UNIVERSITAT JAUME I

PhD Thesis

Proposal for Model Driven Knowledge

Version 7.0

PhD: Reyes Grangel Seguer
Supervisor: Dr. Ricardo Chalmeta Rosaleñ

PhD Programme 'Advanced Computer Systems'
Universitat Jaume I
24^{th} of June of 2007

The work presented in this Thesis has been carried out partially in two research stays funded by the NoE, 'Interoperability Research for Networked Enterprises Applications and Software' (INTEROP) (IST-2003-508011), the first in the European Software Institute (ESI) of Bilbao and the second in Ecole Centrale de Lille (ECL) of Lille. It has also been developed within the framework of the Spanish Project funded by the 'Comisión Interministerial de Ciencia y Tecnología' (CICYT), 'Gestión del conocimiento en el ámbito de las empresas virtuales' (DPI2003-02515), carried out by the Research Group 'Integración y Re-Ingeniería de Sistemas' (IRIS) at the Universitat Jaume I of Castelló.

Abstract

Enterprise Modelling is defined as the art of externalising enterprise knowledge, which adds value to the business or needs to be shared. This kind of modelling has been used successfully since its apparition in the 80's in many domains and with different purposes, among them the re-engineering of business processes or the implementation of computer systems. Its constant evolution has given as a result a context in which there are numerous languages, methodologies and tools for Enterprise Modelling available and useful for their purpose, even for modelling virtual enterprises. These languages and methodologies allow model the most of the enterprise dimensions (process, product, organisation, decision, etc.), and they cover different development phases (inicialisation and definition of objectives, definition of requirements, design, etc.). Besides, they provide models that can be integrated, obtaining different views of enterprises from several points of view and strategic levels. Therefore, it can be stated that nowadays Enterprise Modelling allows enterprises to obtain a complete vision of its business with different purposes.

However, there exist still some problems without solution in the context of Enterprise Modelling. The great quantity of existing Enterprise Modelling Languages and Tools causes lack of interoperability among them, therefore it is difficult to exchange enterprise models carried out with different languages or tools. Moreover, the problem to obtain enterprise applications from these models, as well as the management of them, it makes difficult the use of enterprise models as an useful tool in knowledge management and continuous improvement of enterprises. Some international initiatives try to solve the problem of interoperability at horizontal level, such as UEML, INTEROP or ATHENA, defining formats that allow the exchange of enterprise models carried out with different languages or tools. On the other hand, in the context of MDE (Model Driven Engineering), approaches such as the MDA (Model Driven Architecture) defined by the OMG try to define a suitable framework for generating software from enterprise models. However, the key question is how Enterprise Modelling can become the really force for managing enterprise knowledge. To achieve this objective, Enterprise Modelling should cover enterprise knowledge as a dimension in itself, and also making possible that the other modelled enterprise dimensions can provide the required knowledge that enterprises need in each moment. Thus, Enterprise Knowledge Modelling should become in a efficient way to represent knowledge that the enterprises have with the objective to process and use it there and when it was needed.

The research work of this thesis is stated in this context, which has been developed from the KM-IRIS Methodology for the implementation of knowledge management systems, using the requirements obtained in the phase I and II of the Methodology and with the purpose of supporting its phase III for knowledge representation. Therefore, the main contribution of the thesis is a proposal to model enterprise knowledge called, Proposal for MDK.

The Proposal for MDK includes the metamodels and UML2 profiles for the representation of enterprise knowledge and a guide to help enterprises with the development of its knowledge map. The main source to develop these metamodels has been the requirements and metamodels defined in the European Projects INTEROP and ATHENA, as well as the unified modelling languages defined from them, UEML and POP* respectively, with the aim of making easy the exchange of models among enterprises that use different enterprise modelling tools. The objective has been to adapt and to extend the results of both projects to the domain of knowledge management systems. On the other hand, this Proposal is related to a MDA approach, modelling enterprise knowledge at the CIM level, with the purpose that the obtained models can be then easily maintained and transformed at the PIM and PSM level. The Proposal for modelling has not consisted of defining a new modelling language, but UML2 and its new definition of profiles has been used, to extend this modelling language to enterprise knowledge context. The diverse metamodels defined to collect the knowledge requirements before mentioned and the UML2 profiles that implement these metamodels can be used to represent enterprise knowledge according to the Methodology KM-IRIS. Moreover, they can be used to carry out Enterprise Modelling of the remaining traditional enterprise dimensions, since these dimensions are overlapped with a detailed representation that can be done on the predefined conceptual blocks in this Methodology. The application of the profiles for representing knowledge has as result a set of diagrams that according to the Proposal for MDK, which is based on the architecture MDA, are grouped at distinct levels of abstraction, and they make up the diverse models that cover the CIM level and show the Enterprise Knowledge Map.

Thesis Organisation

This document presents a summary in English of the Thesis, 'Propuesta para el modelado del conocimiento empresarial'. It has the same chapters that the original document written in Spanish. Several papers published and in revision performed to disseminate the results of the Thesis are shown in order to show the main conclusions and outcomes of each chapter. Besides in the chapter 5, which is related to the main contribution of the Thesis, the reader can fins more information in the original document since the defined metamodels and UML profiles are written in English. Finally, it is presented for each paper a summary of its status, the abstract and the complete published content.

1. In the **first chapter**, a brief description of the origin and motivation of the Thesis, as well as of the framework of work. It is described also the methodology of work, the research objectives and the main expected results, jointly with the structure of the document [1, 2].

2. In the **second chapter**, the state of the art in Enterprise Knowledge Modelling is presented, analyzing related concepts, purpose, evolution, etc. and the main existing standards, reference architecture and frameworks, languages, etc. as much in the context of Enterprise Modelling as in that of Knowledge Modelling. As a conclusion, the diverse dimensions cover by Enterprise Modelling are analysed and the current problems in this context related to virtual enterprises [3].

3. In the **third chapter**, a revision of UML and other standards defined by the OMGs shown, from the point of view of its utility for modelling enterprise knowledge [4, 5].

4. In the **fourth chapter**, the Methodology KM-IRIS developed by the Research Group IRIS is described, in whose development the PhD student has participated and which constitutes the origin of this thesis. In particular, the Thesis is related to the phase III (Phase of representation) of the Methodology KM-IRIS, as an suitable mechanism for modelling enterprise knowledge [6].

5. In the **fifth chapter**, the main contribution of the Thesis is detailed, a Proposal for modelling enterprise knowledge that includes a metamodel of enterprise knowledge, the UML2 profiles UML2 needed for its implementation and a guide that allows the virtual enterprises to model enterprise knowledge [7, 8, 9, 10].

6. In the **sixth chapter**, a real case study is presented, a virtual enterprise, in which has been applied the KM-IRIS Methodology and in particular the metamodel and its guide to model enterprise knowledge in the phase of representation [11].

7. Finally, in the **seventh chapter** the conclusions and results obtained in the Thesis are shown, as well as the future research lines.

Contents

Chapter 1

Introduction

Title:	**A Methodological Approach for Enterprise Modelling of Small and Medium Virtual Enterprises based on UML. Application to a Tile Virtual Enterprise**
Authors:	R. Grangel and R. Chalmeta
Conference:	First International Conference on Interoperability of Enterprise Software and Applications (INTEROP-ESA'05) - Doctoral Symposium
Pages/Year:	51-54/2005
Place:	Geneva (Switzerland)
Date:	21-2-2005

Abstract

Enterprise Modelling has been used successfully for years with different purposes. Nowadays, there are a lot of languages, methodologies and tools related to Enterprise Modelling, even for modelling Virtual Enterprises. However, some of the Enterprise Modelling weaknesses have not been solved yet. One of the most important is the lack of interoperability among enterprises that use different Enterprise Modelling Languages (EML). Such EML are defined in proprietary formats, and they are only implemented by proprietary and expensive tools. So that, this problem is intensified in Small and Medium Enterprises (SMEs), because they have limited resources.

In this context, this paper shows my Ph.D. thesis proposal describing the problematic situation which is the origin of this research and the objectives suggested

to solve it. The thesis goal is to investigate the possibilities of using UML 2.0 and Profiles mechanism in order to provide a methodological approach for solving interoperability problems to Small and Medium Virtual Enterprises in the context of Enterprise Modelling.

A Methodological Approach for Enterprise Modelling of Small and Medium Virtual Enterprises based on UML. Application to a Tile Virtual Enterprise

Reyes Grangel (Ph.D. Student) and Ricardo Chalmeta (Supervisor)

Dep. de Llenguatges i Sistemes Informàtics, Universitat Jaume I, 12071 Castelló, SPAIN, {grangel, rchalmet}@uji.es

Abstract. Enterprise Modelling has been used successfully for years with different purposes. Nowadays, there are a lot of languages, methodologies and tools related to Enterprise Modelling, even for modelling Virtual Enterprises. However, some of the Enterprise Modelling weaknesses have not been solved yet. One of the most important is the lack of interoperability among enterprises that use different Enterprise Modelling Languages (EML). Such EML are defined in proprietary formats, and they are only implemented by proprietary and expensive tools. So that, this problem is intensified in Small and Medium Enterprises (SMEs), because they have limited resources.
In this context, this paper shows my Ph.D. thesis proposal describing the problematic situation which is the origin of this research and the objectives suggested to solve it. The thesis goal is to investigate the possibilities of using UML 2.0 and Profiles mechanism in order to provide a methodological approach for solving interoperability problems to Small and Medium Virtual Enterprises in the context of Enterprise Modelling.

1 Introduction

The objective of this paper is to describe the proposal for my Ph.D. thesis. This document intends to give a first idea about the thesis origin and objectives. It is structured in three sections. The first one shows background and definitions related to the thesis framework. In the second one, the problematic situation that the thesis intends to solve is described. Finally, the main research objectives are presented.

2 Background and definitions

Enterprise Modelling [18] is the art of 'externalizing' enterprise knowledge, which adds value to the enterprise or can be shared, i.e., representing enterprise in terms of its organisation and operations (processes, behaviour, activities, information, objects and material flows, resources and organisation units, and system infrastructure and architectures). Therefore, this art consists of obtaining enterprise models, that are a computational representation of the structure, activities, processes, information, resources, behaviours, etc. of an enterprise, government or any another type of business. This model can be at the same time descriptive and definitional, including that what is and what should be. And its role should be to obtain a design, analysis and operation of the enterprise according to the model, i.e., driven by the model (model-driven) [11]. In conclusion, Enterprise Modelling is the set of activities or processes used to develop the different parts of an enterprise model with a definite objective.

On the other hand, Unified Modeling Language (UML) is a visual language for specifying, constructing and documenting the artifacts of systems. It is a general-purpose modelling language that can be used with all major object and component methods, and that can be applied to all application domains (e.g., health, finance, telecom, aerospace) and implementation platforms (e.g., J2EE, .NET). UML has emerged as the software industry's dominant modelling language. It has been successfully applied to a wide range of domains, ranging from health and finance to aerospace to e-commerce [16]. However, UML has been used mainly so far as a modelling language in order to produce software artifacts. Even though, some works to evaluate UML from point of view of Enterprise Modelling have been carried out by some authors [2, 9].

Moreover, the Profiles package is defined in UML 2.0 as a mechanism that allows metaclasses from existing metamodels to be extended to adapt them for different purposes. This includes the ability to tailor the UML metamodel for different platforms (such as J2EE or .NET) or domains (such as real-time or business process modeling). UML Profiles had been already defined in the previous versions of UML, but their definition has been improved in the UML 2.0, specifying better the relationships allowed among elements of the model and the use of metaclasses of a metamodel inside an UML Profile [12].

3 Problem description

Nowadays, there exist a lot of languages, methodologies and tools related to Enterprise Modelling, even for modelling Virtual or Extended Enterprises [10]. Enterprise Modelling Languages provide constructs to describe and model the people roles, operational processes and functional contents, as well as support information and production and management technologies. There exists great quantity of Enterprise Modelling Languages and they are overlapped. But the integration of the models generated with these languages is complicated, since tools do not exist to integrate models generated with different languages. In this sense, the objective is to achieve a common format, as UEML or POP*, which are valid initiatives in order to enable exchange between different models and the establishment of an environment for reusing existing models [1, 13, 14, 17].

This kind of languages are defined in proprietary formats and they are only implemented by proprietary and expensive tools. Therefore, interoperability problem is intensified in Small and Medium Enterprises (SME), who have limited resources to adapt successfully innovative technologies existing in the market.

So that, SMEs carry out few enterprise models, and moreover the exchange of them among partners is very difficult.

On the other hand, SMEs set up Virtual Enterprises in order to establish flexible collaborations with other partners and to take advantage of new market opportunities. Virtual Enterprise [3] can be define as a temporary network of independent companies, often former competitors, who come together quickly to exploit fast-changing opportunities. The business partners are integrated using information and communication technology. So, interoperability problem at different levels, including enterprise modelling level, can become decisive aspects to reach business success.

Therefore, the main problem at enterprise modelling level for Small and Medium Virtual Enterprises (SMVEs) is focused on the lack of interoperability of existing Enterprise Modelling Languages, and also on the few quantity of enterprise models generated in this kind of enterprises. However, such enterprises use UML to model and generate software artifacts. The idea of this proposal is to provide a methodological approach that can help SMVEs to use successfully UML, not only to generate software models, but also to produce enterprise models that enable them to have a holistic enterprise view and better interoperate with other partners.

4 Research objectives

The IRIS Group of Universitat Jaume I in Castelló (Spain) has been working on several projects related to Virtual Enterprise in different sectors (transport, tile industry, textile, etc.) since 1999 [4–8]. This thesis proposal is motivated inside this framework in order to improve the interoperability of this kind of enterprises at enterprise modelling level.

Therefore, the main research goal is to provide mechanisms to reduce interoperability problems at enterprise modelling level to SMVEs. In this sense, the objective is to investigate the possibilities of UML use for Enterprise Modelling in order to solve this kind of interoperability problems. Besides, the mechanism provided by UML Profiles, redefined in UML 2.0, will be analysed in order to extend and adapt UML for the specific domain of enterprise modelling in SMVEs.

The specific objectives of the research work are the following:

- To perform the state of the art in UML and UML Profiles focused on Enterprise Modelling, and in Virtual Enterprises especially in SME; taking into account the MDA [15] framework defined by OMG and European Projects related to interoperability.
- To obtain a set of requirements for modelling whole enterprise dimensions of SMVEs, in order to define a framework for describing problematic situation.
- To define a methodological approach for enterprise modelling of SMVEs based on UML, which should include the UML Profiles defined in order to extend UML for enterprise modelling, and the guidelines to use this profiles in order to generate interoperable enterprise models.

– To validate the methodological approach defined in a real case of study, applying the methodology to a Tile Virtual Enterprise.

References

1. ATHENA (Advanced Technologies for interoperability of Heterogeneous Enterprise Networks and their Applications) Project (IST-2003-2004). http://www.athena-ip.org, 2005.
2. G. Berio and M. Petit. Enterprise Modelling and the UML: (sometimes) a conflict without a case. In *Proc. of 10th ISPE International Conf. on Concurrent Engineering: Research and applications*, pages 26–30, July 2003.
3. J. Browne and J. Zhang. Extended and virtual enterprises - similarities and differences. *International Journal of Agile Management Systems*, 1/1:30–36, 1999.
4. R. Chalmeta. Virtual Transport Enterprise Integration. *Journal of Integrated Design and Process Science*, 4(4):45–55, December 2000. Publisher IOS Press publishes.
5. R. Chalmeta, C. Campos, and R. Grangel. References architectures for enterprises integration. *Journal of Systems and Software*, July 2001. Publisher Elsevier.
6. R. Chalmeta and R. Grangel. ARDIN extension for virtual enterprise integration. *Journal of Systems and Software*, Setember 2003. Publisher Elsevier.
7. R. Chalmeta and R. Grangel. Virtual Integration of Tile Industry (VITI). In *Conceptual Modeling for Novel Application Domains*, volume 2814 of the serie Lecture Notes in Computer Science. M.A. Jeusfeld and O. Pastor, October 2003.
8. R. Chalmeta and R. Grangel. Performance Measurement Systems for Virtual Enterprise Integration. *Journal Computer Integrate Manufacturing*, January 2005.
9. H.E. Eriksson and M. Penker. B*usiness* M*odeling with* UML*:* B*usiness* P*atterns at* W*ork*. J. Wiley, 2000.
10. EXTERNAL (Extended Enterprise MEthodology) Final version 1-12-d-2002-01-0 (IST-1999-10091). http://research.dnv.com/external/default.htm, 2002.
11. M. S. Fox and M. Gruninger. Enterprise Modeling. *AI Magazine*, 19(3):109–121, 1998.
12. L. Fuentes and A. Vallecillo. Una introducci´n a los perfiles UML. *Nov´tica*, marzo-abril(168):6–11, 2004.
13. IDEAS (Interoperability Development for Enterprise Application and Software) Project. http://www.ideas-roadmap.net, 2005.
14. INTEROP (Interoperability Research for Networked Enterprises Applications and Software) Project. http://interop-noe.org, 2005.
15. Object Management Group, OMG. MDA G*uide* V*ersion 1.0.1*, document number: omg/2003-06-01 edition, 2003.
16. Object Management Group, OMG. U*nified* M*odeling* L*anguage (*UML*)* S*pecification:* I*nfrastructure, version 2.0*, OMG Adopted specification ptc/03-09-15 edition, 2003.
17. UEML (Unified Enterprise Modelling Language) Project (IST-2001-34229). http://www.ueml.org, 2005.
18. F. B. Vernadat. E*nterprise* M*odeling and* I*ntegration:* P*rinciples and* A*pplications*. Chapman and Hall, 1996.

Title:	**A Proposal for Modelling Enterprise Knowledge in Virtual Enterprises**
Authors:	R. Grangel
Conference:	Second International Conference on Interoperability of Enterprise Software and Applications (I-ESA'06) - Doctoral Symposium
Book title:	Interoperability for Enterprise Software and Applications, Panetto, H.; Boudjlida, N. (Eds.)
Publisher:	ISTE Publishing Company
Pages/Year:	359-368/2006
ISBN:	1905209614
Place:	Bordeaux (France)
Date:	20-3-2006

Enterprise Modelling is defined as the art of externalising enterprise knowledge. Many languages, standards and tools have been successfully developed over last few decades to model almost any dimension of an enterprise: process, decision, product, and so forth, and even for modelling Virtual Enterprises. However, some shortcomings of Enterprise Modelling have still not been solved. Some of the most important are related to the interoperability, but also linked to the fact that Enterprise Modelling should be focused on enterprise knowledge, since it provides enterprise models with real value.

In this context, this paper outlines my PhD thesis, which describes the problematic situation that is the origin of this research and the solutions suggested to solve it, as well as the progress made in the research. The aim of the thesis is to investigate the possibilities of using UML 2 and Profiles mechanism in order to provide a framework in which to solve interoperability problems related to Enterprise Modelling and which takes into account the knowledge dimension in the context of Virtual Enterprises, where interoperability problems are greater.

A Proposal for Modelling Enterprise Knowledge in Virtual Enterprises

Reyes Grangel

Grupo de Investigación en Integración y Re-Ingeniería de Sistemas (IRIS), Dept. de Llenguatges i Sistemes Informàtics, Universitat Jaume I, 12071 Castelló, Spain,
grangel@uji.es

Abstract. Enterprise Modelling is defined as the art of externalising enterprise knowledge. Many languages, standards and tools have been successfully developed over last few decades to model almost any dimension of an enterprise: process, decision, product, and so forth, and even for modelling Virtual Enterprises. However, some shortcomings of Enterprise Modelling have still not been solved. Some of the most important are related to the interoperability, but also linked to the fact that Enterprise Modelling should be focused on enterprise knowledge, since it provides enterprise models with real value.

In this context, this paper outlines my PhD thesis, which describes the problematic situation that is the origin of this research and the solutions suggested to solve it, as well as the progress made in the research. The aim of the thesis is to investigate the possibilities of using UML 2 and Profiles mechanism in order to provide a framework in which to solve interoperability problems related to Enterprise Modelling and which takes into account the knowledge dimension in the context of Virtual Enterprises, where interoperability problems are greater.

1 Introduction

Enterprise Modelling can be defined as the art of 'externalising' enterprise knowledge, which adds value to the enterprise or needs to be shared [1]. In this context, many languages, standards and tools have been developed and used in very different domains and with a number of purposes including requirements engineering, the development of information systems, business process re-engineering, and so forth. Such domains include Virtual Enterprises, where Enterprise Modelling can become very useful in order to achieve their objectives.

However, there still exist some weaknesses in this context for Virtual Enterprises. For instance, the problem of interoperability at a horizontal level as well as a vertical level (see Fig. 1), where the main problems are, first, the difficulties involved for exchanging enterprise models among enterprises that use different Enterprise Modelling Languages (EMLs); and, second, the generation of software from these models when different enterprises are involved in this process. Therefore, enterprises, and specially Virtual Enterprises, have troubles for using enterprise models for externalising their enterprise knowledge due to the interoperability problems above described. Some projects attempt to solve these problems. For instance, UEML[1] and POP*[2] provide common exchange formats to make it easy to exchange enterprise models at a horizontal level; and in Model Driven Engineering (MDE), several initiatives have being undertaken, one of the most interesting being Model Driven Architecture (MDA) [5] promoted by OMG.

The thesis project presented in this paper has its origin in this framework and its foremost aim is to provide a proposal for a meta-model that enables enterprises to model enterprise knowledge following the MDA approach. The development of this meta-model will be based on previous works carried out in several European Projects, like INTEROP [3] and ATHENA [4], in which different meta-models, UEML [2, 3] and POP* [4], have been defined in order to solve interoperability problems at a horizontal level. On the one hand, the objective is to adapt and extend both results to the context of knowledge management systems. On the other hand, the meta-model obtained will be integrated into the Reference Architecture ARDIN [6] for the integration of Virtual Enterprise, defined by the IRIS Research Group, with the goal of extending its second dimension to enterprise knowledge modelling.

The objective of this paper, then, is to describe my PhD thesis and its current progress state. This section is intended to give an idea about the research question. The section 2 presents the problematic situation that the thesis intends to solve. In the third, the current knowledge and existing solutions related to these problems are described. Finally, the methodology of work, the main research objectives, and the proposed approach are presented in section 4, section 5 outlines the work carried out so far, together with a discussion on the main contributions provided by the results achieved, and section 6 describes the expected contribution.

2 Description of the Problem

Nowadays, there are many languages, methodologies and tools related to Enterprise Modelling, even for modelling Virtual or Extended Enterprises [7]. But integrating the models generated with these languages is complicated, since no tools exist with which to integrate models generated with different languages (interoperability problem at horizontal level, see Fig. 1) [2, 8, 3, 4].

This kind of languages are defined in proprietary formats and are only implemented by proprietary tools that generally speaking, are only affordable for large enterprises. Therefore, the problem of interoperability is intensified in Small and Medium Enterprises (SME), which have limited resources to successfully adapt innovative technologies existing on the market. Thus, SMEs produce few enterprise models and, moreover, their exchange among partners is very difficult.

[1] Unified Enterprise Modelling Language, developed first by UEML Thematic Network [2], and currently by INTEROP NoE [3].

[2] Acronym of the different enterprise dimensions: Process, Organisation, Product, and so on, represented by a star [4].

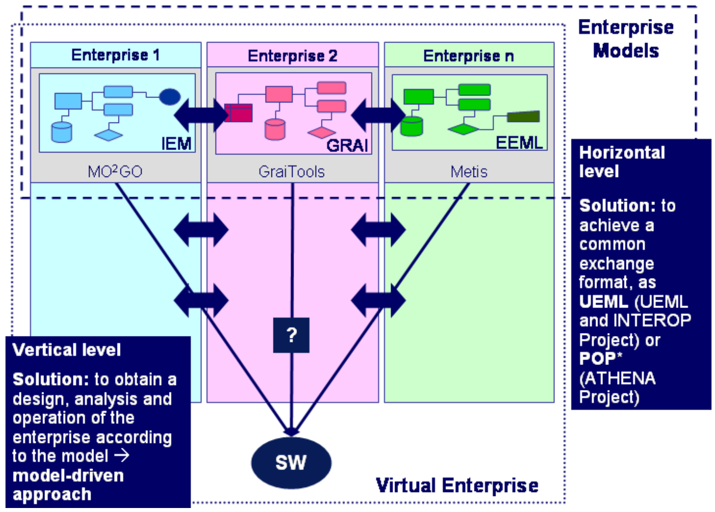


Fig. 1. Interoperability problems and solutions for Virtual Enterprises at the horizontal and vertical level related to Enterprise Modelling

On the other hand, SMEs set up Virtual Enterprises in order to establish flexible collaborations with other partners and to take advantage of new market opportunities. The Virtual Enterprise [9] can be defined as a temporary network of independent companies, often former competitors, which come together quickly to exploit fast-changing opportunities. The business partners are integrated using Information and Communication Technologies. Therefore, the interoperability problems at different levels, including at the Enterprise Modelling level, can become decisive aspects affecting the achievement of business success.

Furthermore, a vertical interoperability problem arises in the Virtual Enterprise's context when its partners intend to use enterprise models to generate software. Since, it is needed to exchange information at different levels (ontological, business, and technological) in order to achieve full interoperability [10, 11] between SMEs that make up the Virtual Enterprise. These inconvenients make hard for Virtual Enterprise[3] to use enterprise models to one of their most valuable purposes, to make explicit enterprise knowledge with the objective of improving performance of enterprise.

[3] The term Virtual Enterprise is used in this paper to concern Virtual Enterprises made up of SMEs.

3 Existing Approaches for Solving Interoperability Problems

Regarding interoperability problem at horizontal level, the objective is to achieve a common format, like UEML or POP*, which are valid initiatives to enabling exchange between different models as well to establish an environment allowing existing models to be reused [2, 8, 3, 4].

On the other hand, different approaches have been proposed to solve the problem of generating software from enterprise models. Such as MDA, which main purpose is to separate the functional specification of a system from the details of its implementation in a specific platform in order to promote the use of models to generate software. Hence, this architecture defines a hierarchy of models from three points of view: Computation Independent Model (CIM), Platform Independent Model (PIM), and Platform Specific Model (PSM) [5].

Different works performed on using **UML for Enterprise Modelling** [12–14] evaluate the possibilities of using UML for modelling enterprises. Consequently, some of them define different types of specific concepts related to business domain, and use extension mechanisms like stereotypes, tagged values, and so forth provided by UML 1.x. However, the new version and specifications developed by the OMG, such as UML 2 and MDA, call for a review of these proposals again, and the works promoted by OMG with Business Enterprise Integration DTF, like Business Semantics of Business Rules (BSBR), Production Rules Representation (PRR), Business Process Definition Metamodel (BPDM), and Organization Structure Metamodel (OSM) that are currently being carried out show this to be the case. In this sense, it is needed to clarify which is the characterisation of the CIM level and, then, to specify which part of CIM models must be transformed into PIM models, since according to [15] there must surely be degrees of CIMness.

Furthermore, the new specification of UML 2 provides profiles with a greater degree of completeness than version 1.5. [16]. Therefore, it will be possible to customise UML in a better way. For instance, UML provides many diagrams for modelling dynamic aspects, but not for direct modelling of business processes in a similar way to how they are represented in an IDEF diagram. Business process modelling with UML is therefore complex [17] and the use of profiles according to UML 2 can make this task easier.

4 Research Objectives and Approach Proposed

This dissertation project is set within two frameworks. The first one, the distinct research projects related to the Virtual Enterprise in different sectors (transport, tile industry, textile, and so forth) [18, 6, 19–21] carried out by the IRIS Research Group at the Universitat Jaume I (Spain). And the second one, the INTEROP NoE [3] in which the IRIS Group is involved and which is focused on interoperability taking into account the following domains: Architecture & Platforms, Enterprise Modelling, and Ontologies. The methodology used for the research has considered the results obtained in these contexts and it has been performed in an iterative and incremental way following the philosophy of the object-oriented methodologies like UP (Unified Process) [22].

The research aims to improve the interoperability of SMEs that promote Virtual Enterprises towards enterprise knowledge modelling. The results obtained will allow enterprise knowledge to be modelled in this kind of enterprises. According to [23] enterprise knowledge can been seen as information made actionable in a way that adds value to the enterprise. Taking into account this definition, enterprise knowledge is defined in this work as the network of connections among data and information that enables people involved in the enterprise to act and to make decisions that add value to the enterprise. Moreover, the meta-model obtained will be integrated into the Reference Architecture ARDIN [6] for the integration of Virtual Enterprises defined by the IRIS Group, with the goal of extending its second dimension to enterprise knowledge modelling.

Therefore, the main research goal is to provide mechanisms that can be used to reduce the interoperability problems related to Enterprise Modelling in a model-driven approach and focused on enterprise knowledge, in the context of Virtual Enterprises. In this regard, the objective is to investigate the possibilities of using UML for Enterprise Modelling in order to solve this kind of interoperability problems. Furthermore, the mechanism provided by UML Profiles, redefined in UML 2, will be analysed in order to extend and adapt UML for the specific domain of enterprise knowledge modelling. The specific objectives of the research work are the following:

- To examine the state of the art in Enterprise Modelling focused on knowledge modelling and UML and UML Profiles focused on Enterprise Modelling, taking into account the MDA [5] framework defined by OMG and European Projects related to interoperability.
- To obtain a set of requirements for modelling the dimensions (process, product, organisation, etc.) of the whole Virtual Enterprise, especially enterprise knowledge, in order to define a framework for describing the problematic situation.
- To define a meta-model based on UML and its extension mechanism, UML Profiles, that allows the knowledge map of a Virtual Enterprise to be represented.
- To define a methodology for enterprise knowledge modelling including the UML Profiles defined, and set out a series of the guidelines of using these profiles in order to generate interoperable enterprise models.
- To validate the methodological approach and UML extension defined in a real case study, by applying the methodology to a Textile Virtual Enterprise.

5 Discussion

The solution suggested here is focused on enterprise knowledge and its modelling at CIM level, i.e. by following the MDA approach and also taking into account previous works on meta-modelling like UEML and POP*. The main contribution

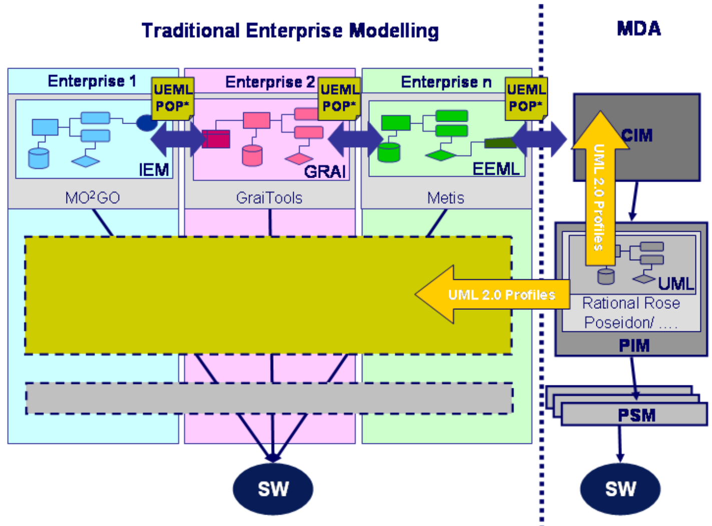


Fig. 2. Research framework: Traditional Enterprise Modelling/MDA

is to combine the following two approaches; traditional Enterprise Modelling [24], such as GRAI [25–27], PERA [28], GERAM [29], IEM [30, 31], EEML [32, 33], and so forth, with the framework defined by OMG with MDA and the new version 2.0 of UML (see Fig. 2). The idea is to take advantage of strengths of the two approaches in order to provide guidelines and mechanisms, which can be apply to SMEs. Moreover, the originality of this work rests on Enterprise Modelling at the CIM level in the representation of knowledge as a new dimension related to existing enterprise dimensions like process, organisation, decision, ans so forth. The main work performed and the results obtained related to this thesis can be summarised in:

- Conclusions on the state of the art in Enterprise Modelling Techniques, Tools and Standards carried out in INTEROP NoE [3], from model-driven point of view [34]. They state the same difficulties in the context of Enterprise Modelling that have been summarised in this paper, focused especially on interoperability problems due to the great number of languages, frameworks, methodologies and tools concerning Enterprise Modelling that exist. Also, many studies are being performed that deal with PIMs, PSMs, UML Profiles, QVT, and so forth in the MDA framework, but the characterisation of CIMs and the features that an enterprise model must satisfy to be considered a CIM and generate appropriate software are still in progress.
- During a research stay at the European Software Institute (Spain) to work on the POP* meta-model within the framework of the ATHENA Project [4], the following work was performed: a comparison among POP*, UEML and other meta-models; participation in the definition of the POP* meta-model;

definition of a UML 2.0 Profiles of POP*; and development of a proof of concept of the POP* meta-model [35]. This work constitute the basis for future work on the development of a meta-model for enterprise knowledge.

– A general methodology [36] obtained as the result of the current IRIS Research Project related to knowledge management. This methodology guides the process of developing and implementing a knowledge management system that allows knowledge to be collected, modelled and applied, while ensuring the quality, security and authenticity of the knowledge provided. The work presented in this paper is concerning with the third phase of this methodology that deals with knowledge representation.
– The definition of the target knowledge [37] useful to establish a common conceptual framework in a Virtual Enterprise, while considering each conceptual block of knowledge (enterprise oriented) proposed in the approach for knowledge management defined by IRIS Group, that is to say, organisation, process, product, and resource. The target knowledge defined has been classified taking into account two points of view, in order to provide a basis that can be used as a reference for further representation of knowledge by Virtual Enterprises that need to model their enterprise knowledge.
– A first proposal for Enterprise Modelling with UML 2 at the CIM level, which takes the model-driven approach into account is presented in Fig. 3 [38, 39]. The proposal describes a profile for Enterprise Modelling, only from the organisational structure point of view. This profile is being improved by including other concepts which are essential for a complete enterprise model, such as process, product, and specially knowledge.

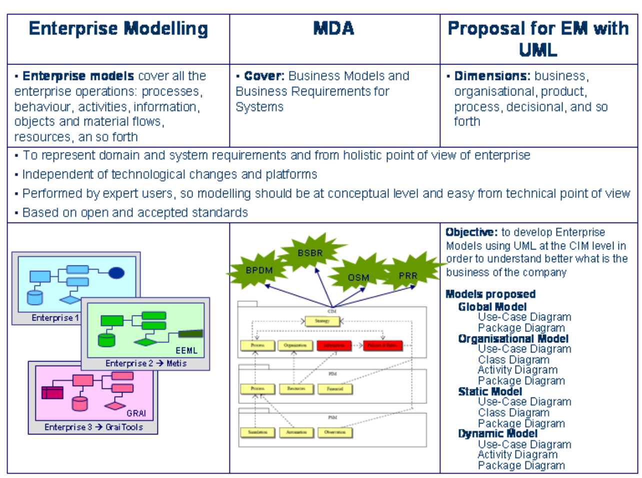


Fig. 3. First proposal for Enterprise Modelling with UML at the CIM level

6 Expected contributions

Nowadays, the main work in progress is, first, related to the customisation of the UEML/POP* meta-models for enterprise knowledge modelling using UML 2 Profiles and refining the proposal above presented, and second, defining the guidelines for using these profiles in order to generate interoperable enterprise models.

In Fig. 4, the current framework proposed to model enterprise knowledge at the CIM level is shown. The framework at the CIM level are divided into three sublevels related to the firsts life-cycle phases defined in GERAM [29], that is to say, 'Global Model' linked to *Identification*, 'Business Models' linked to *Concept*, and 'Business Requirements for Systems' linked to *Requirements*, respectively. Moreover, each model proposed in the framework is being defined at a meta-modelling level in order to provide the UML Profile needed. When this work was finished, the expected contribution will be to provide a practical example applying the defined proposal in a Textile Virtual Enterprise.

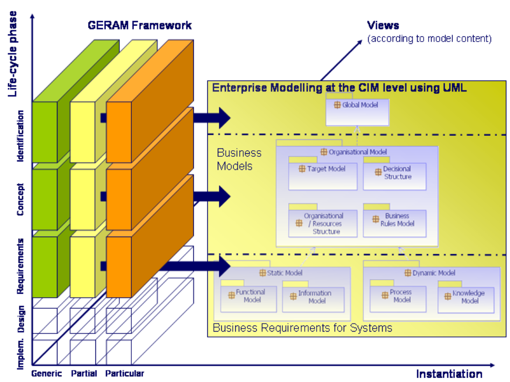


Fig. 4. Relationship between the proposal and the GERAM framework

Acknowledgments

This work was funded by the EC, Interoperability Research for Networked Enterprises Applications and Software (INTEROP NoE) (IST-2003-508011). The authors are indebted to TG2. It was also partially supported by CICYT DPI2003-02515.

References

1. Vernadat, F.B.: Enterprise Modeling and Integration: Myth or reality? In: Proc. 2nd Int. Conf. on Management and Control of Production and Logistics (MCPL2000). (2000) Plenary talk.
2. UEML: Unified Enterprise Modelling Language Themantic Network (IST-2001-34229). http://www.ueml.org (2006)

3. INTEROP: Interoperability Research for Networked Enterprises Applications and Software NoE (IST-2003-508011). http://www.interop-noe.org (2006)
4. ATHENA: Advanced Technologies for interoperability of Heterogeneous Enterprise Networks and their Applications IP (IST-2001- 507849). http://www.athena-ip.org (2006)
5. OMG: MDA Guide Version 1.0.1. Object Management Group. Document number: omg/2003-06-01 edn. (2003)
6. Chalmeta, R., Grangel, R.: ARDIN extension for virtual enterprise integration. The Journal of Systems and Software **67** (2003) 141–152 Elsevier.
7. EXTERNAL: Extended Enterprise MEthodology, Final version 1-12-D-2002-01-0 (IST-1999-10091). http://research.dnv.com/external/default.htm (2002)
8. IDEAS: Interoperability Development for Enterprise Application and Software Project (IST-2001-37368). http://www.cordis.lu/ist/ka2/rmapsmartorg.html (2006)
9. Browne, J., Zhang, J.: Extended and virtual enterprises - similarities and differences. International Journal of Agile Management Systems **1/1** (1999) 30–36
10. Chen, D., Doumeingts, G.: European initiatives to develop interoperability of enterprise applications–basic concepts, framework and roadmap. Annual Reviews in Control **27** (2003) 153–162
11. Ducq, Y., Chen, D., Vallespir, B.: Interoperability in enterprise modelling: requirements and roadmap. Advanced Engineering Informatics **18** (2004) 193–203
12. Marshall, C.: Enterprise Modeling with UML. Designing Successful Software Through Business Analysis. Addison Wesley (2000)
13. Eriksson, H., Penker, M.: Business Modeling with UML: Business Patterns at Work. J. Wiley (2000)
14. Berio, G., Petit, M.: Enterprise Modelling and the UML: (sometimes) a conflict without a case. In: Proc. of 10th ISPE International Conf. on Concurrent Engineering: Research and applications. (2003) 26–30
15. Berrisford, G.: Why IT veterans are sceptical about MDA. In: Second European Workshop on Model Driven Architecture (MDA) with an emphasis on Methodologies and Transformations, Kent, Computing Laboratory, University of Kent (2004) 125–135 http://www.cs.kent.ac.uk/projects/kmf/mdaworkshop/submissions/Berrisford.pdf.
16. Fuentes, L., Vallecillo, A.: Una introducción a los perfiles UML. Novática **marzo-abril** (2004) 6–11
17. Noran, O.: UML vs. IDEF: An Ontology-Oriented Comparative Study in View of Business Modelling. In: ICEIS (3). (2004) 674–682
18. Chalmeta, R., Campos, C., Grangel, R.: References architectures for enterprises integration. The Journal of Systems and Software **57** (2001) 175–191 Elsevier.
19. Chalmeta, R., Grangel, R., Ortiz, A., Poler, R.: Virtual Integration of the Tile Industry (VITI). In Jeusfeld, M., Pastor, O., eds.: Conceptual Modeling for Novel Application Domains. Volume 2814/2003 of Lecture Notes in Computer Science., Springer-Verlag (2003) 65–76
20. Chalmeta, R., Grangel, R., Campos, C., Coltell, O.: An Approach to the Enterprise Integration. In Missikoff, M., ed.: Open INTEROP Workshop on Enterprise Modelling and Ontologies for Interoperability (EMOI-INTEROP 2004) at CAiSE'04. Volume 3. (2004) 253–256
21. Chalmeta, R., Grangel, R.: Performance measurement systems for virtual enterprise integration. International Journal of Computer Integrated Manufacturing **18** (2005) 73–84 Taylor & Francis.

22. Jacobson, I., Booch, G., Rumbaugh, J.: The Unified Software Development Process. Addison-Wesley (1999)
23. Vail, E.F.: Knowledge mapping: getting started with knowledge management. Information Systems Management **Fall** (1999) 16–23
24. Grangel, R., Chalmeta, R., Campos, C., Coltell, O.: Enterprise Modelling, an overview focused on software generation. In Panetto, H., ed.: Interoperability of Enterprise Software and Applications Workshops of the INTEROP-ESA International Conference EI2N, WSI, ISIDI and IEHENA 2005, Hermes Science Publishing (2005)
25. Doumeingts, G., Vallespir, B., Zanittin, M., Chen, D.: GIM-GRAI Integrated Methodology, a Methodology for Designing CIM Systems, Version 1.0. LAP/GRAI, University Bordeaux 1, Bordeaux, France (1992)
26. Doumeingts, G., Vallespir, B., Chen, D.: Decisional modelling GRAI grid. In: International Handbook on Information Systems. Springer-Verlag (1998) 313–337
27. LAP/GRAI: GraiTools. http://www.graisoft.com (2006)
28. Williams, T.: The Purdue Enterprise Reference Architecture. In: Proceedings of the Workshop on Design of Information Infrastructure Systems for Manufacturing, Elsevier (1993)
29. IFIP-IFAC: Generalised Enterprise Reference Architecture and Methodology (GERAM). Technical Report Version 1.6.3 (1999) http://www.cit.gu.edu.au/ bernus/taskforce/geram/versions.
30. IEM: Business process oriented knowledge management. In Mertins, K., Heisig, P., Vorbeck, J., eds.: Knowledge Management. Concepts and Best Practices, Springer-Verlag (2003)
31. IPK: MO^2GO. http://www.ipk.fhg.de (2006)
32. Computas: METIS. http://www.computas.com (2005)
33. Lillehagen, F.M., Dehli, E., Fjeld, L., Krogstie, J., Jørgensen, H.D.: Utilizing Active Knowledge Models in an Infrastructure for Virtual Enterprises. In: PRO-VE. (2002) 353–360
34. Grangel, R., Chalmeta, R.: A Methodological Approach for Enterprise Modelling of Small and Medium Virtual Enterprises based on UML. Application to a Tile Virtual Enterprise. In: Doctoral Symposium at First International Conference on Interoperability of Enterprise Software and Applications (INTEROP-ESA'2005). (2005)
35. Grangel, R., Chalmeta, R., Schuster, S., Penya, I.: Exchange of Business Process Models using the POP* Meta-model. In al., C.B.., ed.: Business Process Management - BPM 2005 Workshops. Volume 3812/2005 of Lecture Notes in Computer Science., Springer-Verlag (2005)
36. Chalmeta, R., Grangel, R., Fernández, V.: Methodology for the implementation of knowledge management systems. IEEE Transactions on Software Engineering (2006) in evaluation.
37. Grangel, R., Chalmeta, R.: Defining of Target Knowledge in Virtual Enterprise. In: evaluation 9th BIS 2006. (2006)
38. Grangel, R., Bourey, J.P., Chalmeta, R., Bigand, M.: UML for Enterprise Modelling: a Model-Driven Approach. In: Interoperability for Enterprise Software and Applications Conference (I-ESA'06). (2006)
39. Grangel, R., Bourey, J.P., Berre, A.: Solving Problems in the Parametrisation of ERPs using a Model-Driven Approach. In: Interoperability for Enterprise Software and Applications Conference (I-ESA'06). (2006)

Chapter 2

State of the Art: Knowledge Enterprise Modelling

Title:	**Enterprise Modelling, an overview focused on software generation**
Authors:	R. Grangel and R. Chalmeta and C. Campos and Ò. Coltell
Conference:	First International Conference on Interoperability of Enterprise Software and Applications (INTEROP-ESA'05) - International Workshop on Enterprise Integration, Interoperability and Networking (EI2N'2005)
Book title:	Interoperability of Enterprise Software and Applications, H. Panetto (Ed.)
Publisher:	HERMES Science Publishing London Ltd.
Pages/Year:	65-76/2005
ISBN:	1 905209 49 5
Place:	Geneva (Switzerland)
Date:	22-2-2005

Abstract

Nowadays, enterprises face to an economic environment controlled by global competitiveness. In this context, enterprises must be more agile and be able to interoperate with their partners, in order to align their objectives with the market needs. Enterprise Modelling can become a way that enables enterprises to know and understand much better their business to achieve these objectives. However, some aspects as the great quantity of existing Enterprise Modelling Languages or the weak connection between Enterprise Modelling and software generation do not make easy this task. In this paper, we present an overview of the current state of the art in Enterprise Modelling Languages from software generation point of view. The main objective is to analyse the existing Enterprise Modelling Languages in order to establish how they can be useful to generate software.

Enterprise Modelling, an overview focused on software generation

Reyes Grangel — Ricardo Chalmeta — Cristina Campos — Òscar Coltell

Grupo de investigación en Integración y Re-Ingeniería de Sistemas (IRIS)
Dep. de Llenguatges i Sistemes Informàtics
Universitat Jaume I
12071 Castelló, SPAIN

grangel@uji.es
rchalmet@uji.es
camposc@uji.es
coltell@uji.es

ABSTRACT. Nowadays, enterprises face to an economic environment controlled by global competitiveness. In this context, enterprises must be more agile and be able to interoperate with their partners, in order to align their objectives with the market needs. Enterprise Modelling can become a way that enables enterprises to know and understand much better their business to achieve these objectives. However, some aspects as the great quantity of existing Enterprise Modelling Languages or the weak connection between Enterprise Modelling and software generation do not make easy this task. In this paper, we present an overview of the current state of the art in Enterprise Modelling Languages from software generation point of view. The main objective is to analyse the existing Enterprise Modelling Languages in order to establish how they can be useful to generate software.

KEYWORDS: Enterprise Modelling, Enterprise Modelling Languages, Overview, Software Generation, Model Driven Architecture (MDA), Computation Independent Model (CIM).

1. Enterprise Modelling

In the 70s the first concepts of modelling were applied to the computer systems (E/R Model, DFD, etc.), but the concept of Enterprise Modelling appears in the USA at the beginning of the 80s, with the initiative Computer Integrated Manufacturing (CIM). Examples of this initiative are the projects Integrated Computer Aided Manufacturing (ICAM) carried out by the US Air Force or the Integrated Computer Aided Manufacturing-International (CAM-I). In the middle of the 80s, different Enterprise Modelling Languages emerge in Europe like for instance GRAI or CIMOSA. Numerous commercial tools appear in the 90s for giving support to a great number of different modelling languages (ARIS ToolSet, FirstSTEP, METIS, KBSI Tools, CimTool, MO2GO, e-MAGIM, etc.).

Enterprise Modelling is defined in [VER 96] as the art of 'externalizing' enterprise knowledge, which adds value to the enterprise or needs to be shared, i.e., representing the enterprise in terms of its organisation and operations (processes, behaviour, activities, information, objects and material flows, resources and organisation units, and system infrastructure and architectures). This art consists of obtaining enterprise models, whose role should be to obtain a design, analysis and operation of the enterprise according to the model, i.e., driven by the model (model-driven) [FOX 98].

Therefore, Enterprise Modelling can be used to select and develop computer systems, to better understand and improve business processes, etc., but the most important benefit of enterprise models is the capacity to add value to enterprise [VER 96]. Since, such models are able to make explicit facts and knowledge, which can be shared for users and different enterprise applications in order to improve enterprise performance [EXT02].

2. Enterprise Modelling Languages

An Enterprise Modelling Language (EML) is a language with an accurate sintaxis and semantics, which can be interpreted and managed by a computer [FUE 04], and it can generate graphical models that represent several dimensions of an enterprise. EMLs should allow building the model of an enterprise according to various points of view such as: function, organisation, process decision, economic, etc. in an integrated way.

Moreover, EMLs define the generic modelling constructs for Enterprise Modelling adapted to the needs of people creating and using enterprise models, according to the definition provide by GERAM [GER99]. In particular EMLs will provide construct to describe and model human roles, operational processes and their functional contents as well as the supporting information, office and production technologies.

Enterprise models are normally composed of submodels such as organisational models, process models, data models, configuration models, etc. The purpose of these models is to provide a common understanding among users about enterprise operations and structure, and decision-making support. In this context, the basis of the standards in Enterprise Modelling should be to achieve the following requirements [BER 99]:

– **To enable three fundamental types of flow inside and among enterprises:** material, information and decision or control.

– **To enable four modelling views:** functional, informational, resources and organisational.

– **To enable three levels of modelling:** definition of requirements, specification of design and implementation description.

Many modelling methods and techniques have been established since 90s, besides there are a great number of initiatives and groups of standardization in Enterprise Modelling [KAL 02]. The greatest part of the standards related to Enterprise Modelling have been developed for the CEN TC310/WG1 (European Standardisation Committee) and ISO TC184/SC5/WG1. They are needed for enterprise integration and interoperability, but they have had really little or null industrial impact.

Next, we show a brief summary of existing EMLs in order to provide a general perspective of existing EMLs. Then, the main weaknesses in the context of EMLs are presented.

2.1. *Overview of existing EMLs*

Nowadays, there exists a great quantity of EMLs and they are widely detailed in several states of the art in Enterprise Modelling carried out in the framework of European Projects, as IDEAS [IDE05], UEML [UEM05], ATHENA [ATH05], and INTEROP [INT05]. Next, we present an overview of EMLs raise in these projects (see tables 1, 2, 3). These tables show the Enterprise Modelling Tools (EMT) associated with these EMLs, the Enterprise Modelling Methodologies (EMM), approaches or standards supported by them, their owner enterprise and their website.

The table 1 shows the EMLs, which enable to represent the three fundamental types of flow among enterprises, the four modelling views and the three levels of modelling above mentioned. This kind of languages could be called traditional EMLs. Next, a brief description of them is shown:

1) **ARIS (ARchitecture of Integrated information Systems)** conceptual design is based on an integration concept which is derived from a holistic analysis of business processes. The result is a highly complex model which is divided into individual views in order to reduce its complexity: data view, function view, organization view and control view.

2) **CIMOSA (CIM Open System Architecture)** is an architecture for enterprise integration consisting of a framework for Enterprise Modelling (reference architecture), an EML and an integrating infrastructure for model enactment all to be supported by a standards based on common terminology.

EML	**EMT**	**EMM**	**Owner**	**www**
ARIS Language	ARIS Process Platform	ARIS UML 1.4	IDS Scheer AG	www.ids-scheer.com
CIMOSA			CIMOSA Association e.V.	www.cimosa.de
	First step designer			www.interfacing.com
	CimTool			www.rgcp.com
GRAI	GraiTools	GIM	LAP/GRAI	www.graisoft.com
IDEF	IDEF Tools	IDEF Method.	KBS	www.kbsi.com
	Business Modelling Workbench			www.idefine.com
	System Architect			www.popkin.com
IEM	MO2GO	IPK Procedure	IPK	www.ipk.fhg.de
ITM BPM UML	Metis	Zachman Framework TOGAF 8 UML 2.0 DoDAF (C4ISR) TEAF/FEAF	Computas AS	www.computas.com
MEML	Metis	EEDO Method.	Computas AS	www.computas.com
Petri Nets	www.daimi.au.dk/PetriNets/tools/		Petri Nets Steering Committee	

Table 1. *Overview of the traditional EMLs (I)*

3) **GRAI** is the set of twelve Methodological Modules. These modules cover the following areas: Re-Engineering and elaboration of target enterprise, Audit, Choice of Information Technology (IT) solutions, Implementation of IT solutions, Performance Indicators, Benchmarking, Business Plan, Relationships between GRAI TM Methodology and quality approach, Management of design department, Management of enterprise evolution, Knowledge management.

4) **IDEF (Integrated DEFinition methodology)** methods are used to create graphical representations of various systems, analyse the model, create a model of a desired version of the system, and to aid in the transition from one to the other. Depending on the IDEF method used, different syntaxes exist to represent the models. The most representative construct of IDEF methodology is the generic IDEF0 diagram (a meta-model). IDEF0 allows the user to depict a view of the process including the inputs, outputs, controls and mechanisms (which are referred to generally as ICOMs).

EML	Owner	www
BPML	BPMI	www.bpmi.org
PIF	PIF Working Group	ccs.mit.edu/pifwg.html
PSL	NIST	www.mel.nist.gov/psl
UEML	UEML European Project	www.ueml.org
XPDL	WfMC	www.wfmc.org

Table 2. *Overview of the main EMLs created to make easy exchange (II)*

5) **IEM (Integrated Enterprise Modelling)** allows different views (information model, process chain, etc.) on one consistent model, in which an enterprise is described by objects, its relations and its behaviour. The generic object classes that can be used are 'product', 'order' and 'resource'. An additional element is the action and a class structure can be defined for it. In the process chain the action connects the input, output states, the controlling order and the necessary resources to perform the process. The modelling of bill of materials and part of relations is also supported.

6) **ITM** is used to implement the four leading EA methodologies. The ITM template also has expressiveness to start modelling of most other enterprise needs, such as project models, business and impact analysis models. **BPM** is a new template aimed at the BPM market and implements most of the BPMN constructs plus integrates it with IDEF and other process modelling language yielding the expressiveness required in practical situations. **UML** implements nine of the diagrams described by OMG as part of the UML version 2.0 specifications, but all has expressiveness to start modelling of most other enterprise needs, such as project models, business and impact analysis models.

7) **MEML** (EEML from EXTERNAL and MEML 1.0, UEML compliant) is made to support process and enterprise modelling across a number of layers. The four layers of interest are: Generic Task Type, Specific Task Type, Manage Task Instances, Perform Task Instances. The modelling language currently includes four modelling domains, in addition to general modelling mechanisms and primitives provided in METIS, like swimlane-diagrams: Process modelling, Resource modelling, Goal modelling, Data modelling (currently implemented with UML Class Diagram).

8) **Petri nets** were initially developed by CA Petri for the specification of concurrent (parallel) systems. The recognised benefits in the context of Enterprise Modelling of Petri Nets are modelling power (resource sharing, conflicts, mutual exclusion, concurrency, non-determinism, visual modelling); analysis (deadlock detection, bottleneck analysis, animation, simulation); and code generation for Controlling Manufacturing Systems.

The table 2 shows languages than could be consider like EMLs, but they have created in order to make easy different kinds of interchanges. Next, a brief description of them is shown:

1) **BPML (Business Process Modelling Language)** is a meta-language for the modelling of business processes, just as XML is a meta-language for the modelling of business data. BPML provides an abstracted execution model for collaborative transactional business processes based on the concept of a transactional finite-state machine.

2) **PIF (Process Interchange Format)**, a PIF process description consists of a file of objects, such as ACTIVITY, ACTOR, and RESOURCE objects. Each object in the file has a unique id that other objects can use to refer to it. Each object type (or class) has a particular set of attributes defined for it; each attribute describes some aspect of the object.

3) **PSL (Process Specification Language)**, the goal of PSL is to create a process interchange language that is common to all manufacturing applications, generic enough to be decoupled from any given application, and robust enough to be able to represent the necessary process information for any given application. This representation would facilitate communication among the various applications because they would all have a common understanding of concepts to be shared.

4) **UEML (Unified Enterprise Modelling Language)**, the main objective of the UEML is to provide industry with a unified and expandable enterprise modelling language. The concept of UEML was born in 1997 in the frame of ICEIMT (Torino conference) organised in cooperation with NIST.

5) **XPDL (XML Process Definition Language)**, the WfMC has identified five functional interfaces to a workflow service as part of its standardization program. This interface includes a common metamodel for describing the process definition (this specification) and also an XML schema for the interchange of process definitions.

Languages showed in the table 3 are based on standards as XML or UML, and they can be used like EMLs. Next, a brief description of them is shown:

1) **BPDM**, this meta-model provides a common language, for describing business processes in an implementation independent manner. This is not to say that the models are abstract from execution details, on the contrary it is our aim to describe executable processes, these models are intended to be abstract from the detailed implementation (platform) mechanisms. The standardization is still in progress.

2) **ebXML (Electronic Business using eXtensible Markup Language)** is a modular suite of specifications that enables enterprises of any size and in any geographical location to conduct business over the Internet. Using ebXML, companies now have a standard method to exchange business messages, conduct trading relationships, communicate data in common terms and define and register business processes.

3) **UML Profile for EAI (Enterprise Application Integration)** intends to solve the EAI problem by defining and publishing a metadata interchange standard for information about accessing application interfaces. The goal is to simplify application integration by standardizing application metadata for invoking and translating application information.

EML	EMM	Owner	www
BPDM		OMG	www.omg.org
ebXML	XML	OASIS	www.ebxml.org
UML Profile for EAI	UML 1.4	OMG	www.omg.org
UML Profile for EDOC	UML 1.4	OMG	www.omg.org

Table 3. *Overview of the main EMLs based on XML and UML (III)*

4) **UML Profile for EDOC (Enterprise Distributed Object Computing)** provides a modelling framework for describing how objects are used to implement enterprise systems. It is based on UML 1.4 and conforms to the OMG Model Driven Architecture.

2.2. *Problems related to EMLs*

Conclusions about EMLs pointed out in the states of the art of mentioned European Projects [IDE05, UEM05, ATH05, INT05] are:

– There exist a great quantity of EMLs and they are overlapped.

– EMLs provide constructs to describe and model the people roles, operational processes and functional contents, as well as support information and production and management technologies.

– The integration of the models generated with these languages is complicated, since tools do not exist to integrate models generated with different languages.

Another European Project, EXTERNAL, provides the main weaknesses related to EMLs, as the following ones [EXT02]:

– **Support to enterprises in dynamic environments:** especially for dynamic roles, cooperation in time and supporting of specific processes. Permanent changes in this kind of enterprises require a controlled way for managing the maturity of structures and processes. Nowadays, Enterprise Modelling Methodologies are not able of dealing with different levels of maturity. Besides, the EMLs are weak in easy and transparent externalization of dynamic roles and policies in extended enterprises.

– **Maintenance of enterprise models:** enterprise models are not updated after its first implementation, which reduces the value for improving the performance of the business processes.

– **Link with software generation:** Enterprise Modelling has the objective to support software implementation. However, few and isolated solutions exist that can link the conceptual level of Enterprise Modelling with the implementation level.

Therefore, the main problems that concern to EMLs can be located on two axis:

– **Horizontal:** the lack of interoperability between EMLs and their corresponding Enterprise Modelling Tools. Almost all kinds of these languages are proprietary specifications and can only be implemented with specific tools designed for this purpose. This problem complicates the interoperability of enterprises at conceptual level. The main solutions provided by the research community to address this problem are focused on defining a common exchange language that can become a standard among the existing EMLs. This is for instance the goal of the UEML Project [UEM05] and one of the objectives of the ATHENA Project [ATH05].

– **Vertical:** the weak connection between enterprise models and the generation of software is one of the major reasons why enterprises only develop few models, which moreover are rarely updated and therefore are not very successful in accomplishing their initial purpose. Initiatives, such as MDA [MDA03] promoted by OMG intend to solve this kind of problems.

3. MDA framework

One of the weaknesses of Enterprise Modelling is the difficulty to software generation from enterprise models. In this section, MDA (Model Driven Architecture) of OMG [MDA03] is described as a reference framework.

MDA was proposed by the OMG (Object Management Group) in 2001 as an architecture for software applications development. This initiative intends to promote the use of models as fundamental way for designing and implementing systems. One of the main objectives of MDA is to separate the operation specification of a system from the details of implementation in a specific platform; so that the computer systems and enterprise can be able to evolve with fast technological changes. In this context, MDA establishes a framework for:

– Specifying a system independently of the platform that supports it.

– Specifying platforms.

– Choosing a particular platform for the system.

– Transforming the system specification into one for a particular platform.

3.1. *Benefits of MDA*

MDA is focused on functionality and behaviour of systems independently of the technology in which they will be implemented. The main advantage of MDA is that it is not necessary to repeat the modelling process of behaviour or functionality of an application or system each time that a new technology appears. Other architectures are connected with a specific technology, with MDA operation and behaviour is modelled only once. The three most important benefits of the use of MDA for enterprises are:

– An architecture based on MDA is always prepared to respond to the future needs.

– MDA makes easy to integrate applications through the boundaries of the middleware.

– The specific facilities of domain in MDA defined by the OMG's Domain Task Force will provide an extensive interoperability, being available on a particular platform or in multiple platforms when necessary.

3.2. *Components of MDA*

A system in MDA can include among others: a program, a single computer system, some combinations of parts of different systems, a federation of systems, people, an enterprise, a federation of enterprises, etc. And a system model is defined as a description or specification of that system and its environment for certain purpose. A model shows often a combination of graphics and text. The text can be in an EML or in natural language.

MDA is focused on the use of models for system development. Therefore, MDA encourages the use of certain classes of models and the relationship among themselves. A system can be observed and analysed from different points of view; MDA specifies three points of view: an independent point of view of the computation, an independent point of view of the platform and a dependent point of view of the platform. In this way, MDA defines three conceptual levels:

– **Computation Independent Model (CIM):** to represent domain and system requirements in the environment in which it is going to operate, concerning business models and a holistic point of view about enterprises.

– **Platform Independent Model (PIM):** to model system functionality but without define how and in which platform will be implemented, centred in information and from a computational point of view.

– **Platform Specific Model (PSM):** the PIM is transformed in a platform dependent model according to selected platform, focused on technological point of view.

CIM specifies the requirements, and the PIM and PSM specify the system design and implementation. The PIM and PSM must not violate the CIM [HEN 03]. The most interesting idea of this approach is the possibility of model transformation by means of tools that automate the transformation process until code generation.

4. CIM characterisation

A CIM describes the domain and requirements of the system in a model that is independent of computation representations and is expressed in the vocabulary of the domain practitioner. The CIM corresponds to the conceptualization perspective and might consist of a model from the informational viewpoint, which captures information about the data of a system.

CIM is an emerging model, not yet formally defined or supported by OMG standards and tools. Using a CIM, an enterprise can capture, manage, and better use some of its most valuable assets: knowledge of its resources, policies, rules, terminology, and processes. Also, enterprises can specify, in an EML, the requirements of their systems and validate that the system design satisfies these requirements. CIM is made up two main subdivisions, which analyse enterprises and their environment from different point of views [HEN 03]:

– **Business Model:** focused on the scope and goals of the business, and the terminology, resources, facts, roles, policies, rules, processes, organizations, locations, and events of concern to the business.

– **Business Requirements for Systems:** based on the purpose, scope, and policies for the system. Business Requirements can be divided into Functional Requirements, Interaction Requirements, and Environment Contract.

In [BER 04], two kinds of CIM are proposed. The first one is a model of a business enterprise, a stand-alone CIM, independent of data processing and of potential software systems. A purely conceptual or domain model of this kind is interesting per se. It can be used to define some business rules. But forward engineering transformation is problematic.

The second one is definitively related to one or more data processing systems. It can be transformed into software systems that consume input data and produce output data. Such a CIM may be thought of as a very abstract PIM. And given there are degrees of PIMness, there must surely be degrees of CIMness.

Some authors [BER 04] do not envisage forward engineering from a purely conceptual CIM. However, they are more optimists about forward engineering from a CIM that abstracts from data processing systems. This kind of CIM can be recognised because it will:

– Acknowledge the divisions between data in discrete loosely-coupled data stores.

– Define what units of work clients invoke or require on each distinct data store, with the preconditions and post conditions of each unit of work.

– Define what data must persist in each discrete data store for those units of work to be completable.

Finally, a standard model framework is required to support the pragmatic association of CIMs with PIMs and PSMs, in order to specify the separation of concerns between different models that make up a complete specification. It will be helpful for MDA to establish normative mappings between other popular frameworks and the standard framework, to promote reuse of models by projects that use different frameworks [HEN 03].

5. Conclusion

Enterprise Modelling must become for enterprises a way for better understanding business, not a final goal. One of the main weaknesses of Enterprise Modelling is the lack of strong links between enterprise models and software generation. A lot of Enterprise Modelling Languages, Standards and Tools exist, but enterprises carried out few enterprise models and it is very hard to maintain them, to use them in order to generate software, or to exchange them among different enterprises.

The main conclusions about state of the art in Enterprise Modelling Techniques, Tools and Standards, in order to understand how it can be useful to software generation from enterprise models, are:

– There is a great number of Languages, Standards, Frameworks, Methodologies and Tools concerning Enterprise Modelling, which cover different parts of the dimensions defined in GERAM and even they are overlapped.

– Enterprise Modelling Tools usually support a particular Enterprise Modelling Language and Methodology; and only a few ones allow the definition of a new language or some adaptation of the languages that implement. Moreover, there not exist tools that can integrate their models with models carried out with other Enterprise Modelling Tools. Therefore, mechanisms for the exchange of enterprise models among enterprise do not exist.

– Enterprise Modelling Standards are necessary for enterprise integration and interoperability, and there are a lot of them concerning Enterprise Modelling. But they have had really little or null industrial impact due to they are associated with a specific platform or technology; which can not be achieved by a great number of enterprises.

– Many works are being performed related to PIMs, PSMs, UML Profiles, QVT, etc., in the MDA framework, but the characterisation of CIMs and the features that a enterprise model must satisfy to be consider CIM and generate appropriate software are still in progress.

Acknowledgements

This work has being founded by CICYT DPI2003-02515. Also, it is partially supported by the European Commission within the 6th Framework Programme (INTEROP Network of Excellence, (IST-2003-508011), www.interop-noe.org). The authors are indebted to WP7.

6. References

[ATH05] "ATHENA (Advanced Technologies for interoperability of Heterogeneous Enterprise Networks and their Applications) Project (IST-2003-2004)", http://www.athena-ip.org, 2005.

[BER 99] BERIO G., VERNADAT F. B., “New developments in enterprise modelling using CIMOSA”, *Computers in Industry*, vol. 40, num. 2-3, 1999, p. 99-114.

[BER 04] BERRISFORD G., “Why IT veterans are sceptical about MDA”, *Second European Workshop on Model Driven Architecture (MDA) with an emphasis on Methodologies and Transformations*, Kent, 2004, Computing Laboratory, University of Kent, p. 125-135, http://www.cs.kent.ac.uk/projects/kmf/mdaworkshop/submissions/Berrisford.pdf.

[EXT02] “EXTERNAL (Extended Enterprise MEthodology) Final version 1-12-D-2002-01-0 (IST-1999-10091)”, http://research.dnv.com/external/default.htm, 2002.

[FOX 98] FOX M. S., GRUNINGER M., “Enterprise Modelling”, *AI Magazine*, vol. 19, num. 3, 1998, p. 109-121.

[FUE 04] FUENTES L., VALLECILLO A., “Una introducción a los perfiles UML”, *Novática*, vol. marzo-abril, num. 168, 2004, p. 6-11.

[GER99] “GERAM: Generalised enterprise reference architecture and methodology. Technical Report Version 1.6.3”, 1999, http://www.cit.gu.edu.au/ bernus/taskforce/geram/versions.

[HEN 03] HENDRYX S., “Integrating Computation Independent Business Modeling Languages into the MDA with UML 2”, http://www.omg.org/docs/ad/03-01-32.doc, 2003.

[IDE05] “IDEAS (Interoperability Development for Enterprise Application and Software) Project”, http://www.ideas-roadmap.net, 2005.

[INT05] “INTEROP (Interoperability Research for Networked Enterprises Applications and Software) Project”, http://interop-noe.org, 2005.

[KAL 02] KALPIC B., BERNUS P., “Business process modelling in industry - the powerful tool in enterprise management”, *Computers in Industry*, vol. 47, num. 3, 2002, p. 299-318.

[MDA03] “MDA Guide Version 1.0.1”, 2003.

[UEM05] “UEML (Unified Enterprise Modelling Language) Project (IST-2001-34229)”, http://www.ueml.org, 2005.

[VER 96] VERNADAT F. B., *Enterprise Modeling and Integration: Principles and Applications*, Chapman and Hall, London, 1996.

Chapter 3

State of the Art: UML as Enterprise Modelling Language

Title:	**UML for Enterprise Modelling: basis for a Model-Driven Approach**
Authors:	R. Grangel and J-P. Bourey and R. Chalmeta and M. Bigand
Conference:	Second International Conference on Interoperability of Enterprise Software and Applications (I-ESA'06)
Book title:	Enterprise Interoperability. New Challenges and Approaches, Doumeingts, G.; Müller, J.; Morel, G.; Vallespir, B. (Eds.)
Publisher:	Springer Verlag Berlin (USA)
Pages/Year:	91-102/2007
ISBN:	978-1-84628-713-8
Place:	Bordeaux (France)
Date:	22-3-2006 to 24-3-2006

Abstract

The Unified Modeling Language (UML) has become a standard visual language for object-oriented modelling that has been used successfully for modelling information systems in very different domains. However, UML is a general-purpose modelling language, which can also be useful for modelling other systems such as, for example, an enterprise. In spite of the distinct works carried out in this area, the OMG's new proposals at the Computation Independent Model (CIM) level call to mind that more practical examples, from the model-driven point of view, are needed to better

understand how it can be applied to model all enterprise dimensions.

In this paper, we present a proposal for Enterprise Modelling with UML 2 at the CIM level, taking into account the model-driven approach, and through some examples, we describe how it can be applied in a real Case Study. In this proposal, we show how UML 2 can be used to provide a holistic vision of an enterprise that considers all its dimensions, that is to say, organisational, process, decisional, ans so forth.

UML for Enterprise Modelling: basis for a Model-Driven Approach

Reyes Grangel[1], Jean-Pierre Bourey[2], Ricardo Chalmeta[1], and Michel Bigand[2]

[1] Grupo de Investigación en Integración y Re-Ingeniería de Sistemas (IRIS), Dept. de Llenguatges i Sistemes Informàtics, Universitat Jaume I, 12071 Castelló, Spain
{grangel, rchalmet}@uji.es
[2] Equipe de Recherche en Génie Industriel, Ecole Centrale de Lille, BP 48, 59561 Villeneuve d'Ascq Cedex, France
{Jean-Pierre.Bourey, Michel.Bigand}@ec-lille.fr

Abstract. The Unified Modeling Language (UML) has become a standard visual language for object-oriented modelling that has been used successfully for modelling information systems in very different domains. However, UML is a general-purpose modelling language, which can also be useful for modelling other systems such as, for example, an enterprise. In spite of the distinct works carried out in this area, the OMG's new proposals at the Computation Independent Model (CIM) level call to mind that more practical examples, from the model-driven point of view, are needed to better understand how it can be applied to model all enterprise dimensions.

In this paper, we present a proposal for Enterprise Modelling with UML 2 at the CIM level, taking into account the model-driven approach, and through some examples, we describe how it can be applied in a real Case Study. In this proposal, we show how UML 2 can be used to provide a holistic vision of an enterprise that considers all its dimensions, that is to say, organisational, process, decisional, ans so forth.

1 Introduction

Enterprise Modelling is externalising and expressing enterprise knowledge [1], which provides a holistic view of an enterprise and considers all its dimensions: process, decision, information, and so forth [2]. Enterprise Modelling has been used for a long time to select and develop computer systems, to better understand and improve business processes, to support decision-making, and so forth, but the most important benefit of enterprise models is their capacity to add value to the enterprise. This is due to the fact that, such models are able to make explicit facts and knowledge which can be shared by users and different enterprise applications in order to improve enterprise performance [1, 3, 4].

Many languages, standards, methodologies and tools for Enterprise Modelling have emerged, since the 70s when the first concepts of modelling were applied to computer systems (E/R Model, DFD, and so forth) so far when modelling concepts and techniques are applied not only to information systems but to the whole enterprise [3]. Nowadays, there exist a great number of languages, standards, and so forth, which cover different enterprise dimensions defined in GERAM [5] and they even overlap. Therefore, interoperability problems are increasing among systems that use different Enterprise Modelling Languages [6]. Moreover, one of the main weaknesses of Enterprise Modelling is the lack of strong links between enterprise models and software generation. For these reasons, some enterprises, especially SMEs, implement few enterprise models and it is very hard to maintain them, to use them to generate software, or to exchange them among different enterprises [7].

One solution, as pointed out in [8], is that the role of enterprise models should be to obtain a design, analysis and operation of the enterprise according to models, i.e., it should be driven by models (model-driven). Nowadays, the model-driven approach is followed by numerous projects like the MODELWARE [9] and INTEROP [10] in the European Union, and Model Driven Architecture (MDA) [11], which is carried out by the OMG.

MDA, for instance, intends to promote the use of models as fundamental way of designing and implementing different kinds of systems. The main purpose of this approach is to separate the functional specification of a system from the details of its implementation on a specific platform. This architecture therefore defines a hierarchy of models from three points of view [11]:

- **Computation Independent Model (CIM):** used to represent domain and system requirements in the environment in which it is going to operate, concerning business models and from a holistic point of view of the enterprise, and independent of the computation.
- **Platform Independent Model (PIM):** used to model system functionality but without defining how and on which platform it will be implemented; it is focused on information and from a computational point of view.
- **Platform Specific Model (PSM):** the PIM is transformed into a platform dependent model according to platform selected for use, and is focused on a technological point of view.

MDA is an emergent paradigm. A lot of work is being carried out within OMG framework related to PIMs, PSMs, QVT, and so forth, but the characterisation of CIMs and the features that an enterprise model must satisfy to be considered CIM and generate appropriate software are still in progress [7]. In this paper, we present an MDA-oriented proposal for modelling enterprises with UML 2 at the CIM level that allows software to be generated from enterprise models in the future, and which also takes into account the works conducted in different traditional Enterprise Modelling Languages [7] like GRAI [12], IEM [13], MEML [4], IDEF [14], and so forth.

The paper is organised as follows. Section 2 shows a review of several works related to the use of UML for Enterprise Modelling. In section 3, the proposal for Enterprise Modelling using UML 2 at the CIM level from the model-driven point of view is presented. Section 4 describes some examples of diagrams performed on a real Case Study applying the proposal explained in the previous section and, finally, section 5 outlines the main conclusions.

2 Existing Proposals for Enterprise Modelling with UML

The Unified Modeling Language (UML) is a graphical language for visualising, specifying, constructing and documenting the artifacts of a software-intensive system [15]. It is a general-purpose modelling language that has been used in different domains, even to model enterprises. In version 1.5 [15], the UML Specification includes an example of **'UML Profile for Business Modelling'** in order to show how UML could be customised to model enterprises. In spite of the fact that all UML concepts can be applied to Enterprise Modelling, the profile includes a number of stereotypes, constraints and tagged values to emphasise several concepts which are specific to the business domain [15]. However, the profile is only an example and other research works have been published on this subject that offer more robustness to Enterprise Modelling.

In [16] Enterprise Modelling is considered as the development of dynamic models that help enterprises to communicate concepts related to business with their stakeholders. These conceptual models make it easy for people to understand the complexity of enterprises in the new global economic order. The features of UML employed to show the relationships among business concepts are inheritance and association (aggregation and composition). In addition, the models also include a definition of several concepts such as '**entities**' to represent the human, material and financial resources of enterprises; '**actions**' to show how entities interact, and thus not only to describe a static structure but the behaviour of the entities; '**plans**', which represent the future actions planned by the enterprise in order to react to a changing environment; '**rules**', which define the standard response to daily situations that occur in an enterprise; and '**organizations**' to represent the legal structure of the enterprise. Finally, the enterprise models proposed in [16] are the following:

- **Purpose:** to define the added value of the enterprise and its reason for existing. In this model, the strategic vision, tactical mission and operational goals and objectives must be depicted. It also has to show the measures that have been implemented for controlling these proposed objectives and planning, which can be either centralised or distributed.
- **Processes:** to show the actions performed by the enterprise in order to achieve an added value that can be offered to its costumers. In this view, the actions are grouped to compile business processes, which are carried out according to workflow rules and are controlled by different actors with different roles inside the organisation.
- **Entities:** to distinguish between the roles of an entity in different business processes in which it participates and the set of values which describe its static structure or state.
- **Organisation:** to represent the structure of the enterprise that enables an understanding of how business processes are carried out inside enterprise or among its partners.

On the other hand, the proposal presented in [17] for Enterprise Modelling is based on providing several views of a business model. These views (business vision, business process, business structure, and business behaviour) made up of one or more diagrams developed in UML, which capture the processes, rules, goals, and objects in the business, and their relationships and interactions with each other. The main concepts included in the Eriksson-Penker Business Extensions are the following:

- **Process:** the set of actions that transform input objects into outputs which have an added value for the customer. Processes have a goal and are affected by events.
- **Events:** a change of state that is caused by a process and is then received by one or more processes.
- **Resources:** all kinds of things that are used in the enterprise, whether they are either physical or abstract, for example, information.
- **Goals:** defined for the enterprise and each of its processes; they represent the desired state of each enterprise resource.
- **Business rules:** define the conditions under which business activity is to be performed and enterprise knowledge should be represented.
- **General mechanism:** mechanisms to be used in any diagram.

Other works in this context such as [18] point out the possibility of use UML as a language for Enterprise Modelling, even though in [19] it is qualified how and under which conditions this can be performed. Hence, we can conclude that it is possible and advisable to use UML for modelling enterprises. To do so, they define different types of specific concepts related to business domain, and use extension mechanisms like stereotypes, tagged values, and so forth provided by UML 1.x. However, the new version and specifications developed by the OMG, such as UML 2 and MDA, call for a review of these proposals again, and the works promoted by OMG within Business Enterprise Integration DTF, like Business Semantics of Business Rules (BSBR), Production Rules Representation (PRR), Business Process Definition Metamodel (BPDM), and Organization Structure Metamodel (OSM) that are currently being carried out show this to be the case. Moreover, taking into account the number of diagrams provided in UML 2 and that the previous works use mainly 'Class Diagrams', it would be interesting to clarify which UML 2 diagrams are useful at the CIM level and then to specify which part of CIM models must be transformed into PIM models, since according to [20] there must surely be degrees of CIMness.

Furthermore, despite the weakness of the stereotype mechanism is pointed out in [19], the new specification of UML 2 provides profiles with a greater degree of completeness than version 1.5. Therefore, it will be possible to customise UML in a better way [21]. For instance, UML provides a lot of diagrams for modelling dynamic aspects but not for direct modelling of business processes in a similar way that to how they are represented in an IDEF diagram. Hence, business process modelling with UML is complex [22] and the use of profiles according to UML 2 can make this task easier.

3 UML Proposal for Enterprise Modelling at the CIM Level

Enterprise models are normally composed of submodels such as organisational models, process models, information models, and so forth. These models must cover at least the following requirements [23], which are also compliant with the GERAM framework [5]:

- **Enable three fundamental kinds of flows inside and among enterprises:** material, information and decision or control.
- **Enable four modelling views:** functional, informational, resources and organisational.
- **Enable three levels of modelling:** definition of requirements, specification of design and implementation description.

On the one hand, these requirements have been established and accomplished by models developed with traditional Enterprise Modelling Languages. On the other hand, CIM models must describe the domain and requirements of the system in a model that is independent of computation representations and is expressed in the vocabulary of the domain practitioner.

Nevertheless, CIM characterisation is an ongoing work that is not yet formally defined or supported by OMG standards and tools. Using a CIM, an enterprise can capture, manage, and make better use some of its most valuable assets: knowledge of its resources, policies, rules, terminology and processes. Moreover, enterprises can specify, in an Enterprise Modelling Language, the requirements of their systems and check that the system design satisfies these requirements. CIM is made up two main subdivisions [24]:

- **Business Model:** a view of the enterprise and its environment that focuses on the scope and goals of the business, and the terminology, resources, facts, roles, policies, rules, processes, organisations, locations and events of concern to the business.
- **Business Requirements for Systems:** a view of the system and its environment that focuses on the purpose, scope, and policies for the system. Business Requirements can be divided into Functional Requirements, Interaction Requirements and Environment Contract.

These two characterisations and the comparison performed in Table 1 are the basis of our proposal. The table shows a general mapping among different approaches for modelling enterprises, such as traditional Enterprise Modelling (EM) taking into account its requirements [5, 23], and the UML framework (either MDA or approaches for Enterprise Modelling with UML as [16] and [17] summarised in section 2).

Furthermore, UML 2 is a language with a very broad scope that covers a large and diverse set of application domains. Not all of its modelling capabilities are necessarily useful in all domains or applications. For this reason, the new specification of UML provides a structure that will allow selection of only those parts of language that are of direct interest and will also take into account the

Table 1. Comparison among different frameworks for Enterprise Modelling

Item	Traditional EM [5, 23]	MDA	Marshall [16]	Eriksson [17]
Flows	Material	N/A	N/A	N/A
	Information	N/A	N/A	N/A
	Decision/Control	N/A	N/A	N/A
Views	Functional	N/A	Processes	Processes/Events
	Informational	N/A	Entities	Resources
	Resources	N/A	Entities	Resources
	Organisational	N/A	Purpose/Organisation	Goals/Business rules
Modelling levels	Definition of requirements	CIM	N/A	N/A
	Specification of design	PIM	N/A	N/A
	Implementation description	PSM	N/A	N/A

need to exchange UML models among different tools that use distinct subsets of the language [25].

Therefore, we consider as well this feature of UML in defining our proposal for modelling enterprises at the CIM level. To do so, we specify which models should make up the enterprise model of a company and which UML 2 diagrams are useful for this purpose. The main objective is to provide a framework to develop the enterprise models proposed using UML in order to gain a better understanding of what the business of the company is. Figure 1 depicts the position of our proposal inside GERAM framework. The models and UML 2 diagrams proposed for Enterprise Modelling at the CIM level are the following:

– **Global model:** used to give a general view on the other models performed. **Diagrams proposed:** Use-Case Diagram, and Package Diagram.
– **Organisational model:** this must represent both the static structure of enterprise and the dynamic structure at strategic and tactic level. The static structure should depict the departments and organisation established by enterprise. The dynamic structure should show the target model that the enterprise has (vision, mission, and so on), and the desicional structure and business rules existing within the context of the enterprise. **Diagrams proposed:** Use-Case Diagram, Class Diagram, Activity Diagram, Package Diagram, and the use of OCL will also be needed to describe restrictions.
– **Static model:** used to describe the informational view of the enterprise. Information about products or services provided by enterprise should therefore be represented. Furthermore, this must show the activities carried out in the enterprise to transform inputs into outputs, as well as the resources and restrictions related to these activities. **Diagrams proposed:** Use-Case Diagram, Class Diagram, and Package Diagram.
– **Dynamic model:** it should depict business, support and decisional processes from dynamic point of view taking into account events and logical operators at high level. **Diagrams proposed:** Use-Case Diagram, Activity Diagram, and Package Diagram.

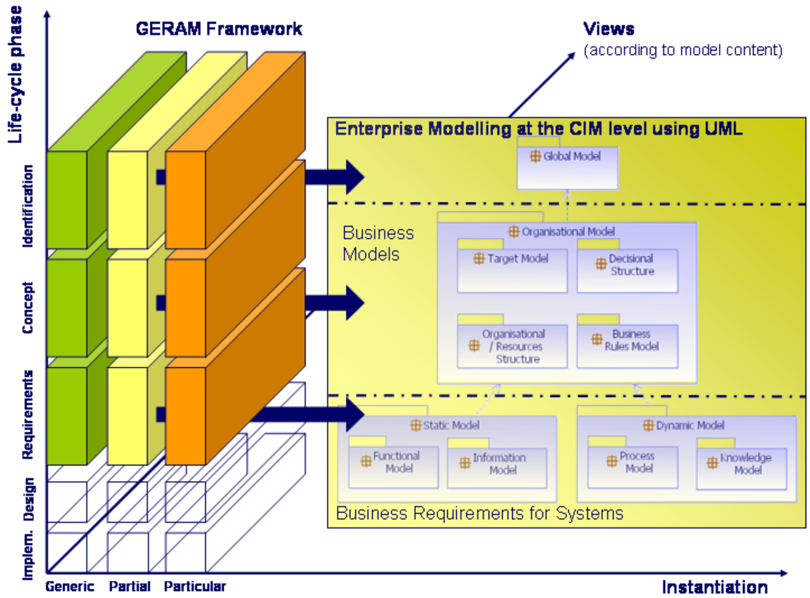


Fig. 1. Position of our proposal inside GERAM framework

The following step is to perform a UML profile for Enterprise Modelling (EM-Profile) in order to establish specific constructs to model enterprises, which are not usually provided with the standard UML, for each model defined in the proposal. In this paper, we show like an example the static structure for **Organisational Model** to better represent the organisation chart of the enterprise. Currently, within an enterprise the organisation elements are responsible for enterprise functions. Since, the main enterprise functions can be represented by use cases, the organisation elements could be represented by standard 'Actors'. In order to emphasise the different types of organisation elements and the hierarchical nature of their relationships, it is interesting to describe the generic constructs of an organisation structure within an UML profile for Enterprise Modelling. To describe this organisation we have developed the **'Organisation Breakdown Structure'** adding the following constructs at the meta-model level to the EM-Profile (see Fig. 2) [25]:

- **OrganisationElt**, subclass of **Kernel::Class**, for the description of nodes in the organisation chart.
- **Position**, subclass of the class **OrganisationElt**, for the description of the leaves of the organisation chart.
- **OrganisationUnit**, subclass of the class **OrganisationElt**, for the description of composite nodes in the organisation chart.
- **ObsAggregation** subclass of **Kernel::Association**, for the edge description of the Organisation Breakdown Structure.

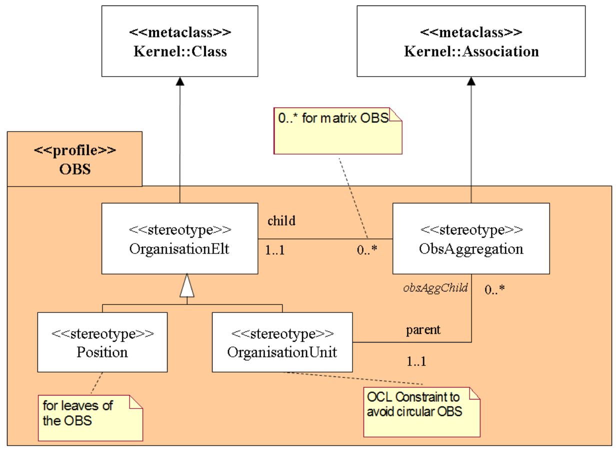


Fig. 2. Organisational constructs of the EM-Profile

Next, OCL constraints [26] can be added to refine the semantics, for example, to specify that the organisation chart cannot be cyclic:

```
context OrganisationUnit

--first helper to get the list of direct children of a node by navigating
--through the obsAggregation association
def: getListOfDirectChildren() : Set(OrganisationElt) =
     self.obsAggChild.child -> asSet()

--second helper to get all the acyclic children
def: getListOfAcyclicChildren(aList: Set(OrganisationElt)) : Set(OrganisationElt) =
     --first step: get the direct children of each element of the parameter 'aList'
     self.getListOfDirectChildren() ->
          --for each child
          collect(child |
              --test if the parameter 'aList' contains the child
              if aList->includes(child)
              then
                --if yes, return the list
                aList
              else
                --test if the child is a leaf of the organisation structure
                if child.oclIsTypeOf(Position)
                then
                   --if yes, add it to the parameter 'aList',
                   aList->including(child)
                else
                --recursive call to the method with the child
                --added to the parameter 'aList'
       child.oclAsType(OrganisationUnit).getListOfAcyclicChildren(aList -> including(child))
                endif
              endif)
          -> flatten() -> asSet()

--invariant definition
inv NoCircularContainment:
     -- the list of acyclic Children of the current node must not contain the current node
     self.getListOfAcyclicChildren(SetSelf) -> excludes(self)
```

In this constraint the helper `getListOfDirectChildren()` returns a Set of the **OrganisationElt** which are the direct children of the current **OrganisationUnit** instance. The helper `getListOfAcyclicChildren` recursively builds the set of all the (direct and transitive) children of the current **OrganisationUnit** instance. Finally, the invariant `NoCircularContainment` checks that the current instance does not belong to the set of its children.

4 Case Study

The main objective of this section is to show, by means of a Case Study, how to use UML for Enterprise Modelling at the CIM level, following the proposal outlined in the previous section. For modelling the Case Study, several diagrams have been developed as shown in Table 2. This table emphasises the representation capabilities of UML with regard to the main models proposed to depict an enterprise: Global, Organisation, Dynamic, and Static. For each point of view, both the potential diagrams (row 'Diagrams') and the work performed on the Case Study (row 'Case Study') are mentioned. Only the organisational model, with a Class Diagram for representing static structure is presented and discussed according to the excerpts from the case description (in italic font) provided by Singular Software[1].

Table 2. Models and UML diagrams performed for the Case Study

Model		**UML**
Global	**Diagrams**	Package Diagram, and (Business) Use-Case Diagram
	Case Study	Organisational, Business Use-Case, Use-Case, Static, and Dynamic Model
Organisation	**Diagrams**	Class Diagram stereotyped
	Case Study	OrganisationalUnit
Dynamic	**Diagrams**	Activity Diagram (AS-IS and TO-BE views)
	Case Study	Order Management Process from shops, franchisees, and dealers
Static	**Diagrams**	Class Diagram
	Case Study	Product, Service, Supplier Management, and Sales Management Models

The organisational aspects are depicted in the description document by the following sentences:

Demo TelCo S.A. is part of the Greek group of companies TelCo, which is specialised in telecommunications, in the production and distribution of batteries, as well as in retail sales of everyday technology products. TelCo is an official TelCarrier partner and sells TelCarrier products (SIM cards and sets).

[1] This Case Study was proposed within the framework of Task Group 2 of the INTEROP NoE [10] by Singular Software (http://www.singularsoftware.gr).

There are six main departments in Demo TelCo: Commercial, Sales, Financial, Logistics, Sunlight and IT. The commercial department of TelCo can be divided into three sub-departments: Products, Services and Administration.

In the 'Class Diagram' in Fig. 3, the aggregation between all the departments are an instance of the **ObsAggregation** meta-class defined in the EM-Profile, whereas the association named 'is a partner' between *TelCo* and *TelCarrier* is an instance of the 'classical' **Association** meta-class.

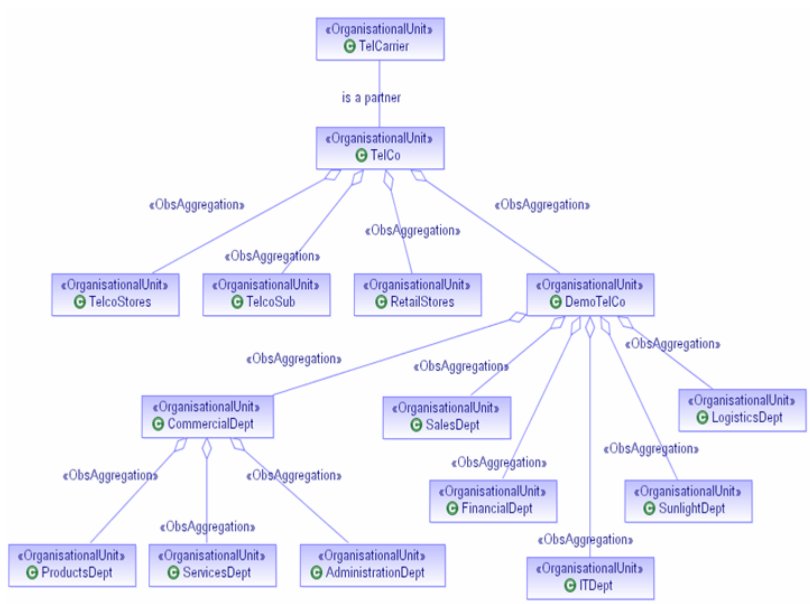


Fig. 3. Excerpt of a 'Class Diagram' describing the organisation of the Case Study

After this first modelling step, the model can be refined taking into account new information, for example, about the commercial department:

Product managers are responsible for creating new items in the system. Commercial department is responsible for price-formation.

In this case a leaf element of the organisation structure must be introduced (see Fig. 4): the position *ProductManager*. Product managers are members of the commercial department, therefore an **obsAggregation** is introduced between the organisation unit *CommercialDept* and the position *ProductManager*. Two new business use cases are added to depict the responsibilities of product managers. These use cases extend the main use case *Manage Products and Services*.

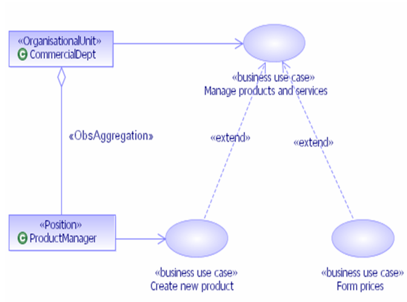


Fig. 4. Relationship between Business Use-Case Diagram and Organisational Diagram

5 Conclusion

In this paper we have analysed the previous works carried out within the context of Enterprise Modelling with UML. The benefits of model-driven approaches and the new specification of UML 2 provided by the OMG suggest the need to provide more practical examples for Enterprise Modelling with UML based on these recent works. A proposal for Enterprise Modelling at the CIM level using UML 2, based on previously described works and on traditional Enterprise Modelling Languages, has been presented along with a Case Study.

We have also shown the interest of defining a UML Profile 2.0 for Enterprise Modelling. In this way we have proposed an initial draft of a UML Profile for Enterprise Modelling. In this profile, only the organisational structure point of view, which allows us to describe the 'Organisational Breakdown Structure' of an enterprise, has been presented. This profile is going to be improved by including other concepts which are essential for a complete enterprise model, such as business rules, business process, and so forth.

Acknowledgments

This work was funded by CICYT DPI2003-02515 and INTEROP NoE [10]; the authors are indebted to Task Group 2.

References

1. Vernadat, F.B.: Enterprise Modeling and Integration: Principles and Applications. Chapman and Hall (1996)
2. Doumeingts, G., Chen, D.: Interoperability development for enterprise applications and software. In Cunningham, P., Cunningham, M., Fatelnig, P., eds.: Building the Knowledge Economy: Issues, Applications, Case Studies. eBusiness, IOS Press Amsterdam (2003)
3. UEML: Unified Enterprise Modelling Language Project (IST-2001-34229). http://www.ueml.org (2004) Deliverable 1.1.
4. EXTERNAL: Extended Enterprise MEthodology, Final version 1-12-D-2002-01-0 (IST-1999-10091). http://research.dnv.com/external/default.htm (2002)

5. IFIP-IFAC: Generalised Enterprise Reference Architecture and Methodology (GERAM). Technical Report Version 1.6.3 (1999) http://www.cit.gu.edu.au/ bernus/taskforce/geram/versions.
6. Chen, D., Doumeingts, G.: European initiatives to develop interoperability of enterprise applications–basic concepts, framework and roadmap. Annual Reviews in Control **27** (2003) 153–162
7. Grangel, R., Chalmeta, R., Campos, C., Coltell, O.: Enterprise Modelling, an overview focused on software generation. In Panetto, H., ed.: Interoperability of Enterprise Software and Applications Workshops of the INTEROP-ESA International Conference EI2N, WSI, ISIDI and IEHENA 2005, Hermes Science Publishing (2005)
8. Fox, M.S., Gruninger, M.: Enterprise Modelling. AI Magazine **19** (1998) 109–121
9. MODELWARE: Modeling solution for software systems Project (IST-2004-511731). http://www.modelware-ist.org/ (2006)
10. INTEROP: Interoperability Research for Networked Enterprises Applications and Software NoE (IST-2003-508011). http://www.interop-noe.org (2006)
11. OMG: MDA Guide Version 1.0.1. Object Management Group. Document number: omg/2003-06-01 edn. (2003)
12. Doumeingts, G., Vallespir, B., Zanittin, M., Chen, D.: GIM-GRAI Integrated Methodology, a Methodology for Designing CIM Systems, Version 1.0. LAP/GRAI, University Bordeaux 1, Bordeaux, France (1992)
13. Spur, G., Mertins, K., Jochem, R.: Integrated Enterprise Modelling. Beuth Verlag GmbH (1996)
14. IDEF: Integrated DEFinition Methods. http://www.idef.com/ (2006)
15. OMG: OMG Unified Modeling Language Specification, version 1.5. Object Management Group. formal/03-03-01 edn. (2003)
16. Marshall, C.: Enterprise Modeling with UML. Designing Successful Software Through Business Analysis. Addison Wesley (2000)
17. Eriksson, H., Penker, M.: Business Modeling with UML: Business Patterns at Work. J. Wiley (2000)
18. Panetto, H.: UML Semantics Representation of Enterprise Modelling Constructs. In: ICEIMT. (2002) 381–387
19. Berio, G., Petit, M.: Enterprise Modelling and the UML: (sometimes) a conflict without a case. In: Proc. of 10th ISPE International Conf. on Concurrent Engineering: Research and applications. (2003) 26–30
20. Berrisford, G.: Why IT veterans are sceptical about MDA. In: Second European Workshop on Model Driven Architecture (MDA) with an emphasis on Methodologies and Transformations, Kent, Computing Laboratory, University of Kent (2004) 125–135
21. Fuentes, L., Vallecillo, A., Troya, J.: Using UML Profiles for Documenting Web-Based Application Frameworks. Annals of Software Engineering **13** (2002) 249–264
22. Noran, O.: UML vs. IDEF: An Ontology-Oriented Comparative Study in View of Business Modelling. In: ICEIS (3). (2004) 674–682
23. Berio, G., Vernadat, F.B.: New developments in enterprise modelling using CIMOSA. Computers in Industry **40** (1999) 99–114
24. Hendryx, S.: Integrating Computation Independent Business Modeling Languages into the MDA with UML 2. http://www.omg.org/docs/ad/03-01-32.doc (2003)
25. OMG: Unified Modeling Language: Superstructure, version 2.0. Object Management Group. version 2.0 formal/05-07-04 edn. (2005)
26. OMG: OCL 2.0 Specification. Object Management Group. Adopted specification ptc/05-06-06 edn. (2005)

Title:	**Exchange of Business Process Models using the POP* Meta-model**
Authors:	R. Grangel and R. Chalmeta and S. Schuster and I. Peña
Conference:	Third International Conference on Business Process Management (BPM 2005) - International Workshop on Enterprise and Networked Enterprises Interoperability (ENEI'2005)
Book title:	Business Process Management - BPM 2005 Workshops, Chris Bussler & al. (Eds.)
Publisher:	Springer Verlag Berlin (USA)
Vol/Pages/Year:	LNCS3812/233-244/2006
ISSN:	0302-9743
JCR:	COMPUTER SCIENCE, THEORY & METHODS (0,402; 62; 38.975) (data of 2005)
Place:	Nancy (France)
Date:	5-9-2005

Abstract

Enterprise Modelling, in general, and Business Process Modelling, in particular, have been used for decades for different purposes and with interesting results. However, a variety of problems can be identified in this context and many enterprises find it difficult to leverage the full potential and benefits of these technologies. One of the most important problems in this sense is the lack of interoperability among enterprises at the modelling level. Quite a lot of efforts has been carried out in this domain to improve enterprise interoperability at this level. The development of the POP* meta-model is one of these initiatives, which aim to establish a meta-model and a corresponding methodology that enable enterprises to exchange their enterprise models, despite the fact that they use different Enterprise Modelling Tools.

In this paper, we present a 'proof of concept' of the POP* meta-model focused on the process dimension, which is expected to further our understanding of how this meta-model can be used to exchange different business process models among the partners in networks of collaborative enterprises. Moreover, the work performed in this 'proof of concept' has been a valuable aid to validate and improve the development of the POP* meta-model.

Exchange of Business Process Models using the POP* Meta-model

Reyes Grangel[1], Ricardo Chalmeta[1], Stefan Schuster[2], and Iñaki Peña[2]

[1] Grupo de Investigación en Integración y Re-Ingeniería de Sistemas (IRIS), Dept. de Llenguatges i Sistemes Informàtics, Universitat Jaume I, Campus del Riu Sec s/n, 12071 Castelló, Spain
{grangel, rchalmet}@uji.es
[2] European Software Institute (ESI), Parque Tecnológico de Zamudio # 204, 48170 Zamudio, (Bizkaia) Spain
{Stefan.Schuster, Inaki.Pena}@esi.es

Abstract. Enterprise Modelling, in general, and Business Process Modelling, in particular, have been used for decades for different purposes and with interesting results. However, a variety of problems can be identified in this context and many enterprises find it difficult to leverage the full potential and benefits of these technologies. One of the most important problems in this sense is the lack of interoperability among enterprises at the modelling level. Quite a lot of efforts has been carried out in this domain to improve enterprise interoperability at this level. The development of the POP* meta-model is one of these initiatives, which aim to establish a meta-model and a corresponding methodology that enable enterprises to exchange their enterprise models, despite the fact that they use different Enterprise Modelling Tools.
In this paper, we present a 'proof of concept' of the POP* meta-model focused on the process dimension, which is expected to further our understanding of how this meta-model can be used to exchange different business process models among the partners in networks of collaborative enterprises. Moreover, the work performed in this 'proof of concept' has been a valuable aid to validate and improve the development of the POP* meta-model.

1 Introduction

Enterprise Modelling is defined in [1] as the art of 'externalising' enterprise knowledge, that is to say, by representing the enterprise in terms of its organisation and dimensions (process, decision, product, resource, and so forth) [2]. Therefore, Enterprise Modelling enables enterprises to gain a much deeper knowledge and understanding of their business so that their objectives can be aligned with the market needs.

In the 70s, the first concepts of modelling were applied to the computer systems (E/R Model, DFD, and so forth), but the concept of Enterprise Modelling appeared in the USA at the beginning of the 80s, with the Computer Integrated Manufacturing (CIM) initiative. Examples of this initiative are the Integrated

Computer Aided Manufacturing (ICAM) Project carried out by the US Air Force or the Integrated Computer Aided Manufacturing-International (CAM-I) Project. In the mid 80s, different Enterprise Modelling Languages, such as GRAI or CIMOSA, emerged in Europe. In addition, numerous commercial tools appeared in the 90s to lend support to a great number of different modelling languages (ARIS ToolSet, FirstSTEP, METIS, KBSI Tools, MO^2GO, e-MAGIM, and so forth.) [2].

Today, the use of Enterprise Modelling is widely extended and many languages, methodologies and tools related to Enterprise Modelling exist, even for modelling Virtual or Extended Enterprises [3]. Enterprise Modelling Languages provide constructs with which to describe and model peoples' roles, operational processes and functional contents, as well as support information, and production and management technologies. However, integration of the models generated with these languages is complicated, since tools for exchanging models generated with different languages do not exist [4–7]. In summary, the main problems with respect to Enterprise Modelling can be seen as lying along two axes [8]:

- **Horizontal:** the lack of interoperability between Enterprise Modelling Languages and their corresponding Enterprise Modelling Tools. Almost all languages of this sort are proprietary specifications and can only be implemented with specific tools designed for this purpose. This problem complicates the interoperability of enterprises at the conceptual level. The main solutions provided by the research community to address this problem are focused on defining a common exchange format. This was, for instance, the goal of the UEML Project [7] and one of the objectives of the INTEROP [6] and ATHENA [4] Projects.
- **Vertical:** the weak connection between enterprise models and the generation of software is one of the major reasons why enterprises develop only a few models, which, moreover, are rarely updated and are therefore not very successful in accomplishing their initial purposes. Initiatives, such as MDA [9] promoted by OMG and MDI within INTEROP [6], are intended to solve this kind of problems.

These same problems can also be observed in the business process context. The number of modelling techniques and tools available for supporting Business Process Modelling is growing rapidly, because of the increasing popularity of business process orientation [10]. In recent years, many advantages of using Business Process Modelling have been pointed out [11].

Nevertheless, collaborative enterprises face a number of problems when attempting to harvest the benefits of Business Process Modelling. A collaborative enterprise is an enterprise where teams work together across boundaries, e.g. life-cycle phases, sharing results and knowledge to improve their common understanding and enable better performance and higher quality results [12]. The main reason for this situation is the large number of techniques and tools [10] that support and that are used for Business Process Modelling; as a result, collaborative enterprises find it difficult to exchange business process models in an efficient way.

Taking the problem of interoperability as its main inspiration, the objective is to achieve a common format, like POP* or UEML, which are valid initiatives allowing enterprises to exchange different kinds of models and to set up an environment in which existing models can be reused [4–7]. In particular, within the framework of the ATHENA Project [4], the POP* methodology was developed with the aim of solving this kind of problems and improving enterprise interoperability. In this context, then, this paper presents the work carried out in the 'proof of concept' of the POP* meta-model in order to validate it and it describes how the POP* meta-model could be used to exchange business process models among different partners from a process-oriented point of view.

The paper is organised as follows. Section 2 gives a brief description of the ATHENA Project as the framework in which this research was carried out, and also discusses the main issues regarding the POP* meta-model and especially its process dimension. Section 3 describes the research work performed and the main results obtained in the 'proof of concept' of the POP* meta-model. Finally, the main conclusions are outlined in section 4.

2 ATHENA Project

ATHENA (Advanced Technologies for interoperability of Heterogeneous Enterprise Networks and their Applications) is an Integrated Project sponsored by the European Commission in support of the Strategic Objective 'Networked businesses and government' set out in the IST 2003-2004 Work Programme of FP6 [4]. ATHENA aims to make a major contribution to interoperability by identifying and meeting a set of inter-related business, scientific and technical, and strategic objectives. In ATHENA, different Research and Development projects are executed in an integrated way. The research work presented in this paper was developed within the framework of one of these projects, called A1, and which focuses on 'Enterprise Modelling in the Context of Collaborative Enterprises'.

The overall goal of this project is the development of methodologies, core languages and architectures as models, model-generated workplaces, services and execution platforms for establishing collaborative on-demand Extended Enterprises and Networked Organisations.

2.1 POP* Meta-model

One of the main goals of the A1 Project is to develop a methodology that provides a set of basic modelling constructs to support model exchange in the context of collaborative enterprises. The methodology includes [12]:

1. The **POP* meta-model**, which describes the set of basic modelling constructs defined and their relationships.
2. The **guidelines**, which describe the management and use of the POP* meta-model.

With respect to this goal and business process orientation, the work performed in the project has similar objectives, but at the same time a different scope, to other approaches like UEML [7] or BPDM [13]. Although the development of the POP* meta-model is based on the adoption of a holistic point of view of an enterprise which takes into account its different dimensions, that is to say, process, organisation, decision, and so forth, this first version is developed in a more comprehensive manner and focuses on the process dimension.

Moreover, POP* was developed taking into account how enterprises need to establish flexible relationships with other partners in order to achieve some competitive advantage, and also with a top-down approach that allows for definition of the constructs needed to depict the particular features of this kind of enterprises. On the other hand, the POP* meta-model was also developed with a bottom-up approach, which involved reviewing some of the most important Enterprise Modelling Languages like IEM, EEML, GRAI, and so forth, and as a result it covers the common concepts identified in these languages. However, the POP* meta-model is neither the merge of the meta-models of these specific Enterprise Modelling Languages, nor the addition of them, but the mapping of the main constructs of these languages in order to identify common concepts and to avoid redundancies. In this sense, the POP* meta-model is a valid mechanism with which to exchange enterprise models among partners in a collaborative enterprise that use different enterprise modelling platforms and languages.

Therefore, the POP* meta-model is a first, but necessary, step in order to achieve enterprise interoperability at the conceptual level. Furthermore, the POP* meta-model will be useful for developing the architecture specification of the Modelling Platform for Collaborative Enterprise (MPCE) within the ATHENA Project. This platform will facilitate the exchange of different kinds of enterprise models, based on the POP* meta-model, and allow them to be managed in a better fashion.

A thorough explanation of the POP* meta-model and its corresponding methodology can be found in [12]. This work includes the description of the POP* meta-model in its first version, with the dimensions defined so far:

- **Process dimension:** representing the activities and tasks carried out in an enterprise and the different objects that are needed to perform them.
- **Organisation dimension:** expressing the formal and informal organisational structures of an enterprise, as well as the different stakeholders and relationships that form part of this organisation.
- **Product dimension:** representing the products or services that an enterprise offers to the market.
- **Decision dimension:** expressing the decision-making process and the structure needed in an enterprise to perform it.
- **Infrastructure dimension:** depicting the ICT infrastructure of an enterprise.

Furthermore, it also provides guidelines illustrating the management and potential use of the POP* meta-model in a cross-organisational setting. The main goal of these guidelines is to explain how the POP* meta-model can be used to exchange enterprise models among or inside enterprises that use different Enterprise Modelling Tools.

2.2 Process Dimension

The process dimension of the POP* meta-model, shown in Fig. 1, is concerned with the activities and elements needed to enact and execute processes in a collaborative enterprise. Its objective is to provide the basic constructs with which to model the tasks and the main enterprise objects that participate in these tasks with different roles, such as input, output, control, and so forth. The process dimension also supports the representation of the process flow, as well as conditions or associated decisions.

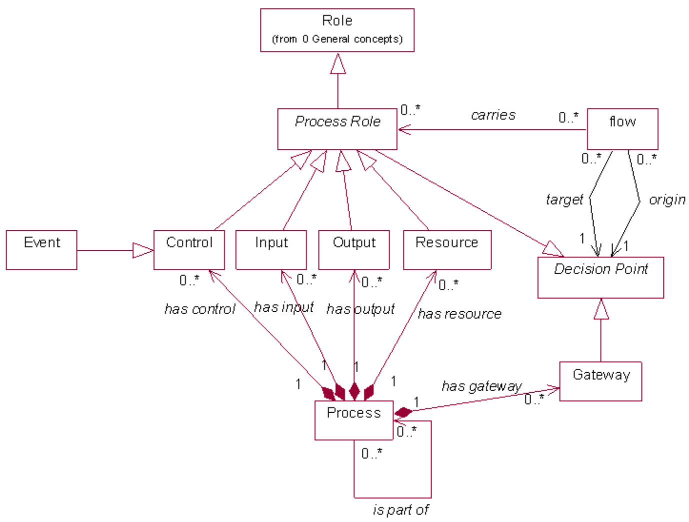


Fig. 1. POP* meta-model: process dimension

In this section, we present a brief description of the main constructs in the process dimension of the POP* meta-model (see Fig. 1). A complete description of these constructs can be found in [12].

- **Process:** this represents a task or an activity performed in an enterprise. A Process can be derived into different subprocesses in order to depict the desired level of detail.
- **Role/Process Role:** this is used to express the function of the diverse enterprise objects in the execution of a Process. Consequently, the subclasses of the Process Role are: Control, Input, Output and Resource.
- **Decision Point:** this depicts a conditional point used to solve the process flow and continuation, i.e., the process sequence. A Decision Point can be a

Process Role, which can have an object attached to it, or a Gateway, which is a true decision point without attached object, and is owned by a process.
– **Flow:** this construct represents the connection of Processes across two Decision Points, which can be either Gateways or Process Roles played by different enterprise objects.

3 'Proof of Concept' of the POP* Meta-model

Within the framework of the above-mentioned ATHENA Project, this paper describes the work performed in the 'proof of concept' of the POP* meta-model. The main objective of this research work is to demonstrate that the POP* meta-model is well defined, as it provides a common and standard language to exchange models among different Enterprise Modelling Tools.

3.1 Process Description

Our demonstration method includes two main steps, as shown in the diagram in Fig. 2. First, an existing model compliant with a specific Enterprise Modelling Tool (MO^{2}GO) [14] is transformed by hand into a POP* model using a UML Profile 2.0. Second, the POP* model is imported into different Enterprise Modelling Tools (GraiTools [15] and Metis [16]).

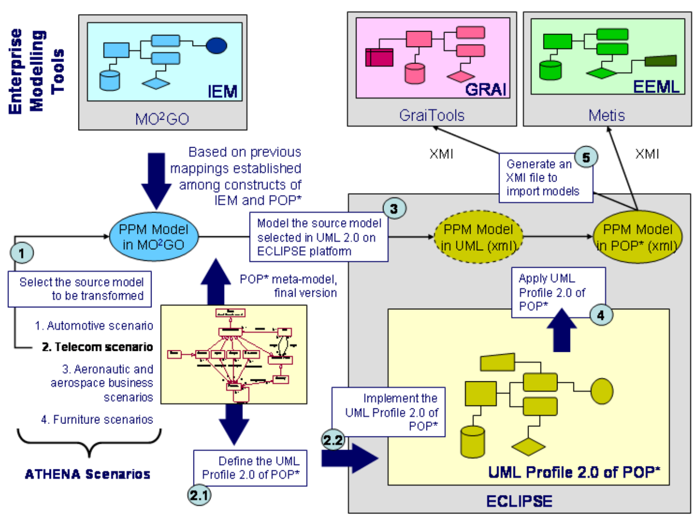


Fig. 2. Diagram showing tasks performed in the 'proof of concept' of the POP* meta-model

In order to achieve this goal, a UML Profile 2.0 of the POP* meta-model was implemented using the ECLIPSE platform. UML Profiles 2.0 is a mechanism that allows the metaclasses of an existing meta-model to be extended, in order to adapt it for different purposes. Therefore, this mechanism includes the ability to tailor the UML meta-model to different platforms (such as J2EE or .NET) or domains (such as real-time or BPM) [17].

In our case, we will use this mechanism to define a UML Profile 2.0 of the POP* meta-model with the aim of carrying out a 'proof of concept' of POP*. The idea is to extend the UML meta-model within a specific domain by means of our profile. This profile can then be used to model collaborative enterprises according to the POP* meta-model.

Therefore, the first task to be carried out in this process is the definition of the UML Profile 2.0 of the POP* meta-model. Following the recommendations given in [18], the main steps involved in defining this profile are:

1. To include one stereotype for each element of the POP* meta-model in a 'profile' package.
2. To specify what elements of the UML meta-model are extended by the stereotypes.
3. To define the attributes of the POP* meta-model as tagged values.
4. To define the constraints of the domain.
5. To implement the profile defined by using the ECLIPSE UML 2.0 plug-in.

On the other hand, the remainder tasks shown in Fig. 2, which are needed to complete the 'proof of concept', are explained in more detail in the following section.

3.2 Work Performed

The 'proof of concept' of the POP* meta-model was performed in order to validate it and to demonstrate a real application of the POP* meta-model as an exchange format. Thus, the work performed and explained in this section can be useful to gain a better understanding of how the POP* meta-model could be used to exchange business process models. This work was carried out according to the steps proposed by the guidelines defined in [12] for applying and managing the POP* meta-model. In what follows, the main steps performed and illustrated in Fig. 2 are presented.

STEP 1. Select the source model to be transformed. For the 'proof of concept' we selected one of the ATHENA scenarios, from the Telecom sector. In particular, the scenario is related to the Product Portfolio Management Process (PPM). We used the PPM scenario modelled in MO^2GO, and we chose only a part of this model in order to ensure that the work could be performed in a short amount of time.

The part of the PPM model selected was the 'WIBAS[3] Project development' process (see Fig. 3), because it illustrates some crucial POP* concepts. It includes almost all the elements that can be represented in a MO^2GO model, and it is sufficiently complex to demonstrate the use of POP* as an exchange format.

[3] WIBAS is the name of a particular product development project.

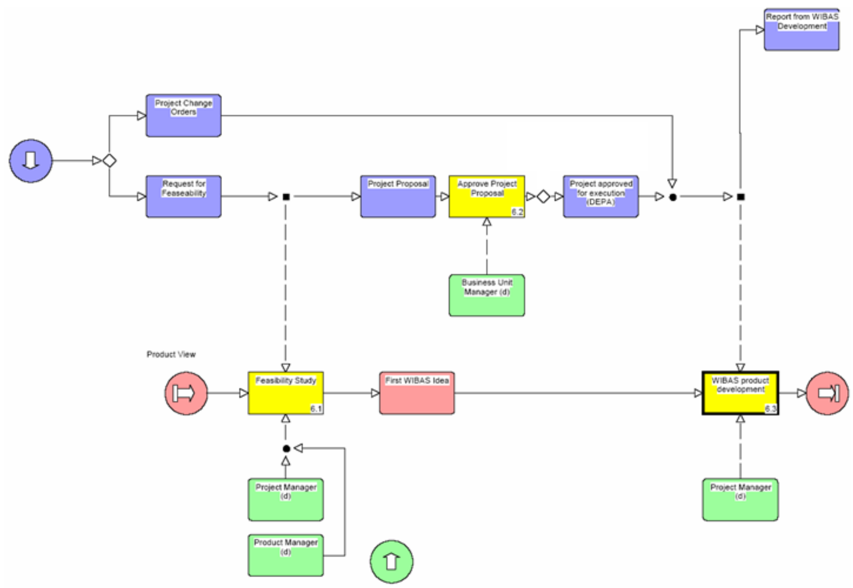


Fig. 3. 'WIBAS project development' process of PPM model developed in MO^{2}GO

STEP 2. Define and implement the UML Profile 2.0 of POP*. The result of the tasks performed in this step is a description of the UML Profile 2.0 of the POP* meta-model. This profile could be used as a basis for further implementation of POP* as an Enterprise Modelling Language. The profile identifies a subset of the UML meta-model elements but does not remove any of the UML meta-model functionalities, and therefore all the utilities of UML remain available for the final users.

Three components are needed to create UML profiles: stereotypes, restrictions and tagged values. **Stereotypes** are defined by their names and the elements of the meta-model that are associated to them. They establish the features that designers assign to the elements that are extended by the profile. **Restrictions** are used to establish conditions over the stereotyped elements, and **tagged values** are additional meta-attributes that are associated to a meta-class in the extended meta-model. This profile specification was developed in accordance with the latest version of the Unified Modelling Language, UML 2.0 [17, 19].

STEP 3. Model the source model selected in UML 2.0. Prior to modelling, it is necessary to select UML diagrams that are useful for our 'proof of concept'. We focused on the most expressive UML diagrams that can be used for business processes modelling, which are class and activity diagrams. This step was carried out using the Rational Rose modeller from the Rational division of

IBM on ECLIPSE platform. This tool was chosen in order to take advantage of the ECLIPSE UML 2.0 plug-in and to support advanced UML profile 2.0 management and XMI 2.0 interchange.

STEP 4. Stereotype the model developed in UML 2.0 with the UML Profile 2.0 of POP*. Using the UML Profile 2.0 of POP* thus implemented, all components of the model previously developed in UML 2.0 were extended using stereotypes (see Fig. 4). In this way we obtained a full, semantically equivalent model but which is now UML 2.0 compliant, that is, it is fully compliant with XMI 2.0 and therefore easily interchangeable.

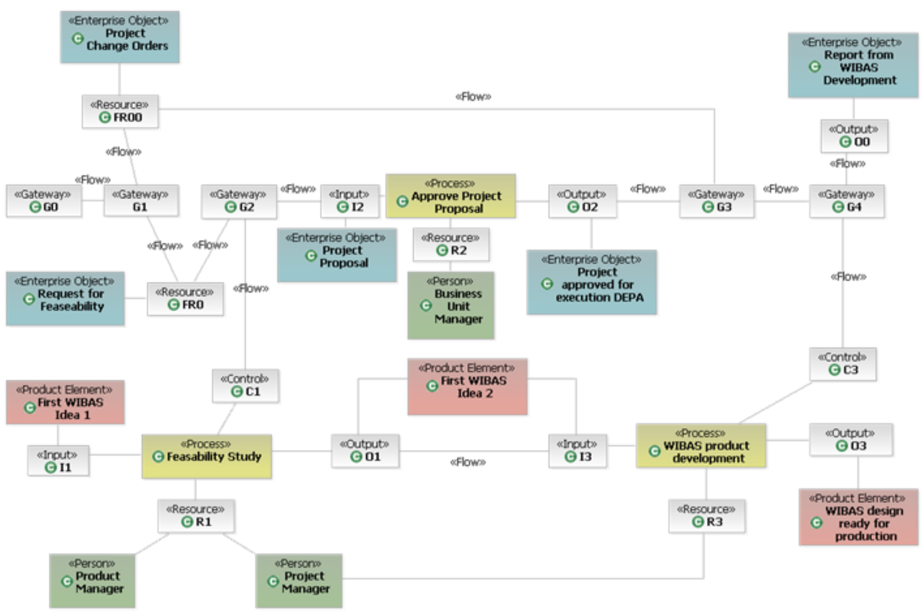


Fig. 4. 'WIBAS project development' process of PPM model developed in POP*

Elements in the MO^2GO diagram were replaced by POP* concepts, but obviously translating native models to POP* involves more than simply replacing each element in the native models by its corresponding element in POP*. The translated model should follow the rules that define how POP* concepts can be related (that is, the syntactic rules defined by the POP* meta-model). This could entail having to include new elements, as can be seen in Fig. 5. For example, in order to develop the class diagram of the translated POP* model:

– Processes were defined to include their interfaces with the outside world. These are specialisations of the Process Role: Input, Control, Output and Resource.
– 'Split' or 'Join' in the MO^{2}GO diagram were transformed into Gateways in the POP* class diagram.
– Flows were stereotyped as associations in order to simplify the diagram and give it more expressiveness.
– According to the POP* meta-model, Flows can connect only Decision Points (this means Gateways or Process Roles). For example, we cannot connect two Processes (or an Object with a Process) directly by means of a Flow.

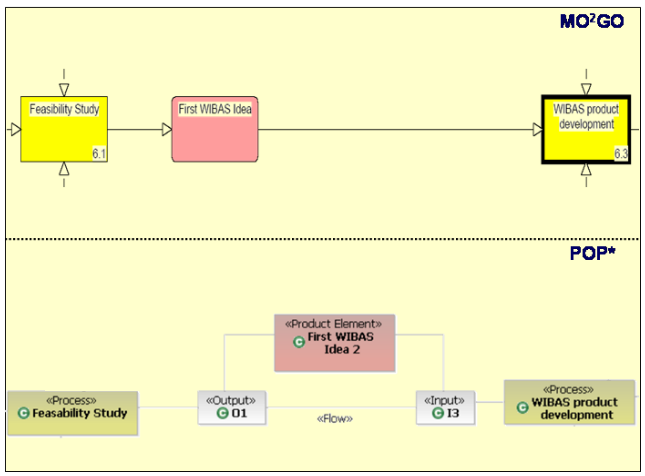


Fig. 5. Modelling of flows in POP*

STEP 5. Generate an XMI file to be imported. Finally, the objective is to generate an XMI file of the POP* model generated in the step 4 by means of the capabilities from the ECLIPSE platform, which will be imported into different Enterprise Modelling Tools, like GraiTools or Metis, for instance.

3.3 Results and Lessons Learned

The 'proof of concept' of the POP* meta-model fulfilled its initial purpose. It assisted in the final development of the POP* meta-model, clarifying some concepts of the meta-model and proving that it is possible to transform models developed in different Enterprise Modelling Tools by means of POP*. Moreover, the tangible results obtained in this research work are:

- The definition of the UML Profile 2.0 of the POP* meta-model, and its implementation in the ECLIPSE UML 2.0 plug-in.
- A real-use case modelled in POP*, based on a source model developed in a specific Enterprise Modelling Tool.
- The XMI files of the real-use case modelled in POP* that can be imported into other Enterprise Modelling Tools.

Finally, the main lessons learned in performing the 'proof of concept' of POP* can be summarised in the following points:

- Major problems were encountered in understanding the source model, especially because it was not developed by one of the team members. In spite of knowing the constructs of a specific Enterprise Modelling Language, the modelling process is sometimes subjective and hence it is hard to interpret a source model that is to be transformed into another model. To this regard, POP* can be useful since it establishes a mapping among the constructs of the most important Enterprise Modelling Languages.
- When transforming a source model into another one by means of POP*, it will sometimes be necessary to include some additional elements in the target model, as shown in Fig. 5. However, these new elements should not modify the semantics of the source model. Hence, it is possible to have some concepts in a specific Enterprise Modelling Language which do not have any correspondence with others. As a consequence, the transformation process must sometimes be performed in a semi-automatic way and with expertise human collaboration.

4 Conclusion

We can conclude that it is possible to use the POP* meta-model as an exchange format among enterprises that use different Enterprise Modelling Languages. Hence, it is a first step on the way to achieving interoperability in the context of collaborative enterprises at the modelling level, and a valid result to be taken into account in further works that are going to be developed in the ATHENA Project, such as the specification of the MPCE, for instance.

On the other hand, and even though it was not the initial objective of the ATHENA Project, the POP* meta-model is now sufficiently well defined to be able to use it as the basis for the further development of an Enterprise Modelling Language, which could be used by providers of tools with meta-modelling capabilities. However, the work within the ATHENA Project will continue to improve and refine the POP* meta-model, particularly the less mature dimensions like the decision dimension, and also to add new dimensions with the objective of providing an exchange format for Enterprise Modelling from a holistic point of view.

Finally, the 'proof of concept' of the POP* meta-model was useful as an aid to understanding how it is possible to exchange business process models among different partners in the context of collaborative enterprises using the POP*.

Acknowledgments

This work was funded by ATHENA IP (IST-2003-507849, [4]) and represents part of the European Software Institute's contribution to the project. The authors are indebted to the A1 Project. In addition, it is partially supported by INTEROP NoE (IST-2003-508011, [6]) and by CICYT DPI2003-02515.

References

1. Vernadat, F.B.: Enterprise Modeling and Integration: Principles and Applications. Chapman and Hall (1996)
2. UEML: Deliverable D1.1. Report on the State of the Art in Enterprise Modelling. http://www.ueml.org (2002)
3. EXTERNAL: Extended Enterprise MEthodology Project, Final version 1-12-d-2002-01-0 (IST-1999-10091). http://research.dnv.com/external/default.htm (2002)
4. ATHENA: Advanced Technologies for interoperability of Heterogeneous Enterprise Networks and their Applications) Project (IST-2003-2004). http://www.athena-ip.org (2005)
5. IDEAS: IDEAS (Interoperability Development for Enterprise Application and Software) Project. http://www.ideas-roadmap.net (2005)
6. INTEROP: Interoperability Research for Networked Enterprises Applications and Software NoE (IST-2003-508011). http://www.interop-noe.org (2005)
7. UEML: UEML (Unified Enterprise Modelling Language) Project (IST-2001-34229). http://www.ueml.org (2005)
8. Grangel, R., Chalmeta, R., Campos, C., Coltell, O.: Enterprise Modelling, an overview focused on software generation. In Panetto, H., ed.: Interoperability of Enterprise Software and Applications, Hermes Science Publishing (2005)
9. Object Management Group, OMG: MDA Guide Version 1.0.1. Document number: omg/2003-06-01 edn. (2003)
10. Aguilar-Saven, R.S.: Business process modelling: Review and framework. International Journal of Production Economics **90** (2004) 129–149
11. Kalpic, B., Bernus, P.: Business process modelling in industry–the powerful tool in enterprise management. Computers in Industry **47** (2002) 299–318
12. ATHENA: Deliverable DA1.3.1. Report on Methodology description and guidelines definition. http://www.athena-ip.org (2005)
13. BPDM: Business Process Definition Metamodel. http://www.omg.org (2005)
14. IPK: MO^2GO. http://www.ipk.fhg.de (2005)
15. LAP/GRAI: GraiTools. http://www.graisoft.com (2005)
16. Computas: METIS. http://www.computas.com (2005)
17. Object Management Group, OMG: Unified Modeling Language (UML) Specification: Superstructure, version 2.0. Document: ptc/04-10-02 (convenience document) edn. (2004)
18. Fuentes, L., Vallecillo, A.: Una introducción a los perfiles UML. Novática **marzo-abril** (2004) 6–11
19. Object Management Group, OMG: Unified Modeling Language (UML) Specification: Infrastructure, version 2.0. OMG Adopted specification ptc/03-09-15 edn. (2003)

Chapter 4

KM-IRIS Methodology for Knowledge Management

Title:	**Methodology for the Implementation of Knowledge Management Systems**
Authors:	R. Chalmeta and R. Grangel
Revue:	Journal of the American Society for Information Science and Technology
Publisher:	John Wiley & Sons
ISSN:	1532-2882
JCR:	COMPUTER SCIENCE, INFORMATION SYSTEMS (1,583; 20; 2552) (data of 2005)
Status:	In evaluation

Abstract

Managing knowledge means managing the processes of creation, development, distribution and utilisation of knowledge in order to improve organisational performance and increase competitive capacity. However, serious difficulties arise when attempts are made to implement knowledge management in enterprises. One of the reasons behind this situation is the lack of suitable methodologies for guiding the process of development and implementation of a Knowledge Management System, which is a computer system to make the processes of creating, collecting, organising, accessing and using knowledge as automatic as possible.

In this paper we propose a methodology for directing the process of development and implementation of a Knowledge Management System in any type of organisation. The methodology is organised in phases and outlines the activities to be performed, the techniques to be used, the supporting tools and expected results for each phase. In addition, an example of a specialised version of this methodology adapted to the specific characteristics of an enterprise is also presented. This specialised version can in turn be tailored even further to adapt it to each type of business.

METHODOLOGY FOR THE IMPLEMENTATION OF KNOWLEDGE MANAGEMENT SYSTEMS

Ricardo Chalmeta[a], Reyes Grangel[a]

[a] Grupo de Investigación en Integración y Re-Ingeniería de Sistemas (IRIS).

Universitat Jaume I. 12006 Castellón. Spain.

Tel: + 34 964 728329 Fax: + 34 964 728435

{rchalmet, grangel}@uji.es

Abstract

Managing knowledge means managing the processes of creation, development, distribution and utilisation of knowledge in order to improve organisational performance and increase competitive capacity. However, serious difficulties arise when attempts are made to implement knowledge management in enterprises. One of the reasons behind this situation is the lack of suitable methodologies for guiding the process of development and implementation of a Knowledge Management System, which is a computer system to make the processes of creating, collecting, organising, accessing and using knowledge as automatic as possible.

In this paper we propose a methodology for directing the process of development and implementation of a Knowledge Management System in any type of organisation. The methodology is organised in phases and outlines the activities to be performed, the techniques to be used, the supporting tools and expected results for each phase. In addition, an example of a specialised version of this methodology adapted to the specific characteristics of an enterprise is also presented. This specialised version can in turn be tailored even further to adapt it to each type of business.

Keywords: Methodology, Knowledge Management, Knowledge Management Systems, Enterprise, Information Systems

2

1. INTRODUCTION

One of the new tools for improving competitiveness and productivity in organisations is the implementation of Knowledge Management (KM), understood as meaning the capacity to create, collect, organise, access and use knowledge. This is due to the fact that:

- Company decisions and actions require far more information and knowledge due to the more global and complex environment.
- There is an increased demand for greater knowledge intensity in products, processes and services. By applying knowledge to the products and services, its value increases.
- Knowledge management stresses the importance of intangible assets and enables them to be used to advantage.
- The possibilities opened up by Information and Communication Technologies to improve knowledge management both within and among enterprises.

A key factor for achieving correct knowledge management in an organisation is the development and implementation of a Knowledge Management System (KMS), that is to say, a technological information system that supports knowledge management, which allows knowledge to be automatically created, codified, stored and distributed within the organisation (Day, 2001).

Running a KMS development and implementation project in an organisation is an extremely complex process that involves different technological, human and organisational aspects. For the project to succeed, each and every one of the steps taken from the moment it is conceived until the ultimate aim is accomplished must be carried out correctly. To do so, it is essential to follow a methodology that guides users throughout the analysis, development and implementation of the KMS and ensures its success.

The literature contains different methodologies that can be used for Information Systems Development (ISDM). These provide a consistent set of procedures to be followed, as well as tools, techniques and documentation that can be used, to make the process of managing and developing information systems more efficient and effective (Yadav et al., 2001). In all cases, an ISDM embodies some form of philosophical view and implies a time-dependent sequence of thinking and action stages (Walters et al., 1994).

A wide range of such frameworks have been developed over the years. In this regard, in 1994 (Jayaratna, 1994) estimated that there were more than 1000 available for use. In (Avison & Fitzgerald, 2006) there is a good compilation and comparative analysis of the most important ones.

Each of these ISDM has its own acknowledged strengths and weaknesses. However, one ISDM is not necessarily suitable for use in all projects. Each methodology is best suited to a specific type of project due to their different technical, organisational, project and team considerations (Meso et al., 2006).

From our experience in developing KMS in real cases and after reviewing the literature (Viswanathan et al., 2005) we can state that one of the chief reasons for the large number of failures in implementing a KMS is the lack of an ISDM which is specifically oriented towards the development of a KMS that reduces the complexity of the process. For example, when the currently existing ISDM, are applied to the development of a KMS, at some stage it becomes necessary to specify the requirements the future KMS should meet. These ISDM do not, however, help users to identify them in a practical way. It would therefore be very useful for the users who have to define these requirements (which in this case is knowledge) to have a series of templates that include examples of typical items of knowledge that an organisation like theirs will be interested in managing. Thus, the process of specifying the requirements could be carried out more quickly and thoroughly. Another example is that, although existing methodologies make use of modelling languages to create a model of the computer system, they do not employ specific languages with profiles that are expressly oriented towards modelling knowledge. Such profiles would allow the knowledge map to be generated in a simple manner that is at the same time both graphic and intuitive.

Consequently, there are a number of problems concerning the methodologies for developing KMS that remain unsolved and hence there is still room for significant improvement as regards both their theoretical aspects and their practical applicability (McInerney & Day, 2002).

To help solve this problem, in this paper we propose a methodology that is structured in several different phases and can be used to guide projects intended to develop and implement knowledge management systems in an enterprise. The methodology makes it possible to: (1) gather, identify and separate knowledge from information; (2) store knowledge using a common language; and (3) make this knowledge widely available to whoever may need it. To collect data and test the operative capacity of the methodology our work was carried out in collaboration with a large textile company.

This methodology will be of interest to practitioners who are involved in the development, implementation and setting up of KMS, since it will enable them to organise and manage the project better, while also allowing them to enhance the way they carry out each of its different component activities.

The paper is organised as follows: the next section presents a review of what knowledge, knowledge management and knowledge management systems are and how they are related to the use and dissemination of knowledge within an organisation. In addition, the current situation with respect to the development and implementation of knowledge management systems is analysed in order to determine the main reasons why they fail. Section three outlines the methodology proposed here for helping to develop and implement a KMS in any type of organisation. The methodology is organised in phases and outlines the activities to be performed, the techniques to be used, the supporting tools and expected results for each phase. Section four shows an example of how this methodology could be applied in an enterprise. Finally, section five presents a case example, and section six shows the conclusions of the work.

2. LITERATURE REVIEW

There is no universally accepted definition of exactly what knowledge is. Some authors define it, for example, as the information individuals possess in their minds (Drestke, 1981). This definition is argued by saying that data (raw numbers and facts) exist within an organisation. After processing these data they are converted into information and, once it is actively possessed by an individual, this information in turn becomes knowledge. There are also other approaches to defining knowledge that are more independent on the information technologies. One of the most cited is the approach proposed by (Nonaka & Takeuchi, 1995), who defines knowledge as the justified belief that increases the capacity of an entity for effective action. Following this line of reasoning, knowledge can be seen from five different perspectives (Alavi & Leidner, 2001): (1) as a state of mind, (2) as an object, (3) as a process, (4) as a condition for access to information, or (5) as a capability. Taking this context and our own empirical observations as our starting point, we define knowledge as the awareness that enables us to possess the skill or the capacity required in a particular situation (1) to deal with and resolve complex issues in an efficient and creative manner, and (2) to take advantage of opportunities by making the most appropriate decisions.

The process of converting the knowledge from the sources available to an organisation and then connecting people with that knowledge is one of the definitions provided to explain knowledge management (O'Leary et al., 1997; O'Leary, 1998; Myers, 1996). Therefore, the aim of knowledge management is the creation, collecting, storage, access, transfer and reuse of knowledge (Devedzic, 1999).

Knowledge management has been used in different kinds of organisations in order to boost profits, to be competitively innovative, or simply to survive (Abdullah et al., 2002). Different examples of its application are well described in a great number of papers. KM is used, for example, to create or assemble productive resources, including research, manufacturing, design, business, learning and training (Liao, 2003).

However, there are different problems that hamper its application, some of the most important being (Snowden, 2002):

- The complexity of the concept.
- The fact that its introduction requires specific organisational culture and practices, human resource policies, marketing and change management.
- The intangibleness of its benefits: many business people find it difficult to associate investment in knowledge management with improvements in company results.

6

- The fact that it needs to be supported by the information and communication technologies.

Several different theories have been put forward to get to grips with the first three problems cited in the previous paragraph associated with knowledge management. These include the cognitive (Chiu et al., 2006), motivational (King & Marks, 2006; Hall, 2003), economic (Ke & Wei, 2005; Eliasson, 2005) or the organisational theories (Gray & Meister, 2006; Revilla et al., 2005). These theories have been used to deal with the formal aspects and essentially attempt to explain the concept of knowledge, its typology and the actions to be carried out in order to favour its development and management.

As far as the fourth problem is concerned, the generally accepted solution is to develop a Knowledge Management System, that is to say, a specialised system supported by information and communication technologies that interacts with the organisation's computer systems to make the processes of creating, collecting, organising, accessing and using knowledge as automatic as possible (Abdullah et al., 2002).

According to Ernst and Young (2001) organisations are basically putting five types of projects into practice related with KMS implementation: creation of Intranets and corporate portals; data warehouses or knowledge repositories (Inmon, 1996); implementation of decision support tools, Implementation of groupware; and creation of document management systems (Lindvall, 2003).

Thus, the architecture of information systems in enterprises that wish to implement a Knowledge Management System should provide a set of tools for supporting the smart integration of all enterprise computer components.

However, the development and implementation of KMS that embrace the whole organisation, including knowledge resulting from its relations with other institutions that it collaborates with, and which also incorporate the management of tacit knowledge is a more complex affair that has still not been satisfactorily resolved (Heinrichs et al., 2005). In this regard (Schutt, 2003) describes the evolution the different generations of knowledge management systems have undergone and explains why they did not live up to the expectations they had aroused. One of the main reasons, as (Shin et al., 2001) confirms, is the lack of a methodology to guide the KMS development and implementation project.

3. KM-IRIS METHODOLOGY

In order to successfully carry out a project of development and implementation of a KMS, while at the same time reducing the degree of complexity, it would be a great aid to be able to use a stage-based methodology that defines the whole creative process in each phase. This would involve defining, among other things, the tasks to be performed, the techniques to be used, the modelling languages for representing the knowledge and the technological infrastructure that allows knowledge to be stored, processed and distributed, depending on the roles that have been defined.

To solve this problem of a lack of such knowledge management methodologies, the IRIS Group at the Universitat Jaume I in Castellón, Spain, has been working on a project entitled "Methodology for Knowledge Management" since 2003. The objective was to develop and validate a useful, practical methodology that can be used to guide the process of developing and implementing a system for gathering, managing, applying and transfer the knowledge that is generated both inside an enterprise and in the relations it has with the different organisations it works with. At the same time it must also ensure the quality, security and authenticity of the knowledge supplied.

Different qualitative and quantitative methods were used to construct the methodology. In the first place, the literature related to this line of research was reviewed and the results of different projects related to Knowledge Management were analysed. In this way, a clear view and better understanding of the topic was obtained.

Information about KM was then collected through an interview and questionnaires given to owners, managers and employees of the different enterprises which collaborated in the KM-IRIS project. Once this information had been put together, analysed, processed and selected, a first version of the KM-IRIS methodology was drawn up.

Finally, the methodology was applied to a large textile enterprise in order to (1) validate and document the benefits and lessons learned in the form of a properly understandable case study, and (2) to improve the initial results by applying the conclusions extracted from those results to them.

First of all, the methodology, called KM-IRIS, was defined on a general level so that it could be used as a guide to manage knowledge in any kind of organisation that wished to do so. It was later adapted to the specific characteristics of an enterprise.

The general methodology is divided into five phases:

1. Analysis and Identification of the Target Knowledge.
2. Extraction of the Target Knowledge.
3. Classification and Representation.
4. Processing and Storage.
5. Utilisation and Continuous Improvement.

We will now describe each of the phases that go to make up the methodology in more detail, that is, the activities involved in each step, the techniques and tools that can be used to aid the process, and the main results that are expected (see Figure 1).

PHASE I. Identification

One of the aspects that usually generates most confusion in knowledge management is the difference between knowledge and information. This uncertainty is increased by the fact that knowledge management relies on information technologies for support instead of *a set of specific technologies that could be called 'knowledge technologies'*. If information and knowledge are not the same, then there seems to be something strange about the fact that knowledge can be handled using technologies that were designed for processing information.

Figure 2 attempts to unravel this paradox. From our point of view knowledge and information are different. The individual who possesses knowledge (the awareness that he or she has acquired through their training, common sense, experience, and so on) (McInerney, 2002), needs to analyse and assess information so that, in a given situation, they can make the right decisions or carry out the activities that have been proposed. In this context, the goal of the knowledge management system is to identify existing knowledge and extract, collect and codify it as information so that it can be stored and distributed using a computer system. Thus, the knowledge management system transforms the organisation's knowledge into information that will later be utilised by individuals to make better decisions or to better carry out their tasks and duties. The quantity and quality of information that is used by the individuals in the organisation to make decisions based on their knowledge therefore increases, since now it is not only produced by processing data but also comes from already existing knowledge. Moreover, the KMS helps to generate new knowledge because having more information available means that, when faced with the same situation, individuals are more likely to make a different kind of decision or to solve problems in a more efficient way, which in turn is a source of feedback for the system.

In this context, we call the organisation's knowledge that will be extracted, processed and codified in a KMS (thereby converting it into information) *target knowledge* (Grangel et al., 2006).

Therefore, the aim of this first phase of the methodology is to identify the knowledge that is going to be managed by the system, that is to say, the target knowledge. In order to identify this knowledge we need to use a pragmatic vision by directing the search towards the knowledge that is useful to the organisation and will provide an added value when utilised. To make it easier to identify in an organised fashion, it is better to begin by defining blocks of knowledge, which are understood as being any elements belonging to the organisation or to its surroundings that contain a particular type of knowledge. These conceptual blocks of knowledge are different for each type of organisation, and may even differ within the same kind of organisation, since such blocks can only be defined by taking into account the strategic objectives of the organisation and its core activities.

Once the elements of the organisation we want to know about (conceptual blocks of knowledge) have been defined, we have to identify what target knowledge will need to be extracted, represented and utilised in each of these conceptual blocks.

Finally, after identifying the knowledge in each block we must provide a detailed description of the knowledge that has been defined as target knowledge and, depending on the volume, perhaps build up an ontological classification so that it can be represented, processed and utilised at a later stage.

Valuable aids to carry out this phase include resources such as templates, questionnaires and reference models that help organisations of the same type or sector to define their conceptual blocks of knowledge, as well as to identify, describe and classify the target knowledge.

PHASE II. Extraction

The aim of this phase is to define suitable mechanisms with which to obtain the target knowledge that was identified in the previous step. To achieve this, first we must define the input variables that we are going to have to use in order to obtain the target knowledge. These input variables may be data or documents that are in the organisation's information system, that is to say, in sources of explicit knowledge, in which case they will be called *explicit input variables*. On the other hand, they might consist of information or knowledge held by people related to the organisation, that is, they lie in sources of tacit knowledge, in which case they will be termed *tacit input variables*. However, in our opinion, it will not be possible to extract and codify all tacit variables. In principle, only technical tacit variables (which refer to know-how and skills that apply to a specific context) can be documented (Day, 2005). Since it is difficult to record, process and operate with cognitive tacit variables such as beliefs or personal values using computers, they are not taken into account within the management information system that is to be developed.

Another source of variables will be the actual knowledge management system itself, since one or several of the input variables could be target knowledge that is generated by the knowledge management system that has been implemented in the organisation, and which can be used to generate new knowledge. So it must therefore be capable of providing itself with feedback.

Once the variables have been defined we must identify the sources of knowledge, which are understood to mean any components within or outside an organisation that supply those variables.

Finally, we have to define the procedure that is going to be used to extract the variables from the sources and also the method of calculation – *the algorithm* – that allows target knowledge to be obtained by combining the input variables. These procedures will vary according to the conceptual block of knowledge that is being dealt with and the input variables that have been defined (see Figure 3).

At this point it is important to draw attention to the difference between what we call conceptual blocks of knowledge and sources of knowledge. Whereas the former refers to an ontological grouping of knowledge, the latter is concerned with the starting point that will be used to extract it. For example, in the first phase of the KM-IRIS methodology an organisation might identify the conceptual block of knowledge 'customer', and from there it can specify the list of target knowledge it wishes to know about its customers. In the next phase of the methodology it will have to define how that target knowledge is going to be extracted. The extraction procedure will not have just data and information from customers as input; it will also utilise other sources of knowledge, such as employees in the organisation, the administration, and so forth. Therefore, in order to obtain the knowledge in a block, the block itself is not going to be the only element used as a source of knowledge, or the origin of that knowledge.

PHASE III. Representation

In the third phase of the methodology, after identifying and extracting the knowledge, the target knowledge will be represented in such a way as to provide us with a model of the knowledge map of the organisation (Lin & Hsueh, 2006).

In the KM-IRIS methodology, in line with the Model Driven Architecture (MDA) approach proposed by (Object Management Group, 2003), the knowledge map is represented at different levels of abstraction. Initially, a model of the knowledge map is created at the CIM (Computation Independent Model) level, that is to say, independent of the computation. Later, transformation mechanisms are used to obtain the corresponding model at the PIM (Platform Independent Model) level. Modelling of the knowledge map, both at the CIM and the PIM level, is performed by means of the set of profiles developed for this purpose using the extension mechanisms provided by the latest version 2.0 of UML (Unified Modeling Language) (Object Management Group, 2004).

The CIM model of the knowledge map must include the conceptual blocks of knowledge that have been identified within the organisation, the target knowledge of each block, their location and the way they interrelate with the other elements on the map, as well as what input variables are required to obtain them, and the procedure for calculating or obtaining them. At this level, the CIM model is aided by the use of conceptual and ontological maps as a step prior to setting out a common framework of the concepts inherent to the organisation.

The PIM model will result from the transformation of the model of the CIM level knowledge map. This phase involves determining what part of the CIM model it is worthwhile computerising and then running the previously defined transformation mechanisms.

PHASE IV. Processing

Once the PIM model of the knowledge map has been obtained, the next step is to generate an executable model for it that can be run on a certain technological platform. This model, called a PSM (Platform Specific Model) in the MDA approach, is produced as the result of processing the knowledge map on a specific computer platform to allow the enterprise to obtain and utilise the knowledge wherever and whenever it is requested.

The activities to be carried out in this phase are similar to those proposed in any other object-oriented methodology for developing a computer system, but based on the previously obtained PIM models. The final result will be a knowledge portal that shows the knowledge map of the enterprise and offers different tools with which to locate and access it.

PHASE V. Utilisation

The last phase is the utilisation of the knowledge, which involves not only making a knowledge portal available to the organisation, but also providing it with the mechanisms it needs to make efficient use of the knowledge management system that has been developed. This involves performing different types of tasks related to training, evaluation, continuous improvement and maintenance, some of the most notable of which include:

- Establishing policies and procedures to allow self-maintenance of the system (Tsai, 2003). In order to achieve this objective the knowledge portal must be integrated with the different computer systems used in the enterprise. In this way all the explicit input variables will be extracted automatically. It is also important to introduce organisational changes so that technical tacit knowledge is codified and stored in such a way as to make it automatically available from the portal. For example, templates and forms must be defined for storing know-how, skills, experience and so forth, so that what was previously kept inside people's minds, in specific documents or was jotted down on a piece of paper is now integrated within the portal.
- Establishing a system of interrelated indicators that keep us permanently informed about the status of the knowledge management system, both at a strategic and a technological and organisational level. There are a number of different KM performance measurement methods that can be used to achieve this goal and which can be classified into three types: qualitative and quantitative, financial and non-financial, and internal and external performance approaches (Liao, 2003). From a practical point of view, one of the most useful of these is the one proposed by Chen & Chen (2005), who developed a model that consists of a set of interrelated indicators to evaluate knowledge management activities from the following perspectives: knowledge creation, knowledge conversion, knowledge circulation, and knowledge execution.
- Consideration of cultural aspects to facilitate the participation and cooperation of all members of the staff at the organisation, as well as all the agents involved in the organisation's objectives, that is, interactions with customers, suppliers, administration, trade unions, and so forth.

4. ADAPTATION OF THE GENERAL METHODOLOGY TO THE PARTICULAR CASE OF AN ENTERPRISE

As far as the activities, tasks and results in each phase are concerned, the methodology described above can be applied to any type of organisation. Nevertheless, in order to make it easier to apply, specialised versions can be created by modifying the templates, questionnaires, reference models and so forth, in order to adapt them to the specific characteristics of each type of organisation. The adaptation of the general methodology to the specific case of enterprises can be seen below (see Figure 4). The methodology was applied to a large textile enterprise so as to be able to validate and refine it.

PHASE I. Identification

A set of blocks of knowledge that are sure to appear in any enterprise, and which the enterprise will need to define its target knowledge, were defined for use when the organisation is an enterprise. These conceptual blocks are: owners, suppliers and customers, employees, administration and trade unions, organisation, product or service, process and resource. The target knowledge we seek to know was identified for each of these blocks and grouped in different ontological categories (Newman, 2000).

PHASE II. Extraction

The variables used to obtain the target knowledge that was previously identified, as well as the sources of tacit and explicit knowledge, were determined in this phase. The more notable explicit sources include databases, document databases, and business intelligence information systems, data warehousing, OLAP systems and data mining information systems. Tacit sources of knowledge are to found in the personnel that collaborate with the enterprise (customers, employees, suppliers, and so forth), as well as in organisations such as trade unions, business associations, and so forth. Lastly, the extraction and calculation procedures were defined for each item of target knowledge.

Table 1 shows an example of the results obtained in Phase I and Phase II of the KM-IRIS methodology after tailoring it for knowledge management in an enterprise. Employee and process deal with tacit sources of knowledge, and customer and product are concerned with explicit sources.

PHASE III. Representation

In order to facilitate the creation of the knowledge map for an enterprise, the KM-IRIS methodology includes a reference model that represents the target knowledge that is to be managed within a typical enterprise. Two aspects were taken into account during the development of this model. The first involved the use of ontologies (Holsapple & Joshi, 2004) as a way to provide a common basis for understanding throughout the whole enterprise, while the second considered the utilisation of the MDA approach and UML to obtain a visual representation of the map of enterprise knowledge that can be turned into an executable model.

Thus, in building the reference model of the knowledge map a new business ontology was defined that took into account (1) the different business concepts explained in Bertolazzi (2001); (2) the different conceptual blocks of knowledge proposed in phase I of the KM-IRIS methodology; and (3) the different dimensions defined within the context of the modelling of the business so as to provide a holistic representation of the enterprise – business, organisation, process, product and resource.

This generic business ontology can also be used so that any enterprise may tailor it to its own domain according to the target knowledge it identifies.

The MDA approach proposed by the OMG (Object Management Group, 2003) was also used to develop a graphic model of the knowledge map at both the CIM and PIM levels which, in the fourth phase, can be transformed into the corresponding PSM. UML was used as the modelling language in the creation of the models, since it has become a commonly accepted standard for the object-oriented modelling of all kinds of systems. However, because UML is somewhat limited as a business modelling language, we took advantage of the new capabilities offered by UML 2.0 and used the profiles mechanism to extend the UML metamodel to the specific domain of enterprise knowledge. A profile was therefore defined in UML 2.0 that allowed the enterprise knowledge to be modelled in different views that took into account both the previously defined generic business model and the conceptual blocks of knowledge and target knowledge specified in earlier phases.

Figure 5 shows the conceptual diagram that was followed to elaborate the reference model of the map of enterprise knowledge at the CIM level, which represents the target knowledge that is to be managed in a typical enterprise and will later be used as a reference model in the development of the knowledge map of a particular enterprise.

In Figure 5 it can be seen how the generic business ontology is taken as the starting point to establish the views needed to configure the map of enterprise knowledge in accordance with the conceptual blocks of knowledge and target knowledge that were identified at an earlier stage. Each of these views represents a specific conceptual block of knowledge that has been determined within the enterprise and it is linked to its corresponding ontological category. Thus, for example, the product view includes all the knowledge requirements set out in the earlier phases in terms of the products and services of the enterprise. Knowledge about these is represented in terms of facts, rules and attitudes, and is modelled according to the UML 2.0 profile that was developed. In addition, the graphic model of each view offers access to different levels of detail and is connected to the other business views that are linked by means of the different ontological categories.

PHASE IV. Processing

In this phase, the PIM models obtained in the previous phase were taken as the basis to design an information system that enables an enterprise to process, store and present the map of enterprise knowledge in a suitable manner and depending on the user's access privileges, as well as to generate new knowledge (Sutton, 2005).

The computer system is organised around a knowledge portal, understood as being a computer solution that makes it possible to extract and process the information variables from the different sources of knowledge, and to generate and integrate the target knowledge required by the enterprise. Thus, the portal will enable us to gather knowledge generated about: the different collaborations, projects/works on the way, different activities, different ways of going about things, and the results that are gradually obtained, together with recommendations and both formal and non-formal *best practices*.

The corporate knowledge portal is built upon a technological infrastructure based on the intelligent integration of technological and functional components that allow a connection to be established among the following systems:

- FrontSide: WebServices interfaces in each one of the applications designed for corporate management and for each of the conceptual blocks of knowledge: Customers/sales, Suppliers/purchases and the supply chain, Employees and Owners of the member enterprises (internal relationships), Administration, Trade Unions and Business District (collaborations/external actions).
- Business BackSide: financial, logistic, warehouses, accounting, human resources, and so forth.
- Knowledge Management BackSide.

Thus, using the Internet as a means of interconnection together with other technologies for presentation and the interface, the knowledge portal will be the end point of the computer system supporting the knowledge management system within an enterprise (see Figure 6).

Consequently, when designing the knowledge portal, the following technologies must be integrated in a suitable and efficient manner:

- An **Intranet** that makes it possible to implement and integrate the different applications for internal knowledge management, as well as to obtain the target knowledge of the remaining conceptual blocks from internal sources within the enterprise.
- An **Extranet** for managing knowledge about both business (customers and suppliers) and the surrounding environment, that is to say, the administration, trade unions and the business district

itself. It will also be used for extracting part of the employees' and owners' target knowledge from these external sources so that it can be stored in the internal backside knowledge repository.

- An infrastructure consisting in networks and communications within the enterprise, in addition to the systems of control and management of access and authorisation that give rise to the different internal or external sub-portals, as well as endowing them with a suitable degree of security depending on the roles and user profiles that are defined.
- ERP (Enterprise Resource Planning), CRM (Customer Relationship Management) and SCM (Supply Chain Management) for managing business knowledge that will provide useful information for generating new knowledge on the Knowledge Management Intranet (Chalmeta, 2006).
- Workflow tools to control workflow and Groupware as a support for collaboration (Deek, 2003; Ellis, 1991).
- Data Warehousing, business intelligence and other decision support tools, which allow feedback and recommendations from the organisation's broad fundamental experience and from the knowledge stored in the backside knowledge repository to be incorporated into decision-making (Chalmeta & Grangel, 2005).
- Other software applications such as Document management systems allow, among other things, information fixed on some kind of support to be searched swiftly and according to different criteria. At the same time they also make it possible to keep track of versions, control access by levels of security and, finally, avoid redundancy in the documents that are stored.

PHASE V. Utilisation

Although proper utilisation of knowledge management shares a number of common features regardless of the type of organisation in which it is applied (it is based on training, evaluation, continuous improvement and system maintenance), when the organisation is an enterprise the following specific aspects, among others, concerning the utilisation of knowledge management must also be taken into account:

- Cultural aspects to facilitate the participation and cooperation in the system of all the employees and owners, in addition to all the agents involved in the organisation's business operations, the most important of which are its customers, suppliers, administration and trade unions.
- Consider training in this area as part of the strategic investment of the enterprise, like plants and equipment; ranking it as a vital component in the construction of competitiveness.
- Guarantee the entire workforce the right to benefit both collectively and individually from the cognitive enrichment that arises from well-channelled and controlled transfers, and prevent any kind of monopolistic use of knowledge from being carried out by individuals who are driven by purely personal, vested interests.
- Insist positively on interdepartmental interaction by making it possible for the departments in the enterprise to transfer their own explicit knowledge, so that by contrasting it they can also enrich it and complete it to the extent that the increase in efficiency and effectiveness of such transfers contributes to the resolution of management problems in each of the departments.
- Solve the problem of Property rights, by recognising the exclusive property rights of the knowledge held by the employee, according to the personal effort they make in carrying out their duties and the economic cost they had to pay, before they were taken on by the enterprise, in order to achieve the cognitive foundations that allowed them to later become part of it.

5. A CASE EXAMPLE

The KM-IRIS methodology was applied to a large textile enterprise. The procedure adopted for the application of the KM-IRIS methodology was as follows. First, a presentation was given at the enterprise so that management staff could see the aims of the knowledge management project. This was accompanied by an explanation of the KM-IRIS methodology, which was to be used to guide the execution of the project. During the presentation it was shown that the methodology has a staged structure and that it includes predefined extraction and calculation procedures, as well as clearly defined tasks and reference models of the target knowledge in a typical enterprise that would only need to be compared with the requirements of this company. These characteristics enabled the directors at the firm to quickly understand (1) the scope of the project; (2) the benefits that it was going to offer them; (3) the activities they would have to collaborate in; (4) the resources that would have to be assigned; and (5) the impact that the project would have on the enterprise. They were therefore already avoiding some of the main causes of failure when implementing KMS.

The enterprise set up a committee that was responsible for decision-making related to the project. This committee was made up of the information systems manager, the quality control manager, the logistics manager and the person in charge of communication and advertising. Other participants in the actual execution of the project included the managers from each department, members of staff from the computer department and, from time to time and as required, other members of the operating staff at the firm.

It is interesting to note that each of the members of the committee identified the benefits of the project according to his or her own background. For example, the information systems manager was the first to realise that the KMS was going to lead to the integration of the firm's computer systems. The enterprise had many corporate IT systems that were heterogeneous and not integrated with one another. Each of them was used to meet different needs in specific areas or aspects of the firm, such as ERP, CRM or SCM. Yet, these systems did not offer the organisation what it was looking for, that is to say, homogeneity, interoperability, easy access and knowledge of its possibilities throughout the different departments in order to prevent duplication of information, data, etc. The decision to implement a new system centred on the knowledge portal, which was the entrance to all the knowledge in the organisation, was to be the factor that integrates the different technological solutions within the firm. On the other hand, the head of communications, saw the portal as the ideal place to centralise all the useful knowledge the firm possessed regarding marketing, internal regulations, public news about the firm and its competitors, and so forth; and at the same it could also be used to disseminate such knowledge among employees, customers, suppliers and other collaborators.

The project was actually carried out following the steps set out in the KM-IRIS methodology. First, the reference model was compared with the real situation of the textile enterprise so as to allow for definition of the target knowledge they wished to manage. The most significant changes were the addition of a new conceptual block (the vision of the enterprise from outside) and incorporating, eliminating or renaming the predefined target knowledge.

Once the target knowledge had been defined, the extraction and calculation procedure for each item of target knowledge was identified, together with the sources they could be obtained from. Explicit sources refer to the firm's IT system, which in this case consisted of the transactional computer system (ERP, CRM, specific logistics systems, etc.), the data warehouse, which provided reports and management control indicators, and the documentary information system. Tacit sources refer to persons and in order to extract their knowledge we drew up a number of surveys (for example, concerning the organisational climate and culture, employees' motivation and satisfaction, training needs), forms (for example, for actions deriving from a claim made by a customer; hence, from now on these are no longer contained in a person's experience, on a piece of paper or in an isolated document written out on the computer, but will instead be stored in a computer system), and collaborative tools.

All the results thus obtained were then recorded and used to generate a map of the knowledge of the enterprise. To do so the methods of representation defined in the KM-IRIS methodology were used.

The next stage was to start to develop the technological solution. This takes the form of a knowledge portal that can be accessed by the firm's collaborators. From a functional point of view, the portal is divided into five areas. One area allows access to the different blocks of knowledge the firm has. Another one is a search engine that allows us to find the target knowledge when we do not know the exact route. The search engine indexes not only the contents of this portal but also those from other external sources such as the corporate websites of customers, suppliers or competitors. A third area of the portal concerns collaborative environments, where members of staff from different departments can work on joint projects. The fourth area includes news related to the enterprise. Lastly, the fifth area is for administering the portal (definition of profiles, contents, services, configuration, etc.). From a technological point of view, the portal is connected to all the computer systems in the firm, so that it can extract the explicit input variables, and also to the forms, surveys, and so forth, to enable it to extract the tacit input variables.

Finally, the implementation of the knowledge management system was carried out. The first step was to invite the top management staff at the firm to a presentation and to present the project publicly in the press (the enterprise thought that its having this sort of knowledge management system would enhance its image as an innovating firm). The next stage was to train users and they are currently running the system.

As well as improving the methodology as a result of applying it to different companies, the potential for developing research in this area has been proved and a series of lessons have been learned:

- In order for enterprises to integrate knowledge management effectively with all their existing business processes, both management and employees must understand and assimilate the strategic business value of knowledge management. These key participants must understand that knowledge management is not simply a technological strategy, but rather an essential business strategy for the success of their individual departments and of the organisation as a whole.
- The knowledge-oriented business model is seldom practised and poorly known, whether it be at the operational or management level.
- Limitations concerning the systemic vision of knowledge management. This behaviour is the result of historical factors that conditioned people and companies not to share knowledge.
- The need for more scientific production showing knowledge management KM methodologies and business experiences. As (Blair, 2002) says, experts learn from case studies.

- The need to encourage the training of staff in knowledge management. It has been shown that staff training programmes do not include the participation of employees in courses or other types of events related to knowledge management.

All these difficulties are related to the low level of awareness of the importance of knowledge management and, therefore, of the benefits that proper knowledge management can generate.

6. CONCLUSIONS

To successfully carry out a project aimed at developing and implementing a knowledge management system, it is essential to have a step-by-step methodology that directs the development and implementation processes. However, existing methodologies for developing computer systems are not oriented towards the specific problems arising in this type of systems.

Within this framework, this paper has offered a description of a methodology obtained as the result of the KM-IRIS project. This methodology guides the process of developing and implementing a knowledge management system that allows knowledge to be collected, managed and applied, while ensuring the quality, security and authenticity of the knowledge provided. The methodology was first presented on a general level so that it could be used as a guide to manage knowledge in any kind of organisation that wished to do so. It was then adapted to the specific characteristics of an enterprise.

As a result, the practitioners who follow this specific methodology for developing a KMS in an enterprise will benefit from a series of advantages, including the following:

- a better definition of the vision and strategy of the project, because those in charge in the organisations in which the KMS is to be implemented will be in a better position to understand the scope and consequences of the project, as well as the important opportunities that can be obtained by having a knowledge management system. This stems from the fact that they will have an initial reference model of the typical target knowledge of an enterprise with a specific modelling language for representing the knowledge map of a company in a graphic and intuitive manner. In this way the definition of the knowledge requirements will fit the needs of the organisation better.

- better planning and management of the project, because, for example, the phases, tasks, outcomes, techniques and documents to be used in each of them are all clearly defined.
- a separation of the needs of the organisation from the technical solution, since this is only taken into account after the organisational aspects have been perfectly well defined

Acknowledgements

This work was funded by DPI2006-14708, CICYT DPI2003-02515 and European Commission within the 6th Framework Programme (INTEROP Network of Excellence, IST-2003-508011).

References

Abdullah, M.S., Benest, I., Evans, A., & Kimble, C. (2002) Knowledge Modelling Techniques for Developing Knowledge Management Systems. In Proceedings of the 3rd European Conference on Knowledge Management (pp. 15 –25). Dublin.

Alavi, M., & Leidner, D. E. (2001). Knowledge Management and Knowledge Management Systems: Conceptual Foundations and Research issues. Management Information Systems Quarterly, 25(1), 107-136.

Avison, D., & Fitzgerald, G. (Ed.) (2006) Information Systems Development Methodologies, Techniques & Tools, fourth edition, McGraw-Hill Education, Maidenhead.

Bertolazzi, P., Krusich, C., & Missikoff M. (2001). An Approach to the Definition of a Core Enterprise Ontology: CEO. In A. D'Atri, & M. Missikoff, (Eds.), Proceedings of the International Workshop on Open Enterprise Solutions: Systems, Experiences, and Organizations (OES-SEO 2001) (pp. 14-15). Rome: Luiss Publications.

Blair, D.C. (2002). Knowledge management: Hype, hope, or help?. Journal of the American Society for Information Science and Technology, 53(12), 1019-1028.

Chalmeta, R. (2006). Methodology for customer relationship management. Journal of Systems and Software, 79(7), 1015-1024.

Chalmeta, R., & Grangel, R. (2005). Performance measurement systems for virtual enterprise integration. Journal of Computer Integrated Manufacturing. 18, 73-84.

Chiu, C.-H., Hsu, M.-H., & Wang, E.T.G. (2006). Understanding knowledge sharing in virtual communities: An integration of social capital and social cognitive theories. Decision Support Systems, 42(3), 1872-1888.

Chen, A., & Chen, M. (2005). Measurement Practices for Knowledge Management: An Option Perspective. Lecture Notes of Computer Science, 3520, 387-399.

Day, R.E. (2001). Totality and representation: A history of knowledge management through European documentation, critical modernity, and post-Fordism. Journal of the American Society for Information Science and Technology, 52(9), 725-735.

Day, R.E. (2005). Clearing up implicit knowledge: Implications for Knowledge Management, information science, psychology, and social epistemology. Journal of the American Society for Information Science and Technology, 56(6), 630-635.

Deek F.P., & McHugh J.A.M. (Ed.). (2003). Computer-Supported Collaboration with Applications to Software Development. Norwell: Kluwer Academic Publishers.

Devedzic, V. (1999). A survey of modern knowledge modeling techniques. Expert Systems with Applications, 17(4), 275–294.

Dretske, F.I. (Ed.). (1981). Knowledge and the Flow of Information. Cambridge, MA: MIT Press.

Eliasson, G. (2005). The nature of economic change and management in a new knowledge based information economy. Information Economics and Policy, 17(4), 428-456.

Ellis, C.A., Gibbs, S.J., & Rein, G.L. (1991). Groupware: Some issues and experiences. Communications of the ACM, 34(1), 38-58.

Ernst & Young Center for Business Innovation and Business Intelligence. (2001). An Executive Perspectives on Knowledge in the Organization. Working Paper.

Grangel, R., Chalmeta, R., & Campos, C. (2006). Definition of Target Knowledge in Virtual Enterprises. In W. Abramowicz, & H.C. Mayr (Eds.), Proceedings of the 9th International Conference on Business Information Systems (BIS 2006) (pp. 256-266). Klagenfurt: Lecture Notes in Informatics.

Gray, P.H., & Meister, D. B. (2006). Knowledge sourcing methods. Information & Management, 43(2), 142-156.

Hall, H. (2003). Applying exchange theories in information science research. Library & Information Science Research, 25(3), 287-306.

Heinrichs, J.H., & Lim, J.-S. (2005). Model for organizational knowledge creation and strategic use of information. Journal of the American Society for Information Science and Technology, 56(6), 620-629.

Holsapple, C.W., & Joshi, K.D. (2004). A formal knowledge management ontology: Conduct, activities, resources, and influences. Journal of the American Society for Information Science and Technology, 55(7), 593-612.

Inmon, B. (Ed.). (1996). Building the data warehouse. New York: John Wiley & Sons.

Jayaratna, N., (1994). Understanding and Evaluating Methodologies, NISAD: A Systematic Framework, McGraw-Hill, Maidenhead

Ke, W., & Wei, K.K. (2005). Critical factors affecting the firm to share knowledge with trading partners: a comparative exploratory case study. In Q. Li, & T.-P. Liang (Eds.), Proceedings of the 7th International Conference on Electronic Commerce (ICEC 2005) (pp. 177-183). Xi'an: ACM.

King, W.R., & Marks, P.V.Jr., (2006) Motivating knowledge sharing through a knowledge management system, Omega, In Press, Corrected Proof, Available online 8 February 2006.

Liao, S. (2003). Knowledge management technologies and applications - literature review from 1995 to 2002. Expert Systems with Applications, 25(2), 155-164.

Lin, F., & Hsueh, C. (2006). Knowledge map creation and maintenance for virtual communities of practice. Information Processing & Management, 42(2), 551-568.

Lindvall, M., Rus, I., & Sinha, S. (2003). Technology Support for Knowledge Management. Lecture Notes in Computer Science. 2640, 94-103.

McInerney, C., & Day, R. (2002). Introduction to the JASIST special issue on Knowledge Management. Journal of the American Society for Information Science and Technology, 53(12), 1008-1009.

McInerney, (2002). Knowledge Management and the Dynamic Nature of Knowledge. Journal of the American Society for Information Science and Technology, 53(12), 1009-1019.

Meso, P., Madey, G., Troutt, M., Liegle, J., (2006). The knowledge management efficacy of matching information systems development methodologies with application characteristics—an experimental study. Journal of Systems and Software, 79(1), 15-28

Myers, P.S. (1996). Knowledge Management and Organizational Design: An Introduction. In P. S. Myers (Ed.) Knowledge Management and Organizational Design (pp 1-6). Boston: Butterworth-Heinemann.

Newman, B.D., & Conrad, K.W. (2000). A Framework for Characterizing Knowledge Management Methods, Practices, and Technologies. In U. Reimer (Ed.) Proceedings of the Third Int. Conf. on Practical Aspects of Knowledge Management (PAKM2000). Basel.

Nonaka, I., & Takeuchi, H. (Eds.). (1995). The Knowledge Creating Company: How Japanese Companies Create the Dynamics of Innovation. New York: Oxford University Press.

Object Management Group. (2003). MDA Guide Version 1.0.1, document number: omg/03-06-01 edition. Retrieved November 29,2006 from http://www.omg.org/cgi-bin/doc?omg/03-06-01

Object Management Group. (2004). Unified Modeling Language (UML) Specification: Superstructure, version 2.0., document: formal/05-07-04 edition. Retrieved November 29, 2006 from http://www.omg.org/technology/documents/formal/uml.htm

O'Leary, D.E., Kuokka, D., & Plant, R. (1997). Artificial Intelligence and Virtual Organizations. Communications of the ACM, 40(1), 52-59.

O'Leary, D.E. (1998). Guest Editor's Introduction: Knowledge-Management Systems-Converting and Connecting. IEEE Intelligent Systems, 13(3), 30-33.

Revilla, E., Acosta, J., & Sarkis, J. (2005). Value perceptions and performance of research joint ventures: An organizational learning perspective. The Journal of High Technology Management Research, 16(2), 157-172.

Schütt, P. (2003). The post-Nonaka Knowledge Management. Journal of Universal Computer Science, 9(6), 451-462.

Shin, M., Holden, T., & Schmidt, R.A. (2001). From knowledge theory to management practice: towards an integrated approach. Information Processing & Management, 37(2), 335-355.

Snowden, D. (2002). Complex acts of knowing: paradox and descriptive self-awareness. Journal of Knowledge Management, 6(2), 100-111.

Sutton, M. (2005). Review of "Integrated document and content management: Strategies for exploiting enterprise knowledge" by Len Asprey and Michael Middleton. Information Processing & Management, 41(5), 1299-1301.

Tsai, B. (2003). Information landscaping: information mapping, charting, querying and reporting techniques for total quality knowledge management. Information Processing & Management, 39(4), 639-664.

Viswanathan M., Yang, Y. K., Whangbo, T. K., Kim, N. B., & Garner B. (2005). Knowledge-Based Compliance Management Systems - Methodology and Implementation. Proceedings of the Fourth Annual ACIS International Conference on Computer and Information Science (ICIS'05).

Walters, S.A., Broady J.E., & Hartley, R.J. (1994). A Review of Information Systems Development Methodologies. Library Management, 15(6) 5-19

Yadav, S, Shaw, N. Webb, L & Sutcu, C. (2001). Comments on "Factors that Impact Implementing a System Development Methodology". IEEE Transactions on Software Engineering, 27(3) 279-281

PHASE I	**Conceptual Block**	EMPLOYEE	PROCESS	CUSTOMER	PRODUCT
	Ontological Category	Satisfaction	Sales	Profit	Cost
	Target Knowledge	Economic Satisfaction	Receive an order	Economic profitability Customers	Economic profitability of Product
	Description	Extent to which the employee is satisfied with the salary he or she is paid	Best practices in accepting orders	Classification of customers according to their economic profitability	Classification of customers according to their economic profitability
PHASE II	**Input Variables**	• Opinion about employees and immediate bosses • Average salary in the sector	• Information (Documents + Data) that is needed or generated to carry out the task, and identification of its origin or destination • Human and technological resources that are involved • Controls or associated regulations	• Annual sales turnover • Average price of products acquired • Average quality of products acquired • Number of claims lodged • Average length of payment period in days • Customer's	• Average cost of the raw materials and labour used to manufacture the product • Average profit obtained from sale of the product • Average cost assigned to the product as advertising costs • Average cost deriving from financial expenses arising from marketing the product

	Knowledge Source	Employee Consultancy firms, business associations, trade unions in the business sector	Employee	Databases and document databases Data Warehouses	Databases and document databases Data Warehouses
	Calculation Procedure	Statistical calculation	Detailed description of the procedure for running the task using the IDEF0 modelling language	Statistical calculation	Statistical calculation
	Extraction Procedure	Questionnaires and personal enquiries	Templates for defining profiles of work positions drawn up by the IRIS group	ETL, OLT and OLAP techniques Data Mining techniques	ETL, OLT and OLAP techniques Data Mining techniques

Table 1. Example of Phases I and II of the KM-IRIS methodology after tailoring it to the needs of an enterprise.

METHODOLOGY FOR THE IMPLEMENTATION OF KNOWLEDGE MANAGEMENT SYSTEMS

Ricardo Chalmeta[a] (Corresponding author), Reyes Grangel [a]

[a] Grupo de Investigación en Integración y Re-Ingeniería de Sistemas (IRIS).

Universitat Jaume I. 12006 Castellón. Spain.

Tel:+ 34 964 728329 Fax: + 34 964 728435

{rchalmet, grangel}@uji.es

PHASES	ACTIVITIES	TECHNIQUES	EXPECTED RESULTS	COMPUTER SUPPORT TOOLS
PHASE I. Identification	• Identify the conceptual blocks of knowledge • Classify into ontological categories • Define the target knowledge (knowledge requirements)	• Templates and questionnaires to identify blocks of knowledge • Reference models concerning the target knowledge	• Conceptual blocks of knowledge • Target knowledge • Categories	• Office automation tools • Modelling tools
PHASE II. Extraction	• Extract knowledge from sources in order to define the input variables and categorise it • Define the extraction and calculation procedures	• Templates to define the input variables • Reference models for extracting and calculating target knowledge	• Set of input variables • Extraction and calculation procedures	• Office automation tools • Modelling tools
PHASE III. Representation	• Establish the relations within the target knowledge • Draw up the knowledge map	• Metamodelling (UML) • Ontologies • Conceptual maps	• Model of the Knowledge map	• Modelling tools • Ontology engineering tools
PHASE IV. Processing	• Develop the technological infrastructure supporting the knowledge map by following an object-oriented methodology for the development of computer systems	• BPM techniques • ETL techniques •Document/DBMS • Data warehouse • OLAP • Data mining	• Knowledge portal (Executable knowledge map)	• BPM tools • ETL tools • Document/DBMS • Data warehouse • OLAP • Data mining
PHASE V. Utilisation	• Establish training and continuous improvement mechanisms among the members of the organisation • Carry out maintenance and the feedback process on the knowledge management system	• e-Learning • Groupware • TQM • ISO standard of quality	• Efficient use of knowledge within the organisation	• Office automation tools • Modelling tools • Learning tools

Fig. 1. KM-IRIS methodology for knowledge management in an organisation.

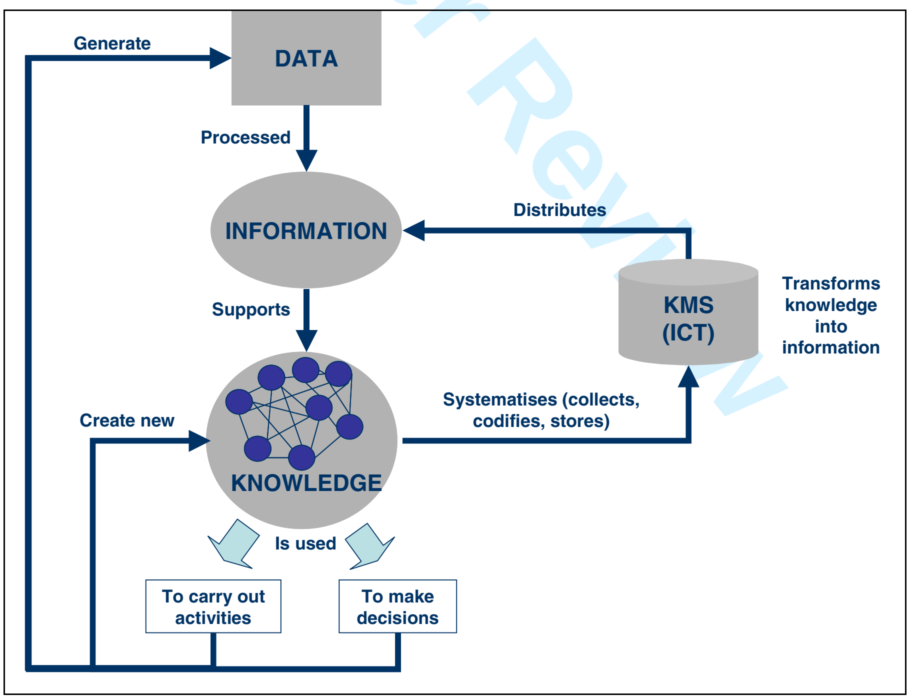


Fig. 2. KMS relation with information and knowledge.

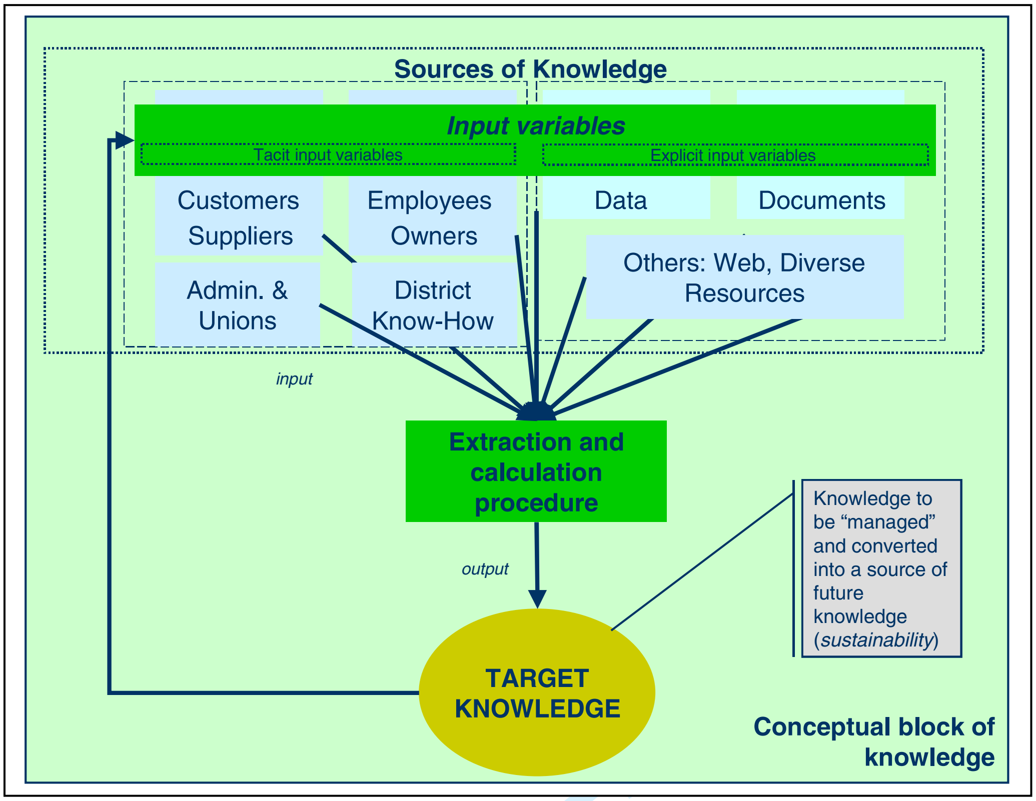


Fig. 3. Phase II of the KM-IRIS methodology for knowledge management.

4

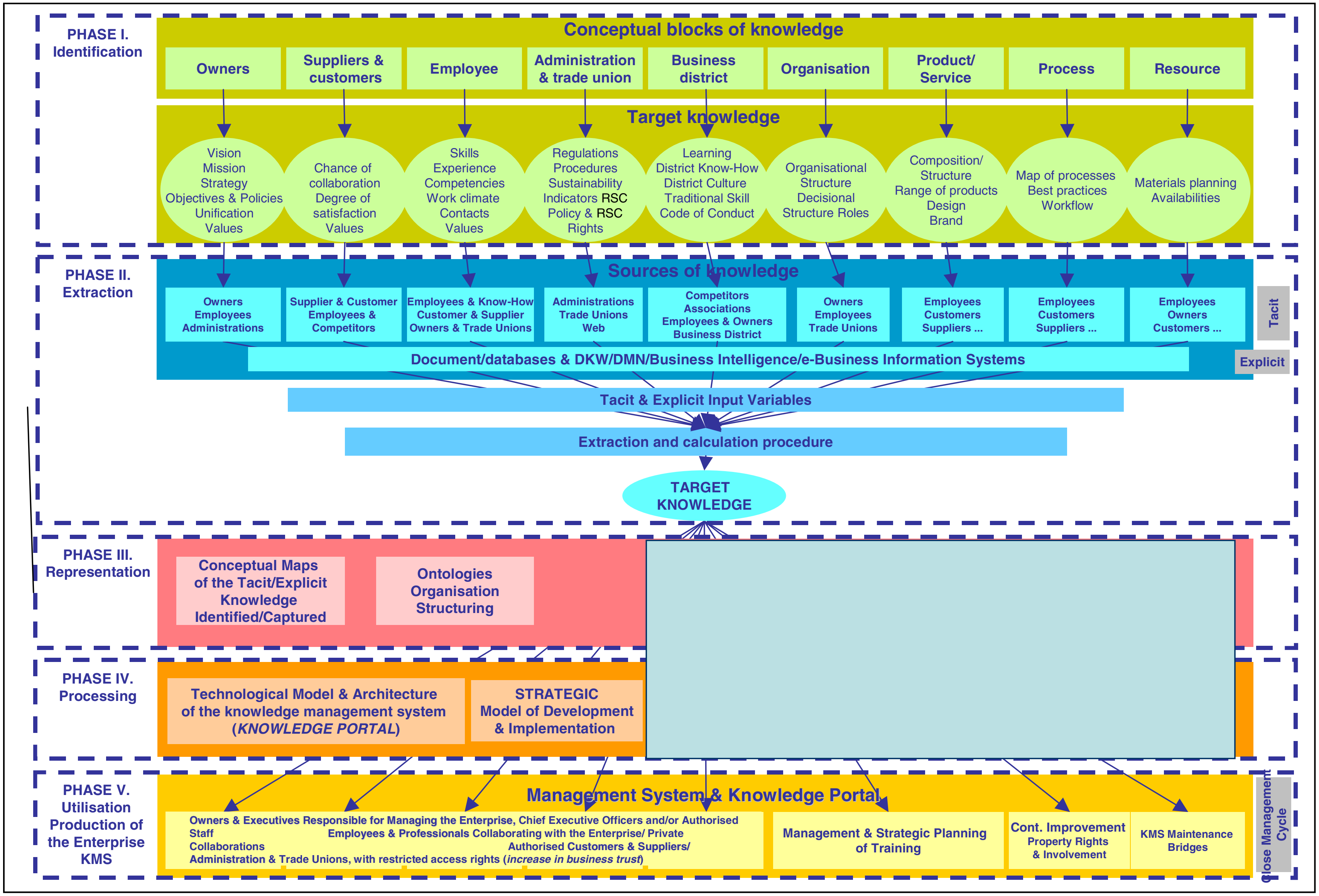


Fig. 4. Specialised version of the KM-IRIS methodology for knowledge management in an enterprise

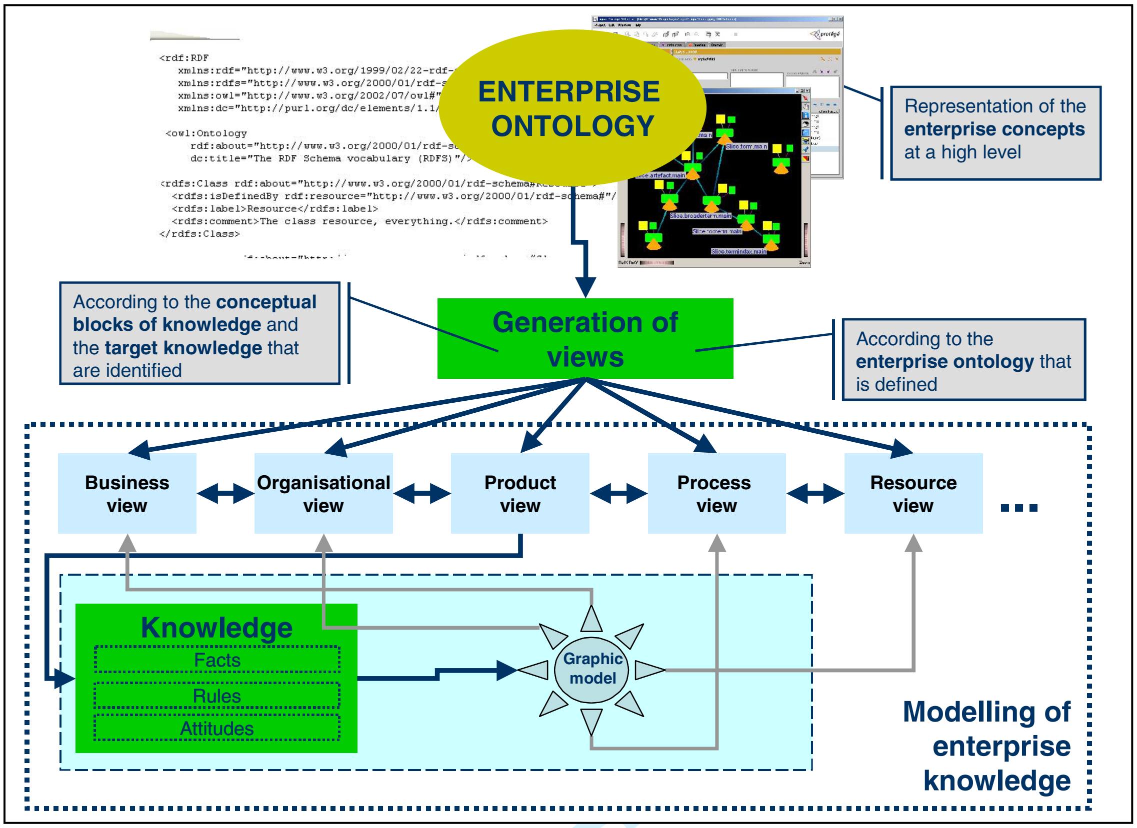


Fig. 5. Conceptual diagram for obtaining the map of enterprise knowledge at the CIM level.

6

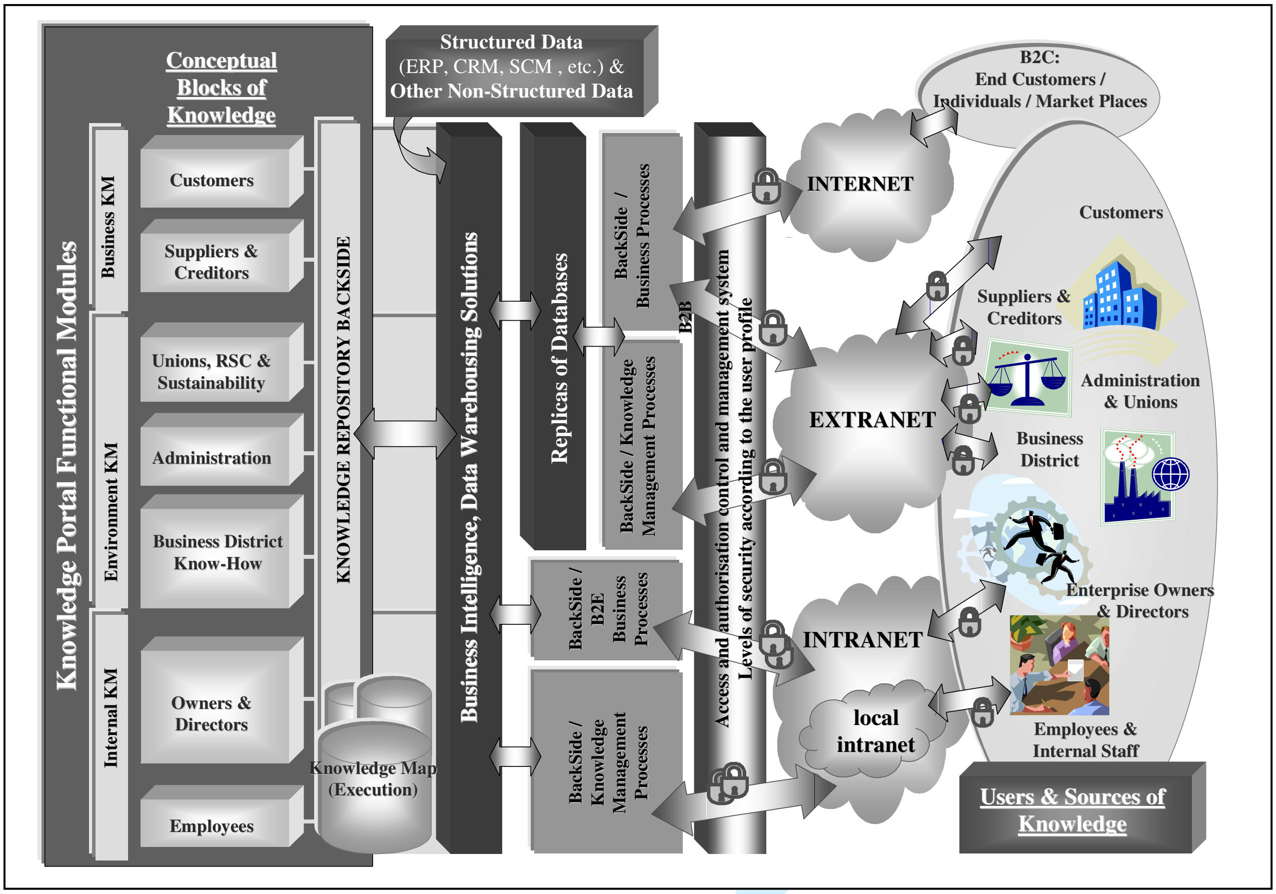


Fig. 6. Technological infrastructure proposed to support a knowledge portal.

Chapter 5

Model Driven Knowledge Proposal

Title:	**Definition of Target Knowledge in Virtual Enterprises**
Authors:	R. Grangel and R. Chalmeta and C. Campos
Conference:	UNISCON 2006 - 9th International Conference on Business Information Systems (BIS 2006)
Book title:	Business Information Systems - 9th International Conference on Business Information Systems (BIS 2006), W. Abramowicz, W.C. Mayr (Eds.)
Publisher:	Gesellschaft für Informatik (GI) (Bonn)
Pages/Year:	256-266/2006
ISBN:	3-88579-179-X
Place:	Klagenfurt (Austria)
Date:	31-5-2006 to 2-6-2006

Abstract

A virtual enterprise is a new organisational paradigm which requires novel technological approaches to managing data, information and knowledge in an efficient way. Knowledge management systems have been adopted as a solution to deal with enterprise systems which generate a huge amount of data, and also need to manage information in an appropriate manner and to share knowledge for decision-making. However, these systems are even more essential in the context of virtual enterprises, where business success is based on interoperability achieved by means of ICT, and therefore there is a need for a common conceptual framework that enables partners in the virtual enterprise to share data, information, and knowledge.

Thus, the implementation of this kind of technologies in virtual enterprises demands new, more specific requirements. In this paper, we propose a conceptual framework that introduces the concept of target knowledge as a first step for implementing efficient knowledge management systems, and for further knowledge representation in the context of virtual enterprises. Finally, a classification of target knowledge defined taking into account several enterprise dimensions is provided.
Keywords. Knowledge Representation, Knowledge Management Systems, Enterprise Knowledge, Enterprise Modelling.

Definition of Target Knowledge in Virtual Enterprises

Reyes Grangel, Ricardo Chalmeta, and Cristina Campos

Grupo de Investigación en Integración y Re-Ingeniería de Sistemas (IRIS), Dept. de Llenguatges i Sistemes Informàtics, Universitat Jaume I, Campus del Riu Sec s/n, 12071 Castelló, Spain
{grangel, rchalmet, camposc}@uji.es

Abstract. A virtual enterprise is a new organisational paradigm which requires novel technological approaches to managing data, information and knowledge in an efficient way. Knowledge management systems have been adopted as a solution to deal with enterprise systems which generate a huge amount of data, and also need to manage information in an appropriate manner and to share knowledge for decision-making. However, these systems are even more essential in the context of virtual enterprises, where business success is based on interoperability achieved by means of ICT, and therefore there is a need for a common conceptual framework that enables partners in the virtual enterprise to share data, information, and knowledge.
Thus, the implementation of this kind of technologies in virtual enterprises demands new, more specific requirements. In this paper, we propose a conceptual framework that introduces the concept of target knowledge as a first step for implementing efficient knowledge management systems, and for further knowledge representation in the context of virtual enterprises. Finally, a classification of target knowledge defined taking into account several enterprise dimensions is provided.
Keywords. Knowledge Representation, Knowledge Management Systems, Enterprise Knowledge, Enterprise Modelling.

1 Introduction

The global economy, customer orientation and the swift evolution of Information and Communication Technologies (ICT) are some of the factors that have produced a new economic scenario, where information and knowledge have became strategic resources for enterprises [1]. The virtual enterprise arises in this context as a new organisational paradigm in which valuable cooperation can be established among partners in order to exploit competitive advantages by sharing resources, skills and costs, and by establishing a new model of interoperability [2].

A virtual enterprise is a network of independent enterprises, often competitors, that form a temporary alliance with the aim of developing a product or service so as to be able to take advantage of new market opportunities and to make it easier to achieve their objectives by sharing resources and costs [3].

Traditional knowledge management systems have been introduced by enterprises as a good solution to enable them to share and distribute knowledge among their employees [4–6]. Nevertheless, in the context of virtual enterprises, where several partners with different procedures, methods, rules, culture and so on are integrated within a single virtual enterprise, the implementation of a knowledge management system is a far more complex task and it cannot be developed only by applying technological issues. Thus, a common conceptual framework that enables partners in a virtual enterprise to represent and to share data, information, and knowledge is needed before establishing a knowledge management system.

Such a framework should be focused on a holistic point of view of the enterprise and it is the basis for providing a common understanding about business for the partners that make up a virtual enterprise. In this paper, we present a set of knowledge requirements, called target knowledge, that are needed to develop this kind of systems. They are related to the KM-IRIS methodology [7], which has been developed by the IRIS Group in order to implement knowledge management systems in virtual enterprises.

The paper is organised as follows. Section 2 shows a review of the concepts related to knowledge framework and states the problems related to knowledge management systems within the context of the virtual enterprise. The knowledge management approach developed by the IRIS Group is briefly presented in section 3, as the framework in which the target knowledge is proposed. Section 4 describes the target knowledge defined within this approach, as well as the classification and analysis performed about it. Finally, section 5 outlines the conclusions.

2 Knowledge Perspective

The concept of knowledge has been defined from very different points of view, but in the field of enterprise information systems it has been usually linked to the concepts of data and information. In this section, we present some definitions of knowledge in order to provide a characterisation of enterprise knowledge as the basis for defining the target knowledge that is needed to implement knowledge management systems in the context of virtual enterprises. Moreover, a brief review of knowledge management systems, as well as the problems concerning the virtual enterprise are also detailed.

2.1 The Concept of Enterprise Knowledge

Data become information when they add value to the enterprise, and information becomes knowledge when insight, abstraction and a better understanding are added to it. Thus, knowledge can be defined as the capacity for effective action in a domain of human actions [8].

On the other hand, Nonaka [9] defines knowledge as the justified belief that increases the capacity of an entity for effective action. The conventional creation of knowledge is usually performed following the model in which data are transformed into information, and information is transformed into knowledge, but it can also follow the reverse model in which knowledge precedes information and data [8]. As a result, knowledge can be represented by means of links among data, information and knowledge inside a system, but other data, information and knowledge can also come from outside the system through other connections.

The process of converting this knowledge from the sources available to an organisation and then connecting people with that knowledge is one of the definitions provided to explain knowledge management [10, 11]. Furthermore, knowledge management facilitates creation, access and reuse of knowledge, typically by using advanced technology, such as the World Wide Web, Lotus Notes, the Internet and intranets [12].

According to [13] enterprise knowledge can been seen as information made actionable in a way that adds value to the enterprise. Taking into account this context, we defined enterprise knowledge as the network of connections among data and information that enables people involved in the enterprise to act and to make decisions that add value to the enterprise. Moreover, two dimensions can be defined in enterprise knowledge, explicit and tacit, following the current interpretation [5] that defines a fuzzy borderline between explicit and tacit knowledge.

2.2 Knowledge Representation

A knowledge management system is a specialised system that interacts with the organisation's systems to facilitate all aspects of knowledge engineering [4]. The benefits of Knowledge Management Systems are well-described in a great number of papers [14], many of which also deal with the context of virtual enterprises. In spite of different generations of knowledge management systems are described in [5], where it is also explained why they did not live up to the expectations they had aroused.

One of the weak points of these systems is the need to link conceptual framework with technological level, especially for knowledge representation. In [15], it is stated that knowledge representation is a multidisciplinary subject that needs to apply theories and techniques from logic, to provide formal structure and rules of inference; ontology, to define the kinds of things that exit in the application domain; and computation, to support the applications that distinguish knowledge representation from pure philosophy.

Therefore, to communicate and distribute knowledge among the partners in a virtual enterprise not only technological approach is required. The definition of a common conceptual framework that enables partners to gain a common understanding about the business and goals of the virtual enterprise is also needed. The main problems in establishing this kind of systems in virtual enterprises are:

- The partners that make up a virtual enterprise implement different processes with distinct rules and procedures to perform the main activity of their businesses.

- The partners in a virtual enterprise usually have different types of infrastructure, organisational structure, decisional units, and so forth.
- The success of each of the partners that make up a virtual enterprise is due to several factors, such as know-how, the use of resources, core skills, and so on.
- The data, notations, documents, and so forth managed by each partner are diverse and sometimes the same documents are used for different purposes.

3 Proposed Approach to Knowledge Management in Virtual Enterprises

The IRIS Research Group at the Universitat Jaume I in Castelló (Spain) has been working on several projects related to the virtual enterprise in different sectors (transport, tile industry, textile, and so forth) since 1999 [16–18]. The main aim of these projects has been to define and apply an architecture, called ARDIN [16], capable of supporting the design and creation of a virtual enterprise in an integrated way.

Taking into account the problems mentioned above, the group's research is currently focused on adding a new dimension to the ARDIN architecture that enables knowledge to be compiled, managed, and applied within a virtual enterprise. The new dimension has been formally organised according to the following issues:

1. A methodology for directing the process of development and implementation of a knowledge management system in a virtual enterprise called KM-IRIS [7].
2. A set of models to allow the identification, representation, and communication of the knowledge inherent to a virtual enterprise.
3. The design of a technological infrastructure that allows knowledge to be stored, processed, and distributed inside a virtual enterprise.

The results shown in this paper are concerned with the second of these issues, the aim of which is to identify what knowledge is useful to an enterprise in general and to provide a conceptual framework that enables to represent enterprise knowledge.

4 Target Knowledge

In section 2, we have defined enterprise knowledge as actions that allow people to act and to make decisions with the result of adding value to the company. Each enterprise has its own vision, mission, and strategies and thus the knowledge that adds value to its business is different in each particular case. However, and bearing in mind that some common concepts are to be found in any enterprise, the framework proposed in this paper provides several conceptual blocks of knowledge defined according to the dimensions of enterprises in order to help them identify the most useful knowledge for them, that is to say, their target knowledge.

In this framework, the first grouping of distinct kinds of knowledge is called a conceptual block of knowledge. This first classification is made by identifying the big items related to the enterprise and on which it wishes to develop its knowledge management system, since these are the most interesting subjects that the enterprise needs to know in order to gain a deeper knowledge of its businesses and the capacity to improve them. Furthermore, the aim of improving knowledge management in the virtual enterprise by establishing a common conceptual knowledge framework that allows the knowledge representation is also considered.

The main conceptual blocks of knowledge defined in this framework are proposed, first, taking into account several enterprise dimensions (organisation, resources, process, and so forth) suggested in the context of Enterprise Modelling [19–21], and second the explicit and tacit dimensions of knowledge. This conceptual blocks of knowledge can be classified into two categories considering the two criteria above mentioned:

– **Enterprise oriented blocks:** the blocks defined are: organisation, process, product, and resource. They have their origin in the enterprise dimensions proposed in the context of Enterprise Modelling. These blocks are primary related to **explicit knowledge**. However, despite the use of the adjective 'explicit' it must be pointed out that in these blocks we can find both explicit and tacit target knowledge, since explicit and tacit knowledge are not two separate forms of knowledge, but instead inseparable, necessary components of all knowledge [22].
– **Human oriented blocks:** the blocks defined are: owner, supplier and customer, administration and trade union, and environment. They are originated in the **tacit dimension of knowledge**. At the same way in the previous case, we can find as tacit as explicit target knowledge in these blocks, however the most target knowledge defined within these blocks will be usually tacit knowledge.

The target knowledge presented in this paper are related to **enterprise oriented blocks**, and, despite the fact that each enterprise should identify its own target knowledge, they can be useful for enterprises like a pattern in the process of identifying their target knowledge for the conceptual block of knowledge above proposed. Therefore, in this section, we present a general definition of target knowledge with the objective of establishing a conceptual knowledge framework that allows for common understanding among the partners in a virtual enterprise - something that is needed before the implementation of its knowledge management system.

This definition is made from the user's point of view, taking into account, for each **enterprise oriented block** defined, the knowledge that partners need to improve the performance and interoperability of the virtual enterprise. The target knowledge for each block is defined in the following subsections.

4.1 Conceptual block of knowledge: ORGANISATION

This conceptual block details the knowledge about the structure of the organisation, providing different visions: administrative, systemic, and from the human resources point of view. Moreover, it captures the target structure, decisional structure and rules structure of the enterprise. Therefore, the target knowledge related to this block can be organised in four ontological categories as it is shown in Table 1.

Table 1. Target knowledge for conceptual block of ORGANISATION

Ontological category	Target knowledge	Description
Target structure	Strategic level	To know which is the enterprise's vision and mission, and also to identify clearly which are the strategic objectives and strategy established in the enterprise in the long term to reach its mission.
	Tactic level	To know what decisions are taken and how resources are assigned in the medium term to follow the strategy defined at the strategic level.
	Operative level	To know how the enterprise's daily activities and operations are planned, coordinated and executed in the short term and who is in charge of these activities.
Organisational structure	Administrative view	To know first which is the structure from administrative and executive point of view taking into account the different kinds of virtual enterprises: in star, in network, ans so forth and, second, which is the organisation chart for individual enterprise as well as virtual enterprise.
	Human Resources view	To know which is the hierarchic organisation established in enterprise, defining the different levels that exist, that is to say, departmental units, departments, sections, and so on.
	System view	To know from a system point of view which are the systems identified in enterprise and which are its main functions and relationships.
Decisonal structure	Decisional centres	To know which is the structure of enterprise taking into account the decision taken by employees at distinct levels.
	Cost centres	To know enterprise's costs associated a each element that exists in enterprise to analyse them considering different clusters performed according to the strategy adopted.
Business rules	Lines of action	To know which are the main guidelines and directives of behaviour established in enterprise to achieve a good functioning of all elements involved in it.
	Rules	To know which are strict rules provided by the company in order to perform all the enterprise activities.

4.2 Conceptual block of knowledge: PROCESS

This block provides knowledge on general issues about processes in enterprises such as ICOMs (Input, Control, Output, and Mechanisms), documents, rules, know-how, and so forth; and on flows (of work, documents, of material, and so on). Different levels of processes are then defined including decisional and collaborative processes. The ontological categories proposed for this block are shown in Table 2.

Table 2. Target knowledge for conceptual block of PROCESS

Ontological category	Target knowledge	Description
General	ICOMs	To know for each process which are the elements needed to perform a process, that is to say, the inputs needed and output obtained as well as the constraints and the mechanisms to carry out the process.
	Splitting of processes	To identify the main macroprocesses performed in enterprise and how they are divided into microprocesses, activities, tasks, and so on.
	Documents	To identify the primary documents that are used for each process, such as orders, delivery notes, invoices, and so forth.
	AI-IS and TO-BE views	To understand which is the current situation of enterprise processes and which should be the desired situation.
	Procedures	To know for each process the specific procedures that it is needed to perform in enterprise.
	Know-how	To identify specific, special skills and capabilities that enterprise has in each process.
	Cost	To analyse which are the costs linked to processes, and their profitability and added value for customers.
Flow	Of materials	To know the different ways in which the materials are transformed in enterprise and in which processes are involved these materials.
	Of data / information / knowledge	To know which are the main track that data run in enterprise to be transformed into information and knowledge, in order to identify the main mechanisms, techniques, and methods to obtain information and knowledge.
	Of decision / control	To understand step by step how decisions are taken in enterprise and they control the enterprise performance.
	Workflow	To know which is the sequence of the different tasks that make up one activity, and how they are carried out and by who.
	Of documents	To know which is the sequence and possibilities of transforming documents involved in processes and by means of what rules this transformation is performed.
Process level	Operative processes	To know which are the processes developed by enterprise at the operative level, realizing which are the core processes, in which enterprise is the leadership; which are the added value processes that add value to enterprise and to its products/services; and which are the supporting processes.
	Decisional processes	To know how the processes related to decision-making at the strategic, tactic and operational level are implemented.
	Collaborative processes	To know what are the processes that involve other partners of virtual enterprise and how they are carried out.

4.3 Conceptual block of knowledge: PRODUCT

The main knowledge about the products and/or services provided by the enterprise are described in this block, taking into account the process of achievement and marketing, the composition options, the quality, the cost and so forth. Thus, the target knowledge related to this block can be organised in five ontological categories as it is shown in Table 3[1].

Table 3. Target knowledge for conceptual block of PRODUCT

Ontological category	**Target knowledge**	**Description**
Generation process	Manufacturing	To identify which is the way to obtain the product in enterprise (manufacturing, assembly, project, an so forth) and to know the main features of the corresponding process to generate the product.
Composition	Bill of materials	To identify the components and materials needed to generate one product.
	Composition levels	To know the different levels of composition in which the product can be divided into.
	Optionality	To identify the possibilities of product configuration in order to provide customers different versions of the same product or the same product with distinct customisation, assembly or labeling.
Quality	Standards	To know the standards that are linked to products developed in enterprise.
	Documentation	To identify the documentation performed about quality product in enterprise and the main links with the other documentation generated in enterprise.
Marketing	Samples	To know which are the possibilities of offering samples of products to customers, identifying which are the more useful, more profitable, and so forth.
	Catalog	To identify which is the list of products with their references, main features, prices, special conditions, and so on.
	Advertisement	To understand which is enterprise's philosophy for advertisement and which are the main mechanisms of publicity that it uses in order to reach the planned objectives.
	Trademark	To identify which is the philosophy of trademark and which are the primary symbols to show it.
	Labels	To know the diverse possibilities of putting labels to products in order to customisation them taking into account customer's wishes.
Cost	Rough / After taxes	To analyse which are the costs as rough as after taxes related to products.
	Profitability	To classify products taking into account economical profitability of products developed by the enterprise.
	Added value for customers	To classify products according to the added value that they provide to customers.

[1] The target knowledge is only shown for products, it would be really similar for services.

4.4 Conceptual block of knowledge: RESOURCE

Knowledge about human resources and material resources is classified in this block into three main categories: location of these resources, potential use of them, and finally, the cost associated. The target knowledge related to human resources is shown in Table 4.

Table 4. Target knowledge for conceptual block of RESOURCE (Human resource)

Ontological category	Target knowledge	Description
Location	Internal	To identify which are the human resources that enterprise has in order to perform its activities.
	Inter-enterprise	To identify other human resources that they belong to other partners in virtual enterprise and how they can be useful collaborating in the enterprise's tasks.
	External	To identify feasible human resources that they do not belong to enterprise, but that could be useful in the future to reach its objectives.
Potential	Availability	To know which is the availability of the external human resources in order to cover different jobs.
	Curriculum	To know which is the people's curriculum vitae involved in individual and virtual enterprise as well as external human resources' curriculum vitae.
	Knowledge	To analyse which are the main knowledge that people involved in enterprise have.
	Capacity	To know which is the volume of work with which people could contribute to perform the different enterprise processes.
	Ability	To analyse which is the main know-how about products, process and so forth that people involved in enterprise have.
	Experience	To classify people according to their experience in several knowledge categories and for solving different kind of problems.
Cost	Rough / After taxes	To analyse which are the costs as rough as after taxes related to human resources.
	Profitability	To know the economical profitability of human resources.
	Added value for customers	To know which are the human resources that provide an added value to customers.

4.5 Analysis of target knowledge

The target knowledge presented above can be classified, using the ontological categories provided in the previous tables, into several categories, which are defined from two points of views:

1. First, the Enterprise Modelling field [19–21], in which the intention is to analyse the enterprise from a holistic point of view and therefore several dimensions related to the enterprise [23], such as organisation, process, product, and so forth, are defined.
2. Second, Knowledge Learning theory, in which the way to learn is based on three issues: concepts, procedures, and attitudes.

The result of this classification can be seen in Table 5.

Table 5. Framework to classify target knowledge

	Organisation	**Process**	**Product**	**Resource**
Concepts	Target structure Organisational structure Decisional structure	General Process level	Composition Quality	Location Potential
Procedures	Business rules	Flows	Generation process Cost	Cost
Attitudes	Business rules		Marketing	

5 Conclusion

Knowledge management systems can be used in virtual enterprises in a similar way how they are used in an individual enterprise. However, the specific features of this new organisational paradigm requires the introduction of a conceptual framework of knowledge, which enables the partners that make up a virtual enterprise to share the same concepts and to be more familiar with the other partners' procedures and attitudes, in order to implement an efficient knowledge management system.

In this paper, we have defined the target knowledge to establish this framework in a virtual enterprise, while considering each conceptual block of knowledge (enterprise oriented) proposed in the approach for knowledge management defined by IRIS Group, that is to say, organisation, process, product, and resource. The target knowledge defined has been classified taking into account two points of view, in order to provide a basis that can be used as a reference for further representation of knowledge by virtual enterprises that need to implement a knowledge management system.

Acknowledgements

This work was funded by CICYT DPI2003-02515. Also, it is partially supported by the European Commission within the 6^{th} Framework Programme, Interoperability Research for Networked Enterprises Applications and Software (INTEROP NoE), (IST-2003-508011), http://www.interop-noe.org. The authors are indebted to TG1 and TG2.

References

1. Kalpic, B., Bernus, P.: Business process modelling in industry–the powerful tool in enterprise management. Computers in Industry **47** (2002) 299–318
2. Kosanke, K.: Enterprise Inter- and Intraorganizational Integration, Building International Consensus. In: Proceedings of the ICEIMT'02 Conference. (2002)

3. Browne, J., Zhang, J.: Extended and virtual enterprises - similarities and differences. International Journal of Agile Management Systems **1/1** (1999) 30–36
4. Abdullah, M.S., Benest, I., Evans, A., Kimble, C.: Knowledge modelling techniques for developing knowledge management systems. In: 3rd European Conference on Knowledge Management. (2002) 15–25
5. Schutt, P.: The post-Nonaka Knowledge Management. Journal of Universal Computer Science **9** (2003) 451–462
6. Ramesh, B., Tiwana, A.: Supporting Collaborative Proccess Knowledge Management in New Product Development Teams. Decision Support Systems **27** (1999) 213–235
7. Chalmeta, R., Grangel, R., Fernández, V.: Methodology for the implementation of knowledge management systems. (2006) in evaluation.
8. Splieger, I.: Technology and Knowledge: bridging a 'generating' gap. Infomation and Management **40** (2003) 533–539
9. Nonaka, I., Takeuchi, H.: The Knowledge Creating Company: How Japanese Companies Create the Dynamics of Innovation. Oxford University Press (1995)
10. O'Leary, D.E., Kuokka, D., Plant, R.: Artificial Intelligence and Virtual Organizations. Communications of the ACM **40** (1997) 52–59
11. O'Leary, D.E.: Guest Editor's Introduction: Knowledge Management Systems: Converting and Connecting. IEEE Intelligent Systems **13** (1998) 30–33
12. Devedzic, V.: A survey of modern knowledge modeling techniques. Expert systems with applications **17** (1999) 275–294
13. Vail, E.F.: Knowledge mapping: getting started with knowledge management. Information Systems Management **Fall** (1999) 16–23
14. Liao, S.: Knowledge management technologies and applications - literature review from 1995 to 2002. Expert Systems with Applications **25** (2003) 155–164 Elsevier.
15. Sowa, J.F.: Knowledge Representation. Logical, Philosophical, and Computational Foundations. Books/Cole (2000)
16. Chalmeta, R., Grangel, R.: ARDIN extension for virtual enterprise integration. The Journal of Systems and Software **67** (2003) 141–152 Elsevier.
17. Chalmeta, R., Grangel, R., Ortiz, A., Poler, R.: Virtual Integration of the Tile Industry (VITI). In Jeusfeld, M., Pastor, O., eds.: Conceptual Modeling for Novel Application Domains. Volume 2814/2003 of Lecture Notes in Computer Science., Springer-Verlag (2003) 65–76
18. Chalmeta, R., Grangel, R.: Performance measurement systems for virtual enterprise integration. International Journal of Computer Integrated Manufacturing **18** (2005) 73–84 Taylor & Francis.
19. Vernadat, F.B.: Enterprise Modeling and Integration: Principles and Applications. Chapman and Hall (1996)
20. IFIP-IFAC: Generalised enterprise reference architecture and methodology (GERAM). Technical Report Version 1.6.3 (1999) http://www.cit.gu.edu.au/ bernus/taskforce/geram/versions.
21. Lillehagen, F.M., Dehli, E., Fjeld, L., Krogstie, J., Jørgensen, H.D.: Utilizing Active Knowledge Models in an Infrastructure for Virtual Enterprises. In: PRO-VE. (2002) 353–360
22. Tsoukas, H.: The tyranny of light. Futures **29** (1997) 827–843
23. Grangel, R., Chalmeta, R., Schuster, S., na, I.P.: Exchange of Business Process Models using the POP* Meta-model. In: International Workshop on Enterprise and Networked Enterprises Interoperability (ENEI'2005). Volume 3812/2005 of Lecture Notes in Computer Science., Springer-Verlag (2005)

Title:	**A Modelling Framework for Sharing Knowledge**
Authors:	R. Grangel and R. Chalmeta and C. Campos
Conference:	11th International Conference on Knowledge-Based and Intelligent Information and Engineering Systems (KES2007)
Publisher:	Springer Verlag Berlin (USA)
Vol:	LNCS
JCR:	COMPUTER SCIENCE, THEORY & METHODS (0,402; 62; 38.975) (data of 2005)
Place:	Vietri sul Mare (Italy)
Date:	12-9-2007 to 14-9-2007
Status:	Accepted

Abstract

Enterprise Modelling can be used successfully for different purposes, which includes capturing enterprise knowledge. However, one of the weaknesses of Enterprise Modelling is the lack of strong links with software generation. Model Driven Engineering attempts to solve this situation by promoting the use of models and their transformations in the software development process. In this context, the use of enterprise models that are able to capture knowledge and help to implement Knowledge Management Systems would be an important step forward.

In this paper, we present a proposal for Enterprise Modelling focused on enterprise knowledge. It starts from the CIM level and follows a model-driven approach. The modelling proposal provides a conceptual framework that allows enterprises to share knowledge by using a defined UML2 Profile for Modelling Enterprise Knowledge.

A Modelling Framework for Sharing Knowledge

Reyes Grangel[1], Ricardo Chalmeta[1], and Cristina Campos[1]

[1] Grupo de Investigación en Integración y Re-Ingeniería de Sistemas (IRIS), Dept. de Llenguatges i Sistemes Informàtics, Universitat Jaume I, 12071 Castelló, Spain
{grangel, rchalmet, camposc}@uji.es

Abstract. Enterprise Modelling can be used successfully for different purposes, which includes capturing enterprise knowledge. However, one of the weaknesses of Enterprise Modelling is the lack of strong links with software generation. Model Driven Engineering attempts to solve this situation by promoting the use of models and their transformations in the software development process. In this context, the use of enterprise models that are able to capture knowledge and help to implement Knowledge Management Systems would be an important step forward.
In this paper, we present a proposal for Enterprise Modelling focused on enterprise knowledge. It starts from the CIM level and follows a model-driven approach. The modelling proposal provides a conceptual framework that allows enterprises to share knowledge by using a defined UML2 Profile for Modelling Enterprise Knowledge.

1 Introduction

Enterprise Modelling can be used successfully for different purposes, such as capturing enterprise knowledge [1–4]. However, enterprise models are not normally used in these processes, due to the fact that one of the weaknesses of Enterprise Modelling is the lack of strong links to software generation [4–6].

Model Driven Engineering (MDE) or Model Driven Development (MDD) approaches are a new paradigm in the context of Software Engineering. Such perspective attempt to improve the software development process by focusing on models as the primary artefacts and transformations as the primary operation carried out on models (which are used to map information from one model to another). As an example, Model Driven Architecture (MDA) defined by the OMG in 2001 [7]. The main purpose of this approach is to separate the functional specification of a system from the details of its implementation on a specific platform. This architecture therefore defines a hierarchy of models from three points of view [4, 7, 8], namely: **Computation Independent Model (CIM)**, **Platform Independent Model (PIM)** and **Platform Specific Model (PSM)**.

Enterprise models can be considered to be CIM models. Thus, new proposals for Enterprise Modelling that are focused on following a model-driven approach are needed in order to improve the connection between the CIM level and software generation. Based on this initial analysis, the problem dealt with in this paper is that of how to improve existing Enterprise Modelling proposals for capturing knowledge by following a model-driven approach, such as MDA, in order to implement a Knowledge Management System. The objective of this paper is, therefore, to present a proposal for Enterprise Modelling that is focused on knowledge, and to show how it is possible to capture this knowledge in models at the CIM level so that it can then be transformed into the PIM level.

In this section, the problem dealt with by the research presented in this paper is described. Section 2 outlines the analysis of the state of the art related to the problem being considered here. It outlines the main problems in the context of Enterprise Modelling, which concern the complexity of knowledge modelling and finally existing UML approaches to model enterprises and knowledge are examined. In section 3, the framework proposed for knowledge modelling is presented. The section also includes an explanation of the principles of the proposal and the mechanism used for modelling from technological point of view, that is to say, the defined UML 2.0 Profile. Finally, section 4 outlines the main conclusions.

2 Modelling at the CIM Level

Modelling a system at the CIM level involves developing a model of it at a conceptual level by using the abstraction mechanisms of a specific language or formalism. This provides a defined set of constructs and rules in order to suppress certain details so that a simplified model can be established independently from the computation viewpoint.

2.1 Enterprise Modelling Perspective

Enterprise Modelling refers to the externalisation and expression of enterprise knowledge [1], which provides a holistic view of an enterprise and considers all its dimensions, i.e. process, decision, information, behaviour, resources and so forth [9]. Nowadays, there are a great number of languages, standards, methodologies and their corresponding tools, which are classified as traditional Enterprise Modelling Languages (EMLs) in [4]. These EMLs cover different dimensions of the enterprise defined in GERAM [10] and they can even overlap. Moreover, other EMLs exist that have been created in order to make different kinds of exchanges easier, since interoperability problems are increasing among systems that use different EMLs [11]. This last category, among them UEML [12–14] and POP* [15, 16], provide common exchange formats to smooth the exchange of enterprise models at a horizontal level.

However, one of the main weaknesses of Enterprise Modelling is the lack of strong links between enterprise models and software generation. To solve this gap, one solution, as pointed out in [17], is that the role of enterprise models should be that of facilitating the design, analysis and operation of the enterprise according to models, i.e. it should be driven by models (model-driven). Nowadays, the model-driven approach is followed by numerous projects such as MODELWARE [18], ATHENA [15], and INTEROP [13] in the European Union, and Model Driven Architecture (MDA) [7], which is carried out by the OMG.

MDA is an emerging paradigm. A lot of work is being carried out within the OMG framework in relation to PIMs, PSMs, QVT, and so forth, but the characterisation of CIMs and the features that an enterprise model must satisfy in order to be considered a CIM and generate appropriate software are still in progress [4]. This gap is specially remarkable when the purpose of modelling is to capture and to make enterprise knowledge explicit.

2.2 Knowledge Complexity and Representation

But, what do we understand by 'enterprise knowledge'? First of all, there is no universally accepted definition of exactly what knowledge is. Some authors define it, for example, as the information individuals possess in their minds [19]. This definition is argued by saying that data (raw numbers and facts) exist within an organisation. After processing these data they are converted into information and, once it is actively possessed by an individual, this information in turn becomes knowledge. There are also other approaches to defining knowledge that are less dependent on the information technologies. One of the most cited is the approach proposed by Nonaka [20], who defines knowledge as the justified belief that increases the capacity of an entity to take effective action. Following this line of reasoning, knowledge can be seen from five different perspectives [21]: (1) as a state of mind, (2) as an object, (3) as a process, (4) as a condition for access to information, or (5) as a capability. Taking this context into account and based on our own empirical observations, we define knowledge as the awareness that enables us to possess the skill or the capacity required in a particular situation (1) to deal with and resolve complex issues in an efficient and creative manner, and (2) to take advantage of opportunities by making the most appropriate decisions; and, enterprise knowledge as the network of connections among data and information that gives the people involved in an enterprise and insight into its workings and enables them to act and to make decisions that add value to the enterprise [22].

A key factor for achieving correct Knowledge Management in an enterprise is the development and implementation of a special kind of Information System, called a Knowledge Management System (KMS). That is to say, a technological system that allows enterprise knowledge to be created, codified, stored and distributed within the organisation [23]. One of the weak points of these kinds of systems is the need to link the conceptual framework with the technological level, especially for knowledge representation [22].

2.3 UML for Knowledge Modelling

The Unified Modeling Language (UML) has become a standard visual language for object-oriented modelling that has been used successfully for modelling information systems in very different domains [24]. However, UML is a general-purpose modelling language that can also be useful for modelling other types of systems such as, for example, an enterprise [25, 26]. Other works, such as [27], point out the possibility of using UML as a language for Enterprise Modelling.

However, the benefits of model-driven approaches and the new specification of UML 2 provided by the OMG suggest the need to provide more practical examples for Enterprise Modelling with UML based on these recent works [28], and especially for Enterprise Knowledge Modelling.

Furthermore, despite the fact that the weakness of the stereotype mechanism is pointed out in [29], the new specification of UML 2 [24] provides profiles that are more complete than version 1.5 [30]. It will therefore be possible to customise UML in a better way [31].

The objective of the research presented in this paper was to consider the possibility of using UML as a knowledge representation language on the basis of two positive factors: first, that it is a visual language which has become a standard object-oriented language and thus there are a lot of tools available on the market; and, second, that it is commonly used by engineers in enterprises for software development. To make this possible, the capacity of UML 2.0 to extend the language to a specific domain was used. A UML 2 Profile for Modelling Enterprise Knowledge was then defined in an attempt to achieve a common understanding within the context of Enterprise Modelling. This profile takes into account enterprise dimensions and previous works leading to initiatives such as UEML and POP*.

3 Proposal for Enterprise Knowledge Modelling

3.1 Principles of the Proposal

The objective of the proposal presented in this paper is to represent Enterprise Knowledge at the CIM level and, thus, the result of using it in enterprises is a graphical model of the enterprise knowledge map that allows enterprises to share knowledge. In general terms, this proposal is based on Model Driven Engineering (MDE), which promotes the design and development of computer systems using different types of models built at different levels of abstraction. More particularly, it is also supported by the MDA defined by the OMG, which is an instantiation of MDE. According to this approach, the process of developing a computer system is based on the separation of the functional characteristics of the system from the details of its specification on a specific platform. The proposal follows this premise as a fundamental concept together with the following principles:

- **Model-driven approach**. The separation of the functional specification of a system from the specific kind of technology that will be used to develop it, improve interoperability, portability, maintenance and usability of the resulting system, in this case, a KMS.
- **Proposal focused on Enterprise Modelling**. Traditionally, Enterprise Modelling has been able to make enterprise knowledge explicit by modelling enterprise processes, products, organisation, etc. If correctly managed these models help enterprises in knowledge management, since graphical models provide a better understanding of enterprise functions and allow decisions to be made in a more efficient way. In this proposal, the traditional enterprise dimensions were adapted so as to take enterprise knowledge into account as a dimension in itself, which can be represented at the CIM level and then transformed at different levels of abstraction.

– **User-oriented modelling framework**. This proposal is carried out at the CIM level, as a starting point from which a KMS based on models has to be built. Bearing in mind that one of the prerequisites of Enterprise Modelling is to establish the objective and the scope of modelling, this proposal attempts to provide the simplicity that is needed at this level. In fact, the enterprise should be represented so that it can be understood by users not specialised in modelling and, at the same time, with a sufficient level of detail to allow the definition of requirements for the future computer system. Thus, the proposal presents a set of models oriented towards improving user understanding, but they also set down the bases for future transformation of the models developed at the CIM level towards lower levels.

3.2 Modelling Framework of the Proposal

In order to achieve the objectives detailed in the previous section, the proposal for Enterprise Knowledge Modelling makes use of the following components:

– A metamodel of Enterprise Knowledge at the CIM level, as well as diverse metamodels for representing the other enterprise dimensions such as process, product and so on.
– A UML2 Profile for Enterprise Knowledge Modelling.
– A guide that can help enterprises to use this profile with the objective of obtaining their knowledge map.

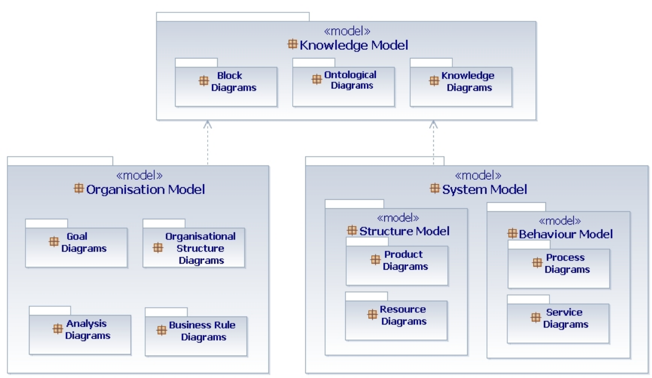


Fig. 1. Framework for Modeling Enterprise Knowledge at the CIM level

This general structure defines a framework for modelling enterprise knowledge on the CIM level at two levels of abstraction, which are required due to the great complexity of this level (see Figure 1):

1. **Knowledge Model.** This corresponds to the top level of the model at the CIM level; the enterprise is represented from a holistic point of view, thus providing a general vision of the enterprise and its business that will later

be detailed in a local way in successive lower levels. Since this proposal is focused on representing enterprise knowledge, at this level the following elements are modelled: the conceptual blocks of knowledge defined by the enterprise, together with target knowledge defined for each block, the ontological categories that make it possible to connect to target knowledge, and the variables and knowledge sources needed for the extraction procedure.

2. **Business Model.** Here, the vision of the business is represented according to three types of models, i.e. the Organisational Model, the Structure Model and the Behaviour Model. The main objective of these models is to represent a company from an organisational, structure and behavioural point of view, respectively. The **Organisational Model** is used to model objectives, organisational structures and business rules of an enterprise; the **Structure Model** represents enterprise products and resources; and, finally, the **Behaviour Model** shows how enterprise activities are carried out and information flows among them, that is, it is a representation of processes and services.

From a technological point of view (see Table 1), this proposal was implemented using the capacity of UML2 to extend a metamodel, that is to say, using a UML2 Profile. The UML2 Profile was defined for Enterprise Knowledge Modelling at the CIM level, following an MDA approach. This Profile is developed from the principles and the conceptual framework defined above. The profile provides the constructs needed to perform the models proposed earlier and it was implemented using IBM Rational Software Modeller. Finally, this modelling proposal can be applied in an enterprise following the KM-IRIS Methodology for the Implementation of KMS described in [32].

Table 1. Framework for Modeling Enterprise Knowledge from technological viewpoint

Abstraction Level	Metamodel	UML Profile	Model	Diagram
CIM-Knowledge	Knowledge	UML Profile for KM	Knowledge	Blocks Ontological Knowledge
CIM-Business	Organisation	UML Profile for GM UML Profile for OSM UML Profile for AM UML Profile for BRM	Organisation	Goals Organisational Structure Analysis Business Rules
	Structure	UML Profile for SM	Structure	Product Resource
	Behaviour	UML Profile for BM	Behaviour	Process Service

4 Conclusion

This research work intends to offer a systematic view of what Enterprise Knowledge is and how it can be modelled from a Model-Driven Engineering approach in order to provide a conceptual framework of sharing knowledge. The idea is not to define yet another Enterprise Modelling Language, but to adapt and to extend an existing standard modelling language like UML, while also taking into account the work carried out in the context of Enterprise Modelling, such as UEML and POP*.

The model-driven approach followed by the proposal is a promising beginning, but to take advantage of it, further research is needed. This modelling framework was applied in a real Case Study on a audit company to test its feasibility, and a first set of improvements was obtained. However, further research will be needed in order to improve the proposal through the feedback from applications in other domains, and providing a method of transforming CIM models of Enterprise Knowledge proposed at the CIM level into PIM models.

Acknowledgments

This work was funded by CICYT DPI2003-02515, CICYT DPI2006-14708 and the European Commission within the 6^{th} Framework Programme, INTEROP NoE (IST-2003-508011). The authors are indebted to TG2 [13].

References

1. Vernadat, F.B.: Enterprise Modeling and Integration: Principles and Applications. Chapman and Hall (1996)
2. Vernadat, F.B.: Enterprise modeling and integration (EMI): Current status and research perspectives. Annual Reviews in Control **26** (2002) 15–25
3. Bernus, P.: Enterprise models for enterprise architecture and ISO 9000:2000. Annual Reviews in Control **27** (2003) 211–220
4. Grangel, R., Chalmeta, R., Campos, C.: Enterprise Modelling, an overview focused on software generation. In Panetto, H., ed.: Interoperability of Enterprise Software and Applications Workshops of the INTEROP-ESA International Conference EI2N, WSI, ISIDI and IEHENA 2005, Hermes Science Publishing (2005)
5. UEML: Unified Enterprise Modelling Language Project (IST-2001-34229). http://www.ueml.org (2004) Deliverable 1.1.
6. ATHENA: Deliverable DA1.3.1. Report on Methodology description and guidelines definition. http://www.athena-ip.org (2005)
7. OMG: MDA Guide Version 1.0.1. Object Management Group. Document number: omg/2003-06-01 edn. (2003)
8. Berrisford, G.: Why IT veterans are sceptical about MDA. In: Second European Workshop on Model Driven Architecture (MDA) with an emphasis on Methodologies and Transformations, Kent, Computing Laboratory, University of Kent (2004) 125–135 http://www.cs.kent.ac.uk/projects/kmf/mdaworkshop/submissions/Berrisford.pdf.
9. Doumeingts, G., Chen, D.: Interoperability development for enterprise applications and software. In Cunningham, P., Cunningham, M., Fatelnig, P., eds.: Building the Knowledge Economy: Issues, Applications, Case Studies. eBusiness, IOS Press Amsterdam (2003)
10. IFIP-IFAC: Generalised enterprise reference architecture and methodology (GERAM). Technical Report Version 1.6.3 (1999) http://www.cit.gu.edu.au/ bernus/taskforce/geram/versions.
11. Chen, D., Doumeingts, G.: European initiatives to develop interoperability of enterprise applications–basic concepts, framework and roadmap. Annual Reviews in Control **27** (2003) 153–162
12. UEML: Unified Enterprise Modelling Language Themantic Network (IST-2001-34229). http://www.ueml.org (2006)

13. INTEROP: Interoperability Research for Networked Enterprises Applications and Software NoE (IST-2003-508011). http://www.interop-noe.org (2006)
14. Opdahl, A., Henderson-Sellers, B.: Chapter 6. Template-Based Definition of Information Systems and Enterprise Modelling Constructs. In: Ontologies and Business System Analysis. Idea Group Publishing (2005)
15. ATHENA: Advanced Technologies for interoperability of Heterogeneous Enterprise Networks and their Applications IP (IST-2001- 507849). http://www.athena-ip.org (2006)
16. Grangel, R., Chalmeta, R., Schuster, S., Penya, I.: Exchange of Business Process Models using the POP* Meta-model. In: International Workshop on Enterprise and Networked Enterprises Interoperability (ENEI'2005). Volume 3812/2005 of Lecture Notes in Computer Science., Springer-Verlag (2005)
17. Fox, M.S., Gruninger, M.: Enterprise Modelling. AI Magazine **19** (1998) 109–121
18. MODELWARE: Modeling solution for software systems Project (IST-2004-511731). http://www.modelware-ist.org/ (2006)
19. Dretske, F.: Knowledge and the Flow of Information. Cambridge, MA: MIT Press (1981)
20. Nonaka, I., Takeuchi, H.: The Knowledge Creating Company: How Japanese Companies Create the Dynamics of Innovation. Nueva York: Oxford University Press (1995)
21. Alavi, M., Leidner, D.: Knowledge management and knowledge management systems: Conceptual foundations and research issues. Management Information Systems Quarterly **25(1)** (2001) 107–136
22. Grangel, R., Chalmeta, R., Campos, C.: Defining of Target Knowledge in Virtual Enterprise. In Abramowicz, W., Mayr, W., eds.: Business Information Systems - 9th International Conference on Business Information Systems (BIS 2006), Gesellschaft für Informatik (GI) (2006)
23. Day, R.: Totality and representation: A history of knowledge management through european documentation, critical modernity, and post-fordism. Journal of the American Society for Information Science and Technology **52(9)** (2001) 725–735
24. OMG: Unified Modeling Language: Superstructure, version 2.0. Object Management Group. version 2.0 formal/05-07-04 edn. (2005)
25. Eriksson, H., Penker, M.: Business Modeling with UML: Business Patterns at Work. J. Wiley (2000)
26. Marshall, C.: Enterprise Modeling with UML. Designing Successful Software Through Business Analysis. Addison Wesley (2000)
27. Panetto, H.: UML Semantics Representation of Enterprise Modelling Constructs. In: ICEIMT. (2002) 381–387
28. Grangel, R., Bourey, J.P., Chalmeta, R., Bigand, M.: UML for Enterprise Modelling: a Model-Driven Approach. In: Interoperability for Enterprise Software and Applications Conference (I-ESA'06). (2006)
29. Berio, G., Petit, M.: Enterprise Modelling and the UML: (sometimes) a conflict without a case. In: Proc. of 10th ISPE International Conf. on Concurrent Engineering: Research and applications. (2003) 26–30
30. OMG: OMG Unified Modeling Language Specification, version 1.5. Object Management Group. formal/03-03-01 edn. (2003)
31. Fuentes, L., Vallecillo, A., Troya, J.: Using UML Profiles for Documenting Web-Based Application Frameworks. Annals of Software Engineering **13** (2002) 249–264
32. Chalmeta, R., Grangel, R., Fernández, V.: Methodology for the Implementation of Knowledge Management Systems. In evaluation: Journal of the American Society for Information Science and Technology (2007)

Title:	**A Proposal to Develop Conceptual Models of Knowledge Management Systems**
Authors:	R. Grangel and R. Chalmeta and C. Campos
Conference:	26th International Conference on Conceptual Modeling (ER 2007)
Publisher:	Springer Verlag Berlin (USA)
Vol:	LNCS
JCR:	COMPUTER SCIENCE, THEORY & METHODS (0,402; 62; 38.975) (data of 2005)
Place:	Auckland (New Zeland)
Status:	In evaluation

Abstract

A Knowledge Management System (KMS) is a complex computer system that interact with the organisation's systems to facilitate the organisational knowledge management. A suitable conceptual model that identifies and represents all the knowledge that has to be processed and managed within the KMS should be created during the development process of this kind of systems. In addition, it is needed to link this conceptual model with the technological level in order to improve the process productivity.

In this paper, we present a Proposal to develop conceptual models of Knowledge Management Systems using UML as modelling language, that meets the above requirements. We describe the framework and the components of the Proposal, and an excerpt of the metamodel and the UML2 profile developed for the organisational dimension together with its application in a Case Study.

A Proposal to Develop Conceptual Models of Knowledge Management Systems

Reyes Grangel[1], Ricardo Chalmeta[1], and Cristina Campos[1]

[1] Grupo de Investigación en Integración y Re-Ingeniería de Sistemas (IRIS), Dept. de Llenguatges i Sistemes Informàtics, Universitat Jaume I, 12071 Castelló, Spain
{grangel, rchalmet, camposc}@uji.es

Abstract. A Knowledge Management System (KMS) is a complex computer system that interact with the organisation's systems to facilitate the organisational knowledge management. A suitable conceptual model that identifies and represents all the knowledge that has to be processed and managed within the KMS should be created during the development process of this kind of systems. In addition, it is needed to link this conceptual model with the technological level in order to improve the process productivity.
In this paper, we present a Proposal to develop conceptual models of Knowledge Management Systems using UML as modelling language, that meets the above requirements. We describe the framework and the components of the Proposal, and an excerpt of the metamodel and the UML2 profile developed for the organisational dimension together with its application in a Case Study.

Key words: Knowledge Management Systems, Knowledge Modelling, Model Driven Architecture, UML 2.0 Profile

1 Introduction

A Knowledge Management System (KMS) is a specialised system that interacts with the organisation's systems in order to generate new knowledge, distribute it among the members of the organisation and put it to use in products, services and systems [1]. Although the benefits of KMSs are well-described in a great number of papers [2], they did not live up to the expectations they had aroused [3]. One of these reasons is that although the implementation projects of KMSs are generally well developed from a technological point of view, they fail because the knowledge management needs have not well defined in a suitable conceptual model. Therefore, some organisations implement solutions that exceed their needs and are too sophisticated, or they choose solutions that are too basic and barely improve the efficiency of their knowledge extraction and acquisition processes.

Trying to solve this problem, in this paper we show how is possible to develop a conceptual model of the future KMS that (1) represents, using a graphical modelling language, all the knowledge that one organisation needs to identify, collect and store, and (2) can be used to guide in the development process of the KMS.

The paper is organised as follows. Section 2 outlines the analysis of the state of the art related to the problem being considered here, how to link conceptual models of knowledge with KMS and existing techniques for knowledge modelling. In section 3, the Proposal for Enterprise Knowledge Modelling is presented. One of the knowledge models of this Proposal, in particular for Organisational Structure Modelling, is shown in section 4 describing the developed metamodel, the implemented UML2 Profile and its application in a Case Study. Finally, section 5 outlines the main conclusions and possible lines for further research.

2 Literature Review

Starting from the concepts of knowledge and what means its management, this section presents a brief summary on how model-driven approaches could be applied to KMSs, and existing techniques for knowledge modelling at the CIM level.

2.1 Knowledge and Knowledge Management

There is no universally accepted definition of exactly what knowledge is. Some authors define it, for example, as the information individuals possess in their minds [4]. This definition is argued by saying that data (raw numbers and facts) exist within an organisation. After processing these data they are converted into information and, once it is actively possessed by an individual, this information in turn becomes knowledge. There are also other approaches to defining knowledge that are less dependent on the information technologies. One of the most cited is the approach proposed by Nonaka [5], who defines knowledge as the justified belief that increases the capacity of an entity to take effective action. Following this line of reasoning, knowledge can be seen from five different perspectives [6]: (1) as a state of mind, (2) as an object, (3) as a process, (4) as a condition for access to information, or (5) as a capability. Taking this context into account and based on our own empirical observations, we define knowledge as the awareness that enables us to possess the skill or the capacity required in a particular situation (1) to deal with and resolve complex issues in an efficient and creative manner, and (2) to take advantage of opportunities by making the most appropriate decisions.

On the other hand, according to [7] enterprise knowledge can been seen as information made actionable in a way that adds value to the enterprise. Taking this context into account, we defined enterprise knowledge as the network of connections among data and information that gives the people involved in an enterprise and insight into its workings and enables them to act and to make decisions that add value to the enterprise [8].

In addition, we consider knowledge management as the process of converting the knowledge from the sources available to an organisation and then connecting people with that knowledge [9, 10]. Therefore, the aim of knowledge management is the creation, access and reuse of knowledge [11].

2.2 Model Driven Knowledge Management Systems

A key factor for achieving correct knowledge management in an enterprise is the development and implementation of a special kind of Information System, called a Knowledge Management System (KMS). That is to say, a technological system that allows enterprise knowledge to be created, codified, stored and distributed within the organisation [12].

One of the weak points of these kinds of systems is the need to link the conceptual framework with the technological level [8]. One solution, as pointed out in [13], is that the role of enterprise models should be that of facilitating the design, analysis and operation of the enterprise according to models, i.e. it should be driven by models (model-driven). Nowadays, the model-driven approach is followed by numerous projects such as MODELWARE [14], ATHENA [15], and INTEROP [16] in the European Union, and Model Driven Architecture (MDA) [17], which is carried out by the OMG.

Model Driven Architecture (MDA) defined by the OMG in 2001 [17], is intended to promote the use of models as a fundamental way of designing and implementing different kinds of systems. The main purpose of this approach is to separate the functional specification of a system from the details of its implementation on a specific platform. This architecture therefore defines a hierarchy of models from three points of view [18, 17, 19], namely:

- **Computation Independent Model (CIM):** used to represent domain and system requirements in the environment in which it is going to operate. It is based on business models and sees the enterprise from a holistic point of view, that is independent of the computation.
- **Platform Independent Model (PIM):** used to model system functionality but without defining how it will be implemented and on what platform; it is focused on information and sets out from a computational point of view.
- **Platform Specific Model (PSM):** the PIM is transformed into a platform-dependent model according to the platform selected for use and is focused on a technological point of view.

A lot of work is being carried out within the OMG framework in relation to PIMs, PSMs, QVT, and so forth, but the characterisation of CIMs are still in progress [18]. This delay is more noteworthy in the characterisation and definition of the features that a Computer Independent Model of Knowledge (CIMK) must satisfy in order to generate an appropriate KMS. The main problem involved in modelling enterprises at the CIM level is how to accomplish a clear definition of the various aspects that the actors want to take into account. The domain and purpose of modelling, together with the aspects that must be highlighted, should be defined, and then the most suitable Enterprise Modelling Language should be chosen [20]. Therefore, the number of issues about knowledge that can be modelled at the CIM level increases the complexity of CIM models of knowledge and their transformations.

2.3 Knowledge Modelling at the CIM Level

From a knowledge modelling perspective, in [21], it is stated that knowledge representation is a multidisciplinary subject that needs to apply theories and techniques from (1) logic to provide a formal structure and rules of inference; (2) ontology, to define the types of things that exist in the application domain; and (3) computation, to support the applications that distinguish knowledge representation from pure philosophy. Moreover, according to [22], there is no single best theory or language for knowledge representation; rather, it is necessary to choose the technique(s) that can be best adapted for each kind of knowledge (procedural, declarative, metaknowledge, heuristic, etc.). The traditional techniques used in Artificial Intelligence for knowledge representation are the following [22]:

1. **Object-Attribute-Value-Triplets:** these are used to represent facts about objects and their attributes; they state the value of an attribute of an object.
2. **Uncertain Facts:** this is an extension of the previous O-A-V technique to allow uncertainty of facts to be described.
3. **Fuzzy Facts:** these represent uncertainty using the imprecise and ambiguous terms of the natural language.
4. **Rules:** these relate one or more premises or situations to one or more conclusions or actions.
5. **Semantic networks or concept maps:** these attempt to reflect cognition (following the psychological model of the human associative memory) by means of graphs that include objects, concepts and situations for a specific domain of knowledge.
6. **Frames:** these represent stereotypical knowledge of certain concepts or objects.
7. **Ontologies:** these represent a set of knowledge terms, including the vocabulary, the semantic interconnections and some simple rules of inference and logic, from a particular topic.

These knowledge representation techniques are supported by different knowledge representation languages, which are used to represent knowledge in a KMS. A knowledge representation language should be able to represent entities, events, actions, processes and time from syntactic and semantic points of view. An overview of the existing paradigms is given in [22]: (1) Logic-Based Representation Languages, (2) Frame-Based Representation Languages, (3) Rule-Based Representation Languages, (4) Visual Languages for Knowledge Representation, (5) Natural Languages and Knowledge Representation, and (6) Ontology Knowledge Representation.

In the fourth category, Unified Modeling Language (UML) is pointed out as being a suitable language for knowledge representation, even though it was originally developed for the software engineering domain. UML has become a standard visual language for object-oriented modelling that has been used successfully for modelling information systems in very different domains [23]. However, UML is a general-purpose modelling language that can also be useful for modelling other types of systems such as, for example, an enterprise [24, 25]. Other works, such as [26], point out the possibility of using UML as a language for Enterprise Modelling, even though in [27] it is explained how and under which conditions this can be performed. However, the benefits of model-driven approaches and the new specification of UML2 provided by the OMG suggest the need to provide more practical examples for Enterprise Modelling with UML based on these recent works [28], and especially for Enterprise Knowledge Modelling. In this line, some works, like [29], has been carried out following the possibility suggested in the previous section, but this proposal is not enterprise oriented and thus it does not take into account the different enterprise dimensions for modelling [30, 31].

Furthermore, despite the fact that the weakness of the stereotype mechanism is pointed out in [27], the new specification of UML 2 [23] provides profiles that are more complete than version 1.5 [32]. It will therefore be possible to customise UML in a better way [33]. For instance, UML provides a lot of diagrams for modelling behaviour aspects (but not for direct modelling of business processes) in a similar way to how they are represented in an IDEF diagram. Hence, business process modelling with UML is complex [34] and the use of profiles according to UML 2 can make this task easier.

Taking into account the state of the problems and solutions analysed in this section, the objective of the research presented in this paper was to consider the possibility of using UML as a knowledge representation language on the basis of two positive factors: first, that it is a visual language which has become a standard object-oriented language and thus there are a lot of tools available on the market; and, second, that it is commonly used by engineers in enterprises for software development. To make this possible, the capacity of UML to extend the language to a specific domain was used, and a UML2 Profile for Modelling Enterprise Knowledge was then defined. Moreover, this profile takes into account enterprise dimensions and previous works leading to initiatives from such as UEML[1] [36–39] and POP*[2] [40, 41].

3 Conceptual Framework of the Proposal for Enterprise Knowledge Modelling

The development and implementation of Knowledge Management Systems that embrace the whole enterprise is a more complex issue that has still not been satisfactorily resolved [42]. Trying to solve this problem, the IRIS Group at the Universitat Jaume I in Castelló, Spain, has been working on a project entitled 'Knowledge Management in Virtual Enterprises' since 2003.

The main results of this project are: (1) a useful, practical methodology that can be used to guide the process of developing and implementing a system

[1] Unified Enterprise Modelling Language, first developed by the UEML Thematic Network [35] and currently being worked on by INTEROP NoE [16].

[2] Acronym of the different enterprise dimensions: Process, Organisation, Product, and so on (represented by a star), proposed by ATHENA IP [15].

for gathering, managing and applying the knowledge that is generated both inside an enterprise and in the relations it has with the different organisations it works with. The result is the KM-IRIS Methodology for the Implementation of KMS [43]; (2) a Proposal for Enterprise Knowledge Modelling that makes it possible to represent and communicate the knowledge inherent to a virtual enterprise. The objective of this paper is concerned with the second of these results, the **Proposal for Enterprise Knowledge Modelling**, which aim is to represent Enterprise Knowledge at the CIM level obtaining a graphical model called **Enterprise Knowledge Map**.

In general terms, this Proposal is based on Model Driven Engineering (MDE), which promotes the design and development of computer systems using different types of models built at different levels of abstraction. More particularly, it is also supported by the MDA defined by the OMG, which is an instantiation of MDE. According to this approach, the process of developing a computer system is based on the separation of the functional characteristics of the system from the details of its specification on a specific platform. Therefore, the Proposal defines a framework for developing conceptual models of KMSs at the CIM level, that is to say, that allows to model enterprise knowledge on the CIM level at two levels of abstraction, which are required due to the great complexity of this level (see Table 1):

1. **CIM-Knowledge:** this corresponds to the top level of the model at the CIM level; the enterprise is represented from a holistic point of view, thus providing a general vision of the enterprise focused on representing enterprise knowledge that will later be detailed in a local way in successive lower levels.
2. **CIM-Business:** here, the vision of enterprise knowledge is detailed by means of a representation of its business, according to three types of models, i.e. the Organisational Model, the Structure Model and the Behaviour Model.

Table 1. Proposal for Enterprise Knowledge Modeling.

Abstraction Level	Metamodel	UML Profile	Model	Diagram
CIM-Knowledge	Knowledge	UML Profile for KM	Knowledge	Blocks Ontological Knowledge
CIM-Business	Organisation	UML Profile for GM UML Profile for OSM UML Profile for AM UML Profile for BRM	Organisation	Goals Organisational Structure Analysis Business Rules
	Structure	UML Profile for SM	Structure	Product Resource
	Behaviour	UML Profile for BM	Behaviour	Process Service

The Proposal follows the MDE premise as a fundamental concept together with the following principles: (1) it is focused on **Enterprise Modelling**, since it takes into account enterprise dimensions and previous works leading to initiatives such as UEML and POP*; and (2) it is a **user-oriented modelling framework**, since it should be developed at the CIM level by domain experts. A summary of the Proposal showing which are its abstraction levels, metamodels and profiles developed, as well as models and diagrams proposed for each level are shown in Table 1.

From a technological point of view, this proposal was implemented using the capacity of UML2 to extend a metamodel, that is to say, using a UML2 Profile. The **UML2 Profile** was defined for **Enterprise Knowledge Modelling** at the CIM level, following an MDA approach and the principles detailed above. The Profile provides the constructs needed to perform the models proposed earlier (see Table 1), and it was developed following these steps:

1. **Definition of the metamodels** shown in Table 1, with the objective of representing at conceptual levels the elements used for Enterprise Knowledge Modelling.
2. **Definition of the models and diagrams** that can be used to obtain the enterprise knowledge map. Figure 1 shows the models and diagrams defined within the Proposal by means of a Class Diagram.

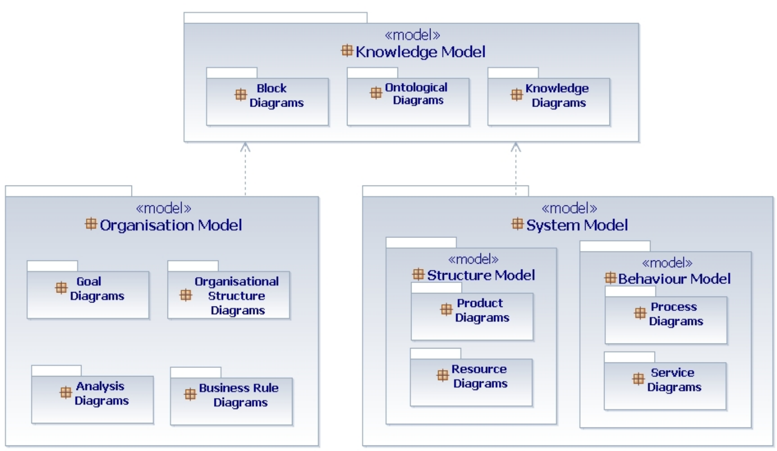


Fig. 1. Models and diagrams defined within the Proposal.

3. **Definition of the UML Profile for Enterprise Knowledge Modeling**, following for each of the profiles detailed in Table 1 the steps that follows:
   - Definition of stereotypes, tag values and constraints of the profile.
   - Extension of the metaclasses of the UML2 Metamodel.
   - Detail description of the profile.

4. **Implementation of the Profile** using a UML tool (IBM Rational Software Modeler [44] or MagicDraw [45] for example); in this case, the tool selected was IBM Rational Software Modeler.
5. **Validation of the Profile** by means of real Case Study.

4 Knowledge Model of Organisational Structure

This section shows an excerpt of the Proposal in order to make it understandable. In particular, it is described how is possible to model the enterprise knowledge related to organisational dimension. So that, an excerpt of the Organisation Metamodel, the UML Profile for OSM, and the Organisational Structure Diagram applied in a Case Study are shown.

4.1 Organisational Structure Metamodel

Definition of the metamodels shown in Table 1, with the objective of representing at conceptual levels the elements used for Enterprise Knowledge Modelling. Figure 2 shows an excerpt of the Organisational Structure Metamodel. The constructs needed to represent the knowledge for modelling Organisational Structure are the following (see Figure 2).

- **Enterprise:** it represents any type of organization with some of the existing legal forms for the businesses. This builder permits to represent so much an individual business as a remote entity, like a business extended or virtual, formed by different businesses with different legal personality. For this class the following attributes are defined:
  - **collaboration:** it specifies the type of cooperation that a business maintains with the remainder of businesses with the ones that can be related. Its values can be one of them you defined in the enumeration **'EnterpriseCollaborationType'**: single, extended or virtual.
  - **legalStatus:** it specifies the legal form that possesses the business.
  - **legalName:** it indicates the legal name that has assigned the business.
  - **cif:** it specifies the fiscal identifier of the business.
- **Unit:** it represents each one of the logical groups that are carried out in the business to negotiate their organization, being able to be of one of the following types: department, organizing unit, section and subsections. Besides, this builder permits to describe hierarchical structures in the shape of tree on the organizing structure of the business, therefore he permits to obtain his chart. For this class the following attributes are defined:
  - **type:** it specifies why categories of them you defined in the enumeration, **'UnitType'**, belongs the unit: department, organisationalUnit, section or subsection.
  - **isLeaf:** it indicates if the business unit no longer breaks down in no another level, that is to say, if is a matter of a leaf in the hierarchical tree that would form the chart with its diverse organisational units.

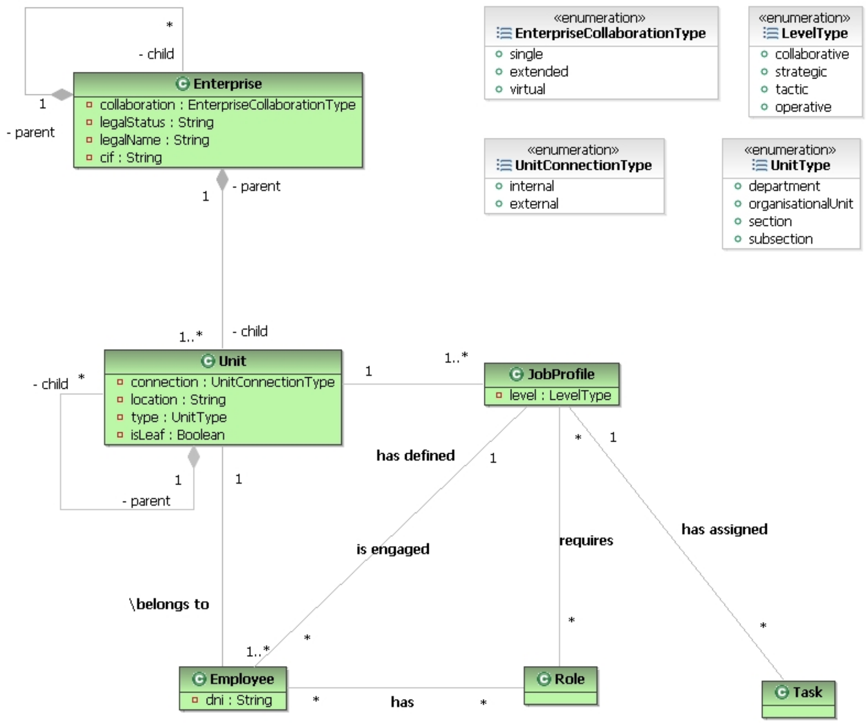


Fig. 2. An excerpt of the Organisational Structure Metamodel.

  - **connection:** It indicates that type of connection exists among the definite units in the organization between the two possible you defined by the enumeration **'UnitConnectionType'**: internal or external.
  - **location:** it specifies the physical location of the unit.
- **JobProfile:** it represents the group of an assembly of tasks that are related and they require of complementary and specific competences for their execution. The profile or placed of work defines the tasks and roles that should perform who occupy it. For this class the following attributes are defined:
  - **level:** it indicates the hierarchical level in which is definite the position of work, being possible one from among the following you defined by the enumeration **'LevelType'**: collaborative, strategic, tactic or operative.
- **Employee:** it represents the people that develop a determined work in the business, occupying a position of work and that have a determined role. For this class the following attributes are defined:
  - **dni:** it specifies the identifier unique of each employee.
- **Role:** it represents the attitudes and abilities that are required for a determined placed of work.

– **Task:** it represents the individual actions that are responsibility of a single individual and that are assigned to a determined placed of work.

4.2 UML Profile for Organisational Structure Modelling (OSM)

Definition of the UML Profile for Enterprise Knowledge Modeling, following for each of the profiles detailed in Table 1. From a technological point of view, this proposal was implemented using the capacity of UML2 to extend a metamodel, that is to say, using a UML2 Profile. The **UML2 Profile** was defined for **Enterprise Knowledge Modelling** at the CIM level, following an MDA approach and the principles detailed above. The Profile provides the constructs needed to perform the models proposed earlier (see Table 1), and it was developed following these steps:

– Definition of stereotypes, tag values and constraints of the profile.
– Extension of the metaclasses of the UML2 Metamodel.
– Detail description of the profile.

This profile has been called **'UML Profile for OSM'** and it allows the representation of the organizing structure of the business, showing which is the division of labor carried out in departments, sections, subsections, etc. as well as the different positions of work in each one of them, the employees that occupy them and the roles and associated tasks.

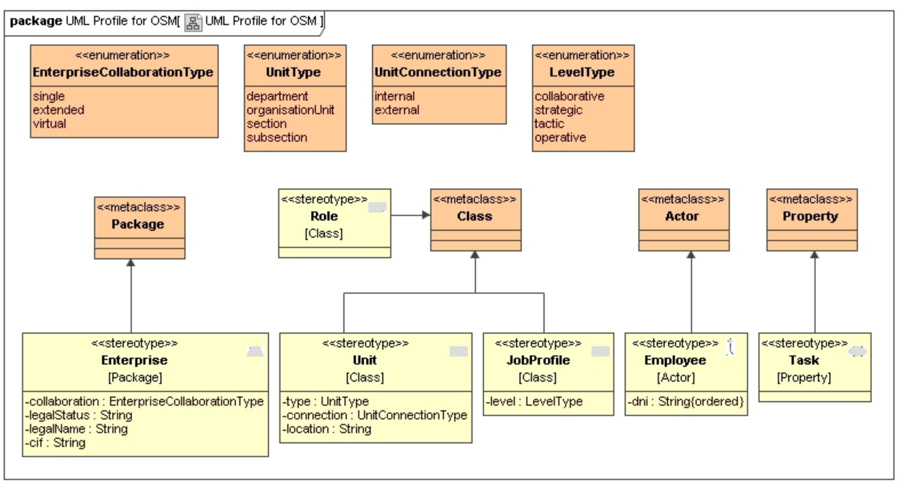


Fig. 3. Diagram of the 'UML Profile for OSM'.

In Figure 3, the diagram of the profile can be observed in which the definite stereotypes from the builders they are detailed relating to the organizing structure of the **Organisation Metamodel**, as well as the corresponding metaclasses of the UML2 Metamodel that extend.

In this case neither the attribute has been added **isLeaf** of the metaclasse **Unit** as a value labeling of the stereotype **Unit**, by the same reason commented in the previous profile.

Implementation of the Profile using IBM Rational Software Modeler. Figure 3 shows, as an example, one of the profiles that makes up of the **UML Profile for Enterprise Knowledge Modelling.**

The **'Organisational Structure Diagram'** form part of the **'Organisation Model'** and it allows to represent the chart of a business. This chart can include so much its organizing units as the positions of work, roles and employees in each one of them. The main stereotypes of the **'UML Profile for OSM'** that can be utilized to carry out this diagram they are shown in Table 2:

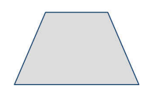
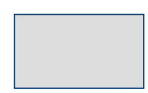
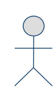
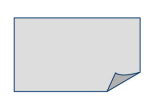
Stereotype	**Elements to model**	**Icon**
<<**Enterprise**>>	Individual or collaborative enterprise	
<<**Unit**>>	Any of the organisational units of an enterprise: departments, organisational units, sections, sub-sections, etc.	
<<**JobProfile**>>	Job profiles	
<<**Employee**>>	Employees of the enterprise	
<<**Role**>>	Roles of job	
<<**Task**>>	Tasks of a determined placed of work	<<**Task**>>

Table 2. Stereotypes and icons that is possible to use within the 'Organisational Structure Diagram'.

4.3 Validation with a Case Study

Validation of the Profile by means of real case study. An application of the Proposal for Enterprise Knowledge Modelling to a real **case study** was carried out in order to validate the basis of the Proposal empirically, and to test the UML2 Profile implemented for Enterprise Knowledge Modelling in a practical case from a definitional and a technological point of view.

Figure 4 presents an example of the application of the Proposal to an **audit enterprise**. In particular, the **UML Profile for OSM** was applied to this company in order to obtain the **Organisational Structure Diagram**. This diagram represents enterprise's organisational units and their corresponding job profiles, tasks, roles and employees. It was developed at the **CIM-Business level** and it is able to provide a detailed vision of the enterprise knowledge related to its organisational structure.

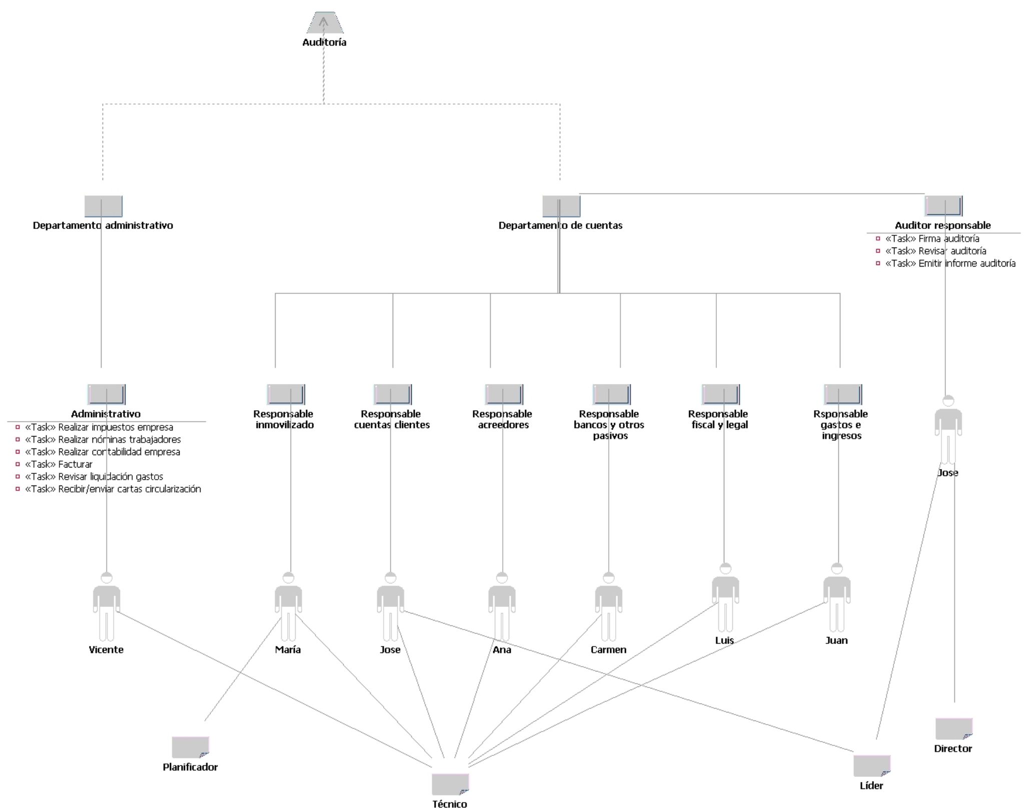


Fig. 4. 'Organisational Structure Diagram' for the Case Study.

Acknowledgments

This work was funded by CICYT DPI2003-02515, CICYT DPI2006-14708 and the European Commission within the 6^{th} Framework Programme, INTEROP NoE (IST-2003-508011). The authors are indebted to TG2 [16].

References

1. Abdullah, M.S., Benest, I., Evans, A., Kimble, C.: Knowledge modelling techniques for developing knowledge management systems. In: 3rd European Conference on Knowledge Management. (2002) 15–25
2. Liao, S.: Knowledge management technologies and applications - literature review from 1995 to 2002. Expert Systems with Applications **25** (2003) 155–164 Elsevier.
3. Shin, M., Holden, T., Schmidt, R.: From knowledge theory to management practice: towards and integrated approach. Information Processing & Management **37** (2001) 335–355
4. Dretske, F.: Knowledge and the Flow of Information. Cambridge, MA: MIT Press (1981)
5. Nonaka, I., Takeuchi, H.: The Knowledge Creating Company: How Japanese Companies Create the Dynamics of Innovation. Nueva York: Oxford University Press (1995)
6. Alavi, M., Leidner, D.: Knowledge management and knowledge management systems: Conceptual foundations and research issues. Management Information Systems Quarterly **25(1)** (2001) 107–136
7. Vail, E.F.: Knowledge mapping: getting started with knowledge management. Information Systems Management **Fall** (1999) 16–23
8. Grangel, R., Chalmeta, R., Campos, C.: Defining of Target Knowledge in Virtual Enterprise. In Abramowicz, W., Mayr, W., eds.: Business Information Systems - 9th International Conference on Business Information Systems (BIS 2006), Gesellschaft für Informatik (GI) (2006)
9. O'Leary, D.E., Kuokka, D., Plant, R.: Artificial Intelligence and Virtual Organizations. Communications of the ACM **40** (1997) 52–59
10. O'Leary, D.E.: Guest Editor's Introduction: Knowledge Management Systems: Converting and Connecting. IEEE Intelligent Systems **13** (1998) 30–33
11. Devedzic, V.: A survey of modern knowledge modeling techniques. Expert systems with applications **17** (1999) 275–294
12. Day, R.: Totality and representation: A history of knowledge management through european documentation, critical modernity, and post-fordism. Journal of the American Society for Information Science and Technology **52(9)** (2001) 725–735
13. Fox, M.S., Gruninger, M.: Enterprise Modelling. AI Magazine **19** (1998) 109–121
14. MODELWARE: Modeling solution for software systems Project (IST-2004-511731). http://www.modelware-ist.org/ (2006)
15. ATHENA: Advanced Technologies for interoperability of Heterogeneous Enterprise Networks and their Applications IP (IST-2001- 507849). http://www.athena-ip.org (2006)
16. INTEROP: Interoperability Research for Networked Enterprises Applications and Software NoE (IST-2003-508011). http://www.interop-noe.org (2006)
17. OMG: MDA Guide Version 1.0.1. Object Management Group. Document number: omg/2003-06-01 edn. (2003)

18. Grangel, R., Chalmeta, R., Campos, C.: Enterprise Modelling, an overview focused on software generation. In Panetto, H., ed.: Interoperability of Enterprise Software and Applications Workshops of the INTEROP-ESA International Conference EI2N, WSI, ISIDI and IEHENA 2005, Hermes Science Publishing (2005)
19. Berrisford, G.: Why IT veterans are sceptical about MDA. In: Second European Workshop on Model Driven Architecture (MDA) with an emphasis on Methodologies and Transformations, Kent, Computing Laboratory, University of Kent (2004) 125–135 http://www.cs.kent.ac.uk/projects/kmf/mdaworkshop/submissions/Berrisford.pdf.
20. Grangel, R., Ben-Salem, R., Bourey, J., Daclin, N., Ducq, Y.: Transforming GRAI Models into UML Models, a First Step to Model Driven Interoperability. In: evaluation 3rd Interoperability for Enterprise Software and Applications Conference (I-ESA'07). (2007)
21. Sowa, J.F.: Knowledge Representation. Logical, Philosophical, and Computational Foundations. Books/Cole (2000)
22. Gasevic, D., Djuric, D., Devedzic, V.: Model Driven Architecture and Ontology Development. Springer (2006)
23. OMG: Unified Modeling Language: Superstructure, version 2.0. Object Management Group. version 2.0 formal/05-07-04 edn. (2005)
24. Eriksson, H., Penker, M.: Business Modeling with UML: Business Patterns at Work. J. Wiley (2000)
25. Marshall, C.: Enterprise Modeling with UML. Designing Successful Software Through Business Analysis. Addison Wesley (2000)
26. Panetto, H.: UML Semantics Representation of Enterprise Modelling Constructs. In: ICEIMT. (2002) 381–387
27. Berio, G., Petit, M.: Enterprise Modelling and the UML: (sometimes) a conflict without a case. In: Proc. of 10th ISPE International Conf. on Concurrent Engineering: Research and applications. (2003) 26–30
28. Grangel, R., Bourey, J.P., Chalmeta, R., Bigand, M.: UML for Enterprise Modelling: a Model-Driven Approach. In: Interoperability for Enterprise Software and Applications Conference (I-ESA'06). (2006)
29. Abdullah, M., Evans, A., Benest, I., Kimble, C.: Developing a UML Profile for Modelling Knowledge Based Systems (2006)
30. Berio, G., Vernadat, F.B.: New developments in enterprise modelling using CIMOSA. Computers in Industry **40** (1999) 99–114
31. IFIP-IFAC: Generalised enterprise reference architecture and methodology (GERAM). Technical Report Version 1.6.3 (1999) http://www.cit.gu.edu.au/ bernus/taskforce/geram/versions.
32. OMG: OMG Unified Modeling Language Specification, version 1.5. Object Management Group. formal/03-03-01 edn. (2003)
33. Fuentes, L., Vallecillo, A., Troya, J.: Using UML Profiles for Documenting Web-Based Application Frameworks. Annals of Software Engineering **13** (2002) 249–264
34. Noran, O.: UML vs. IDEF: An Ontology-Oriented Comparative Study in View of Business Modelling. In: ICEIS (3). (2004) 674–682
35. UEML: Unified Enterprise Modelling Language Thermantic Network (IST-2001-34229). http://www.ueml.org (2006)
36. UEML: Unified Enterprise Modelling Language Project (IST-2001-34229). http://www.ueml.org (2004) Deliverable 1.1.
37. Berio, G., Opdahl, A., Anaya, V., Dassisti, M.: Deliverable DEM 1: UEML 2.1. Technical report, INTEROP-DEM (2005)

38. Opdahl, A., Henderson-Sellers, B.: A Template for Defining Enterprise Modelling Constructs. Journal of Database Management **15(2)** (2004)
39. Opdahl, A., Henderson-Sellers, B.: Chapter 6. Template-Based Definition of Information Systems and Enterprise Modelling Constructs. In: Ontologies and Business System Analysis. Idea Group Publishing (2005)
40. Ohren, O.P.: Deliverable DA1.3.1. Report on Methodology description and guidelines definition. Technical report, ATHENA (Advanced Technologies for interoperability of Heterogeneous Enterprise Networks and their Applications) Project (IST-2003-2004) (2005)
41. Grangel, R., Chalmeta, R., Schuster, S., Penya, I.: Exchange of Business Process Models using the POP* Meta-model. In: International Workshop on Enterprise and Networked Enterprises Interoperability (ENEI'2005). Volume 3812/2005 of Lecture Notes in Computer Science., Springer-Verlag (2005)
42. Heinrichs, J., Lim, J.S.: Model for organizational knowledge creation and strategic use of information. Journal of the American Society for Information Science and Technology **56(6)** (2005) 620–629
43. Chalmeta, R., Grangel, R., Fernández, V.: Methodology for the Implementation of Knowledge Management Systems. In evaluation: Journal of the American Society for Information Science and Technology (2007)
44. IBM: IBM Rational Software Modeler Development Platform 6.0.1. http://www-306.ibm.com/software/rational/ (2007)
45. NoMagic: MagicDraw UML 12.0. http://www.magicdraw.com/ (2007)

Title:	**Using UML Profiles for Enterprise Knowledge Modelling**
Authors:	R. Grangel and R. Chalmeta and C. Campos
Conference:	26th International Conference on Conceptual Modeling (ER 2007) - 3rd International Workshop on Foundations and Practices of UML (FP-UML 2007)
Publisher:	Springer Verlag Berlin (USA)
Vol:	LNCS
JCR:	COMPUTER SCIENCE, THEORY & METHODS (0,402; 62; 38.975) (data of 2005)
Place:	Auckland (New Zeland)
Status:	In evaluation

Abstract

Knowledge representation is a multidisciplinary subject that needs to apply theories and techniques from logic, ontology and computation. There are a great number of languages able to represent knowledge, among them UML can be considered a suitable language for modelling knowledge. It could be included within the category of visual languages for knowledge representation.

On the other hand, numerous efforts are being carried out in the context of Enterprise Modelling to improve the capacity of enterprise models for externalising enterprise knowledge. The Proposal presented in this paper combines both approaches, UML and Enterprise Modelling, in order to make possible Enterprise Knowledge Modelling using UML. It shows a summary of this Proposal describing its principles, the main steps of its development and an example of one of the UML Profiles implemented with the objective of modelling knowledge.

Using UML Profiles for Enterprise Knowledge Modelling

Reyes Grangel[1], Ricardo Chalmeta[1], and Cristina Campos[1]

[1] Grupo de Investigación en Integración y Re-Ingeniería de Sistemas (IRIS), Dept. de Llenguatges i Sistemes Informàtics, Universitat Jaume I, 12071 Castelló, Spain
{grangel, rchalmet, camposc}@uji.es

Abstract. Knowledge representation is a multidisciplinary subject that needs to apply theories and techniques from logic, ontology and computation. There are a great number of languages able to represent knowledge, among them UML can be considered a suitable language for modelling knowledge. It could be included within the category of visual languages for knowledge representation.
On the other hand, numerous efforts are being carried out in the context of Enterprise Modelling to improve the capacity of enterprise models for externalising enterprise knowledge. The Proposal presented in this paper combines both approaches, UML and Enterprise Modelling, in order to make possible Enterprise Knowledge Modelling using UML. It shows a summary of this Proposal describing its principles, the main steps of its development and an example of one of the UML Profiles implemented with the objective of modelling knowledge.

1 Knowledge Modelling Perspective

In [1], it is stated that knowledge representation is a multidisciplinary subject that needs to apply theories and techniques from **logic** to provide a formal structure and rules of inference; **ontology**, to define the types of things that exist in the application domain; and **computation**, to support the applications that distinguish knowledge representation from pure philosophy. Moreover, according to [2], there is no single best theory or language for knowledge representation; rather, it is necessary to choose the technique(s) that can be best adapted for each kind of knowledge (procedural, declarative, metaknowledge, heuristic, etc.). The traditional **techniques used in Artificial Intelligence** for knowledge representation are the following [2]:

- **Object-Attribute-Value-Triplets:** these are used to represent facts about objects and their attributes; they state the value of an attribute of an object.
- **Uncertain Facts:** this is an extension of the previous O-A-V technique to allow uncertainty of facts to be described.
- **Fuzzy Facts:** these represent uncertainty using the imprecise and ambiguous terms of the natural language.
- **Rules:** these relate one or more premises or situations to one or more conclusions or actions.

- **Semantic networks or concept maps:** these attempt to reflect cognition (following the psychological model of the human associative memory) by means of graphs that include objects, concepts and situations for a specific domain of knowledge.
- **Frames:** these represent stereotypical knowledge of certain concepts or objects.
- **Ontologies:** these represent a set of knowledge terms, including the vocabulary, the semantic interconnections and some simple rules of inference and logic, from a particular topic.

These knowledge representation techniques are supported by different **knowledge representation languages**, which are used to represent knowledge in a KMS. A knowledge representation language should be able to represent entities, events, actions, processes and time from syntactic and semantic points of view. An overview of the existing paradigms is given in [2]: (1) Logic-Based Representation Languages, (2) Frame-Based Representation Languages, (3) Rule-Based Representation Languages, (4) Visual Languages for Knowledge Representation, (5) Natural Languages and Knowledge Representation, and (6) Ontology Knowledge Representation. In the fourth category, **UML** is pointed out as being a suitable language for knowledge representation, even though it was originally developed for the software engineering domain. This is one of the starting points for the research shown in this paper.

On the other hand, the second is **Enterprise Modelling** defined as the externalisation and expression of enterprise knowledge [3], which provides a holistic view of an enterprise and considers all its dimensions, i.e. process, decision, information, behaviour, resources and so forth [4]. Both, UML and Enterprise Modelling, are the basis of the Proposal presented in this paper to model Enterprise Knowledge.

The paper is organised as follows. **Section 2** outlines the analysis of the state of the art related to the basis of the Proposal, the main problems in the context of Enterprise Modelling and existing UML approaches to model enterprises and knowledge. In **section 3**, the Proposal for Enterprise Knowledge Modelling using UML, and one of the UML Profiles implemented are presented. Finally, **section 4** outlines the main conclusions.

2 Background for Enterprise Knowledge Modelling

According to [5] enterprise knowledge can been seen as information made actionable in a way that adds value to the enterprise. Taking this context into account, we defined **enterprise knowledge** as the network of connections among data and information that gives the people involved in an enterprise and insight into its workings and enables them to act and to make decisions that add value to the enterprise [6]. In the sections that follow, an overview of Enterprise Modelling and UML are presented, taking into account the main problems and approaches related to Enterprise Knowledge.

2.1 Enterprise Modelling Framework

Enterprise Modelling has been used for a long time to select and develop computer systems, to better understand and improve business processes, to support decision-making, for example [3, 7, 8]. Many languages, standards, methodologies and tools for Enterprise Modelling have emerged since the 70s, when the first concepts of modelling were applied to computer systems (E/R Model, DFD, and so forth), and modelling concepts and techniques are now applied not only to information systems but to the whole enterprise [7].

Nowadays, there are a great number of languages, standards, methodologies and their corresponding tools, such as GRAI [9], IEM [10], MEML [8], or IDEF [11], which are classified as **traditional Enterprise Modelling Languages** (EMLs) in [12]. These EMLs cover different dimensions of the enterprise defined in GERAM [13] and they can even overlap. Moreover, other **EMLs** exist that have been **created in order to make different kinds of exchanges easier**, since interoperability problems are increasing among systems that use different EMLs [14]. This last category could also be considered to be EMLs, and, among them UEML[1] [7, 17] and POP*[2] [19, 20] provide common exchange formats to smooth the exchange of enterprise models at a horizontal level. Finally, another category is made up of the **EMLs that are based on standards** such as XML or UML, and they can be used as EMLs [12].

Therefore, taking into account that the problem of interoperability is being solved by initiatives like **UEML** and **POP***, the most important benefit of enterprise models is their capacity to add value to the enterprise. This is due to the fact that such models are able to make facts and **knowledge explicit** so that they can be shared by users and different enterprise applications in order to improve enterprise performance [3, 7, 8]. One of the tasks that has still to be solved in this domain is how to achieve dynamic, interactive enterprise models that are capable of capturing enterprise knowledge and making it explicit [21].

On the other hand, one of the main weaknesses of Enterprise Modelling is the lack of strong links between enterprise models and **software generation**. For these reasons, some enterprises, especially SMEs, implement few enterprise models and, if they do, it is very hard for them to maintain them or to use them to generate software [12]. One solution, as pointed out in [22], is that the role of enterprise models should be that of facilitating the design, analysis and operation of the enterprise according to models, i.e. it should be driven by models (model-driven). Nowadays, the model-driven approach is followed by numerous projects such as MODELWARE [23], ATHENA [18], and INTEROP [16] in the European Union, and Model Driven Architecture (MDA) [24], which is carried out by the OMG.

MDA is an emerging paradigm. A lot of work is being carried out within the OMG framework in relation to PIMs, PSMs, QVT, and so forth, but the

[1] Unified Enterprise Modelling Language, first developed by the UEML Thematic Network [15] and currently being worked on by INTEROP NoE [16].

[2] Acronym of the different enterprise dimensions: Process, Organisation, Product, and so on (represented by a star), proposed by ATHENA IP [18].

characterisation of CIMs and the features that an enterprise model must satisfy in order to be considered a CIM and generate appropriate software are still in progress [12]. The main problem involved in modelling enterprises at the CIM level is how to accomplish a clear definition of the various aspects that the actors want to take into account. The domain and purpose of modelling, together with the aspects that must be highlighted, should be defined, and then the most suitable EML should be chosen [25]. Therefore, the number of issues that can be modelled at the CIM level increases the complexity of CIM models and their transformations, especially when the final aim is to capture enterprise knowledge.

2.2 UML for Enterprise Knowledge Modelling

The **Unified Modeling Language** (UML) has become a standard visual language for object-oriented modelling that has been used successfully for modelling information systems in very different domains [26]. However, UML is a general-purpose modelling language that can also be useful for modelling other types of systems such as, for example, an enterprise [27, 28]. Other works, such as [29], point out the possibility of using UML as a language for **Enterprise Modelling**, even though in [30] it is explained how and under which conditions this can be performed. However, the benefits of model-driven approaches and the new specification of UML2 provided by the OMG suggest the need to provide more practical examples for Enterprise Modelling with UML based on these recent works [31], and especially for Enterprise Knowledge Modelling. In this line, some works, like [32], has been carried out following the possibility suggested in the previous section, but this proposal is not enterprise oriented and thus it does not take into account the different enterprise dimensions for modelling [33, 13].

Furthermore, despite the fact that the weakness of the **stereotype mechanism** is pointed out in [30], the new specification of UML2 [26] provides profiles that are more complete than version 1.5 [34]. It will therefore be possible to customise UML in a better way [35]. For instance, UML provides a lot of diagrams for modelling dynamic aspects (but not for direct modelling of business processes) in a similar way to how they are represented in an IDEF diagram. Hence, business process modelling with UML is complex [36] and the use of profiles according to UML2 can make this task easier.

Taking into account the state of the problems and solutions analysed in this section, the objective of the research presented in this paper was to consider **the possibility of using UML as a knowledge representation language** on the basis of two positive factors: first, that it is a visual language which has become a standard object-oriented language and thus there are a lot of tools available on the market; and, second, that it is commonly used by engineers in enterprises for software development. To make this possible, the capacity of UML 2.0 to extend the language to a specific domain was used. A UML2 Profile for Enterprise Knowledge Modelling was then defined in an attempt to achieve a common understanding within the context of Enterprise Modelling.

3 Proposal for Enterprise Knowledge Modelling

The objective of the Proposal presented in this paper is to represent Enterprise Knowledge at the CIM level and, thus, the result of using it in enterprises is a graphical model of the **Enterprise Knowledge Map**.

In general terms, this proposal is based on Model Driven Engineering (MDE), which promotes the design and development of computer systems using different types of models built at different levels of abstraction. More particularly, it is also supported by the MDA defined by the OMG, which is an instantiation of MDE. According to this approach, the process of developing a computer system is based on the separation of the functional characteristics of the system from the details of its specification on a specific platform. Therefore, the Proposal defines a **framework for modelling enterprise knowledge on the CIM level** at two levels of abstraction, which are required due to the great complexity of this level (see Table 1):

1. **CIM-Knowledge:** this corresponds to the top level of the model at the CIM level; the enterprise is represented from a holistic point of view, thus providing a general vision of the enterprise focused on representing enterprise knowledge that will later be detailed in a local way in successive lower levels.
2. **CIM-Business:** here, the vision of enterprise knowledge is detailed by means of a representation of its business, according to three types of models, i.e. the Organisational, the Structure and the Behaviour Models.

Table 1. Proposal for Enterprise Knowledge Modeling.

Abstraction Level	Metamodel	UML Profile	Model	Diagram
CIM-Knowledge	Knowledge	UML Profile for KM	Knowledge	Blocks Ontological Knowledge
CIM-Business	Organisation	UML Profile for GM UML Profile for OSM UML Profile for AM UML Profile for BRM	Organisation	Goals Organisational Structure Analysis Business Rules
	Structure	UML Profile for SM	Structure	Product Resource
	Behaviour	UML Profile for BM	Behaviour	Process Service

The Proposal follows the MDE premise as a fundamental concept together with the following principles, it is focused on **Enterprise Modelling**, since it takes into account enterprise dimensions and previous works leading to initiatives such as UEML and POP*; and it is a **user-oriented modelling framework**, since it should be developed at the CIM level by domain experts. A summary of the Proposal with its abstraction levels, metamodels and profiles developed, as well as models and diagrams proposed for each level are shown in Table 1.

3.1 Steps for Developing the Proposal

From a technological point of view, this proposal was implemented using the capacity of UML2 to extend a metamodel, that is to say, using a UML2 Profile.

The **UML2 Profile** was defined for **Enterprise Knowledge Modelling** at the CIM level, following an MDA approach and the principles detailed above. The Profile provides the constructs needed to perform the models proposed earlier (see Table 1), and it was developed following these steps:

1. **Definition of the metamodels** shown in Table 1, with the objective of representing at conceptual levels the elements used for Enterprise Knowledge Modelling.
2. **Definition of the models and diagrams** that can be used to obtain the enterprise knowledge map. Figure 1 shows the models and diagrams defined within the Proposal by means of a Class Diagram.

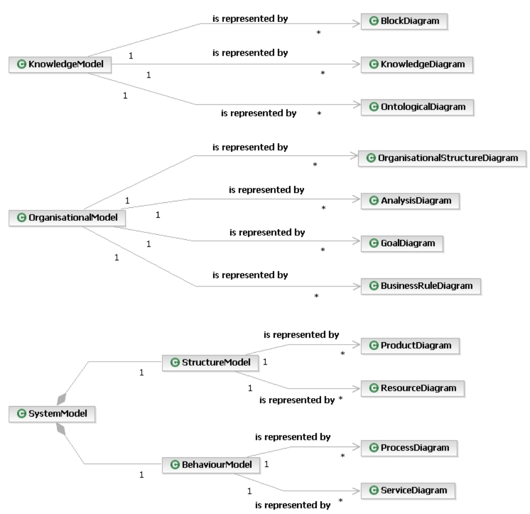


Fig. 1. Models and diagrams defined within the Proposal.

3. **Definition of the UML Profile for Enterprise Knowledge Modeling**, following for each of the profiles detailed in Table 1 the steps that follows:
   – Definition of stereotypes, tagged values and constraints of the profile.
   – Extension of the metaclasses of the UML2 Metamodel.
   – Detail description of the profile.
4. **Implementation of the Profile** using a UML tool (IBM Rational Software Modeler [37] or MagicDraw [38] for example). Figure 2 shows, as an example, one of the profiles that makes up of the **UML Profile for Enterprise Knowledge Modelling**.
5. **Validation of the Profile** by means of real case study.

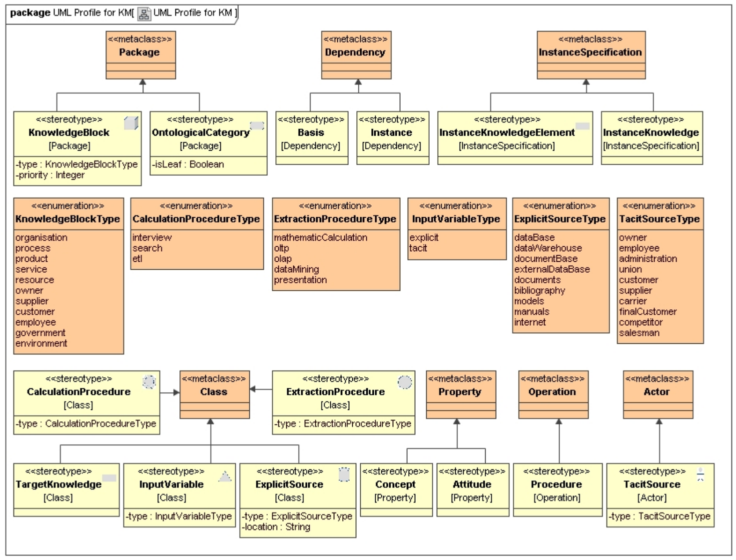


Fig. 2. Diagram of the 'UML Profile for KM'.

3.2 Discussion on the 'UML Profile for KM'

Figure 2 shows the diagram of the implemented **'UML Profile for KM'** by means of the MagicDraw UML 2.0 [38]. In this section, some relevant comments about this UML Profile are provided, taking into account that the mapping between the constructs of the metamodel proposed to implement this Profile and the constructs of the Profile is not always one to one. The main reason is that there exist some elements that it is necessary to represent at conceptual level in the metamodel for example for a better understanding, but then again it is not needed to represent them in a specific implementation, such as stereotypes in a UML Profile.

- In the proposed metamodel the class **OntologicalCategory** has a property named **isLeaf** to indicate that one element is a leaf within the ontological hierarchy. It is necessary to add this property in the UML Profile as a tagged value, since the stereotype **OntologicalCategory** extends the metaclass **Package**, and this is not a subclass of **RedefinableElement**. In the case, that the extended class of UML2 Metamodel was the metaclass **Class**, which is a subclass of **RedefinableElement**, and therefore inherits the property **isLeaf** that has the class **RedefinableElement**, it would not be necessary to add this property.
- **KnowledgeSource** in the proposed metamodel owns two subclasses, **ExplicitSource** and **TacitSource**, therefore we could add it in the Profile as an abstract stereotype with the aim of having a superclass of the stereotypes **ExplicitSource** and **TacitSource**, and in this way both could inherit its tagged values. However, **KnowledgeSource** has not been added as an abstract stereotype, since in this case there is not any property that we need

to add as tagged value. Moreover, the two subclass in the metamodel, **ExplicitSource** and **TacitSource**, extend as you can see in Figure 2 distinct UML2 metaclasses.

- **Concept** and **Attitude** extend **Property** since they are features of the target knowledge from structural point of view, whereas **Procedure** extend **Operation** since it represents the behaviour which is needed to learn concerning an specific knowledge.
- **Basis** is added as an stereotype that extends **Dependency** in order to model the reflexive relationships that exist in the metamodel for the classes **KnowledgeBlock**, **OntologicalCategory** and **TargetKnowledge**, and also the relationships between **KnowledgeBlock** and **OntologicalCategory**, and so on. In this way, it is possible for example to represent the relationship between target knowledge or between ontological categories and target knowledge.
- **Instance** is added as an stereotype that extends **Dependency** in order to model the relationship between target knowledge and its instances. Taking into account that the stereotype **TargetKnowledge** extends the metaclass **Class**, and the stereotype **InstanceKnowledge** the metaclass **InstanceSpecification**, both elements can only be linked by means of a dependency, which has been stereotyped in this case to represent one of the main features of the Proposal presented in this paper, that is to say, the instantiation of target knowledge.

4 Conclusion

The benefits of this proposal with respect to other proposals for Enterprise Modelling or Knowledge Representation could be summarised as follows:

- It provides a graphical model for representing knowledge that allows employees who are not specialised in knowledge engineering to gain a better understanding of the enterprise and its operations from a knowledge point of view. This is a feature that other non-visual representation languages cannot provide.
- Regarding traditional Enterprise Modelling, this proposal is knowledge-oriented and based on a model-driven approach in order to implement a KMS, but at the same time it takes into account traditional enterprise dimensions, such as organisation, process, product, and so forth.
- Enterprises can use a formalism, like UML, that is well known by engineers and is normally used to develop software. They are therefore familiar with the use of this modelling language, as well as with the corresponding development process and the tools that are currently available.
- It is possible to have a number of commercial tools at one's disposal for implementing the profile which can support the process of modelling and model management.

Acknowledgments

This work was funded by CICYT DPI2003-02515, CICYT DPI2006-14708 and the European Commission within the 6^{th} Framework Programme, INTEROP NoE (IST-2003-508011). The authors are indebted to TG2 [16].

References

1. Sowa, J.F.: Knowledge Representation. Logical, Philosophical, and Computational Foundations. Books/Cole (2000)
2. Gasevic, D., Djuric, D., Devedzic, V.: Model Driven Architecture and Ontology Development. Springer (2006)
3. Vernadat, F.B.: Enterprise Modeling and Integration: Principles and Applications. Chapman and Hall (1996)
4. Doumeingts, G., Chen, D.: Interoperability development for enterprise applications and software. In Cunningham, P., Cunningham, M., Fatelnig, P., eds.: Building the Knowledge Economy: Issues, Applications, Case Studies. eBusiness, IOS Press Amsterdam (2003)
5. Vail, E.F.: Knowledge mapping: getting started with knowledge management. Information Systems Management **Fall** (1999) 16–23
6. Grangel, R., Chalmeta, R., Campos, C.: Defining of Target Knowledge in Virtual Enterprise. In Abramowicz, W., Mayr, W., eds.: Business Information Systems - 9th International Conference on Business Information Systems (BIS 2006), Gesellschaft für Informatik (GI) (2006)
7. UEML: Unified Enterprise Modelling Language Project (IST-2001-34229). http://www.ueml.org (2004) Deliverable 1.1.
8. EXTERNAL: Extended Enterprise MEthodology, Final version 1-12-D-2002-01-0 (IST-1999-10091). http://research.dnv.com/external/default.htm (2002)
9. Doumeingts, G., Vallespir, B., Zanittin, M., Chen, D.: GIM-GRAI Integrated Methodology, a Methodology for Designing CIM Systems, Version 1.0. LAP/GRAI, University Bordeaux 1, Bordeaux, France (1992)
10. Spur, G., Mertins, K., Jochem, R.: Integrated Enterprise Modelling. Beuth Verlag GmbH (1996)
11. IDEF: Integrated DEFinition Methods. http://www.idef.com/ (2006)
12. Grangel, R., Chalmeta, R., Campos, C.: Enterprise Modelling, an overview focused on software generation. In Panetto, H., ed.: Interoperability of Enterprise Software and Applications Workshops of the INTEROP-ESA International Conference EI2N, WSI, ISIDI and IEHENA 2005, Hermes Science Publishing (2005)
13. IFIP-IFAC: Generalised enterprise reference architecture and methodology (GERAM). Technical Report Version 1.6.3 (1999) http://www.cit.gu.edu.au/ bernus/taskforce/geram/versions.
14. Chen, D., Doumeingts, G.: European initiatives to develop interoperability of enterprise applications–basic concepts, framework and roadmap. Annual Reviews in Control **27** (2003) 153–162
15. UEML: Unified Enterprise Modelling Language Thematic Network (IST-2001-34229). http://www.ueml.org (2006)
16. INTEROP: Interoperability Research for Networked Enterprises Applications and Software NoE (IST-2003-508011). http://www.interop-noe.org (2006)
17. Opdahl, A., Henderson-Sellers, B.: A Template for Defining Enterprise Modelling Constructs. Journal of Database Management **15(2)** (2004)

18. ATHENA: Advanced Technologies for interoperability of Heterogeneous Enterprise Networks and their Applications IP (IST-2001- 507849). http://www.athena-ip.org (2006)
19. Ohren, O.P.: Deliverable DA1.3.1. Report on Methodology description and guidelines definition. Technical report, ATHENA (Advanced Technologies for interoperability of Heterogeneous Enterprise Networks and their Applications) Project (IST-2003-2004) (2005)
20. Grangel, R., Chalmeta, R., Schuster, S., Penya, I.: Exchange of Business Process Models using the POP* Meta-model. In: International Workshop on Enterprise and Networked Enterprises Interoperability (ENEI'2005). Volume 3812/2005 of Lecture Notes in Computer Science., Springer-Verlag (2005)
21. Lillehagen, F., Dehli, E., Fjeld, L., Krogstie, J., Jorgensen, H.: Utilizing Active Knowledge Models in an Infrastructure for Virtual Enterprises. In: PRO-VE. (2002) 353–360
22. Fox, M.S., Gruninger, M.: Enterprise Modelling. AI Magazine **19** (1998) 109–121
23. MODELWARE: Modeling solution for software systems Project (IST-2004-511731). http://www.modelware-ist.org/ (2006)
24. OMG: MDA Guide Version 1.0.1. Object Management Group. Document number: omg/2003-06-01 edn. (2003)
25. Grangel, R., Ben-Salem, R., Bourey, J., Daclin, N., Ducq, Y.: Transforming GRAI Models into UML Models, a First Step to Model Driven Interoperability. In: evaluation 3rd Interoperability for Enterprise Software and Applications Conference (I-ESA'07). (2007)
26. OMG: Unified Modeling Language: Superstructure, version 2.1.1. Object Management Group. version 2.1.1 formal/2007-02-05 edn. (2007)
27. Eriksson, H., Penker, M.: Business Modeling with UML: Business Patterns at Work. J. Wiley (2000)
28. Marshall, C.: Enterprise Modeling with UML. Designing Successful Software Through Business Analysis. Addison Wesley (2000)
29. Panetto, H.: UML Semantics Representation of Enterprise Modelling Constructs. In: ICEIMT. (2002) 381–387
30. Berio, G., Petit, M.: Enterprise Modelling and the UML: (sometimes) a conflict without a case. In: Proc. of 10th ISPE International Conf. on Concurrent Engineering: Research and applications. (2003) 26–30
31. Grangel, R., Bourey, J.P., Chalmeta, R., Bigand, M.: UML for Enterprise Modelling: a Model-Driven Approach. In: Interoperability for Enterprise Software and Applications Conference (I-ESA'06). (2006)
32. Abdullah, M., Evans, A., Benest, I., Kimble, C.: Developing a UML Profile for Modelling Knowledge Based Systems (2006)
33. Berio, G., Vernadat, F.B.: New developments in enterprise modelling using CIMOSA. Computers in Industry **40** (1999) 99–114
34. OMG: OMG Unified Modeling Language Specification, version 1.5. Object Management Group. formal/03-03-01 edn. (2003)
35. Fuentes, L., Vallecillo, A., Troya, J.: Using UML Profiles for Documenting Web-Based Application Frameworks. Annals of Software Engineering **13** (2002) 249–264
36. Noran, O.: UML vs. IDEF: An Ontology-Oriented Comparative Study in View of Business Modelling. In: ICEIS (3). (2004) 674–682
37. IBM: IBM Rational Software Modeler Development Platform 6.0.1. http://www-306.ibm.com/software/rational/ (2007)
38. NoMagic: MagicDraw UML 12.0. http://www.magicdraw.com/ (2007)

Chapter 6

Application of the Proposal for MDK

Title:	**A Proposal for Goal Modelling Using a UML Profile**
Authors:	R. Grangel and R. Chalmeta and C. Campos and J-P. Bourey
Conference:	26th International Conference on Conceptual Modeling (ER 2007) - First International Workshop on Requirements, Intentions and Goals in Conceptual Modeling (RIGiM 2007)
Publisher:	Springer Verlag Berlin (USA)
Vol:	LNCS
JCR:	COMPUTER SCIENCE, THEORY & METHODS (0,402; 62; 38.975) (data of 2005)
Place:	Auckland (New Zeland)
Status:	In evaluation

Abstract

UML has become the standard object-oriented language for modelling systems in Software Engineering domain. More and more relationships are being established between this domain and Enterprise Modelling context. Some recent research works, such as GORA methods and MDE approaches, suggest the interest in providing concrete methods and mechanisms to make possible the needed link between enterprise's goals and the requirements defined to develop the computer system.

UML is a good candidate to connect these two levels, that is to say, CIM level

and PIM level from a MDA perspective. In this paper, we present a Proposal for Enterprise Goal Modelling based on UML, which is focused on modelling knowledge. This Proposal is developed at the CIM level and presents different models to capture software requirements at the CIM level. In particular, the metamodel concerning goal dimension and the UML Profile implemented from it are shown. Finally, the resulting Goal Diagram is explained by means of an example.

A Proposal for Goal Modelling Using a UML Profile

Reyes Grangel[1], Ricardo Chalmeta[1], Cristina Campos[1], and Jean-Pierre Bourey[2]

[1] Grupo de Investigación en Integración y Re-Ingeniería de Sistemas (IRIS), Dept. de Llenguatges i Sistemes Informàtics, Universitat Jaume I, 12071 Castelló, Spain
{grangel, rchalmet, camposc}@uji.es
[2] Laboratoire de Génie Industriel de Lille, Ecole Centrale de Lille, 59561 Villeneuve d'Ascq Cedex, France
jean-pierre.bourey@ec-lille.fr

Abstract. UML has become the standard object-oriented language for modelling systems in Software Engineering domain. More and more relationships are being established between this domain and Enterprise Modelling context. Some recent research works, such as GORA methods and MDE approaches, suggest the interest in providing concrete methods and mechanisms to make possible the needed link between enterprise's goals and the requirements defined to develop the computer system.
UML is a good candidate to connect these two levels, that is to say, CIM level and PIM level from a MDA perspective. In this paper, we present a Proposal for Enterprise Goal Modelling based on UML, which is focused on modelling knowledge. This Proposal is developed at the CIM level and presents different models to capture software requirements at the CIM level. In particular, the metamodel concerning goal dimension and the UML Profile implemented from it are shown. Finally, the resulting Goal Diagram is explained by means of an example.

1 Introduction

Enterprise Modelling refers to the externalisation and expression of enterprise knowledge [1], which provides a holistic view of an enterprise and considers all its dimensions, i.e. process, decision, information, behaviour, resources and so forth [2]. Nowadays, there are a great number of languages, standards, methodologies and their corresponding tools, such as GRAI [3], IEM [4], MEML [5], IDEF [6], etc.

On the other hand, **Unified Modeling Language** (UML) has become a standard language for object-oriented modelling that has been used successfully for modelling software systems in very different domains [7]. However, UML is a general-purpose modelling language that can also be useful for modelling other types of systems such as, for example, an enterprise [8, 9]. Other works, such as [10], point out the possibility of using UML as a language for **Enterprise Modelling**, even though in [11] it is explained how and under which conditions this can be performed. However, the benefits of model-driven approaches and the new specification of UML2 suggest the need to provide more practical examples for Enterprise Modelling with UML based on these recent works [12], and especially for Enterprise Knowledge Modelling. In this line, some works, like [13], has been carried out, but this proposal is not enterprise oriented and thus it does not take into account the different enterprise dimensions for modelling [14].

The main weaknesses of Enterprise Modelling is the lack of strong links between enterprise models and software generation [15]. One solution, as pointed out in [16], is that the role of enterprise models should be that of facilitating the design, analysis and operation of the enterprise according to models, i.e. it should be driven by models (model-driven). In this context, UML is a good candidate to establish the needed links between enterprise models and systems models in general, and requirement engineering in particular, using the extension mechanism of UML Profiles.

Taking into account this context, the objective of the research presented in this paper was to consider **the possibility of using UML for Enterprise Modelling** with two objectives: first, to provide an extension of UML, one of the modelling languages used commonly by engineers to develop software, focused on representing enterprise goals; and second, to establish the basis for connecting enterprise goals and systems models. To make this possible, the capacity of UML2 to extend the language to a specific domain was used, and a UML2 Profile for Enterprise Goal Modelling was implemented.

The paper is organised as follows. **Section 2** outlines two approaches related to the aim of establishing connections between enterprise and system models. In **section 3**, the Proposal for Enterprise Knowledge Modelling using UML is described. **Section 4** presents one of the metamodels and UML Profiles implemented in this Proposal, in particular which is related to goal dimension. Finally, **section 5** outlines the main conclusions.

2 Linking Enterprise Models and System Models

Linking enterprise models in general, and enterprise goals and strategies in particular, to the first step for software development, that is to say, requirements elicitation, is one of the recent research trends bridging Enterprise Modelling and Software Engineering domains. This section gives a brief summary of two initiatives developed to bridge enterprise and system models.

2.1 MDA

Model-driven approaches are a good solution to try to put to rights the shortcomings of Enterprise Modeling for generating code from enterprise models. Model Driven Engineering (MDE) or Model Driven Development (MDD) approaches are a new paradigm in the context of Software Engineering. Such perspective attempt to improve the software development process by focusing on models as the primary artifacts and transformations as the primary operation carried out on models (which are used to map information from one model to another). As a result, they may have important consequences on the way information systems are built and maintained [17, 18].

As an example, Model Driven Architecture (MDA) defined by the OMG [19], is intended to promote the use of models and their transformations as a fundamental way of designing and implementing different kinds of systems. The main purpose of this approach is to separate the functional specification of a system from the details of its implementation on a specific platform. This architecture therefore defines a hierarchy of models from three points of view [15, 19, 20], namely:

- **Computation Independent Model (CIM):** used to represent domain and system requirements. It is based on business models and shows the enterprise from a holistic point of view, that is independent of the computation.
- **Platform Independent Model (PIM):** used to model system functionality but without defining how it will be implemented and on what platform; it is focused on information and sets out from a computational point of view.
- **Platform Specific Model (PSM):** the PIM is transformed into a platform-dependent model according to the platform selected for use and is focused on a technological point of view.

Nowadays, the model-driven approach is followed by numerous projects such as MODELWARE [21], ATHENA [22], and INTEROP [23] in the European Union, and Model Driven Architecture (MDA) [19] carried out by the OMG.

MDA is an emerging paradigm. A lot of work is being carried out within the OMG framework in relation to PIMs, PSMs, and so forth, but the characterisation of CIMs and the features that an enterprise model must satisfy in order to be considered a CIM and generate appropriate software are still in progress [15]. The main problem involved in enterprise modelling at the CIM level is how to accomplish a clear definition of the various aspects that the actors want to take into account. The domain and purpose of modelling, together with the aspects that must be highlighted, should be defined, and then the most suitable Enterprise Modelling Language (EML) should be chosen [24]. Therefore, the number of issues that can be modelled at the CIM level increases the complexity of CIM models and their transformations, especially when the final aim is to capture enterprise knowledge.

2.2 GORA Methods

At the same time, another areas of research have emerged that recognise the importance of guaranteeing requirements quality by goals, especially Goal Oriented Requirements Analysis (GORA) methods aiming at bridging the gaps between stakeholders needs and requirements specifications [25]. These methods use mainly progressive top-down approaches [26–28]. They start from the definition of the customers needs and, by refining and decomposing the needs into more concrete goals, make it possible the elicitation of the system requirements by a top-down approach. The result is generally structured as a directed

AND-OR graph. Its upper parts show the needs and its lower parts show the requirements. These approaches can be combined or weaved with use case modelling techniques [29–31, 25] in order to get a clear connection between the goal-oriented methods and the requirements elicitation processes. For example, [25] proposes such an approach enabling the support of collaborative tasks and a goal decomposition from multiple perspectives.

All these methods and techniques are devoted to information systems and software engineering but are not limited to them and can be used in a broader context such as Enterprise Modelling. As pointed out by [32] enterprise modelling is in connection with requirements engineering and Goal-oriented approaches can be used in this context. In this way, it is possible to establish links between goals of the enterprise defined at several levels of granularity, for example from strategic, tactic, and operative level, and the requirements system to be implemented in order to reach these goals. However, one weakness of the these approaches is that they generally use different formalisms at the enterprise level for expressing strategic goals, and at the IT system development level. For example, a specific formalism is developed in [32] for describing a Strategic Dependency model. In the following sections, a Proposal founded on the definition of a UML Profile, which allows to develop an integrated approach based on a unique formalism, is presented.

3 Proposal for Enterprise Knowledge Modelling

The research presented on this paper focuses on Enterprise Goal Modelling. It belongs to a wider research project [33] which aims at modelling Enterprise Knowledge at the CIM level and, thus, the result of using it in enterprises is a graphical model of the **Enterprise Knowledge Map**.

In general terms, this Proposal is based on Model Driven Engineering (MDE) and, more particularly, on the MDA defined by the OMG. According to this approach, the process of developing a computer system is based on the separation of the functional characteristics of the system from the details of its specification on a specific platform. Therefore, the Proposal defines a **framework for modelling enterprise knowledge on the CIM level** at two levels of abstraction, which are required due to the great complexity of this level (see Table 1):

1. **CIM-Knowledge:** this corresponds to the top level of the model at the CIM level; the enterprise is represented from a holistic point of view, thus providing a general vision of the enterprise focused on representing enterprise knowledge that will later be detailed in a local way in successive lower levels.
2. **CIM-Business:** here, the vision of enterprise knowledge is detailed by means of a representation of its business, according to three types of models, i.e. the Organisational, the Structure and the Behaviour Model.

The Proposal follows the MDE premise as a fundamental concept together with the following principles, it is focused on **Enterprise Modelling**, since it

Table 1. Proposal for Enterprise Knowledge Modeling.

Abstraction Level	Metamodel	UML Profile	Model	Diagram
CIM-Knowledge	Knowledge	UML Profile for KM	Knowledge	Blocks Ontological Knowledge
CIM-Business	Organisation	UML Profile for GM UML Profile for OSM UML Profile for AM UML Profile for BRM	Organisation	Goal Organisational Structure Analysis Business Rules
	Structure	UML Profile for SM	Structure	Product Resource
	Behaviour	UML Profile for BM	Behaviour	Process Service

takes into account enterprise dimensions and previous works leading to initiatives such as UEML[1] [35] and POP*[2] [33]; and it is a **user-oriented modelling framework**, since it should be developed at the CIM level by domain experts. From a technological point of view, this Proposal was implemented using the capacity of UML2 to extend a metamodel, that is to say, by defining a UML2 Profile for each enterprise aspect to take into account. A summary of the Proposal showing its abstraction levels, metamodels and profiles developed, as well as models and diagrams proposed for each level are shown in Table 1.

The Proposal was developed following these steps:

1. **Definition of the models and diagrams** that can be used to obtain the Enterprise Knowledge Map. The models and diagrams defined within the Proposal are presented in Table 1.
2. **Definition of the metamodels** shown in Table 1, with the objective of representing at conceptual level the elements used for Enterprise Knowledge Modelling.
3. **Definition of the UML Profile for Enterprise Knowledge Modeling**, following for each of the profiles detailed in Table 1 these steps:
   - Definition of stereotypes, tagged values and constraints of the profile.
   - Extension of the metaclasses of the UML2 Metamodel.
   - Detailed description of the profile.
4. **Implementation of the Profile** using a UML tool (IBM Rational Software Modeler Development Platform [3] or MagicDraw UML 12.0.[4] for example).
5. **Validation of the Profile** by means of real case study.

In the next section, one of the profiles that makes up the **UML Profile for Enterprise Knowledge Modelling** is presented as an example of how goal

[1] Unified Enterprise Modelling Language, first developed by the UEML Thematic Network [34] and currently being worked on by INTEROP NoE [23].

[2] Acronym of the different enterprise dimensions: Process, Organisation, Product, and so on (represented by a star), proposed by ATHENA IP [22].

[3] http://www-306.ibm.com/software/rational/

[4] http://www.magicdraw.com/

dimension is directly taking into account in this Proposal, since there is some implicit concepts related to GORA concepts inside the other models of the Proposal. Therefore, the main steps above depicted are shown in the next section for goal dimension, that is to say, the suggested Goal Metamodel, the implemented 'UML Profile for GM', and an example to illustrate the Goal Diagram.

4 'UML Profile for GM'

The Goal Metamodel was defined with the objective of representing at conceptual level the elements related to goal dimension in enterprises. At conceptual level, the main elements that are possible to represent based on [36] are shown in Table 2.

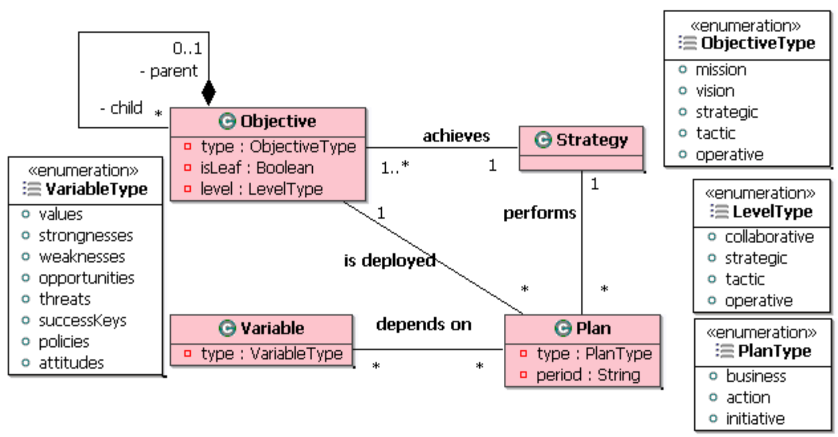


Fig. 1. Goal Metamodel: an excerpt of the Organisational Metamodel.

Figure 1 shows an excerpt of the Organisational Metamodel (the Goal Metamodel), showing only the constructs needed to represent enterprise goals defined from Table 2, which are the following:

- **Objective:** this represents any target that enterprises want to achieve, it is possible to define it at different hierarchical levels: strategic, tactic and operative. At the strategic level, this constructor is also used to represent the enterprise's mission and vision. For this class, the following properties are defined:
  - **type:** this specifies the category of the objective, which is one of the following defined in the enumeration **'ObjectiveType'**: mission, vision, strategic, tactic or operative.
  - **isLeaf:** this indicates if it is not possible to divide the objective in other subobjectives.

Constructor	Question	Target Knowledge
Objective	Why? For what?	Mission, Vision, Strategic Objectives, Tactic Objectives, Operative Objectives
Strategy	How?	Strategy, Know-how
Plan	Where?	Business Plan, Action Lines
Variable	With?	Values, Strengths, Weaknesses, Opportunities, Threats, Success Key, Policies, Attitudes

Table 2. Conceptual elements to represent in goal dimension.

  - **level:** this indicates the hierarchical levels in which the objective is defined, it is possible one of the following levels defined in the enumeration **'LevelType'**: collaborative, strategic, tactic or operative.
- **Strategy:** this represents how the enterprise wants to achieve the objectives proposed at strategical level.
- **Plan:** this represents the organisation of the work at different hierarchical levels in order to accomplish the objectives and strategy defined in the enterprise. For this class, the following properties are defined:
  - **type:** this specifies the kind of the plan, which can be one of the types defined in the enumeration **'PlanType'**: business, action or initiative.
  - **period:** this specifies the interval of time for what the plan is defined.

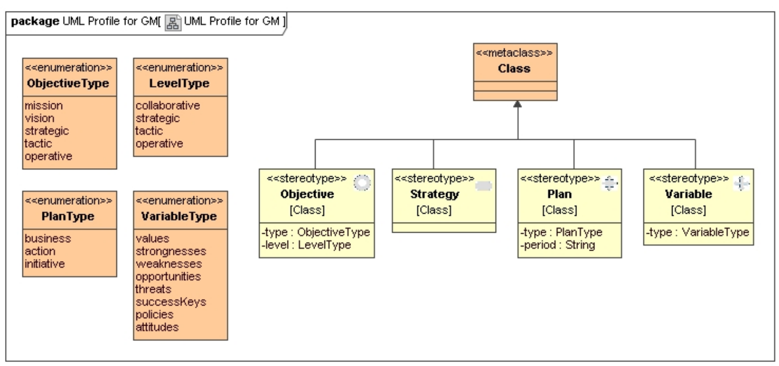


Fig. 2. Diagram of the 'UML Profile for GM'.

- **Variable:** this represents any factor that is able to make influence in the execution of the plans defined in the organisation. For this class, the following properties are defined:
  - **type:** this specifies one of the categories defined in the enumeration **'VariableType'**: values, strengths, weaknesses, opportunities, threats, successKeys, policies or attitudes.

Figure 2 shows the diagram of the implemented **'UML Profile for GM'** by means of the MagicDraw UML 12.0., which was developed from the Goal Metamodel shown in Figure 1. Finally, figure 3 shows the **Goal Diagram** for a real case, in which is possible to notice some of the needed requirements for the computer system, which can be mapped onto use case at system level.

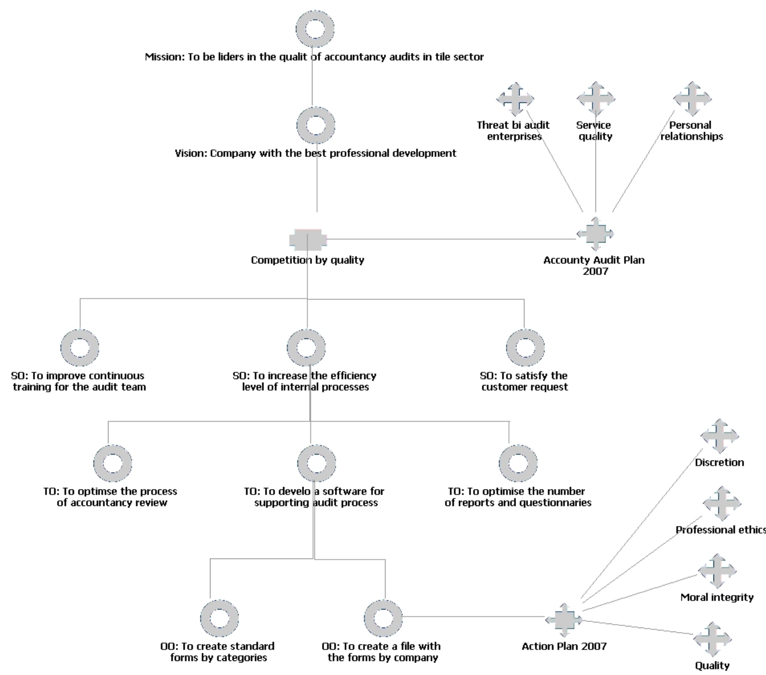


Fig. 3. Goal Diagram for an audit enterprise.

5 Conclusion

The Proposal for Goal Modelling presented in this paper is a first attempt to establish links between enterprise and system models. This Proposal is a part of a wider research work aiming at defining a set of UML profiles for bridging the Enterprise Modelling domain to the System Development domain. Combining the main advantages of using a common basic formalism (i.e. UML), with its adaptation to specific concerns and viewpoints through the definition of UML Profiles, and, at last, with a MDA approach makes it easier the definition of links between models at enterprise level and at system level.

Acknowledgments

This work was funded by CICYT DPI2006-14708 and the EC within the 6^{th} FP, INTEROP NoE (IST-2003-508011). The authors are indebted to TG2 [23].

References

1. Vernadat, F.B.: Enterprise Modeling and Integration: Principles and Applications. Chapman and Hall (1996)
2. Doumeingts, G., Chen, D.: Interoperability development for enterprise applications and software. In Cunningham, P., Cunningham, M., Fatelnig, P., eds.: Building the Knowledge Economy: Issues, Applications, Case Studies. eBusiness, IOS Press Amsterdam (2003)
3. Doumeingts, G., Vallespir, B., Chen, D.: Decisional modelling GRAI grid. In: International Handbook on Information Systems. Springer-Verlag (1998) 313–337
4. Spur, G., Mertins, K., Jochem, R.: Integrated Enterprise Modelling. Beuth Verlag GmbH (1996)
5. EXTERNAL: Extended Enterprise MEthodology, Final version 1-12-D-2002-01-0 (IST-1999-10091). http://research.dnv.com/external/default.htm (2002)
6. IDEF: Integrated DEFinition Methods. http://www.idef.com/ (2007)
7. OMG: Unified Modeling Language: Superstructure, version 2.1.1. Object Management Group. version 2.1.1 formal/2007-02-05 edn. (2007)
8. Eriksson, H., Penker, M.: Business Modeling with UML: Business Patterns at Work. J. Wiley (2000)
9. Marshall, C.: Enterprise Modeling with UML. Designing Successful Software Through Business Analysis. Addison Wesley (2000)
10. Panetto, H.: UML Semantics Representation of Enterprise Modelling Constructs. In: ICEIMT. (2002) 381–387
11. Berio, G., Petit, M.: Enterprise Modelling and the UML: (sometimes) a conflict withouta case. In: Proc. of 10th ISPE International Conf. on Concurrent Engineering: Research and applications. (2003) 26–30
12. Grangel, R., Bourey, J.P., Chalmeta, R., Bigand, M.: UML for Enterprise Modelling: a Model-Driven Approach. In: Interoperability for Enterprise Software and Applications Conference(I-ESA'06). (2006)
13. Abdullah, M., Evans, A., Benest, I., Kimble, C.: Developing a UML Profile for Modelling Knowledge Based Systems (2006)
14. IFIP-IFAC: Generalised Enterprise Reference Architecture and Methodology (GERAM).Technical Report Version 1.6.3 (1999) http://www.cit.gu.edu.au/ bernus/taskforce/geram/versions.
15. Grangel, R., Chalmeta, R., Campos, C.: Enterprise Modelling, an overview focused on software generation. In Panetto, H., ed.: Interoperability of Enterprise Software and Applications Workshopsof the INTEROP-ESA International Conference EI2N, WSI, ISIDI and IEHENA 2005, Hermes Science Publishing (2005)
16. Fox, M.S., Gruninger, M.: Enterprise Modelling. AI Magazine **19** (1998) 109–121
17. Berre, A., Hahn, A., Akehurst, D., Bezivin, J., Tsalgatidou, A., Vermaut, F., Kutvonen, L., Linington, P.: Deliverable D9.1: State-of-the art for Interoperability architectureapproaches, Model driven and dynamic, federated enterprise interoperabilityarchitectures and interoperability for non-functional aspects. Technical report, INTEROP-D9 (2004)
18. Aagedal, J., Bézivin, J., Linington, P.F.: Model-Driven Development. Lecture Notes in Computer Science **3344** (2005) 148–157
19. OMG: MDA Guide Version 1.0.1. Object Management Group. Document number: omg/2003-06-01 edn. (2003)
20. Berrisford, G.: Why IT veterans are sceptical about MDA. In: Second European Workshop on Model Driven Architecture

(MDA) with an emphasis on Methodologies and Transformations, Kent, Computing Laboratory, University of Kent (2004) 125–135 http://www.cs.kent.ac.uk/projects/kmf/mdaworkshop/submissions/Berrisford.pdf.
21. MODELWARE: Modeling solution for software systems Project (IST-2004-511731). http://www.modelware-ist.org/ (2007)
22. ATHENA: Advanced Technologies for interoperability of Heterogeneous Enterprise Networks and their Applications IP (IST-2001- 507849). http://www.athena-ip.org (2007)
23. INTEROP: Interoperability Research for Networked Enterprises Applications and Software NoE (IST-2003-508011). http://www.interop-noe.org (2007)
24. Grangel, R., Ben-Salem, R., Bourey, J., Daclin, N., Ducq, Y.: Transforming GRAI Models into UML Models, a First Step to Model Driven Interoperability. In: evaluation 3rd Interoperability for Enterprise Software and Applications Conference (I-ESA'07). (2007)
25. Kaiya, H., Saeki, M.: Weaving multiple viewpoint specifications in goal oriented requirements analysis. In: APSEC '04: Proceedings of the 11th Asia-Pacific Software Engineering Conference (APSEC'04), Washington, DC, USA, IEEE Computer Society (2004) 418–427
26. Antón, A.I.: Goal-based requirements analysis. In: IEEE International Conference on Requirements Engineering (ICRE '96), Colorado Springs, Colorado, USA (1996) 136–144
27. van Lamsweerde, A.: Goal-oriented requirements engineering: A guided tour. In: RE '01: Proceedings of the 5th IEEE International Symposium on Requirements Engineering, Washington, DC, USA, IEEE Computer Society (2001) 249
28. Kaiya, H., Horai, H., Saeki, M.: Agora: Attributed goal-oriented requirements analysis method, Los Alamitos, CA, USA, IEEE Computer Society (2002) 13
29. Rolland, C., Souveyet, C., Achour, C.B.: Guiding goal modelling using scenarios. IEEE Transaction on Software Engineering **24** (1998) 1055–1071
30. Anton, A., Carter, R., Dagnino, A., Dempster, J., Siege, D.: Deriving goals from a use-case based requirements specification. Requirements Engineering Journal **6** (2001) 63–73
31. Santander, V.F.A., Castro, J.: Deriving use cases from organizational modeling. In: RE '02: Proceedings of the 10th Anniversary IEEE Joint International Conference on Requirements Engineering, Washington, DC, USA, IEEE Computer Society (2002) 32–42
32. Yu, E.S.K., Liu, L., Li, Y.: Modelling strategic actor relationships to support intellectual property management. In: ER '01: Proceedings of the 20th International Conference on Conceptual Modeling, London, UK, Springer-Verlag (2001) 164–178
33. Grangel, R., Chalmeta, R., Schuster, S., Penya, I.: Exchange of Business Process Models using the POP* Meta-model. In: International Workshop on Enterprise and Networked Enterprises Interoperability (ENEI'2005). Volume 3812/2005 of Lecture Notes in Computer Science., Springer-Verlag (2005)
34. UEML: Unified Enterprise Modelling Language Themantic Network (IST-2001-34229). http://www.ueml.org (2007)
35. Opdahl, A., Henderson-Sellers, B.: Chapter 6. Template-Based Definition of Information Systems and EnterpriseModelling Constructs. In: Ontologies and Business System Analysis. Idea Group Publishing (2005)
36. Williams, T.: The Purdue Enterprise Reference Architecture. In: Proceedings of the Workshop on Design of Information Infrastructure Systems for Manufacturing, Elsevier (1993)

Bibliography

[1] R. Grangel and R. Chalmeta. A Methodological Approach for Enterprise Modelling of Small and Medium Virtual Enterprises based on UML. Application to a Tile Virtual Enterprise. In *Doctoral Symposium at First International Conference on Interoperability of Enterprise Software and Applications (INTEROP-ESA'2005)*, 2005.

[2] R. Grangel. A proposal for modelling enterprise knowledge in virtual enterprises. In *Doctoral Symposium at Second International Conference on Interoperability of Enterprise Software and Applications (INTEROP-ESA'2006)*, 2006.

[3] R. Grangel, R. Chalmeta, C. Campos, and Ò. Coltell. Enterprise Modelling, an overview focused on software generation. In H. Panetto, editor, *Interoperability of Enterprise Software and Applications Workshops of the INTEROP-ESA International Conference EI2N, WSI, ISIDI andIEHENA 2005*. Hermes Science Publishing, 2005.

[4] R. Grangel, R. Chalmeta, S. Schuster, and I. Peña. Exchange of Business Process Models using the POP* Meta-model. In *International Workshop on Enterprise and Networked Enterprises Interoperability (ENEI'2005)*, 2005.

[5] R. Grangel, J-P. Bourey, R. Chalmeta, and M. Bigand. UML for Enterprise Modelling: basis for a Model-Driven Approach.

[6] R. Chalmeta and R. Grangel. Methodology for the Implementation of Knowledge Management Systems. *In evaluation: Journal of the American Society for Information Science and Technology*, 2007.

[7] R. Grangel, R. Chalmeta, and C. Campos. Defining of Target Knowledge in Virtual Enterprise. In W. Abramowicz and W.C. Mayr, editors, *Business Information Systems - 9th International Conference on Business Information Systems (BIS 2006)*. Gesellschaft für Informatik (GI), 2006.

[8] R. Grangel, R. Chalmeta, and C. Campos. A modelling framework for sharing knowledge. In *KES 2007*, 2007.

[9] R. Grangel, R. Chalmeta, and C. Campos. A proposal to develop conceptual models of knowledge management systems. In *ER 2007*, In evaluation.

[10] R. Grangel, R. Chalmeta, and C. Campos. Using uml profiles for enterprise knowledge modelling. In *ER 2007 - FP-UML 2007*, In evaluation.

[11] R. Grangel, R. Chalmeta, C. Campos, and J-P. Bourey. A proposal for goal modelling using a uml profile. In *ER 2007 - RIGiM 2007*, In evaluation.