Unified Resource Modelling:

Integrating knowledge into business processes

João Neves jneves@ieee.org
André Vasconcelos andre@ceo.inesc.pt
Artur Caetano artur.caetano@inesc.pt
José Tribolet jmt@inesc.pt
Pedro Sinogas sinogas@ieee.org
Ricardo Mendes rmendes@ceo.inesc.pt

CEO/INOV

July 2001

Abstract

This paper presents an overview of the state of the art of resource modelling used by either information systems specialists and human resources professionals. A generic framework for resource modelling is used for comparing these two approaches in terms of the context and description of the work the resource does and is able to do. This is a base work to the development of a new tool for human resources management that makes explicit the connection between human resources and business processes.

Keywords: Resource Model, Knowledge, Business Processes, Competences

Introduction

Today there are mainly two approaches to resource modelling on a business environment. The main differences of these approaches are their origins, context and tools. In the software development field, the need to increase the effectiveness ratio of software systems' implementations lead to the need of a better understanding of the business that software serves and the organizations where it was integrated. The objective was to increase the effectiveness of software systems' implementations [1]. This approach uses extensions of the concepts from industrial engineering, that is process based, and applies it to organizations. In terms of resource modelling uses techniques available for system modelling, like Entity-Relationship diagrams and, more recently, the Unified Modelling Language for representing business processes and environment[2,3].

The other main approach used in resource modelling comes from the human resources management field. Coming from social sciences, the focus is on people, how to identify and evaluate what they are able to do. The most used technique has been the use of competences to define what human resources are able to do, using them as a basis for all human resources processes: evaluation, workforce assessment and planning, etc.[4].

So we have an approach that focus in what is done (business processes) and another that focus in what people can do (competences). This paper presents an attempt to model

them together that is based on Selic's framework for resource modelling[5]. The objective of the model is to improve business process execution efficiency in an organization. This improvement should come from better workforce planning and the possibility of interchanging resources between activities.

Unified Resource Modelling Framework

Resources are the basis of all systems. They define the limits of the systems' operations and efficiency. Therefore the ability to accurately model resources is an important facet of business modelling. The Unified Resource Modelling Framework is based on Selic's generic framework for resource modelling[5]. This framework does not define any specific resources - like computer or person - but defines a generic approach for modelling such resources. The model focuses on the notion of an *abstract resource*, which defines the common characteristics of resources regardless of their specific manifestation.

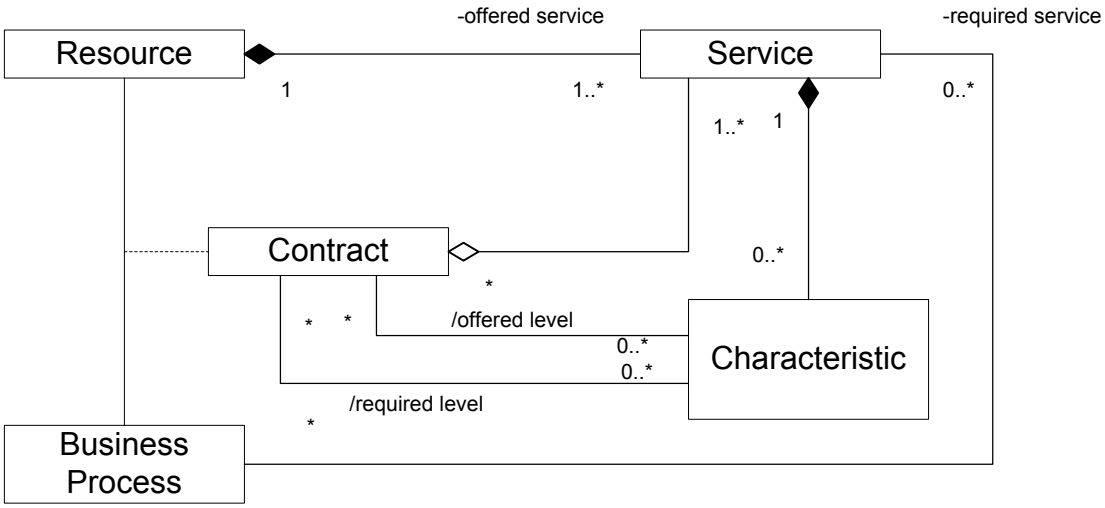

Figure 1 - Unified Resource Model Framework

Resource
any entity for which services can be defined.

Process
the client that defines the purpose of using a resource and the activities it should perform.

Service
the basic activities a resource can perform.

Characteristic
represents some aspect of the performance of a quantifiable service.

Contract
involves one or more services that a resource offers and that the process requires.

The absence of any of these classes (or their equivalents) in a model means that something is being ignored: if a *process* is not present, then the purpose and the work the resource is providing is not taken into account. If *service* is ignored, it is not known what needs to be done, nor what the resource can do. Without *characteristic* we cannot evaluate services and distinguish different levels of service. Without *contract* we aren't able to distinguish the level of service that a resource can provide from the level of service

it is providing in a defined process. Taking this into account we present three different approaches to resource modelling.

Eriksson-Penker Business Extensions

The Unified Modelling Language (UML) provides system architects working on object analysis and design with one consistent language for specifying, visualizing, constructing, and documenting the artefacts of software systems, as well as for business modelling[6]. The main features of the language are to provide a formal definition of a common object analysis and design meta-model to represent semantic, IDL specifications for mechanisms for model interchange between tools and a graphic syntax for representing human-readable models.

Eriksson and Penker built on the impact caused by the use of UML in software systems design and proposed the Eriksson-Penker Business Extensions with the goal of enabling business modelling with a language that has been used mainly for software systems design[3]. The result is not innovative in terms of business modelling, nor was intended to be, but presents a synthesis of concepts that are integrated in a UML model.

A business model is composed of *views*, each of which captures information about one or more specific aspects of the business; *diagrams* that represent a specific part of the business structure inside a view; *objects and processes*, that represent the "things" in a business (objects) or the functions in the business that consume, refine, or use objects to affect or produce other objects (processes).

To illustrate the meaning of resource they have chosen two definitions: *A resource is an entity which can play a role in the realization of a certain class of tasks*[7]; *A resource is a concept used in the business, and represents anything that we choose to evaluate as a whole*[8].

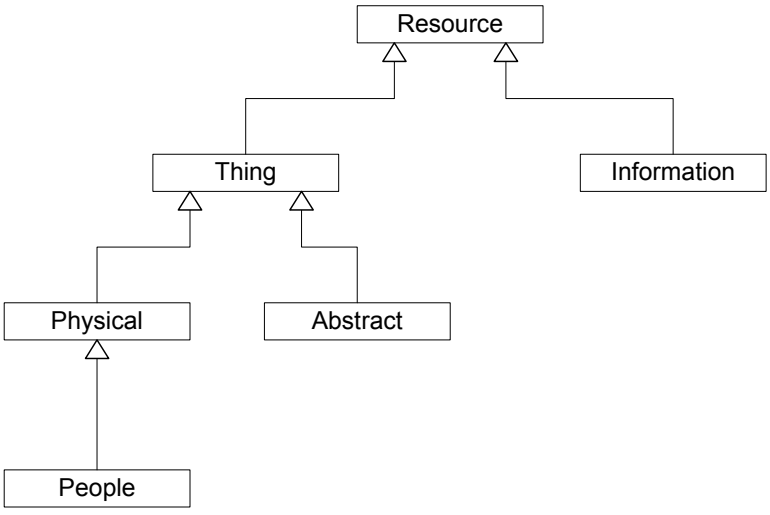

Figure 2 - Types of resources

The resource meta-model defines four resource types:

Physical

entity with material reality that occupies a volume of space like raw materials, parts, etc.

Abstract

idea or concept like a contract, role, account, etc.

Information

representation of concept, thing, or another information object. It holds information about other objects.

People

a human being acting in the process. It's a specialization of a physical resource to emphasize and identify people in the process.

To help analysts, Eriksson and Penker, also provide several patterns of resources usually found in business environments. These patterns include from document specification to the relation between thing and information, including the actor-role pattern and others.

In terms of resource modelling one of the most interesting patterns is the Actor-Role one. This pattern actually separates a resource (actor) from its services (role) and puts it in the context of a process.

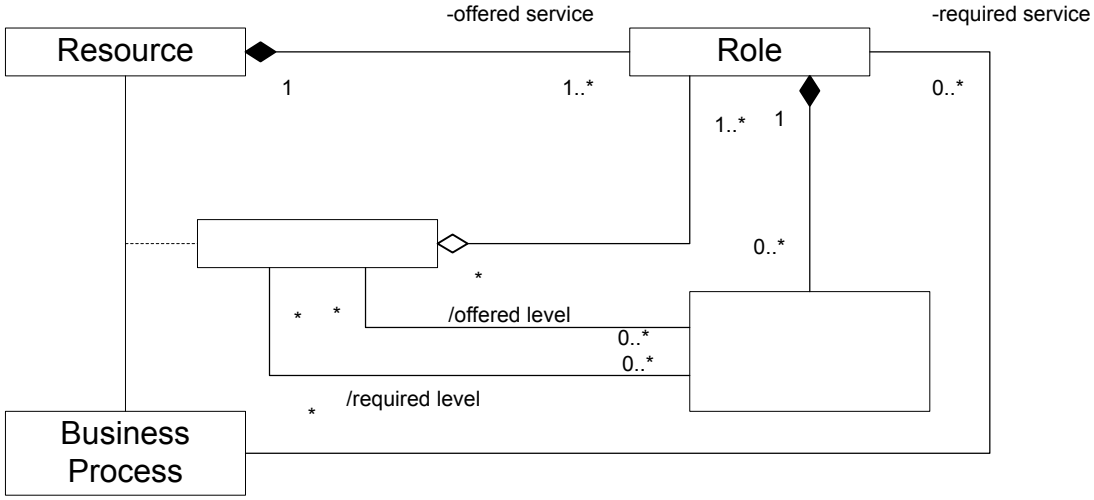

Figure 3 - Framework's classes included in Eriksson-Penker Business Extensions

In terms of resources this model main focus is on how and where they are used. The importance off processes is normal on a model that is supposed to help build systems that will adapt to the working environment. Unfortunately, this means that the descriptions of what resources are able to do is assumed to be what they are doing. This reduces the possibility of replacing the resources that are actually executing a process by others that are in roles that need a lower level of service.

Enterprise Knowledge Development

Enterprise Knowledge Development (EKD) is a collaborative project between the SYSLAB of the Royal Institute of Technology and Stockholm University, the University of Paris-1 and the UMIST in Manchester, UK. EKD is an integrated collection of methods, techniques, and tools that will support the process of analysing, planning, designing, and changing a business. It was developed in the beginning of the eighties and has suffered several iterations throughout the years and several projects. EKD extends capabilities of several traditional software development methodologies, such as ER (Entity-Relationship)

diagrams and data-flow diagrams, by providing additional views of the software, product or business being designed[2].

It includes six models: the enterprise model, the goal model, the rule model, the concept model, the process model, the actor and resource model and the technical component and requirement model. The actor and resource model defines the types of actors and resources, or individual actors, involved in enterprise activities. The model describes how different actors relate between themselves, goals and processes through relationships with the goal and business process model.

The main objective of the Actor and Resource model it to clarify who is/should be performing which processes and tasks and how is the reporting and responsibility structure between actors defined. In this model actors and resources can be individuals, organizational units, non-human resources and roles. *Individual* denotes a person in the enterprise. *Organizational unit* can represent every organisational structure in the enterprise such as a group, department, division, section, project, team, subsidiary, etc. *Non-human resource* can be types of machines, systems of different kinds, equipment, etc. *Role* may be played by the individuals, organisational units and non-human resources in different contexts.

The relationships between the components in the actor and resource model are all binary. The can define *responsibility* between actors, between actors and business processes, goals, business rules and goals. Responsibilities can be delegated or transferred among actors. Responsibilities can be organisational, defining who is responsible for a goal, rule, resource or business process, who defines a goal or a rule, who owns a resource or who monitors another actor; responsibilities can also be operational indicating who performs or should perform a business process. The relationships can also define dependency among enterprise actors. An actor depends on another for something that can be either a resource or a business process.

There are two specific relationships on the Actor and Resource Model: the ISA that represents a generalisation between roles and the PartOF relationship that identifies components of actors and resources.

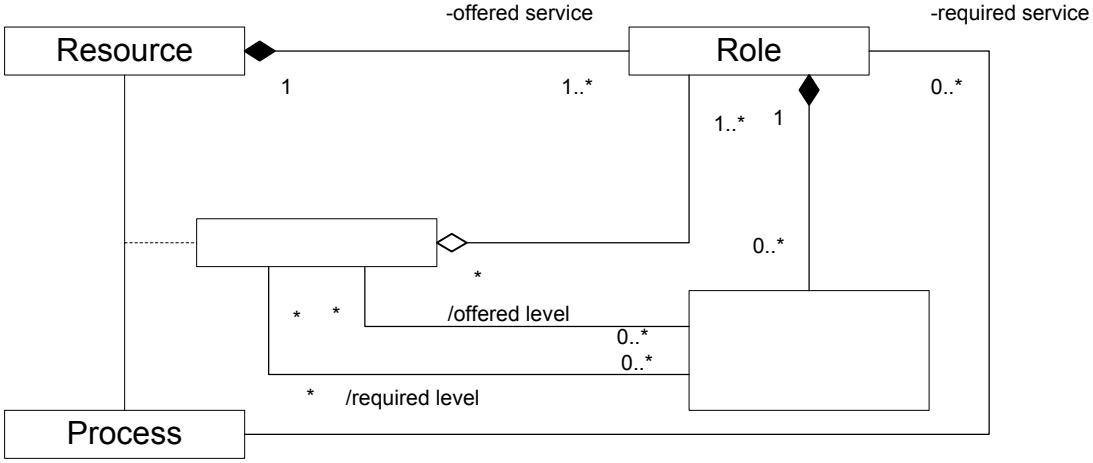

Figure 4 - Framework's classes included in Enterprise Knowledge Development

As can be seen in Figure 4, this model has most of the same characteristics than the one presented earlier.

Human Resources Management

In 1973, the paper “Testing for Competence rather than Intelligence” from David C. McClelland has been credited with launching the competency movement. This movement uses competencies as the basic unit to classify people and relate them to the ability of executing a task in a determined environment. Competencies may be used by most human resources related processes: recruiting, placement, retention, promotion, performance management, succession planning, development and career pathing, pay, etc.

Spencer defines competency has “the underlying characteristic of an individual that is causally related to criterion-referenced effective and/or superior performance in a job or situation”[9]. *Underlying characteristic* means the competency is a deep part of a person and can predict behaviour in a wide variety of situations and job tasks. This also means a competency is part of what Davenport calls tacit knowledge - knowledge that is embedded in people's minds[10]. *Causally related* means that a competency causes or predicts behaviour and performance. *Criterion-referenced* means that, against a defined criterion or standard, the competency actually predicts who does something well or poorly.

There are five types of characteristics, from the most visible to the more tacit: skill, knowledge, self-concept, trait and motives.

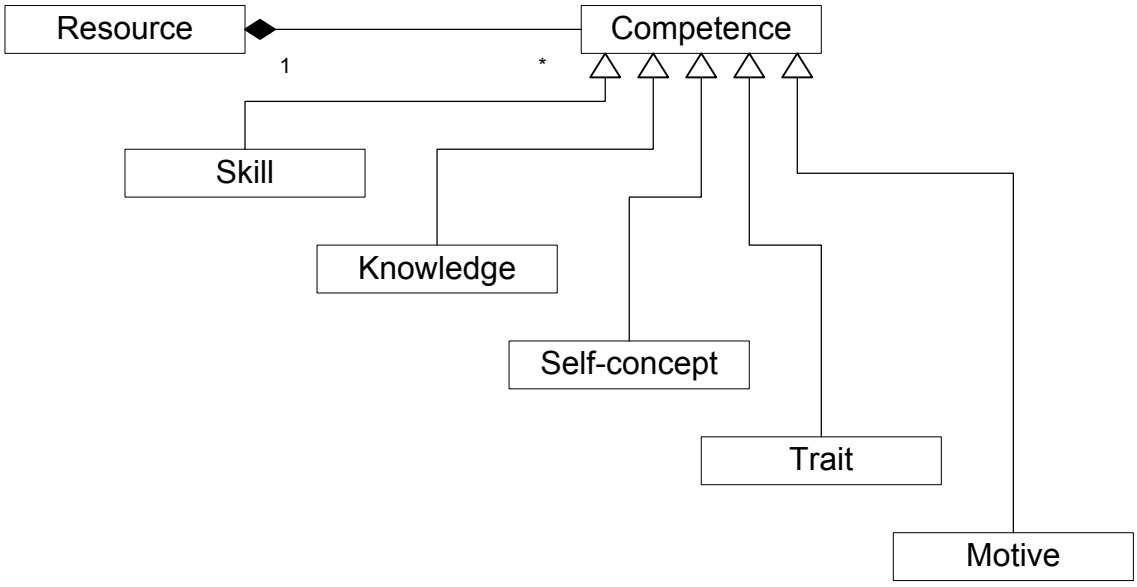

Figure 5 - Relation between resources and competences

The competency definition implies a causal relationship from *intent* to *outcome*. That relationship means that you need to know the intent of someone to be able to predict her actions. That intent can be represented by competencies of the type motive, trait, self-concept and knowledge. Intent originates *action*, which can be represented by skills. Only action generates outcome: the job performance we are trying to improve.

The reason to use criterion-referenced as a defining part of competency is the need to effectively define the impact of competencies on performance. A characteristic is not a competence unless it predicts something meaningful in the real world, so a performance measurement is critical for any kind of competence definition.

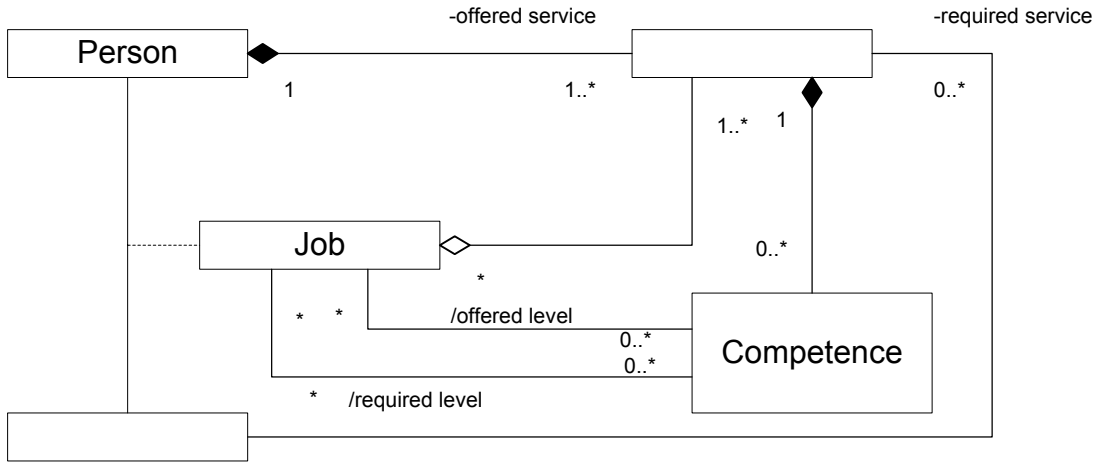

Figure 6 - Framework's classes include in Human Resources Management

The focus of the human resources field is on the resource level. This means that there is an underlying assumption that the contract (job or position) is already well defined, and so are the services needed. In this case one only needs to check if the resource (person) is able to provide the level of services needed. Unfortunately this means that there isn't a direct connection between what is done and a resource, hampering the possibility of feedback that could improve the efficiency of resource allocation.

Conclusions and Speculation

Using the Unified Resource Modelling Framework we confirmed the initial idea that the information systems and human resources have different approaches to resource modelling. We also discussed the problems with the actual resource modelling techniques, especially in what refers to human resources. These problems come from an emphasizes on either affecting resources to a task or following the execution of processes, with neither of the models presented covering both these setup and execution stages of a process. The result is a lack of feedback to the setup stage that hampers the improvement of the methods usage and the impossibility of repeating part of the process setup in the middle of the executing, that is, replacing someone.

We believe that, using this framework as a basic structure, it is possible to build a dynamic model of human resources that integrates both the allocation and work of resources. To test the dynamic model a system will be built to support human resources management processes in an enterprise.

References

[1] K. C. Laudon and J. P. Laudon, *Management Information systems - Organization and technology in the networked enterprise*. Prentice-Hall, sixth edition, 1999.

[2] J. A. Bubenko, Jr., D. Brash, and J. Stirna, "Ekd user guide", tech. rep., Kista, Dept. of Computer and Systems Science, Royal Institute of Technology (KTH) and Stockholm University, 1998.

[3] M. Penker and H.-E. Eriksson, *Business Modeling With UML: Business Patterns at Work*. John Wiley and Sons, 2000.
[4] J. H. Boyett and J. T. Boyett, “Human resources most effective practices across best companies”, tech. rep., PeopleSoft, Inc., May 1999.
[5] B. Selic, “A generic framework for modeling resources with uml”, *Computer*, vol 33, June 2000.
[6] “OMG unified modeling language specification”, tech. rep., Object Management Group, June 1999.
[7] F. Vernadat, *Enterprise Modeling and Integration*. London, England: Chapman and Hall, 1996.
[8] T. Gale and J. Eldred, *Getting Results with the Object-Oriented Enterprise Model*. New York: SIGS Book, 1996.
[9] L. M. Spencer, Jr. and S. M. Spencer, *Competence at Work: models for superior performance*. John Wiley and Sons, Inc., first ed., 1993.
[10] T. H. Davenport and L. Prusak, *Working Knowledge: How Companies Manage What They Know*. Boston, Massachusetts: Harvard Business School Press, 1998.