LKMP: learning and knowledge management products). Competency evaluation results and the products of an actor's activities form the basis of their user model. The *Actor's Model Integration* tool serves to integrate and define the access to the LKMP data, according to a predefined policy that configures this tool. Then the Actor's models can be exploited by a *Generic Advisor* tools to provide personalized assistance to the actors.

Figure 10 presents a global view of a competency management workflow where these tools/services are used. It shows three phases. In the first one, designers edit the LKMA basic components: a scenario workflow and a domain ontology (including entry and target competency). In the second phase, designers configure some evaluation tools that will be used by the scenario actors at delivery time. The last phase is where the evaluation of actual competencies is done and its results are integrated into actor's models and ePortfolios (LKMP). This is also where assistance data is added to activity definition of the learning scenario.

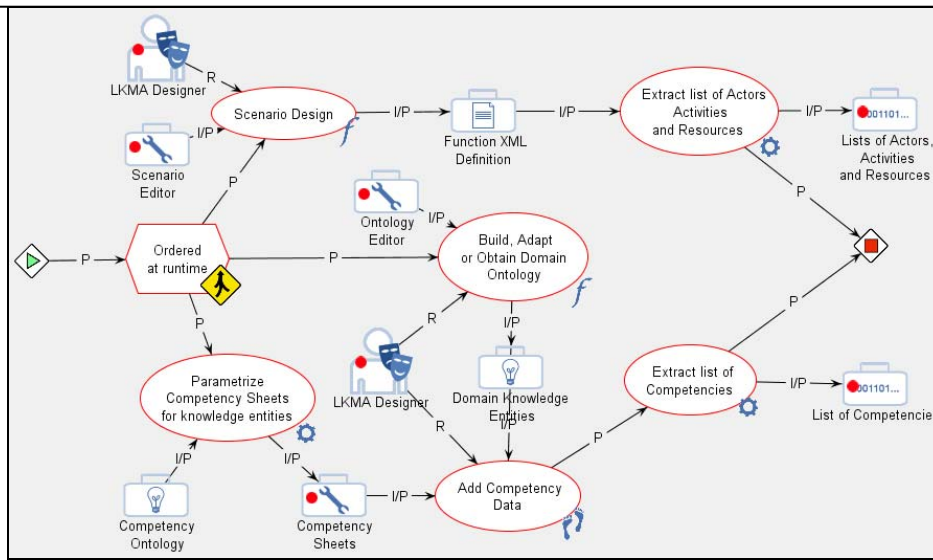
Figure 10 –Global view of the Competency Management Workflow



3.3 Scenario and Knowledge/Competency Design

Figure 11 describes with more details the first phase. It shows the LKMA Designer's task using the Scenario editor and the Ontology Editor with Competency property sheets. These edition activities produce on one hand a list of actors, activities and resources present in the LKMA scenario, sorted by type, and on the other hand a list of competencies with its components. Both lists will serve to parameterize the competency distribution and evaluation weighting tools in the next phase. They can also serve in the third phase to design personalization rules based on competency gaps that will be triggered at runtime to provide to learners advices on resources to use, activities to complete, actors to consult, or simply to adapt their learning or knowledge management environment.

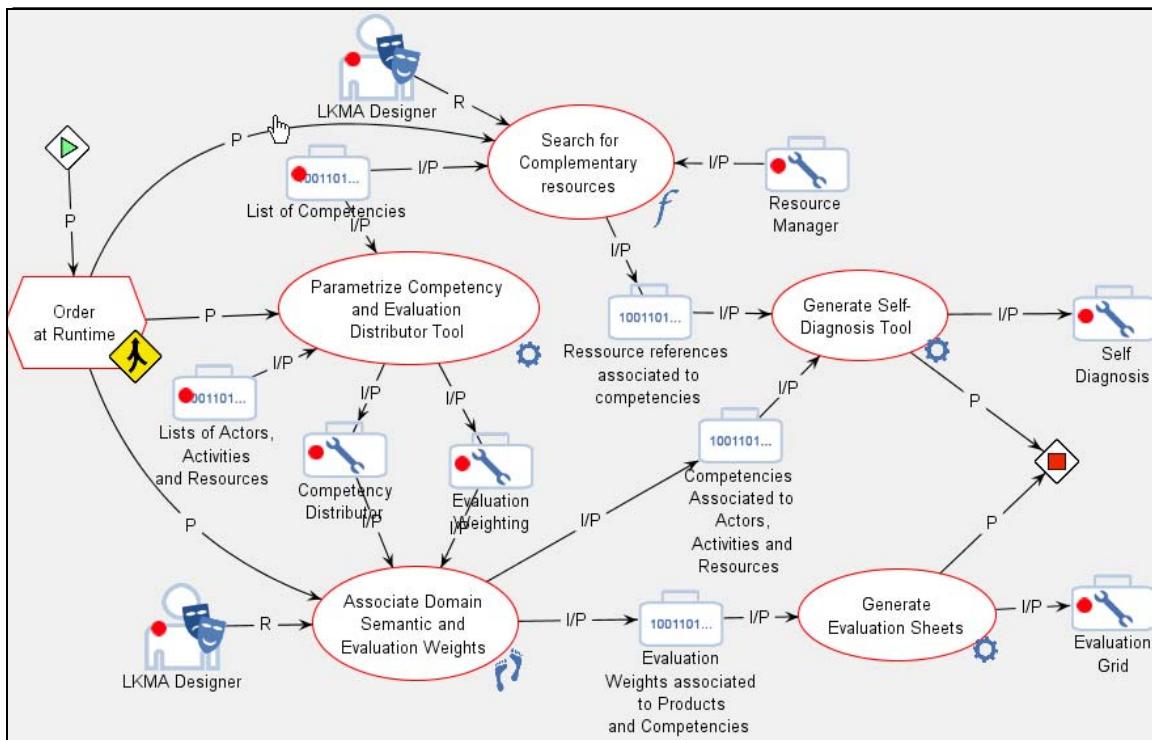
Figure 11 – Design Sub-processes for Competency Management



3.4 Knowledge and Competency Association

Figure 12 shows the second sub-process or phase of the competency management process.

Figure 12 – Knowledge and Competency Association Sub-process



It is possible to annotate a scenario component directly by selecting it and opening a domain semantic interface showing the ontology editor and its competency sheets. But doing this in a piecemeal fashion is not the best way to design a coherent application.

To improve this, we need more synthetic association tools, such as the competency distributor and the evaluation weighting tools shown on figure 12. Basically, they take the form of two dimensional tables with lines filled by a list of competencies already defined in the ontology editor. Columns headings are the activities, the input resources, the products or the actors in the scenario.

In the case of the evaluation weighting tool, only the learner products marked for evaluation by a designer are displayed in the columns. This tool enables the designer to distribute weights between them and, for each product, between target competencies for the product.

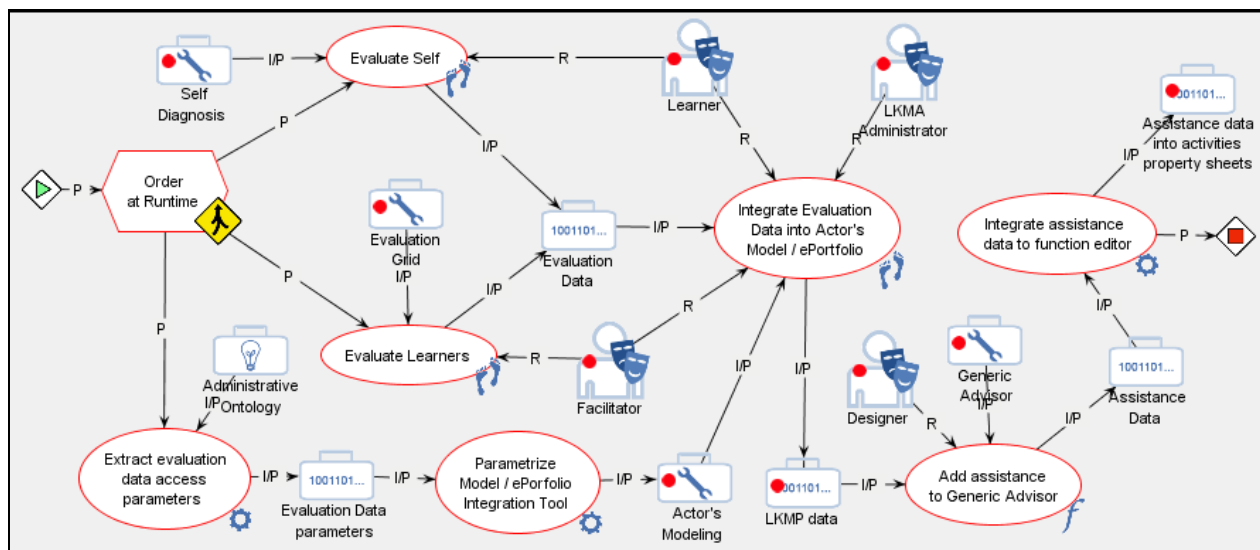
3.5 Domain Semantic Exploitation Sub-process

The above operations require a certain amount of effort by the application’s designers that will be rewarded by the automatic production of corresponding tools for learners’ and facilitators’ to use at runtime for competency evaluation. The tools/services presented on figure 13 are called “Domain Semantic Exploitation Tools” because they use the association of knowledge and competencies to actors, activities and resources, to provide evaluation and personalisation functionalities.

The Self Diagnosis tool used by learners (and possibly by facilitators), help these actors assess their own actual competencies with regard to target competencies defined by designers in the preceding phase of the process. The Evaluation Grid is used by facilitators to evaluate the actual competencies of learners.

Both tools output evaluation data that can be integrated into an actor’s model and or his/her ePortfolio. This is done according to an administrative ontology (possibly including a Digital Rights Management component) that defines how this data can be integrated in an actor’s model and an ePortfolio, and also what persons or computer agents will have access to it. The resulting LKMP data can be used to trigger, personalization rules previously designed in a generic advisor. This assistance data will serve to control the flow of activities at runtime based on differences between actual competencies (stored in evaluation data) and target competencies set by the designers using the competency association tools.

Figure 13 – The Domain Semantic Exploitation Sub-process



Conclusion

The importance given to competency management is well justified. Acquiring new competencies is the central goal of any education or knowledge management process. Thus, it must be embedded in any software framework as an instructional engineering tool, to inform the runtime environment of the knowledge that is processed by actors, and their situation towards achieving competency acquisition objectives. At runtime, the actual competency of actors must be evaluated in different ways and this data can be integrated as a central piece for learner's ePortfolios and models. This data maybe scarce or elaborated, but it can be used by human facilitators or software agents to assist learners in acquiring new competency, and to adapt the learning environments to the learner's characteristics.

We are not too far away from releasing most of the tools presented in this framework. Most of them have already been built in the past, but the major challenge is to integrate them in a coherent, flexible, user-friendly and scalable way, within the new context provided by the semantic Web and the ontology-driven architecture of TELOS. There is still a proof to be made that the general approach presented here can be used at different level by average design practitioners and learners.

References

- Berners-Lee, T., Hendler, Lassila, O. (2001) *The Semantic Web*. Scientific American, Feature article
- Bloom, Benjamin S. (1975) *Taxonomy of Educational Objectives: the Classification of Educational Goals*. New York: D. McKay.
- Breuker, J., Muntjewerff, A. and Bredewej, B. (1999). Ontological modelling for designing educational systems. In Proceedings of the AI-ED 99 Workshop on Ontologies for Educational Systems, Le Mans, France. IOS Press.
- Breuker J. and Van de Velde W. (1994) *CommonKads Library for Expertise Modelling*. IOS Press, Amsterdam, 360 pages.
- Chandrasekaran B. (1987) Towards a Functional Architecture for Intelligence Based on Generic Information Processing Tasks. *IJCAI-87 Proceedings*, Milan Italy, pp 1183-1192
- Davies, J., van Harmelen, F., Fensel, D., eds. (2002) *Towards the Semantic Web: Ontology-driven Knowledge Management*. John Wiley & Sons, Inc., New York, NY, USA
- IMS-RDCEO (2002) *IMS Reusable Definition of Competency or Educational Objective – Best Practice* (2002). 1.0 Final Specification, IMS Global Learning Consortium, Inc. Revision: 25 October 2002, <http://www.imsglobal.org/competencies/index.html>
- IMS-LD (2003) *IMS Learning Design. Information Model, Best Practice and Implementation Guide, Binding document, Schemas*. Retrieved October 3, 2003, from <http://www.imsglobal.org/learningdesign/index.cfm>
- Krathwohl D.R., Bloom, B.S., and Masia, B.B. (1964) *Taxonomy of educational objectives : The classification of educational goals*. Handbook II: Affective domain. New York: Longman, 1964
- Martin, B.L. & Briggs L. (1986) *The Affective and Cognitive Domains: Integration for Instruction and Research*. Educational Technology Publications, New Jersey, 494 pages, 1986
- McDermott J.. (1988) Preliminary steps towards a taxonomy of problem-solving methods. In Marcus, S. (Ed), *Automating Knowledge Acquisition for Expert Systems*, pp. 225-255. Kluwer Academic Publishers, Boston, Mass.

- Merrill M.D. *Principles of Instructional Design*. Educationnal Technology Publications, Englewood Cliffs, New Jersey, 465 pages.
- Paquette G. and O. Marino (2005), “Learning Objects, Collaborative Learning Designs and Knowledge Representation”, in *Technology, Instruction., Cognition and Learning* , Vol. 3, p.85-108, Old City Publishing, Inc.
- Paquette,G., Marino, O., De la Teja, I., Léonard, M., Lundgren-Cayrol, K., (2005). Delivery of Learning Design: the Explor@ System’s Case. In R. Koper & C. Tattersall (Eds.). *Learning Design – A Handbook on Modelling and Delivering Networked Education and Training*, Springer Verlag, pp. 311-326.
- Paquette, G. et Rosca. I. (2004) An Ontology-based Referencing of Actors, Operations and Resources in eLearning Systems. SW-EL/2004 Workshop. Eindhoven, August 2004
- Paquette, G. (2003) Instructional Engineering for Network-Based Learning. Pfeiffer/Wiley Publishing Co, 262 pages.
- Paquette, G. (2002a) *L’ingénierie pédagogique – pour construire l’apprentissage en réseau.*, 456 pages, Presses de l’Université du Québec, mai 2002.
- Paquette, G. (2002b) *Modélisation des connaissances et des compétences, un langage graphique pour concevoir et apprendre*, 357 pages, Presses de l’Université du Québec, mai 2002.
- Paquette, G. (2002c). TeleLearning Systems Engineering – Towards a new ISD model. *Journal of Structural Learning* 14, 1-35.
- Paquette, G. (1999) Meta-knowledge Representation for Learning Scenarios Engineering, in S. Lajoie et M. Vivet (Eds), Proceedings of AI-Ed’99 in AI and Education - Open learning environments, IOS Press, 1999.
- Paquette, G., F. Crevier, C. Aubin. (1994) ID Knowledge in a Course Design Workbench. *Educationnal Technology, USA*, volume 34, n. 9, pp. 50-57, November 1994
- Polya Georges, (1957). *Comment poser et résoudre un problème?* Dunod Paris.
- Popper K. R. (1967) *The Logic of Scientific Discovery*, Harper Torchbooks, New York, 1967..
- Pitrat J. (1991). *Métaconnaissance, avenir de l’Intelligence Artificielle*. Hermès, Paris, 1991.
- Reigeluth C. (1983) *Instructional Theories in Action: Lessons Illustrating Selected Theories and Models*. Hillsdale, NJ: Lawrence Earlbaum, 487pages.
- Romiszowski A. J. (1981) *Designing Instructional Systems* Kogan Page London/Nichols Publising, New York, 415 pages.
- Ruelland, D., Brisebois A. and Paquette G. (2005), *Supporting Self-Assessment in a Competency Approach to Learning*, eLearn-05 Conference, Vancouver, October 2006
- Schreiber G., Wielinga B., Breuker J. (1993) *KADS – A Principled Approach to Knowledge-based System Development*. San Diego : Academic Press. 457 p.
- Steels L. (1990) Components of Expertise. *AI Magazine*, vol. 11, no 2, Summer 1990.
- Thayse, A. (1988) *Approche logique de l’intelligence artificielle*, Dunod, Paris, 1988