UECML: Unified Enterprise Competence Modelling Language

Grégoire Pépiot, Naoufel Cheikhrouhou,
Jean-Marie Furbringer, Rémy Glardon

ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE
LABORATORY FOR PRODUCTION MANAGEMENT AND PROCESSES
CH-1015 LAUSANNE, SWITZERLAND

Abstract

This paper introduces the principles of a unified language devoted to the area of competence based enterprise modelling. The language, named UECML, for Unified Enterprise Competence Modelling Language, is intended to provide a neutral interface to enterprise modelling based on competencies. Therefore, it is built on previous languages, especially UEML and provides constructs to cover process, resource, competence and enterprise entities. A model implementation applied to a producer of railway vehicles is proposed in this paper.

Key words: Competence, Resource, Formalism, Enterprise Modelling, UECML.

1 Introduction

Today, enterprises operate in a constantly moving market, characterized by short product life cycles and increased demand for flexibility. In the same way, the ability to respond rapidly to changing market opportunities by quickly providing the required combination of design and manufacturing capabilities and competencies becomes a key issue in maintaining competitiveness. Moreover, most organizations have access to the same technologies. The unique way of maintaining their competitive advantage is thus to create and exploit a network of adequate competencies. This should be achieved through the integration of the competencies aspects in the enterprise models.

1 This work was carried out under the financial support of the Swiss Innovation Promotion Agency.
The aim of this paper is to propose a new modelling language called Unified Enterprise Competencies Modelling Language (UECML) for competence based enterprise modelling and to present its specialized constructs based on Unified Enterprise Modelling Language (UEML) principles [Vernadat F., 2002].

The paper is structured as follows. Section 2 addresses the overview-related literature of Enterprise Modelling. In section 3, the motivations and objectives for developing a competence based enterprise model are discussed. In section 4, the different entities or constructs that are employed for competence based enterprise modelling are reviewed. In section 5, a model implementation applied to a producer of railway vehicles is proposed. Finally, section 6 has conclusions.

2 State of the art on Enterprise Modelling

An enterprise is a complex system consisting of many constitutive interacting elements. Modelling such a complex system can significantly help understanding and identifying the most critical issues. Enterprise modelling can be defined as the art of externalizing enterprise knowledge, which adds value to the enterprise or needs to be shared [Vernadat F., 2002]. The goal of an enterprise model is [Petrie C., 1992] [Rolstadas A., 1995] [Scheer A.W., 1999] [Vernadat F., 1996]

- to describe the elements of a business entity, a part of a single enterprise, the whole enterprise (or a network of enterprises). The enterprise model usually takes into account functions, behaviors, information, resources and economic aspects,
- to represent or formalize the structure and behavior of enterprises, components and operations in order to understand, to engineer, evaluate, optimize, and even control the business organization and operations.

2.1 Enterprise modelling languages

Research in the field of industrial engineering and software engineering (mainly software development) has widely contributed to enterprise modelling development (languages and tools). Software engineering has also proposed innovative modelling techniques like the method MERISE, or UML supported by OMG (Object Management Group) or other task forces like WfMC (Workflow Management Coalition). Without any pretention on exhaustivity, some of the most relevant contributions of industrial engineering can be summarized as follows (see figure 1):
Fig. 1. Main Enterprise Modelling Languages

- CIMOSA: The CIM Open System Architecture [Vernadat F., 1996] provides an enterprise modelling framework, which contains a rich set of advanced modelling constructs. They cover functions, information, resources and organization aspects at several modelling levels (requirements definition, design specification and implementation description) of an enterprise in an integrated way. The modelling language is based on an event driven process-based approach, which views the business entity as a large set of concurrent processes executed by communicating agents [AMICE, 1993] [Vernadat F., 1998].

- IDEF suite: The IDEF family of languages, comprising IDEF0 (or SADT), IDEF1x, IDEF3 and IDEF4, is the most used set of modelling techniques in North America. [Menzel C. and R. Mayer, 1998]

- GRAI: is a method which provides support in analyzing the decision system of a manufacturing organization. [Doumeingts G., B. Vallespir and D. Chen, 1998]


- ACNOS: is devoted to the analysis of structured and non-structured activities in production systems. This analysis is based on processes models (Extension of IDEF3), on generalized stochastic Petri net models, competencies trees and economical performance evaluation methods.

The developed models cover the same aspects of the enterprise, i.e. model processes, activities and objects of the enterprise. Although enterprise modelling has demonstrated very effective and well-proven capabilities in practice, the situation is made even worse by fact that most of these tools are not interoperable and cannot exchange components [Vernadat F., 2000]. The enterprise modelling approach based on the competencies need specific requirements. First, the proposed approach is based on the concepts of processes and activities. This is due to the used definition of a competence that states a competence \( C(A) \) is the ability to combine in a efficient manner a number of non-material resources (knowledge, know-how etc.) and material resources (instruments, machines etc.) in order to respond to the need of an activity \( A \) [LeBoterf G., 1999].

Secondly, the approach has to provide an enterprise model, able to represent the competencies with the same and single formalism at different levels of abstraction and to valorise the competencies [Pépiot G., N. Cheikhrouhou, J.-M. Fürbringer and R. Glardon, 2005]. Unfortunately most of models previously described doesn’t allow to cover all these requirements at the same time as shown figure 2. However a wide range of concepts, which are used in these different methods and languages, are similar. This common basis leads to the tentative development of the unified enterprise modelling language (UEML). A comparison and consolidation effort was made on the basis of the previously presented languages in order to identify the set of constructs necessary to cover all modelling views. This set of normalized constructs [CEN ENV 12204, 1995] is documented in figure 3 and provides a starting point for UEML [Vernadat F., 2002].
2.2 Some UEML principles

UEML doesn’t constitute an ultimate enterprise modelling language acting as a substitute for all previous ones, but it is a standard meta-model, with its associated ontology. UEML and its developments contribute towards [Vallespir B., V. Chapurlat and C. Braesch, 2003]:

- a clear definition of the semantics common to the formalism and a better delimitation of enterprise modelling and engineering,
- a better inter-operability and capacity of communication between the actors of the heterogeneous modelling environment,
- a better definition of the modelling scientific corpus and of the enterprise engineering and thus an increased visibility,
- a vocabulary accepted and shared by the standardization organisms.

On the basis of UEML, the Unified Enterprise Competence Modelling Language (UECML) is designed and developed.

3 UECML Motivations and objectives

The construction of competence based enterprise models is motivated by pragmatic needs:

- a better definition of the enterprise modelling and engineering based on the competences;
- a clear definition of the concepts used in the enterprise modelling based on the competences;
- a clear identification of the role of the competence and its constituents.
How well an enterprise manages all of these needs determines whether the enterprise remains viable. UEML (modelling language dedicated to Enterprise Modelling), as well as UML (dedicated to Information Systems Modelling) are too general to fill the previous identified gap (figure 2) and thus to satisfy the objectives of competence based enterprise modelling. For this purpose, UECML provides supplementary entities, specialized for the management of enterprises on the basis of the competences. UECML is thus an enrichment of UEML in terms of used terminology and structure of the concepts to be represented. UECML provides thus an enterprise model:

- able to represent the competencies with the same and single formalism at different levels of abstraction;
- able to formalize in a coherent way the concept of competence;
- facilitating the design and the development of soft tools.

The main entities of UECML are presented in the next section.

4 UECML Core Constructs

UECML is based on a set of core constructs and sets of additional constructs. These additional constructs are specialized constructs required by the competence based enterprise modelling needs. The enterprise modelling based on the competences (UECML) has to support all core constructs introduced in the current section. (See figure 4.)

![Fig. 4. UECML Core Constructs](image-url)
Activity: an activity is a sequence of operations which consume resources and are realized by one or several actors. An activity is a elementary step of one process. An activity (A) can be formally defined as:

\[(A) = (A_{Id}, A_{La}, A_{Des}, cond, input/output)\]  \hspace{1cm} (1)

where \(A_{Id}\) is the activity identification, \(A_{La}\) is the activity label, \(A_{Des}\) is the description of the activity, \(cond\) are a set of conditions to be satisfied to enable or stop the activity execution and \(input/output\) defines the input or output object flow.

Process: a process is a ordered set of steps. Process steps can be sub-processes or activities.

A process (P) can be defined as:

\[(P) = (P_{Id}, P_{La}, P_{Des})\]  \hspace{1cm} (2)

where \(P_{Id}\) is the activity identification, \(P_{La}\) is the activity label, \(P_{Des}\) is the description of the process.

Organization Object: an organization object is any entity that is used by activities in the day to day operations of the organization. Organization objects are the elements involved in the input/output flows of activities. The organization object class can be categorized into sub-classes. Two sub-class objects are suggested in UECML are the "resource class" and the "human class". The relationships between the classes are based on two kinds of abstraction mechanisms: generalization and specialization. The generalization mechanism consists in together making the common elements of a set of classes to a more general class named super-class. The super-class is an abstraction of its sub-classes. On the other hand the specialization allows to capture the particularities of a set of objects which are not discriminated by the identified classes. The new characteristics are represented by a new class which is a sub-class of the existing classes. An organization object (OO) can be formally defined as:

\[(OO) = (OO_{Id}, OO_{La}, OO_{Des}, typeofrelation)\]  \hspace{1cm} (3)

where \(OO_{Id}\) is the object identification, \(OO_{La}\) is the object label, \(OO_{Des}\) is the description of the organization object, \(typeofrelation\) defines if the object is a sub-type of a more generic object (generalization) or it is a component of a compound object (specialization).
Resource: this is a special class of organization objects used in support of the execution of activities. This entity will inherit all properties of the object organization just defined previously. Two generic classes of resources can be defined: Non-Material Resources and Material Resources. A resource \((R)\) can be formally defined as:

\[
(R) = \langle R^{(A_i|R_j)}, R_{La}, R_{Des}, type(R_{NM}, R_{M}) \rangle
\]

where \(R^{(A_i|R_j)}\) is the identification of the resource \(R_j\) required for the activity \(A_i\), \(R_{La}\) is the resource label, \(R_{Des}\) is a short description of the resource and \(type(R_{NM}, R_{M})\) defines the type of the resource, material or not.

Non-Material Resource: this is a sub-class inheriting the resource class. A resource \((R_{NM})\) can be formally defined as:

\[
(R_{NM}) = \langle R^{(A_i|R_j)}_{NM}, R_{La}^{NM}, R_{Des}^{NM}, type \rangle
\]

where \(R^{(A_i|R_j)}_{NM}\) is the identification of the non-material resource \(R_j\) required for the activity \(A_i\), \(R_{La}^{NM}\) is the non-material resource label, \(R_{Des}^{NM}\) is a short description of the non-material resource, \(type\) is an attribute representing the type of the non-material resource defined by the following set of values \{knowledge, know-how, social – attitude\}

Material Resource: this is a sub-class inheriting the resource class. A resource \((R_{M})\) can be formally defined as:

\[
(R_{M}) = \langle R^{(A_i|R_j)}_{M}, R_{La}^{M}, R_{Des}^{M}, type \rangle
\]

where \(R^{(A_i|R_j)}_{M}\) is the identification of the material resource \(R_j\) required for the activity \(A_i\), \(R_{La}^{M}\) is the material resource label, \(R_{Des}^{M}\) is a short description of the material resource, \(type\) is an attribute representing the type of the material resource defined by a non-predefined set of values. The following set \{technique, information, process, etc.\} can be suggested.

Competencies: a competence \(C(A)\) is the ability to mobilize in a efficient manner a number of non-material resources (knowledge, know-how etc.) and material resources (instruments, machines etc.) in order to respond to the need of an activity \((A)\) [LeBoterf G., 1999]. The competence can be required (by an activity) and/or acquired (by an actor).
A competence $C(A)$ can be formally defined as [Pépiot G., N. Cheikhrouhou and R. Glardon, 2004]:

$$\langle C(A_i), [R_{NM}], [N_{ctxt}], [I_u], [R_M], [J_v], \text{type} \rangle$$  \hspace{1cm} (7)

where

- $C(A_i)$ is the identification of the competence,
- $[R_{NM}]$ is an array which terms are the different attributes of the non-material resources (identification, label, description and type) required by the activity $(A_i)$ or acquired by the actor involved in this activity,
- $[N_{ctxt}]$ is a vector which indicates the skill level of the non-material resource for the execution of the activity in a given context "$ctxt". This context is defined by $(A_i, P_i)$ or $(P_i)$.
- $[I_u]$ is an array which terms are represented by the value of the indicators for each non-material resource estimated by the set of indicators $I_u$ ($u \in [1 \ldots n]$ where $n$ is the number of evaluating indicators of the non-material resources),
- $[R_M]$ is a matrix which terms are the different attributes of the material resources (identification, label, description and type) required by the activity $(A_i)$ or acquired by the organization involved in the activity,
- $[J_v]$ is a matrix which terms are represented by the value of the indicators for each material resource estimated by a set of indicators $J_v$ ($v \in [1 \ldots n]$ where $n$ is the number of evaluating indicators of the material resources),
- $\text{type}$ is an attribute representing the type of the competence defined by the following set of values [uc (unit competence), ic (individual competence), cc (collective competence)].

The next definition of the sub-classes of the competence class will not take into account the matrix $[I_u]$ and $[J_v]$. These matrix are the basis elements of the development phase of the competencies valorization mechanisms.

**Unit Competencies:** a unit competence $C_{uc}(A)$ or (UC) is a sub-class inheriting from the competence class. A UC represents the lowest level of aggregation of the competencies. Indeed, the UC is directly associated with an activity although it has to be often linked to a workstation. This association (UC $\leftrightarrow$ A) will allow a flexible management of the "human capital" and a competence based management coupled with an activity based management.
If the organization redefines its processes, it will be easier to identify the required competencies for these new processes by examining the activities and their requirements. The \( C_{uc}(A) \) or (UC) can be instantiated under two types: required/acquired which lead the two following formal definitions:

\( C_{uc}^r(A) \): **unit competence required by the activity** (A).

\[
\langle C_{uc}^r(A_i), [R_{NM}], [N_{ctxt}], [R_M] \rangle
\]

where

- \( C_{uc}^r(A_i) \) is the identification of the unit competence required by the activity \( A_i \),

- \( [R_{NM}] = \begin{bmatrix} R_{NM}^{(A_i|R_1)}
& \vdots
& \vdots
& \vdots
& R_{NM}^{(A_i|R_j)} \end{bmatrix} \), \( R_{NM}^{(A_i|R_j)} \) indicates the identification of the non-material resource \( R_j \) required by the activity \( A_i \),

- \( type \in \{\text{knowledge, know-how, social- attitude}\} \),

- \( [N_{ctxt}] = \begin{bmatrix} N_{ctxt}^{(A_i|R_1)}
& \vdots
& \vdots
& \vdots
& N_{ctxt}^{(A_i|R_j)} \end{bmatrix} \), \( N_{ctxt}^{(A_i|R_j)} \) indicates the skill level required by the non-material resource \( R_j \) for the execution of the activity \( A_i \).

In other words an actor must have a knowledge of the \( R_{NM}^{(A_i|R_j)} \) with the skill level \( N_{ctxt}^{(A_i|R_j)} \) (\( N_{ctxt}^{(A_i|R_j)} \in \{1, 2, 3, 4\} \) if the skill level is evaluated with four levels) in order to execute the activity \( A_i \),

- \( [R_M] = \begin{bmatrix} R_M^{(A_i|R_1)}
& \vdots
& \vdots
& \vdots
& R_M^{(A_i|R_j)} \end{bmatrix} \), \( R_M^{(A_i|R_j)} \) indicates the identification of the material resource \( R_j \) required by the activity \( A_i \),

- \( type \in \{\text{technique, information, process, etc.}\} \).
$C_{uc}^{Act}(A)$: unit competence acquired by an actor (Act) for the execution of the activity (A).

\[
\langle C_{uc}^{Act}(A_i), [R_{NM}^{Act}], [N_{ctxt}^{Act}], [R_{M}^{Act}] \rangle
\]

where

- $C_{uc}^{Act}(A_i)$ is the identification of the unit competence acquired by an actor $Act_j$ for the execution of the activity $A_i$,

- $[R_{NM}^{Act}] = \begin{bmatrix} R_{NM}^{Act}(A_i|R_1) & \text{type} \\ \vdots & \vdots \\ R_{NM}^{Act}(A_i|R_k) & \text{type} \end{bmatrix}$, $R_{NM}^{Act}(A_i|R_k)$ indicates the non-material resource $R_k$ acquired by the actor $Act_j$ for the execution of the activity $A_i$, $type \in \{\text{knowledge, know-how, social-attitude}\}$,

- $[N_{ctxt}^{Act}] = \begin{bmatrix} N_{ctxt}^{Act}(A_i|R_1) \\ \vdots \\ N_{ctxt}^{Act}(A_i|R_k) \end{bmatrix}$, $N_{ctxt}^{Act}(A_i|R_k)$ indicates the skill level of the non-material resource $R_k$ acquired by the actor $Act_j$ for the execution of the activity $A_i$,

- $[R_{M}^{Act}] = \begin{bmatrix} R_{M}^{Act}(A_i|R_1) & \text{type} \\ \vdots & \vdots \\ R_{M}^{Act}(A_i|R_k) & \text{type} \end{bmatrix}$, $R_{M}^{Act}(A_i|R_k)$ indicates the material resource $R_k$ at the disposal of actor $Act_j$ and used for the execution of the activity $A_i$, $type \in \{\text{technique, information, process, etc.}\}$.

**Individual Competencies:** An individual competence ($C_{ic}^{Act}$) or (IC) is also a sub-class inheriting from the competence class. An individual competence is the second level of aggregation of the competencies.

This competence represents the set of unit competencies acquired by an actor $j$ ($Act_j$) and his ability to mobilize, in an efficient manner, some of his unit competencies $C_{uc}^{Act}(A)$ (unit competence acquired by an actor (Act) for the execution of the activity (A)). An individual competence ($C_{ic}^{Act}$) or (IC) can be required or acquired and thus formally defined as:
\((C_{ic}^{\text{Act}})\): individual competence acquired by an actor (Act).

\[ (C_{ic}^{\text{Act}}) = \{ C_{ic}^{\text{Act}_j}, [C_{uc}^{\text{Act}_j}(A)], [R_{NM}^{\text{Act}_j}] \} \]

where

- \(C_{ic}^{\text{Act}_j}\) is the identification of the individual competence acquired by the actor \(\text{Act}_j\),

- \([C_{uc}^{\text{Act}_j}(A)]\) = 
  \[
  \begin{bmatrix}
  C_{uc}^{\text{Act}_j}(A_1) \\
  \vdots \\
  C_{uc}^{\text{Act}_j}(A_k)
  \end{bmatrix},
  \]
  \(C_{uc}^{\text{Act}_j}(A_k)\) indicates the unit competence acquired by the actor \(\text{Act}_j\) for the execution of the activity \(A_k\) \((k \in [1, \ldots, n])\), \(n\) being the whole number of activities executed (or that can be executed) by the actor \(\text{Act}_j\),

- \([R_{NM}^{\text{Act}_j}]\) = 
  \[
  \begin{bmatrix}
  R_{NM}^{\text{Act}_j}(1) \\
  \vdots \\
  R_{NM}^{\text{Act}_j}(k)
  \end{bmatrix},
  \]
  \(R_{NM}^{\text{Act}_j}(k)\) indicates a particular non-material resource acquired by the actor \(\text{Act}_j\). This particular non-material resource is the actor’s ability to combine, in an efficient manner, a number \(m\) of his unit competencies \(C_{uc}^{\text{Act}_j}(A_k), k \in [1, \ldots, m]\), the syntax of the expression of this resource is:

  "to know \(\cup\) combining \(\cup\) \(C_{uc}^{\text{Act}_j}(A_k)\)"

\((C_{ic}^{\text{Act}_j})\): individual competence required for an actor (Act).

\[ (C_{ic}^{\text{Act}_j}) = \{ C_{ic}^{\text{Act}_j}, [C_{uc}^{\text{Act}_j}(A)], [R_{NM}^{\text{Act}_j}] \} \]

where

- \(C_{ic}^{\text{Act}_j}\) is the identification of the individual competence required for an actor \(\text{Act}_j\),

- \([C_{uc}^{\text{Act}_j}(A)]\) = 
  \[
  \begin{bmatrix}
  C_{uc}^{\text{Act}_j}(A_1) \\
  \vdots \\
  C_{uc}^{\text{Act}_j}(A_k)
  \end{bmatrix},
  \]
  \(C_{uc}^{\text{Act}_j}(A_k)\) indicates the unit competence required by an actor \(\text{Act}_j\) for the execution of the activity \(A_k\) \((k \in [1, \ldots, n])\), \(n\) being the whole number of activities that must be executed by an actor \(\text{Act}_j\),
Collective competence: A collective competence (CC), $C_{cc}^{\{Act_1,\ldots,Act_n\}}(P_i)$ or $C_{cc}^{Team_j}(P_i)$ is also a sub-class inheriting from the competence class. A collective competence is the highest level of aggregation of the competencies. A (CC) represents the competence arising from a team of actors $\{Act_1, Act_2, \ldots, Act_n\}$ (with $n$ the number of actors involved in the process $P_i$) building a team (Team$_j$) for the execution of a process $P_i$.

$C_{cc}^{al[Team_j]}(P_i)$: collective competence acquired by a team (Team) of actors for the execution of a process ($P_i$). $C_{cc}^{al[Team_j]}(P_i)$ is formally defined as:

$$\langle C_{cc}^{\text{Team}_j}(P_i), [C_{cu}^{al[Act_j]}(A)], [R_{NM}^{al[\cup\{Act\}]}, [N_{\text{ctxt}}^{al[\cup\{Act\}]}, cc] \rangle$$

where

- $C_{cc}^{al[Team_j]}(P_i)$ is the identification of the collective competence acquired by the team Team$_j$ for the execution of the process $P_i$,

- $[C_{cu}^{al[Act_j]}(A)] = \begin{bmatrix} C_{cu}^{al[Act_j]}(A_1) \\ \vdots \\ C_{cu}^{al[Act_j]}(A_m) \end{bmatrix}$
with $C_{cu}^{\text{Act}_{j}}(A_k)$ indicates all unit competencies acquired by the actors ($\text{Act}_{j}$) ($j \in \{1, \ldots, n\}$ and $n$ is the whole number of actors involved in the process) and necessary for the execution of the activities ($\text{Act}_{k}$) ($k \in \{1, \ldots, m\}$ and $m$ indicates the whole number of activities involved in the process $P_i$).

$$- \left[ R_{\text{NM}}^{A} \right] = \begin{bmatrix} R_{\text{NM}}^{\{... \text{Act}_u \cup \ldots \cup \text{Act}_v ...\}} \langle P_i \mid R_1 \rangle \text{ type} \\ \vdots \\ R_{\text{NM}}^{\{... \text{Act}_u \cup \ldots \cup \text{Act}_v ...\}} \langle P_i \mid R_p \rangle \text{ type} \end{bmatrix}$$

with $R_{\text{NM}}^{\{... \text{Act}_u \cup \ldots \cup \text{Act}_v ...\}} \langle P_i \mid R_w \rangle$ (with $w \in \{1, \ldots, p\}$) indicates a non-material resource ($R_w$) acquired by a team of actors $\{... \text{Act}_u \cup \ldots \cup \text{Act}_v ...\}$ (where $u, v \in \{1, \ldots, n\}$ and $n$ is the number of actors sharing a non-material resource ($R_w$)) involved in the execution of the process ($P_i$), $\text{type} \in \{\text{knowledge, know-how, social-attitude}\}$,

$$- \left[ N_{\text{ctxt}}^{A} \right] = \begin{bmatrix} N_{\text{ctxt}}^{\{... \text{Act}_u \cup \ldots \cup \text{Act}_v ...\}} \langle P_i \mid R_1 \rangle \\ \vdots \\ N_{\text{ctxt}}^{\{... \text{Act}_u \cup \ldots \cup \text{Act}_v ...\}} \langle P_i \mid R_p \rangle \end{bmatrix}$$

with $N_{\text{ctxt}}^{\{... \text{Act}_u \cup \ldots \cup \text{Act}_v ...\}} \langle P_i \mid R_w \rangle$ indicating the skill level for the non-material resource ($R_w$) acquired by a set of actors $\{... \text{Act}_u \cup \ldots \cup \text{Act}_v ...\}$ (with $u, v \in \{1, \ldots, n\}$ and $n$ is the number of actors sharing the non-material resource ($R_w$)) involved in the process ($P_i$).

$C_{cc}^{\text{Team}_{j}}(P_i)$: **Collective competence required by a team of actors involved in the execution of a process** ($P_i$).

$C_{cc}^{\text{Team}_{j}}(P_i)$ is formally defined as:

$$\langle C_{cc}^{\text{Team}_{j}}(P_i), [C_{cu}^{\text{Act}_{j}}(A)], [R_{\text{NM}}^{A}], [N_{\text{ctxt}}^{A}], cc \rangle$$

where

- $C_{cc}^{\text{Team}_{j}}(P_i)$ is the identification of the collective competence required for the team Team$_{j}$ for the execution of the process ($P_i$),

$$- \left[ C_{cu}^{\text{Act}_{j}}(A) \right] = \begin{bmatrix} C_{cu}^{\text{Act}_{j}}(A_1) \\ \vdots \\ C_{cu}^{\text{Act}_{j}}(A_m) \end{bmatrix}$$
with $C^{\text{uc}}_{\text{Act}}(A_k)$ indicates the set of the unit competencies required for the actors $(\text{Act}_j)$ ($j \in \{1, \ldots, n\}$ and $n$ is the number of actors involved in the process) in order to execute the activities $(A_k)$ ($k \in \{1, \ldots, m\}$ and $m$ is the whole number of activities of the process $(P_i)$)

$$-[R^{\text{NM}}_{\{\cup\text{Act}\}}] = \begin{bmatrix} R^{\text{NM}}_{\{\ldots \text{Act}_u \cup \ldots \text{Act}_v \ldots\}}(P_i | R_1) & type \\ : & : \\ R^{\text{NM}}_{\{\ldots \text{Act}_u \cup \ldots \text{Act}_v \ldots\}}(P_i | R_o) & type \end{bmatrix}$$

with $R^{\text{NM}}_{\{\ldots \text{Act}_u \cup \ldots \text{Act}_v \ldots\}}(P_i | R_k)$ is the identification of a non-material resource $(R_k)$ ($k \in \{1 \ldots o\}$ and $o$ is the number of $(R_{\text{NM}})$ required for the team) required for a set of actors $(\ldots \text{Act}_u \cup \ldots \text{Act}_v \ldots)$ (with $u, v \in \{1, \ldots, n\}$ and $n$ is the number of actors that should own and share the non-material resource $(R_k)$ for the execution of the process $(P_i)$), $type \in \{\text{knowledge, know-how, social-attitude}\}$

$$[N^{\text{r}}_{\text{ctx}}_{\{\cup\text{Act}\}}] = \begin{bmatrix} N^{\text{r}}_{\text{ctx}}_{\{\ldots \text{Act}_u \cup \ldots \text{Act}_v \ldots\}}(P_i | R_1) \\ : \\ N^{\text{r}}_{\text{ctx}}_{\{\ldots \text{Act}_u \cup \ldots \text{Act}_v \ldots\}}(P_i | R_o) \end{bmatrix}$$

with $N^{\text{r}}_{\text{ctx}}_{\{\ldots \text{Act}_u \cup \ldots \text{Act}_v \ldots\}}(P_i | R_k)$ is the skill level of the non-material resource $(R_k)$ ($k \in \{1 \ldots o\}$) required for a set of actors $(\ldots \text{Act}_u \cup \ldots \text{Act}_v \ldots)$ (with $u, v \in \{1, \ldots, n\}$ and $n$ is the number of actors that should own and share the non-material resource $(R_k)$ for the execution of the process $(P_i)$). In this case the context is only $(P_i)$.

**Human:** this a special class of organization objects used in support of execution of activities. This entity will inherit all properties of the object organization just defined. Two generic classes of resources can be defined: **Actor or Person** class and **Team** class. A human $(H)$ can be formally defined as:

$$\langle H_{\text{Id}}, H_{\text{Na}}, [R_{\text{Identity}}], type(\text{actor, team}) \rangle$$

where $H_{\text{Id}}$ is the human resource identification, $H_{\text{Na}}$ is the human resource name, $[R_{\text{Identity}}]$ is an array which terms represent the usual identification characteristics of human resource (i.e. from a human resource software), $type$ is an attribute representing the type of the human resource defined by the following set of values $[\text{actor (or person), team}]$. 

15
**Actor (or Person):** an actor (or person) (Act) is a sub-class inheriting from the human class. An actor (Act) can be formally defined as:

\[
(\text{Act}) = (\text{Act}_j, \text{Act}_\text{Na}, [\text{Act}_\text{Identity}])
\]  

(9)

where \( \text{Act}_j \) is the actor identification, \( \text{Act}_\text{Na} \) is the name of the actor, \([\text{Act}_\text{Identity}]\) is a matrix which terms represent the usual identification characteristics of a person (i.e. from an human resource software).

**Team:** a team (Team) is a sub-class inheriting from the human resource class. A team (Team) can be formally defined as:

\[
(\text{Team}) = (\text{Team}_j, \text{Team}_\text{Na}, [\text{Team}_\text{Identity}])
\]  

(10)

where \( \text{Team}_j \) is the team identification, \( \text{Team}_\text{Na} \) is the name of the team, \([\text{Team}_\text{Identity}]\) is an array which terms represent the usual identification characteristics of a set of persons (i.e. from a human resource software).

**Organization Unit:** an organization unit defines an element of an organization structure provided with authority and responsibility for identified activities and enterprise objects of the enterprise [Vernadat F., 2002]. It defines a decision center at a certain decision level (position, department, division, direction and so on). An organization unit (\( OU \)) can be formally defined as:

\[
(OU) = (OU_{\text{Id}}, OU_{\text{La}}, OU_{\text{Des}}, \text{responsible},\text{ responsibilities}, \text{authorities},)
\]

where \( OU_{\text{Id}} \) is the organization unit identification, \( OU_{\text{La}} \) is the organization unit label, \( OU_{\text{Des}} \) is the description of the organization unit, \( \text{responsible} \) is the actor in charge of this organization unit, \( \text{responsibilities} \) is the set of responsibilities assigned to the organization unit, \( \text{authorities} \) is the set of authorities exercised by the organization unit.
5 Model implementation

This section presents a implementation of UECML applied to a producer of railway vehicles. The manufacture of passenger railway vehicles require mainly boilermaking activities illustrated by figure 6.

![Diagram of the boilermaking activities]

Fig. 6. Model of the boilermaking activities applied to a producer of railway vehicles

Let us consider two processes of boilermaking

- \( \alpha \): Front wall assembly - 2 windows
- \( \beta \): Back wall assembly - 4 windows (described in figure 6)

respectively involving the activities:

- To set up the metal sheets pos.2 and pos.4 in a gauge (\( A_1 \)) and to weld metal sheets (\( A_2 \)) (for the process \( \alpha \))
- To weld metal sheets (\( A_2 \)) et to rectify the wall pos.1-6 to 10 (\( A_3 \)) (for the process \( \beta \))

(\( A_2 \)) being involved in both processes. Each activity requires a certain number of (\( R_{NM} \)) (the (\( R_M \)) are voluntarily not taken into account here).
5.1 Activities and resources

The activities $A_i$ require the following ($R_{NM}$):

For the activity $A_1$:

- $R_{NM}^{(A_1|R_1)}$: To be able to read technical drawings
- $R_{NM}^{(A_1|R_2)}$: To aim at positioning a metal sheet as close as possible to the other one
- $R_{NM}^{(A_1|R_{12})}$: Knowing locksmith’s trade

For the activity $A_2$:

- $R_{NM}^{(A_2|R_{12})}$: Knowing locksmith’s trade
- $R_{NM}^{(A_2|R_3)}$: To be able to read welding specifications on drawings

For the activity $A_3$:

- $R_{NM}^{(A_3|R_4)}$: To be able to choose the type of heating according to the curves of the metal sheet

This organization employs three actors: ($\text{Act}_1$), ($\text{Act}_2$), ($\text{Act}_3$) who are likely to respectively execute the following activities: $A_1$ for the actor ($\text{Act}_1$), $\{A_1, A_2\}$ for the actor ($\text{Act}_2$) and $\{A_1, A_2, A_3\}$ for the actor ($\text{Act}_3$). Each one of these three actors acquired a certain number of ($R_{NM}$):

For the actor ($\text{Act}_1$):

- $R_{NM}^{(\text{Act}1|(R_1)})$: To be able to read technical drawings
- $R_{NM}^{(\text{Act}1|(R_5)})$: To know the cold welding techniques

For the actor ($\text{Act}_2$):

- $R_{NM}^{(\text{Act}2|(R_{12})})$: Locksmith’s trade knowledge
- $R_{NM}^{(\text{Act}2|(R_3))}$: To be able to read welding specifications on drawings

For the actor ($\text{Act}_3$):

- $R_{NM}^{(\text{Act}3|(R_{12})})$: Locksmith’s trade knowledge
- $R_{NM}^{(\text{Act}3|(R_3))}$: To be able to read welding specifications on drawings
- $R_{NM}^{(\text{Act}3|(R_4))}$: To be able to choose the type of heating according to the curves of the metal sheet
5.2 Unit and individual competencies

As an example, let us consider the unit competence (UC) required by the activity \( A_1: C_{cu}^t(A) \). This (UC) is formalized as indicated in table 1. Similarly, the unit competence (UC) acquired by the actor \( \text{Act}_2 \) and useful for the execution of the activity \( A_2: C_{cu}^{\alpha | Act_2}(A_2) \) will be formalized as shown in table 2.

\[
\begin{bmatrix}
R_{NM}^{(A_1|R_1)} & \text{Knowledge } N_{\alpha}^{(A_1|R_1)} = 1 \\
R_{NM}^{(A_1|R_2)} & \text{Knowledge } N_{\alpha}^{(A_1|R_2)} = 2 \\
R_{NM}^{(A_1|R_{12})} & \text{Knowledge } N_{\alpha}^{(A_1|R_{12})} = 1
\end{bmatrix}
\]

Table 1
\( C_{cu}^t(A_1): \) CU required by the activity \( A_1 \)

\[
\begin{bmatrix}
R_{NM}^{(\alpha | Act_2)(B|R_{12})} & \text{Knowledge } N_{\alpha}^{(\alpha | Act_2)(B|R_{12})} = 2 \\
R_{NM}^{(\alpha | Act_2)(B|R_{12})} & \text{Knowledge } N_{\beta}^{(\alpha | Act_2)(B|R_{12})} = 1 \\
R_{NM}^{(\alpha | Act_2)(B|R_3)} & \text{Knowledge } N_{\alpha}^{(\alpha | Act_2)(B|R_3)} = 3 \\
R_{NM}^{(\alpha | Act_2)(B|R_3)} & \text{Knowledge } N_{\beta}^{(\alpha | Act_2)(B|R_3)} = 2
\end{bmatrix}
\]

Table 2
\( C_{cu}^{\alpha | Act_2}(A_2): \) CU acquired by the actor 2 for the execution of the activity \( A_2 \)

Let us now look at the individual competence (IC) acquired by the actor \( \text{Act}_3: C_{ci}^{(\alpha | Act_3)} \). Three (UC) are involved in the construction of (IC): \( C_{cu}^{(\alpha | Act_2)}(A_1) \), \( C_{cu}^{(\alpha | Act_2)}(A_2) \) and \( C_{cu}^{(\alpha | Act_2)}(A_3) \) described in table 3.

\[
\begin{bmatrix}
R_{NM}^{(\alpha | Act_3)(A_1|R_{12})} & \text{Knowledge } N_{\alpha}^{(\alpha | Act_3)(A_1|R_{12})} = 1 \\
R_{NM}^{(\alpha | Act_3)(A_2|R_{12})} & \text{Knowledge } N_{\alpha}^{(\alpha | Act_3)(A_2|R_{12})} = 3 \\
R_{NM}^{(\alpha | Act_3)(A_2|R_{12})} & \text{Knowledge } N_{\beta}^{(\alpha | Act_3)(A_2|R_{12})} = 2 \\
R_{NM}^{(\alpha | Act_3)(A_2|R_3)} & \text{Knowledge } N_{\alpha}^{(\alpha | Act_3)(A_2|R_3)} = 2 \\
R_{NM}^{(\alpha | Act_3)(A_2|R_3)} & \text{Knowledge } N_{\beta}^{(\alpha | Act_3)(A_2|R_3)} = 1 \\
R_{NM}^{(\alpha | Act_3)(A_3|R_4)} & \text{S-F } N_{\beta}^{(\alpha | Act_3)(A_3|R_4)} = 1
\end{bmatrix}
\]

Table 3
\( C_{ci}^{(\alpha | Act_3)}: \) CI acquired by the actor 3
5.3 Collective Competencies

Non-material resources are also directly required by the 2 processes $\alpha$ et $\beta$ (See figure 8):

For process $\alpha$:

- $R_{NM}^{(\alpha|R_1)}$: To know the safety requirements of the boilermaking workshop

For process $\beta$:

- $R_{NM}^{(\beta|R_2)}$: To be able to collaborate narrowly
- $R_{NM}^{(\beta|R_3)}$: To be able to visualize the deformations of the metal sheet with the rule
Some of the actors are involved in a team which acquired some useful ($R_{NM}$) for the execution of the processes. The actors ($Act_1$) and ($Act_2$) constitute the team ($Team_1$), this team having acquired the following resources:

- $R_{NM}^{Team_1}(R_1)$: To know the safety requirements of the boilermaking workshop
- $R_{NM}^{Team_1}(R_2)$: To be able to collaborate narrowly

while ($Team_2$) mobilizing the actors ($Act_2$) and ($Act_3$) has acquired the resource:

- $R_{NM}^{Team_2}(R_3)$: To be able to visualize the deformations of the metal sheet with the rule

In order to be able to clarify the competences, values must be attributed to the required and acquired levels of the different ($R_{NM}$). For that, it is necessary to understand that a ($R_{NM}$) cannot be evaluated except its context. Indeed if only 5 ($R_{NM}$) are required, it doesn’t imply the evaluation of 5 required levels, but probably more.
Required and acquired levels

For example the necessary level of $R_{NM}(3)$ required by the activity ($A_2$) can be different if this activity proceeds within the process $\alpha$ or $\beta$ and thus to request the evaluation of 2 necessary levels for the same resource $R_{NM}(3)$. Indeed the requirements for being able to read technical drawings are more difficult if the activity ($A_2$) of welding metal sheets takes place in the process $\beta$ of Back wall assembly - 4 windows. The table 4 summarize the required levels of the different $R_{NM}$ taking into account their context. It was decided to evaluate these levels using 4 levels scale 1 to 4: {1 : Basic knowldege, 2 : Intermediate, 3 : Expert, 4 : To be able to teach}, for the type of resource: knowledge, {1 : To understand, 2 : To do, 3 : To supervise, 4 : To be able to teach}, for the type of resource: know-how.

On the same principle, the levels acquired by the actors in the context of their activities within process have to be evaluated. Table 4 describes the different acquired levels. The collective competence can now be formalized. For example the collective competence (CC) acquired by the set of actors $\{\text{Act}_1, \text{Act}_2\}$ involved in the team $Team_1$ for the execution of the process $\alpha$: $C_{cc}^{Team_1}(\alpha)$ will be written as indicated in table 5.
The two following (UC) are involved in the construction of (CC): $C_{cc}^{\alpha|Act_1(A)}$, $C_{uc}^{\alpha|Act_2(B)}$ like the 2 resources:

1. $R_{NM}^{\alpha|\{Act_1\cup Act_2\}(\alpha|R_1)}$ equal to $R_{NM}^{\alpha|Team_1(\alpha|R_1)}$ or $PR_{NM}(1)$
2. $R_{NM}^{\alpha|\{Act_1\cup Act_2\}(\alpha|R_2)}$ equal to $(R_{NM}^{\alpha|Team_1(\alpha|R_2)})$ or to $PR_{NM}(2)$ described at Table 5.

The levels $\mu = N_{\alpha}^{\alpha|\{Act_1\cup Act_2\}(\alpha|R_1)}$ of $PR_{NM}(1)$ and $\nu = N_{\alpha}^{\alpha|\{Act_1\cup Act_2\}(\alpha|R_2)}$ of $PR_{NM}(2)$ can be evaluated on the same principle as the acquired levels. These levels evaluate the skill level of a non-material resource for a team.

6 Discussion and conclusion

This paper has overviewed some of Enterprise Modelling methodologies. Although quite a lot of the academic literature/research reports and enterprise models were reported to have been implemented in industry, to the authors’ knowledge, no paper giving a competence based enterprise model with the same and single formalism at different levels of abstraction and able to valorise the competencies and industrial implementations has been found. The present paper aims to fill this gap. The proposed model (UECML) concerning competence based enterprise modelling is able to formalize in a coherent way the concept of competence in the framework of enterprises. The formalism suggested is independent of any model, able to integrate several levels of abstraction and based on three natures different of competences (CU, CI and CC). UECML thus makes it possible to initiate a step of competences identification according to a desired degree of detail. The principal challenges are:

- How to use UECML for an implementation?
- To identify the right level of detail (level of aggregation) for using UECML.
Indeed, an optimal competences management requires a adequate level of detail in the use of UECML in order to keep a better visibility of competences without making too simplifying assumptions or building a competence based model with a too fine granularity. On the other hand, UECML allows us to also initiate the phase of the competencies valorization. The implementation of UECML applied to a producer of railway vehicles illustrates that the proposed model was effective in competence based enterprise modelling. UECML allows us to develop an experimental prototype for supporting this implementation. The proposed experimental soft prototype is simple and can be easily implemented by both industrial and services enterprises in order to identify the competencies.

Modelling an enterprise in terms of competencies is a very interesting approach as these are more and more recognized as key competitive factors. These competencies need to be carefully managed, maintained and allocated in a flexible and dynamic way for most organizations, reason for which in this article have been introduced the basis of UECML (Unified Enterprise Competencies Modelling Language).

The future research direction should extend to a formalism for defining more precisely the resources and the criteria/indicators used in the valorization phase. This extension will make it possible to take into account the problem of granularity and will facilitate the valorization of competences approach. The use of this "bottom up" competence based approach through UECML seems a promising field of investigation.

Acknowledgment

This work was carried out under the financial support of the Swiss Innovation Promotion Agency (CTI 6126.1). The authors wish to thank the industrial partners for their collaboration.

References


