Chapter XLIII

KC–PLM: Knowledge Collaborative Product Lifecycle Management

João Carlos Amaro Ferreira
ISEL – Instituto Superior de Engenharia de Lisboa, Portugal & GuIAA – Grupo de Investigação em Ambientes Autónomos, Portugal

ABSTRACT

This chapter defines a system and a methodology, the Knowledge Collaborative Product Lifecycle Management (KC-PLM) to better support the complete product lifecycle in the industry. The KC-PLM system intends to reduce the lead-time from new product development to production by providing and integrating knowledge platform, based on a semantic information repository, domain ontology, a domain specific language and on the user collaboration. These characteristics differentiate the KC-PLM system from others PLM systems, because it supports an intelligent rules engine, to extrapolate and make inference with historical solutions that allow the generation of new solutions. A real case study in automobile business shows the current proposal application and its benefits in a product concept phase.

1 INTRODUCTION

The industry is growing rapidly. Globalization continues to introduce new opportunities and increases competition. Manufacturers are faced each day with a rapid globalization of markets and delocalization of suppliers; in this context, companies show a change of capital concerns: not more only physical capital (plants and equipments), but also intellectual capital (global knowledge of products and processes). Companies are trying, globally, to secure the best talent available at the most reasonable cost to serve world-wide markets and are demanding more innovations from suppliers while cutting their prices. Suppliers are asking for better working relationships and more
guarantees to justify their higher R&D expenses. Dealers want better and more extended (options) products range to sell to their customers and better training and tools for their technicians. In this increasing complexity also the expectations of customers have increased, forcing manufacturers and suppliers to develop new products frequently and each new product must be more technological and complex, but also more reliable and safer than the previous product, personalized with high quality and reasonable / low costs (i.e., value for money). As a result, new organizational approaches and new business models are required to handle the increased complexity of simultaneously managing knowledge, products, geographies and customers. Traditional bureaucratic designs that are built around vertical control and lateral segmentation are being replaced by organizational models admitting work to occur through cross-cutting processes that run across the organization. In a transnational company, dynamic configurations of teams carry out the development of products and processes, while lateral linkages coordinate and integrate diverse knowledge across dispersed knowledge centers. In other words, global new product development and production is a complex system, not simply the aggregation of multiple virtual and co-located teams. Besides, another consideration is the accelerating technology development. The rapid evolution of computing technologies has changed the automotive product development process to a “digital business” where many digital information formats, internet, pervasive computing and wireless communication are shaping the business landscape.

The companies are also stating that they need to re-evaluate their value propositions and how they differentiate themselves from competitors. They are focusing resources on their core capabilities to realize their competitive advantage, and leveraging business partners to do the rest. Areas in which companies are striving to differentiate themselves include product development, innovation and cycle time. This involves the processes associated with the research and design of products and services that are sold to the customer. Innovation, fast-time-to-market and development of desirable products are key business goals. This entails the integration and collaboration of business partners to respond to an emerging opportunity, customer need or competitive threat. Goals include developing collaborative working relationships across the value net, integrated processes and systems, dynamic linkages to engage and disengage members of the value net, and to formalize the knowledge.

In this industrial and technological scenario, decreasing product development times and costs while working in collaborative real time virtual environments, fast and smart retrieval and manipulation of past programs knowledge and increasing full product lifecycle managements are a must for industrial companies. This chapter idea appears from the collaboration of CEIIA (www.ceiiia.com) and Pininfarina both design companies, Portuguese and Italian. Development time must be reduced, by using automatic validation actions and optimization of knowledge acquisition and transfer flows, because Pininfarina plays a coacher role over CEIIA young engineers. My University is experienced in UML, Information Retrieval and IMATI-CNR (http://www.imati.cnr.it/) from Pininfarina side with the experience gained on ontology development conducted to FP7 project KREATE CAR. Also my background experience in AutoEuropa and the nightmare of data integration among development, planning and production raised the idea of exploring current advances in Semantic Web, Social networks and UML modeling to produce tools and create a methodology to handle the complete product life cycle.

These considerations already motivated some previous work on the scientific community as a collaborative, or a knowledge system or even in a Web system (Chiu, 2002; Hai-yue, 2004; Rose, 2005; Rosenman, 1999; Shem, 2007). Also the main Software development entities are looking on this
subjects and current advances in Semantic Web. In 2008, following the revolution around Web 2.0, one of the key commercial players in PLM introduced the notion of PLM 2.0 <http://www.cbronline.com/article_news.asp?guid=C472DF6-3DA7-458A-B077-8FCC73A96CE5>, which encompasses a social community approach to PLM.

In this work is proposed a system KC-PLM to reach the goals above described based on an integrated software framework that supports all product lifecycle in a collaborative environment, knowledge reuse oriented, with a special emphasis on the product design, phase that OEMs can we have great savings potential (time and costs). KC-PLM supports the validation and set up of the initial mock-up of a new product at the Kick-off, which is the feasibility phase of a new product development activity. This is achieved by developing tools for creating, modeling and preserving the knowledge deriving from information available in digital format. The knowledge base will also in a standard and manageable format incorporate ergonomics and homologation rules, technological database (with design, manufacturing and production information) and past design experiences. This environment relies on the formalization (through ontology) of the relationships existing among the product requirements, their application on the corresponding product components and the related rules that must be followed. The abstraction of the component information, required to perform any specific analysis, will be also provided; the ontology and data mining techniques are then used to drive the retrieval of the necessary information from existing data bases of both past project parts and rules existing for homologation and ergonomics. Retrievals of information from the repository based on domain ontology will help users to cooperate in a knowledge environment.

One of the KC-PLM objectives is to create new skills and maintain the knowledge; individuals undertaking a new project for an organization might access information resources in KC-PLM to learn best practices and lessons learned for similar projects undertaken previously, access relevant information again during the project implementation to seek advice for issues encountered, and access relevant information afterwards for advice on after-project actions and review activities. KC-PLM offers systems, repositories, and corporate processes to encourage and formalize these activities. Similarly, knowledge may be captured and recorded when the project team learns lessons during the project and, after-action reviews, may lead to further insights and lessons being recorded for future access. KC-PLM will lead to competitive advantage that comes with improved or faster learning and new knowledge creation and to greater innovation, better customer experiences, consistency in industry practices and knowledge access across a global organization. Considerations driving the KC-PLM might include: (1) making available increased knowledge content in the product lifecycle; (2) achieving shorter new product development cycles; (3) facilitating and managing organizational innovation; (4) leverage the expertise of people across the organization; (5) benefiting from “network effects” as the number of productive connections between employees in the organization increases and the quality of shared information increases; (6) managing the proliferation of data and information in complex business environments and allowing employees rapidly to access useful and relevant knowledge resources and best practice guidelines; (7) facilitate organizational learning; (8) managing intellectual capital and intellectual assets in the workforce (such as the expertise and know-how possessed by key individuals) as individuals retire and new workers are hired; (9) increasing the role of SME in projects with OEM; and (10) ensuring high customers satisfaction with high quality transport products.

This chapter is divided in the follow sections: section two describes the main approach guidelines taken based on the best practices of Software Engineering, Semantic Web and Social Networks.
Section three describes KC-PLM main modules; section four is dedicated to PLM lifecycle and the identification of main stakeholders; section five presents a real case study application on product development at Pininfarina and finally in section six the project conclusions are present and the future work.

**MAIN APPROACHES**

Our research proposal is based on using synergies between four main areas: (1) **Semantic Web**, the development started this century around the efforts on the organization of semantic knowledge, based on the use of descriptive technologies such as Resource Description Framework (RDF) and Ontology Web Language (OWL), as well as the data-centric, customizable Extensible Markup Language (XML); (2) **Social Networks**, with user collaboration efforts in terms of semantic annotation used for knowledge reuse; (3) **UML** with a domain specific language (DSL) based on UML2 and follow **MDA** (Model Driven Architecture) approach. We took the most important methodologies from Software Engineering, for the software development, to control and optimize processes and also to automate some semi-automatic routine/standards; (4) **PLM**, existing PLM through their API will integrate current proposals with a CMS (Content Management System).

Our approach methodology is illustrated on Figure 2 and is based on a domain specific ontology (DSO) that we called Vehicle Corporate Ontology (VCO) (Ferreira, 2007a) DSL based on UML2 called Vehicle Development Modeling Language (VDML) (Ferreira, 2007b) are proposed. Modeling activity processes is performed through VDML in a MDA approach and for a complete description see (Ferreira 2007c; Ferreira, 2007d; Ferreira, 2007e). The output of this modeling activity (models) is used to understand and improve the process and also from predefined templates we can generate digital artifacts (automatic or semi-automatic) to integrate user’s applications. This approach is complemented with a VCO which will provide a standard terminology used for data integration and knowledge reuse. Since VCO follows MDA approach, we can perform mapping between different ontology created based on the same approach (Ferreira, 2007a). VCO will benefit of UML classes and visual modeling will decrease syntactic and semantic errors and increases readability. UML itself does not satisfy the needs of the representation of ontology concepts that are imported from descriptive logic and that are in-

---

**Figure 1. Main approaches for the KC-PLM system**

[Image of a diagram showing the main approaches: UML/MDA with Optimization, Process Automation, Standardization, Modeling, Control; Semantic Web with Data Exchange/Integration; PLM with Existing Application; Social Networks with Collaboration, User Annotations; KC-PLM at the center.]
clude in Semantic Web Ontology Language (e.g. OWL, RDF, RDF Schema, ...). The OMG's MDA concept has the ability to create, using appropriate family of languages, a metamodel defined by MOF (Meta-Object Facility) and based on OWL. Since industry application uses different terminology systems, database information exchange is achieved whose semantics are specifying using logic based ontologies. A broader terminology is specified by VCO, and is used as reference terminologies. Relations among terminology are indirect, once each terminology can be mapped into other using DSO. A collaborative annotation process will use VCO and will enrich information repository with new content and relations.

UML/MDA Approach

UML has emerged as the software industry’s dominant language and is already an Object Management Group (OMG) standard. It represents a collection of the best engineering practices that have been proved successful in the modeling of large and complex systems. OMG is proposing the UML specification for international standardization for information technology. Wide recognition and acceptance, which typically enlarge the market for products based on it, will be the major benefit. Therefore specific subjects (e.g. PLM) require making UML models more specific and thus more precise. This in turn can be done by using stereotypes (since they are an extension mechanism inherent in second version of UML) as means of adding necessary information to existing model elements. Stereotypes have been given a special attention together with the idea of the Model Driven Architecture (MDA) UML profiles are UML packages of the stereotype «profile». A profile can extend a metamodel or another profile (Kobryn, 1999) while preserving the syntax and semantic of existing UML elements. It adds elements which extend existing metaclasses. UML profiles consist of stereotypes, constraints and tagged values. A stereotype is a model element defined by its name and by the base class(es) to which it is assigned. Base classes are usually metaclasses from the UML metamodel, for instance the metaclass «Class», but can also
be stereotypes from another profile. A stereotype can have its own notation, e.g. a special icon. Constraints are applied to stereotypes in order to indicate restrictions. They specify pre- or post conditions, invariants, etc., and must comply with the restrictions of the base class (Kobryn, 1999). Constraints can be expressed in any language, such as programming languages or natural language. We use the Object Constraint Language (OCL) (OMG, 2001) in our profile, as it is more precise than natural language or pseudo code, and widely used in UML profiles. Tagged values are additional meta attributes assigned to a stereotype, specified as name-value pairs.

**MDA (Model Driven Architecture)** is a new paradigm in the systems specification (MDA, 2003). MDA can specify systems at all levels, including middleware levels. This new model of systems architecture, offers us a group of very important advantages for distributed environments: (1) supports the whole development life cycle with more precision; (2) decreases development costs; (3) applicable to all languages, platforms, operating systems, networks and middleware; (4) most of the MDA patterns is already available: UML, MOF, XMI, CWM (new metadata standard for data warehousing and business intelligence); (5) a powerful middleware infrastructure: CORBA, IDL and services; built on a success OMG platform technologies OMG, like CORBA and UML; (6) rigorous mapping in the future for another infrastructures as: XML, SOAP, Java, .NET; (7) MDA being independent of middleware, languages or systems (language-, vendor- and middleware-neutral) guarantees that following its requirements and good development practices will have an application with scalability and migration potential.

**Semantic Web Approach**

The idea of the Semantic Web is to extend unstructured information with machine processable descriptions of the meaning (semantics) of information and to provide missing background knowledge when required. The key challenge of the Semantic Web is to ensure a shared interpretation of information in different information repositories. Related information sources should use the same concepts to reference the same real world entities or at least there should be a way to determine if two sources refer to the same entities, but possibly using different vocabularies. Ontologies and ontology languages are the key enabling technology in this respect. An ontology, by its most cited definition in Artificial Intelligent (AI), is a shared, formal conceptualization of a domain, i.e. a description of concepts and their relationships (Borst, 1997; Gruber, 1993). Ontologies are domain models with two special characteristics, which lead to the notion of shared meaning or semantics: (1) Ontologies are expressed in formal languages with a well-defined semantics; (2) Ontologies build upon a shared understanding within a community. This understanding represents an agreement among members of the community over the concepts and relationships that are present in a domain and their usage. The first point underlines that ontology needs to be modeled using languages with a formal semantics. RDF and OWL are the languages most commonly used on the Semantic Web, and in fact when using the term ontology many practitioners refer to domain models described in one of these two languages.

An ontology specifies a rich description of the: (1) Terminology, concepts, nomenclature; (2) Properties explicitly defining the terms, concepts; (3) Relations among concepts (hierarchical and lattice); (4) Rules distinguishing concepts, refining definitions and relations (constraints, restrictions, regular expressions) relevant to a particular domain or area of interest. Ontology can be used for: (1) independently developed services; (2) agreements among companies, organizations to share common services and information; (3) applications and services can work together to share information and processes consistently,
accurately, and completely; (4) improves search accuracy by enabling contextual search using concept definitions and relations among them. The Semantic Web project is the main direction of the future Web development. Domain ontologies are the most important part of Semantic Web applications. AI techniques are used for ontology creation, but those techniques are more related to research laboratories. Recently, there are many proposals to use software engineering techniques, especially the UML since it is the most accepted software engineering standard, in order to bring ontology development process closer to wider practitioners' population. However, UML is based on object oriented paradigm, and has some limitation regarding ontology development. These limitations can be overcome using UML's extensions (i.e. UML profiles), as well as other OMG's standards (i.e. Model Driven Architecture - MDA). Currently, there is an initiative (i.e. RFP) within the OMG aiming to define a suitable language for modeling Semantic Web ontology languages in the context of the MDA.

Social Network Approach

Social Network approach in literature is usually connected to Web 2.0 and deal mainly with users' Web interaction, how they are prepared to provide content as well as metadata, while the Semantic Web opens new technological opportunities for Web developers in combining data and services from different sources. Frequently the combination of Web2.0 and Semantic Web is called Web3.0. This collaborative approach appears in several systems, like: (1) articles and facts organized in tables and categories in Wikipedia; (2) photos organized in sets and according to tags in Flickr; (3) del.icio.us with bookmarks in digital objects of classification; (4) scientific publications are tagged in CiteULike; and (5) 43Things allows users to share their goals and plans (e.g. to travel or lose weight) by annotating their descriptions with keywords and connecting users with similar pursuits. The idea appears in the scientific literature with the name of folksonomy and is based on user's digital objects description (collaborative annotation) to describe a set of shared objects with a set of keywords of their own choice. So we will extend the traditional bipartite model of ontologies (concepts and instances) by incorporating actors in the model, giving a representation as a tripartite graph with hyperedges. The set of vertices is partitioned into the three (possibly empty) disjoint sets $A = \{a_1, \ldots, a_k\}$, $C = \{c_1, \ldots, c_l\}$, $I = \{i_1, \ldots, i_m\}$ corresponding the set of actors (users), the set of concepts (tags, keywords, most of then taken from the VCO ontology) and the set of objects annotated (bookmarks, drawing, process information etc.). Thus the folksonomy is defined by a set of annotations $T \subseteq A \times C \times I$. Such a network is most naturally represented as hypergraph with ternary edges, where each edge represents the fact that a given actor associates a certain instance with a certain concept.

This approach will win from specialized domain users (e.g designers, production/product engineers, logistics and others) with less predisposition to SPAM and non-collaborative tendencies that sometimes reflects Web users. Users can be rewarded regarding their collaboration based on credit mechanisms latter converted in money or other assets (Ferreira, 2008), but we did not implemented this approach yet.

KC-PLM Architecture

As illustrated in Figure 3 the KC-PLM system consists of the following main artifacts: the PLM tool, the K-CMS Web-based system, and the common semantic information repository (SIR).

The SIR is a heterogeneous digital information repository with domain ontology. Available Information types are drawings, normative and ergonomics rules, manufacturing, suppliers, production, technological and molding issues and job templates for automatic retrieval information;
The PLM tool is a desktop application based on a PLM commercial application, with a proper interface to suit the knowledge repository. There is a retrieval engine facility and modules for knowledge acquisition and process and design optimization. Process smart solution selection and retrieval and design optimization are based on evolutionary algorithms. There is also connection to other commercial specific application in CAD, Production areas or other specific systems related with product lifecycle;

The CMS (Content Management System) is a Web-based Content Management System that provides controlled access to the knowledge repository and project management features. The CMS system provides a knowledge and collaboration work environment with some automatic actions based on the job profile description.

As seen in the Figure 3, the KC-PLM system can be used by designers, engineers, production, dealers, project managers, external users (like OEM) and suppliers. This system will serve mainly the project working engineers, but the Web interface will provide access to external users in a controlled access environment, and it will also help project manager to follow and control the related project.

**SIR:** The main source of information are: (1) collection of information related to CAD/CAS drawings in different formats or other product data (e.g production. Logistics, manufacturing); (2) collection and digital conversion of ergonomics and national homologation rules (which are also the ‘objective functions’ for process and design optimization); (3) collection of information regarding past experiences, common mistakes, changes performed in the different phases of vehicle design and production; (4) Technology Data Base (DB) In order to deal with complex rules requiring verification on different process steps (from conception to production), a specialized module related with an experience database as well as with specific domain ontology will be integrated into final solution; (5) supplier database information; (6) production, logistics database. For this task an experience database

---

**Figure 3. KC-PLM system main modules**
will be used and this will permit to fast define a database structure starting from the ontology definition – automatic import of classes. Because a normal relational database will not fulfill the existing needs, a CBR [Case Base Reasoning] tool and Rule Based databases will be used. These two parts will form the database experience module that will capture the past experience and allow experience to be reused during different processes. CBR is a problem solving paradigm that differs from other major AI approaches, in fact, instead of relying solely on general knowledge of a problem domain, or making associations along generalized relationships between problem descriptors and conclusions, CBR is able to utilize the specific knowledge of previously experienced, concrete problem situations (cases). A new problem is solved by finding a similar past case, and reusing it in the new problem situation. A second important difference is that CBR also is an approach to incremental, sustained learning, since a new experience is retained each time a problem has been solved, making it immediately available for future problems. Regarding user collaboration and annotation process a semi-structure information is created based on the Semantic Web languages such as RDF and OWL.

**CMS**, for detailed information see (Ferreira, 2008). This a Web-based platform that supports the creation, management, distribution, publication and search of the information kept by the knowledge repository. The CMS will handle information from the knowledge repository, control users access to project information, provide tools for project managements (e.g Project Workflow Management) and also for domain ontology editing and browsing. The CMS is responsible for the Web site’s contents life cycle management, providing tools for new content creation and edition, oriented by a content creation/edition workflow. Further on, the CMS will also enable complete site structure and visual appearance management, while delivering automatically generated and completely integrated Web site navigation. Main system features are: (1) User management; (2) Permission Management; (3) Content Creation and SIR access; (4) Visual style management; (5) Multi-Language Support; (6) Integrated Content Search Engine and inference rule based knowledge extraction; (7) Viewer for CAD file; (8) Web based Tool for Ontology Browsing and Editing; and (9) Project Workflow Management.

**PLM LIFECYCLE AND MAIN SKATEHOLDERS**

Product Lifecycle Management (PLM) involves three main phases, as illustrated in Figure 4: (1) **Product Development**: Conception based users needs and Innovation concepts, market research,
styling, product guidelines design, test, analyze, product engineer and validation; (2) **Product Production**, which includes manufacturing planning, manufacture; (3) **Service**, which includes use, deliver operate, maintain, support, sustain, phase-out, retire, recycle and disposal.

Main actors of complete PLM cycle is illustrated in Figure 5 are: (1) **Product Development Team**, has the responsibility of development and integration of a certain area of the vehicle as well as its compatibility with the adjacent development teams; (2) **Product Manager**, has the overview of all product development related activities progress and controls the achievement of the various milestones. Identifies, and submits to the project manager approval, special plans for timing achievement if required based on the recommendations of the product development teams; (3) **Industrial Engineer**, concentrates on plant layout, including the arrangement of assembly line stations, material-moving equipment, work standards, and other production matters; (4) **Purchasing Team**, is responsible to find and negotiate the price with supplier for all the outsourced parts identified in a new product; (5) **Manufacturing Engineer**, is responsible to specify and follow up with equipment suppliers the development and installation of all the machinery necessary to produce the defined product; (6) **Supplier** is responsible for the construction of the tool and production of the parts respecting the automotive quality standard. Before delivering to serial production they have to submit master samples for approval and obtain the production release; (7) **Quality Engineer** is trained in various quality tools, including advanced problem solving techniques, to assist in Functional Tests activity. Evaluates several parameters like time, quality and performance (e.g. investigation into water leaks, rattle and squeaks, interior trim, closures, electrical and powertrain, etc) and is responsible for quality assurance at all stages of the process, from an individual part until the release of the final vehicle; (8) **Logistics Planning (part of Manufacturing Activity)** is responsible for planning, implementing an efficient and effective forward and reverse flow and storage of goods and related information between the point of origin and the point of consumption in order to meet customers requirements. Logistics is an integrating function, which coordinates and optimizes all material flow activities, as well as integrates logistics activities with other functions including marketing, sales manufacturing, finance and information technology; (9) **Production Engineer** leads through the productive process and establishes feedback cycles allowing fast detection and correction of repetitive failures avoiding them to reach a further step in the production system or even final customer; (10) **Laboratory Specialist** is responsible for the part material definition taking in account pre-defined targets, homologation rules; (11) **Toolshop Prototype Engineer** is responsible for the construction of the first part taking in account the part definition made by product engineer and laboratory specialist; (12) **Pilot Plant Engineer** is responsible for all validations and test of parts assembled; (13) **Release Engineer** is responsible after the correct validation of part for their release in appropriate system; (14) **CAD Engineers**, demonstrate the ability to interpret and develop a model from a 2D picture or instructions, such as Design Objectives into a 3D model using appropriate techniques to ensure a feasible, proportionally balanced model which meets the design requirements, work with little or no direction to develop a surface displaying an appreciation of shapes, proportions and perception of the final outcome of the 3D model, develop both intricate and surface model surfaces to within 0.2mm of engineering feasibility data, within the required time frame perform feasibility and packaging studies and verify solutions to be in compliance with regulations; (15) **Styling and CAS Engineers** produces sketches to invoke the feeling of acceleration or tension, sharpness or softness according emotional and creative passions and the market segment of a car. Define stylistic properties.
and then creating computer-aided styling (CAS) tools that can capture and produce emotions. The aesthetic shape properties produced, will concur to generate time, energy, and money throughout the design process: from creating and modifying models, to the disconnects in concurrent aesthetic and engineering design, to the broken feedback loop from downstream design/engineering processes; (16) **CAE Engineers** are responsible for predict product performance, drive design, and minimize cost and weight. CAE is also involved in most of the customer functional attributes: durability, NVH, safety, vehicle dynamics, thermal management, aerodynamics, fuel economy and performance, package, weight, electrical systems and electronics; (17) **Testing and Validation Engineers** are responsible, within the Product Development Teams, to identify, perform, and follow up all the tests and validations required to a certain area of the vehicle. Their feedback is essential to the approval of the work produced by the engineers and can originate severe corrections and changes to their work; (18) **Project Manager** has the overall control of the project for timing and costs targets achievement, regarding product development, purchasing, quality and manufacturing activities. Analysis critical situations coming from the various teams and submits recovery plans for approval of the board of management; (19) **Market Researchers** are responsible to collect information about clients’ needs and future tendencies and features for a new car. Provide data that is used to support critical technical and financial decisions during all the product life cycle; (20) **Human Resources**, must prepare recruitment (if applicable) training on new technologies and training on the job actions according to the different areas requirements to the various phases of the project; (21) **Reverse Engineer** is responsible for the production of electronic 3D solid product models from captured surface geometry data for use in CAD/CAM/CAE/CAV environments using Reverse Engineering techniques and processes; (22) **Distribution and Importers/Dealers**, are responsible to make sure that the launch timing is the appropriate one taking into consideration the markets and the time of the year and the presentation of a given model in the international main exhibitions worldwide; (23) **Plant Manager**, is the entity that must be informed about the achievement, or not of all milestones in order to activate recovery plans when ever it is necessary making sure that the launch timing will be achieved with the right

---

**Figure 5. Main actors in PLM lifecycle**
quantity and quality levels and (24) Finance, must install a system that ensures that all the costs are reported in time and are according to the approved budget and the profitability margins are achieved according to the project assumptions.

A description and activity modeling involving all PLM phases can be found at (Ferreira, 2007c; Ferreira, 2007d; Ferreira, 2007e).

The knowledge collaborative system can be defined using a role/task-oriented perspective, in which a set of tasks is performed by different role players: Architect, Requirements Engineer, Designer, Programmer, Tester and Integrator.

CASE STUDY

A complete PLM life cycle case study is complex and difficult to perform due multidisciplinary tasks/process involved and the companies. The most important phase to reduce costs is the design/conception phase, we will conduct a case study in a design company (Pininfarina) in automobile business, mainly in CAD design system and for a detailed view see (Ferreira, 2008). Pininfarina will test in the KC-PLM in a Product Design conception phase in a full one-shot vehicle model frame, applying all of their checking procedures on surfaces in ergonomics, stamping and normative issues. Some of the checking procedures in vehicle ergonomics issues are: Seating Ground Reference Point, A-pillar Obstruction, Windshield Header Sections and Wiper Sweep Design, Manufacturing Feasibility, I/P Hardpoints and Mirror Visibility. For exterior/interior main components the stamping feasibility will be assured by the fulfillment of design rules strictly linked to manufacturing process, like forming axes, flanging appendixes and stamping parameter for major frame components. These smart verification tests on the computer model prevent costly mistakes, since they are taking place already in the planning phase of the car body components, and they are also making the work for the designers easier by reducing the spent time for computing model changes. Another interesting feature is that with the new software it will be possible to test new vehicle architectures against either new CAS definitions or existing models, and with the documented know-how it is possible to achieve component standardization with further benefits in terms of costs and time. In reality, every time that CAS surfaces are changed from the style department, and this may happen many times in a product development of a vehicle, just the feasibility check in ergonomics takes normally one week of work by a design expert. The KC-PLM environment will shorten the time spent for the design process from days to hours, without keeping into account that the CAD-system will use and evaluate at the same time corporate, legal and international rules. This is a validation of the research program more connected to a typical OEM point of view. A general description of the process is the following: The marketing department acquires and aggregates customer needs data and supplies them to the style department that sketches in class B surfaces all the new ideas for the one-shot vehicle production.

The reduction of product development time and costs will be quantified for specified tasks based on the comparison between the classical product development techniques and the new developed KC-PLM based features. The great improvements in manufacturing and quality with strong reduction of technological stamping problems and of customer penalty points will also be localized and quantified compared with classical product development strategy. This increase dramatically the competitiveness of the enterprise: convert the process so it starts out with correct or near correct knowledge-based information, avoiding re-engineering vehicle components near job1-gate, and therefore avoiding wasting money on tooling modifications. Also, the attained quality improvement will avoid producing low quality products. The relevant manufacturing information stored within the database will provide the developed
system with the ability to verify and highlight any case were the technological rules have been violated: the database will provide constraints to the product, for example if the base justify that a radius of curvature is too small to allow the component to be manufactured it will be detected immediately. Another possibility achieved from the time reduction is that engineer now can investigate more “what if” packaging scenarios and ergonomic cases than traditionally would be possible due to the time constraints.

The key to sustaining innovation on new products is thus related to the information management and knowledge sharing activities of the multidisciplinary design teams involved in the KC-PLM environment. Concept designers and product engineers need active support to understand each other and to share their knowledge when resolving design issues and problems: this support could be the KC-PLM environment.

A fast visualization helps the user to locate certain parts in the drawing to modify or add elements from a rendered view; the novel aspect in this case is the exploitation of semantic aspects to generate simplified versions of the model according to certain stages in the design process, enhancing specific parts (e.g. coloring) and adapting the camera position. Experts from each automotive discipline involved use shared information to conduct specialized test analysis with the goal of setting a working methodology to reduce development time by using an iterative model that can perform many key analyses.

CONCLUSION

In this chapter was proposed a methodology and a software framework aimed at supporting the complete cycle of a product. Developing tools for capturing, modelling and preserving the knowledge deriving from past design experiences, based on collaboration processes among users, exploring synergies between current approaches of Web 2.0 (Semantic Web) and Social Networks (user collaboration) for knowledge extraction and reuse. This environment relies on the formalization (through ontology) of the relationships existing among the product requirements as well as their application on the corresponding parts/components and the related rules that must be accomplished. The abstraction of the component information, required to perform any specific analysis, will be also provided; the ontology is then used to drive the retrieval of the necessary information from existing data bases of both past project parts and homologation and ergonomic rules. The modeling process activity through VDML can be used for process improvements and the MDA approach embedded permits the conversion of this process activity description to digital artifacts (e.g code pieces to interact with specific program API, for example macro style to integrate in CAD) in a semi-automatic process. An example of this methodology has been applied in the modeling of ergonomic activities in the product design phase. Based on this approach a complete set of Catia V macro was created to perform automatic ergonomics checks during the product design phase. This approach saves considerable amount of time in product development bringing to an earlier stage ergonomics and homologation rules (usually performed in a late project stage), the usage of past experiences and automatic actions will reduce user’s working time. Reducing working time will conduct to total cost reduction.

This PLM environment is complemented with a Web based interface to access information in a controlled way and these systems will contribute to the development of a collaborative environment among users in the design phase and subsequent phases like engineering and production. Central and flexible data base will give the opportunity of accessing and sharing information among different development and production phases. SIR permits knowledge reuse and the integration of data application among different applications.
REFERENCES


KEY TERMS AND DEFINITIONS

**CMS (Content Management System)** is a Web-based Content Management System that pro-
vides controlled access to the knowledge repository and project management features. The CMS system provides a knowledge and collaboration work environment with some automatic actions based on the description of the job profile.

**Folksonomy**, is an collaborative annotation process performed by the users. Users read internet content and associated metadata chosen from a controlled vocabulary or free chosen terms. This metadata is used for retrieval purposes and mainly created by the producers and the consumers of the information.

**Knowledge Retrieval** is a process of returning (delivering) information in a structured form, consistent with human cognitive processes as opposed to simple lists of data items. Knowledge capture and recycling (Retrieval) is very important is today’s organizations, improves the conditions for training/tutoring young technical staff and engineers and is a key point on organization development and evolution.

**Ontology** is a shared, formal conceptualization of a domain, i.e. a description of concepts and their relationships. This set of concepts is connected in a network or graph and used to described specific knowledge areas and can represent knowledge.

**Product Lifecycle Management (PLM)** is the process of managing the entire lifecycle of a product from its conception, through design and manufacture, to service and disposal. Associated to this process there is specific software to help the users to handle and control the different product stages.

**Semantic Information Repository**, is an information or data collection that links concepts and names together, based on descriptive technologies such as Resource Description Framework (RDF), Web Ontology Language (OWL) and XML. This repository integrates structured, semi-structured and unstructured data in a manageable format.

**Unified Modeling Language (UML)** has emerged as the software industry’s dominant language and is already an Object Management Group (OMG) standard. It represents a collection of the best engineering practices that have been proved successful in the modeling of large and complex systems.