



Science Arts & Métiers (SAM)

is an open access repository that collects the work of Arts et Métiers ParisTech researchers and makes it freely available over the web where possible.

This is an author-deposited version published in: <https://sam.ensam.eu>
Handle ID: <http://hdl.handle.net/10985/7715>

To cite this version :

Dimitri VAN WIJK, Alain ETIENNE, Emmanuel GUYOT, Benoit EYNARD, Lionel ROUCOULES - Enabled virtual and collaborative engineering coupling PLM system to a product data kernel - In: 5th International Conference on Digital Enterprise Technology: from numerical simulation to intelligent cooperation, France, 2008-10 - 5th International Conference on Digital Enterprise Technology: from numerical simulation to intelligent cooperation - 2010

Any correspondence concerning this service should be sent to the repository

Administrator : archiveouverte@ensam.eu



ENABLED VIRTUAL AND COLLABORATIVE ENGINEERING COUPLING PLM SYSTEM TO A PRODUCT DATA KERNEL

Dimitri VAN WIJK

Pi3C
dimitri.van_wijk@pi3c.com

Alain ETIENNE

Université de Technologie de Troyes
alain.etienne@utt.fr

Emmanuel GUYOT

Université de Technologie de
Troyes
emmanuel.guyot@utt.fr

Benoit EYNARD

Université de Technologie de
Compiègne
benoit.eynard@utc.fr

Lionel ROUCOULES

Université de Technologie de
Troyes
lionel.roucoules@utt.fr

ABSTRACT

Nowadays, the use of digital modelling through multiple CAx systems becomes more and more unquestionable. It allows many benefits (3D, automations, simulations, etc.) in spite of the fact that those different systems are individually used and are relatively isolated (i.e. not interoperable) during the product design process. Then more enhancement of product development might be possible if the whole design expertises (and IT applications), could collaborate dynamically (i.e. interoperate) in a complete virtual and collaborative engineering platform. For instance, product definition could be therefore seen as the synthesis of multidisciplinary information defined and assessed with respect to both collaborative activity and individual domain of expertise. At the same time, PDM/PLM systems are developed in most of cases as product data vault (product data referential of a company with internal PDM or projects collaborative referential with collaborative engineering platform). Nevertheless, current PLM are not able to manage more than only files (and not data breakdown) and then are not yet efficient for CAx systems interoperability.

The paper aims at giving some first concepts and software demonstrators in order to tackle the problem of interoperability in virtual engineering design. The proposal is based on the PPO data kernel: a dynamic data structure able to integrate and merge multidisciplinary product data breakdown and to exchange them with other CAx applications. Then the current PDM/PLM systems will be defined as one of the CAx systems which interoperate with the kernel to benefit of both PLM and CAx potential functionalities. A business case will be studied between the PPO product data kernel and the Windchill (from PTC vendor) PLM system. Some recommendations for future work are then discussed to present what could be a generic interoperable software platform.

KEYWORDS

Concurrent Engineering, Collaborative Design, Product Lifecycle Management, Interoperability, Virtual Engineering

1. INTRODUCTION

Thanks to scientific advances and led by industry competitiveness, industrial products become inevitably more and more complex (ex: more functions and associated services) and multidisciplinary (ex: mecatronics). In consequence, the different competencies of a company

(engineering design, mechanical analysis, tolerancing, electrical design, etc.) must henceforth collaborate in order to innovate, to improve quality and to reduce costs (Bianconi et al, 2006). In fact, the project teams in charge of developing a product need to aggregate people of many different skills. But paradoxically, engineering activities are still

mainly centred on a geometric representation (i.e. CAD model) and not on real multidisciplinary representations coming from multi-domain expertises. Moreover, currently, there is no software system that supports such heterogeneous collaboration.

As functional solutions, Product Lifecycle Management (PLM) systems try to allow the collaboration between the different design experts. But they are not adapted to this because: First PLM systems are too strongly linked to the 3D representation; or rather they are not focused enough on a multidisciplinary representation. Secondly, PLM systems are only able to manipulate files instead of data (The difference will be explained in the next part of this paper). Authors try in this paper to answer these problems by proposing another use of PLM systems connected to a PPO kernel (Product, Process, and Organisation).

Obviously, one of the key to support the collaboration is interoperability (from a general point of view). Now lots of researches and investments are concentrated on this topic (Sudarsan et al 2005). That is why, after an explanation of what is understood as “interoperability”, the paper will expose what has been done concretely to make interoperable the systems we used.

In a first time, the paper will give some definitions about the concepts in use. A quick review of related work will be done concerning the actual CAx collaboration and PLM systems use. Then, interoperability and possible interactions between PLM systems and the PPO kernel will be described. Finally an example will be exposed to validate what is explained in the paper.

2. FRAME

To avoid confusions, this first section explains the semantic of the employed terms as they are understood and used in this paper.

2.1. PRODUCT

First, the term “product” can represent many different things. At a high abstraction level, a “product” has many different aspects; for instance: organisational, material, etc. And from now we will especially be interested in the informational aspect. Then, under the “product” term we will refer to all the different data coming from the development process steps of a project to make an item and allowing this concrete item to be made. It means that the product could be seen as a collection of intellectual results coming from the work of all the different experts in the companies. In consequence,

each expert looks at the product with a specific viewpoint, which leads to a corresponding product representation. A product will finally have as many views as expert skills. We could also consider the views as understood in the industry (Example: design, manufacturing assembly, maintenance, etc.) however, they are only groups of the experts points of view we told before.

At a lower abstraction level, the representation of such product information could take different forms. In fact, persistent files which contain the product data could be differentiated from product data itself: Product data are platform and language independent whereas files encrypt data in a specific format, with a specific syntax. Product data could be parameters, behavior law, attributes. One of the advantages of this separation is the possibility to consider the communication between the CAx systems as exchanges of product data and to be avoided of format problems.

2.1. INTEROPERABILITY

We saw previously that from a general point of view interoperability can support the collaboration, but the global term “Interoperability” could have many meanings and consequently many domains like attested in (Ford et al, 2007). The number of interoperability definitions for this term in literature is also a strong proof of this affirmation. In fact, each field of application could provide a particular sense of interoperability. Among all these definitions, the most cited is the definition from IEEE, which give general bases of interoperability: “The ability of two or more systems of elements to exchange information and to use the information that has been exchanged” (IEEE, 1990). Other definitions were proposed to add the concepts of services to this definition, like United States Department of Defence definition (DoD, 2004). Services exchanges have effectively some importance for the interoperability as understood in this paper. Authors are finally attached to the notion of “effort to couple”, even if this definition is not very popular (Ford et al, 2007); Effort to couple is important for the reason that interoperability is quite always possible if enough time and the money is spent to develop a solution for a specific application case. Reduce efforts to interoperate is also clearly an essential purpose of interoperability research. Moreover this notion allows making a strong difference between point-to-point approach and mediator approach.

Under the global term “Interoperability”, we also have to consider different interoperability domains. In some work addressing interoperability problems,

three main levels were identified, namely the technical, semantic and business levels (ATHENA, 2007; I-DEAS, 2002). As the paper focuses on the product (cf. previous paragraph), only the two first levels will be taken into account. Some approaches propose to start with the business level to structure and drive the interoperability levels below. And the PPO (Product, Process and Organisation) kernel also proposes a process dimension, but the goal of this paper is to validate the interaction possibilities between a PDM/PLM system and the PPO kernel. Moreover, the business level is mainly focused on collaboration between organisations, whereas this paper will first consider the collaboration inside a company. So we will not take into account the business processes interoperability.

3. RELATED WORK

3.1. SUPPORT FOR INTEROPERABILITY

Nowadays, interoperability has become a full-fledged research domain (justified by researcher groups like the European INTEROP excellence network). In addition, these last years, many companies have developed technologies and solutions to improve interoperability, especially between existing management systems like, PLM, PDM, ERP, etc.

To give some concrete examples, the OpenPDM software from ProSTEP (and additions to this software) allows to make STEP AP214 based exchanges of product structure definitions between different PLM systems (Yang et al, 2008). Those exchanges are done through XML technologies allowing the definition of a virtual reference database that will be used by the concerned actors.

Another example could be, the Share-A-Space software from EuroSTEP as presented in (CIMDATA, 2004), that offer a STEP AP239 based environment allowing translation and conciliation of data produced during the whole product lifecycle. In opposition to the first given example, the reference database will be real and receive the data through connectors adapted to the different systems. Share-A-Space implements an interesting approach, which has some similar points with the PPO kernel in the sense that a repository with a generic data model is created. Then this repository is able to receive heterogeneous data, to conciliate and give out adapted data to the other heterogeneous systems.

In PLM systems domain mainly, more and more works are turned towards the PLM services to encapsulate the heterogeneity (Gunpinar et al, 2006 and Lämmer et al, 2007) and to allow the communication between heterogeneous systems.

Nevertheless, all these works are, so far, only turned toward the PLM systems and does not take into account interoperability between CAx expert software.

3.2. COLLABORATION AROUND THE PRODUCT

Concerning research work on collaboration between design experts during product development, there are many approaches. One of them is the VIM (View Information Modelling) methodology proposed by (Million, 1998). The principle of VIM is to have a unique mono-block data model composed of a reference model and completed by sub models. These sub models are bridged to the reference model by common concepts. The so added sub data models, specific to design expertises, allow the dynamic building of the whole data model. So each data model translates an expert "Viewpoint" and the representation of the product from this viewpoint led to a view of this product (Harik, 2005).

Other works try to address the problem exhaustively treating all the interfaces between the different experts tools (Twigg D., 2002). This approach is opposed to the approach used in PPO kernel. In fact, there is also one reference data model, but this one is generic enough to allow any mapping of the different concepts that come from the different design/manufacturing digital systems. If elements are missing, it is possible to create them on the base of the generic elements. There is then new specifically typed elements based on generic ones.

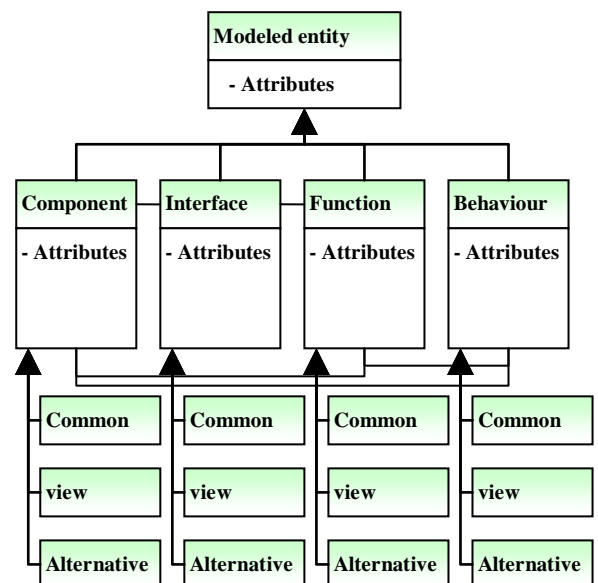


Figure 1 – Reminder of a part of the PPO data model

This approach has similarities with norms, but norms are functional models and try to allow this representation but are not powerful enough (because too specific. There is an example with the different Application protocols of STEP).

4. INTERACTIONS BETWEEN THE PPO KERNEL AND PLM SYSTEMS

4.1. POSITION OF PPO AND PLM SYSTEM

4.1.1. Position relative to industry

Among the solutions presented before, many of them are essentially concentrated on PLM system environment or systems relatively close to this environment (CAD systems, ERP, etc.). Moreover, they are focused on managing files and not product data. Following those remarks, PPO aims to contribute and bring a missing brick in current interoperability works and the interactions with a PLM system will allow managing product data container as well as product data content.

In industry it can be observed that collaborative platforms based on PLM systems are developed to support the collaboration between partners (suppliers, customers, others) of a project (NGuyen Van T., 2006, Guyot et al, 2007 and Van Wijk et al, 2008). So the PLM system manages an “external” collaboration (as shown on figure 2) because PLM systems allow managing accurately access right, data flow, and configuration. But in many cases, external collaboration is done through the intermediate of Enterprise PDM systems and then integration with PLM collaborative platform may be done by solutions we presented in paragraph 3.1.

On the same figure, it is shown that the collaboration “inside” a company between all the competencies should be supported by a system able to manipulate product data at a more fine level of granularity. Such a role should be given to systems like the PPO kernel. Anyway, the two collaboration systems (PLM system and PPO kernel) have to be synchronized because even if the collaboration is done at a different level it is about the same product. In fact, we do not consider the case of collaboration between the different competences of different companies though the PPO kernel because there are administrative and property rights constraints that make for the moment this type of collaboration very difficult.

To summarize, the PLM system could be seen as an organisational mediator, it will structure and officially formalize the different generated parts. However, the content of product data will be inaccessible to the expert systems (We could see it

as “black boxes”). The PPO kernel could then be seen as competencies mediator but managing product data informally (“White boxes” collaboration).

In addition, such architectures beg the question of document property and document validity. In fact, product data in PPO does not have really formal representation. So the validity, property and uniqueness of the product data are difficultly managed. Moreover, files may be present in the PPO kernel (for example a CAD Model) and in this case, copies of files between PPO and PDM will be redundant. In this case, a simple link to the file should be the best way to preserve the data uniqueness. However, authors have first validated the PPO-PDM interoperability copying files during the exchanges.

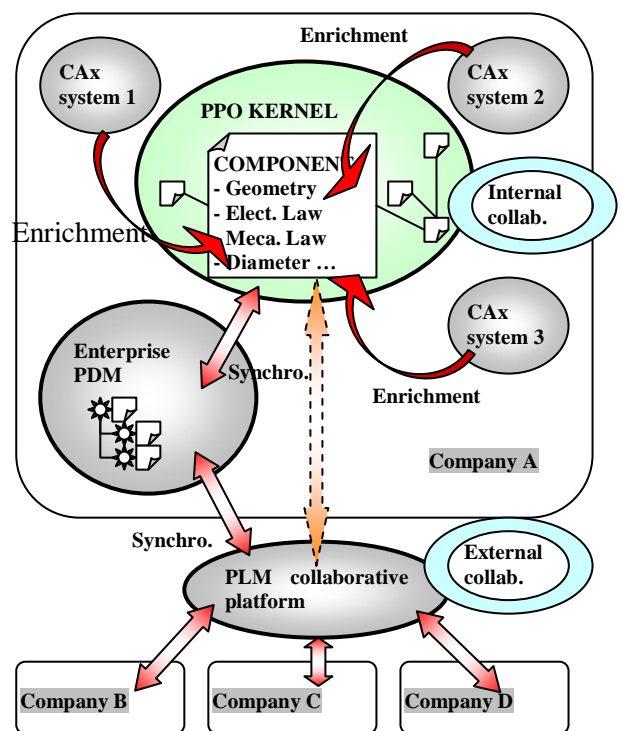


Figure 2 – Positioning of collaboration tools

4.1.2. Functionalities

Following the previous paragraph of this paper, functionalities of both PLM system and PPO kernel have to be differentiated. Thus, the PPO kernel is in charge to concentrate the product data (format independently) and to assure the coherence between those data. It will then structure and distribute the data during collaborative design tasks. When the data reach a sufficient maturity and that a part (“assembly” or “component”) can be concretized then this formalisation of the competencies concentration can be managed by the container inside the company (Case of simple PDM) or for

exchanges and collaborative work (Case of PLM Platform). The PLM system will then assure some functions at a “company level” like the examples in the following table.

1 - Functionalities repartition (Non exhaustive)

PLM	PPO
Support access rights (ACL, Access Control List)	Changes management / Changes impact calculation
Support project processes	Act as reference for product data
Trace parts maturity	Distribute data and offer communication technologies
Manage Components Relationships	Link product data files and exchanged parameters (ie : product data)
Manage document property rights	Manage relation between heterogeneous data (Transform data)
Manage configuration	Organize, synchronize CAx systems work

4.2. CORRESPONDENCE DESCRIPTION

The semantic correspondence of the two data models has been possible thanks to the genericity of the PPO data model (Like shown below for the case of PLM parts). However, due to the generic aspect of the reference data model, create more specific object (cf data model reminder) based on the existing generic objects is quite mandatory (cf. paragraph 3.2). For instance, the link between the two systems is kept registering the unique ID for each entity on both sides. In consequence we have to create a specific object of type “Attribute” for each entity with adapted characteristics in the PPO model before instantiate it (like shown in the figure below). Thus, semantic interoperability could have at a minimum two sub problematics: Correspondence between existing concepts of data model and Enrichment of data model. Those two problematics are not differentiated in the next figure, but it has an impact in the implementation level (cf. paragraph 4.3).

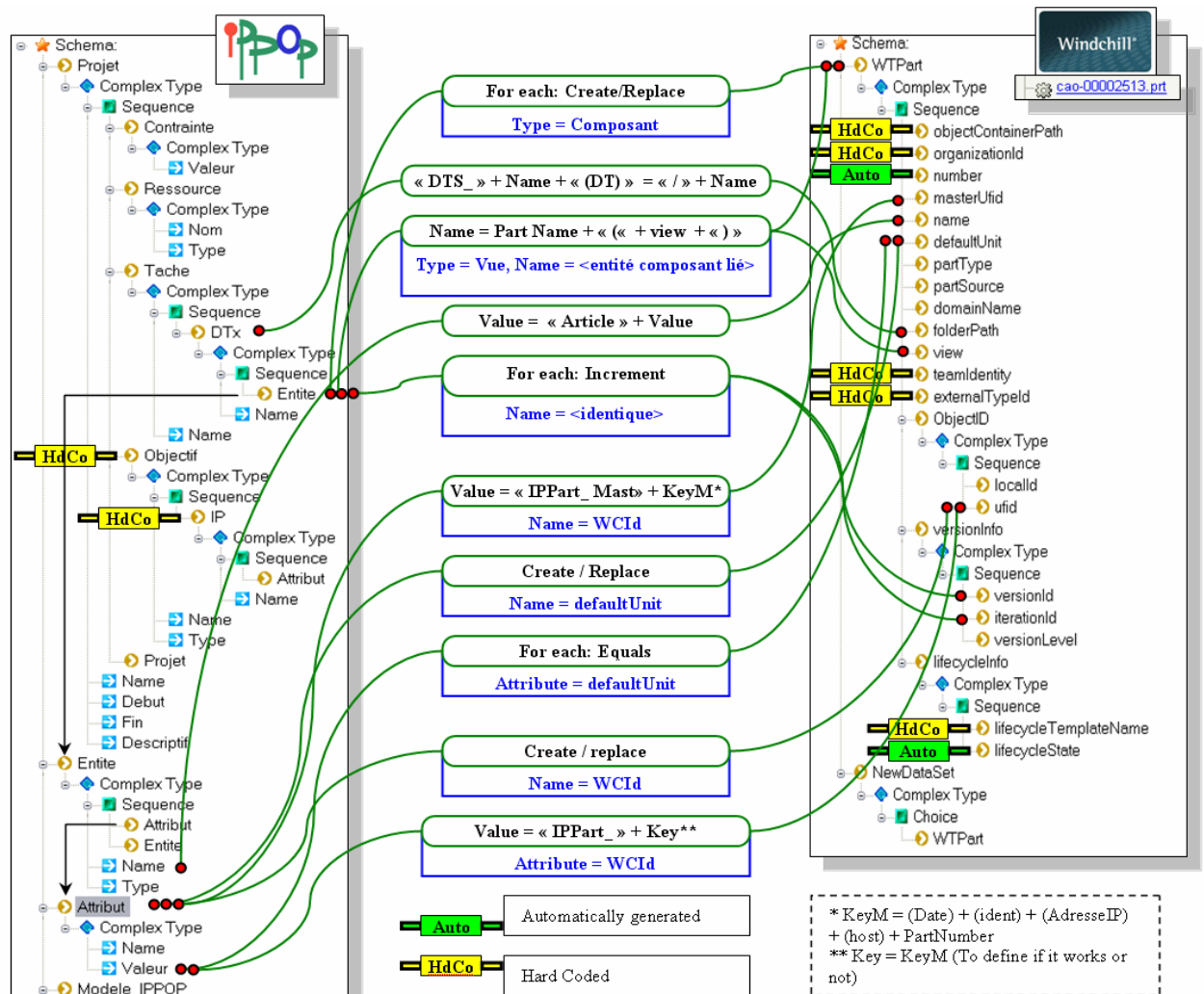


Figure 3 – Nom de la figure

At a higher scale, the parent root concepts that were mapped are those in the following table:

It could be considered that a project is linked to a product, and a product is an output technical data of an enterprise task. All the components appearing in the PPO kernel in this task reflect a creation desire of a concrete item. That is why; common components are mapped to parts in corresponding Engineering Bill of Material structure (EBOM). Then each expert viewpoint is ordered in the PLM system views (cf. 3.2). Product data contained in PPO kernel are concretised in part specification document and attached to it.

2- Mapped root instances

Domain	PPO	WINDCHILL
Project	"Output Technical Data" of a task	Project/Product
		Document structure / Folder structure
Product	"Component" entity, "Common" type	Part
Links	"Component" entity relations	Product Structure
Attributes	Entity "Attributes"	Part / Doc Relationships
		Specification Doc.
Views	"Component" entity, "View" type	Part (Same as previous)

Sometimes, the mapping of the main concepts is a correspondence between different levels. For example the type of an object (like "common" for element) could be map to a part which is a higher level definition. It is also true for the "technical data" of a task which could be affected to an entire product or project. In this case, it could be called an inspiration more than a correspondence. Indeed, the lower level (Often PPO) is used to structure the higher level (Often Windchill). Then the complementarities are about links and organization more than data and information itself. This organization done, it could be feed by correspondence between product information, like attributes and so on.

It is maybe the fact that the two models don't address the same level that makes them complementary.

4.3. TECHNICAL IMPLEMENTATION

From a technical point of view, a problem encountered was how to extract the data from the different CAx systems. Two main choices were taken into account.

There is the possibility to use SOA (Services Oriented Architecture) capabilities. In fact, some projects addressing interoperability problems make

the hypothesis all systems have such architecture. However, even if such technologies allow encapsulating heterogeneities, it is still a recent approach and in consequence not often implemented in the CAx systems. That's why another solution will be explored but the SOA possibility must be considered.

In order to allow the communication between every CAx system, the choice was to use XML import/export. In fact, most of the systems are already able to give XML data as output, and even if there are not those functions, it is often possible to quickly develop it via APIs. Moreover, the data model information can be extracted at the same time.

In every case, the output and input data can be written in XML format that make easier the correspondence between the considered CAx system data model and the PPO data model.

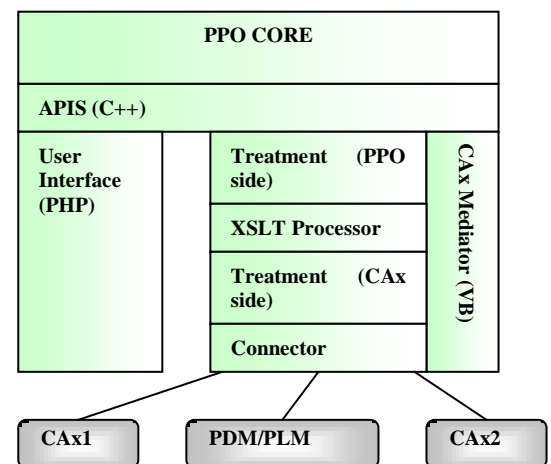


Figure 4 – Functional architecture

To operate the correspondence between the data and to manage the exchanges between the PPO kernel and the other systems, a communication module has been developed (See the figure 4). This module is composed of a PPO connector, a PPO intermediate processor (to change the PPO original schema in a more comprehensible schema), a XSLT processor, a post treatment engine and a communication manager.

The use of XSLT will allow some portability of the mapping described. It will also be reusable (for example, if we decide to use a product like OpenPDM, cited above). From a dynamic point of view, the evolution of the XSLT mapping in the context of subscription as described in (Etienne A. et al, 2008) will also be studied. The possibilities offered by this technology are quite powerful but

there is missing functions due to the actual implementation of the W3C recommendation (functions in XSLT 2.0 would be useful but are still not implemented (W3C, 1999 and W3C, 2007)).

In consequence, complete rules have been added to the code to answer those missing functionalities. At the same time a post treatment has been applied to the file to prepare the reintegration (By instance: cutting, archiving, renaming the files).

The choice of XSLT is not incompatible with exploitation of SOA architecture. In fact, web services were used to operate the import / export of the mapped elements on both sides. Then, web services are used to answer the technical level interoperability for this specific case of PLM/PPO communication. As already said in this paper, the systematic use of web services to answer the semantic interoperability need to take the hypothesis that every CAX tool dispose of SOA architecture to cooperate with the PPO kernel. It could work for the case of PLM system but not for many other systems. However, web services use for direct semantic correspondence could solve problems encountered (like document version increment). Moreover, conforming to the interoperability definitions authors adhere to (cf. paragraph 2.1), the “service” dimension of interoperability have to be taken in account and these possibilities will be considered deeply in further works.

5. APPLICATION CASE

For our application case, the paper will observe the product data exchanges PPO-PDM of a portative electric drill. After simplifications, it will be supposed that the drill is composed of a battery block, an electric motor, a reducer, a drill chuck, a carter, an electrical command and a couple-reducer mechanism (See figure 6).

From now the paper will follow a scenario of product design step by step. And, in the scenario,

the use of the PPO system with other CAX systems will be inspired by the descriptions given in (Noël et al, 2004 and Dufaure et al, 2006). The scenario is nearby the scenario described in the following figure 5. This represents one of the possible uses of the couple PPO kernel/PLM system. Shortly, engineers express their first ideas on IPPOP. That will constitute a beginning of structure which will be managed in the PLM system.

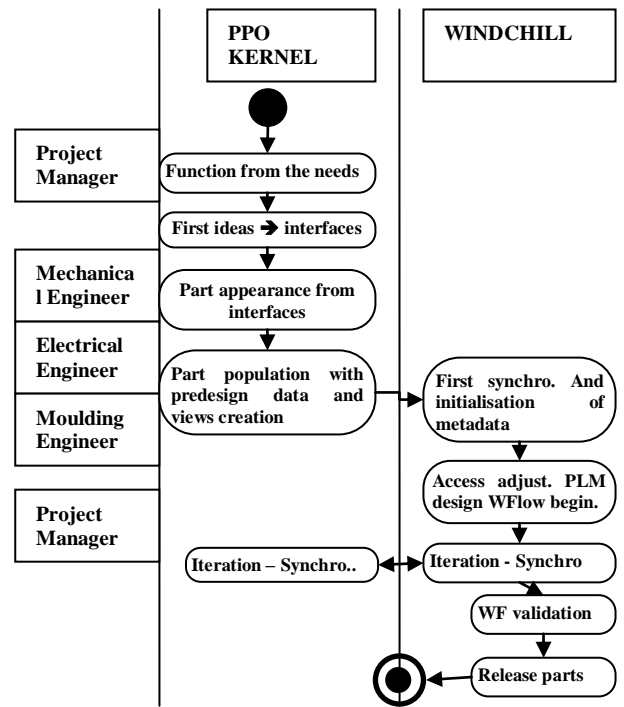


Figure 5 – A scenario of use

At the very beginning, a task “Design a drill” as been entered in PPO kernel and the needs for the designed drill are translated in “Function” objects. Then experts (Electrical designer, Mechanical designer, moulding expert, etc.) created a “component” entity of “common” type for each element of the product in the PPO kernel (See figure 6).

EXAMPLES OF CREATED OBJECTS: BATTERY BLOCK

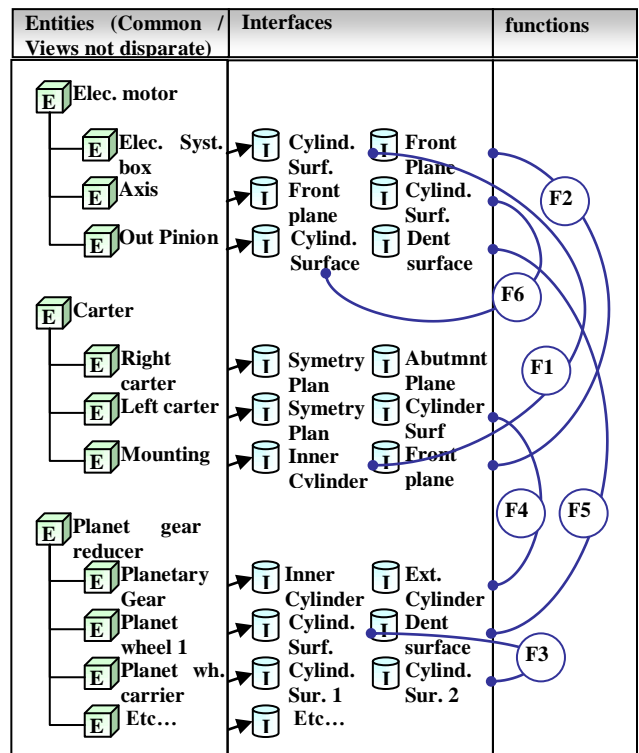
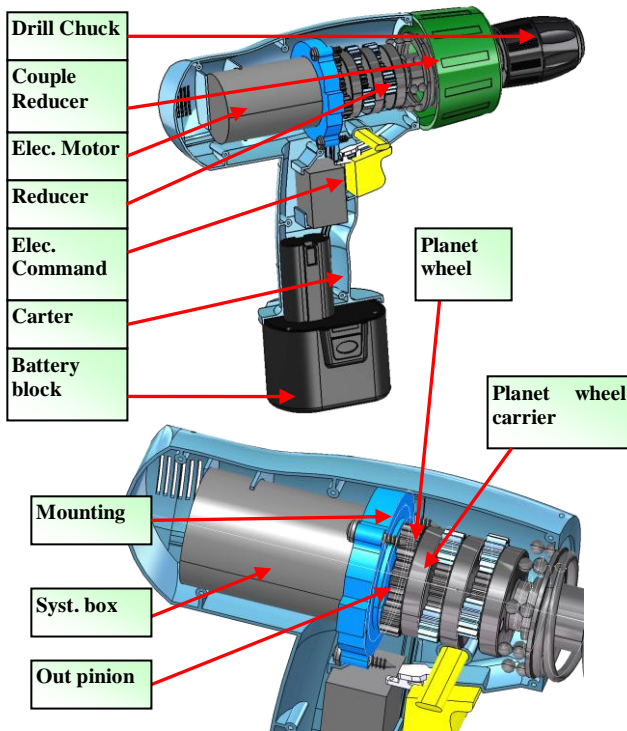


Figure 6 – Drill and PPO modelisation

After that, each “Common component” entity is loaded with pre-calculation and pre-project information. So for each “common” component entity and linked to it, experts create new “component” entity of “view” type (cf. data model reminder). There is here the notion of Viewpoint (cf. paragraph 3.2). New attributes are added whose values are filled by first parameters. “Behaviour” objects are also created then linked to the components. This step allows the mechanical designer to take in account the electrical designer parameters using the parameters coming from the electrical view (for instance the power and then the dimensions of the batteries in order to dimension the battery shell). All this drives to the next step.

From the different parameters and behaviours linked to the different “component” objects, the mechanical designer is able to build a skeleton. Conforming to the skeleton, the experts create “Interfaces” objects relatives to the different components. Those interfaces are frontier geometry of a component that will be in contact of the interface of another component. The interfaces (ie the skeleton elements) linking the components confirm the intention of engineering of the different parts and structure an assembly.

As design intention is confirmed, skeleton elements are described in CAD files. From this moment, the traceability of just created part takes a certain importance. Moreover the links between the different parts are formalized and have to be managed and justify the need of the PLM system. A remark can be that the need of the PLM system use only begin when files are produced, especially CAD files.

A first export and translation allow the creation of a product structure with the skeleton files in Windchill. In this case, the engineering BOM (Bill of Material) has been generated following the PPO entity hierarchy: this choice is adapted in the case of product constituted by low number of component. In the case the product has a lot of components; it could be chosen to structure first the engineering BOM and then design the part specification through the PPO kernel, but this is another workflow. All the created elements lifecycle is set to the “in work” step. And the version, revision and iteration are set to their initial position. At the same time, the attributes “Windchill ufid”, “Windchill master ufid”, “Windchill localid” are added as children of the attribute object in the PPO data model. Those attributes are essential to keep the links between the two systems. In fact, the names of the parts are attributes like others. They are no reliable references for both Windchill and PPO.

The entity views described in PPO does not correspond directly to the views as defined in Windchill. In the PLM system, views are assumed much as lifecycle main states (as design, as manufactured, etc.) than experts viewpoints expression. But the PPO views entity have been ordered and classified in the PLM system views of the products. Then for example, the mechanical engineer view, the electrical engineer view and the moulding expert view have been classified in the “As design” view of the product structure.

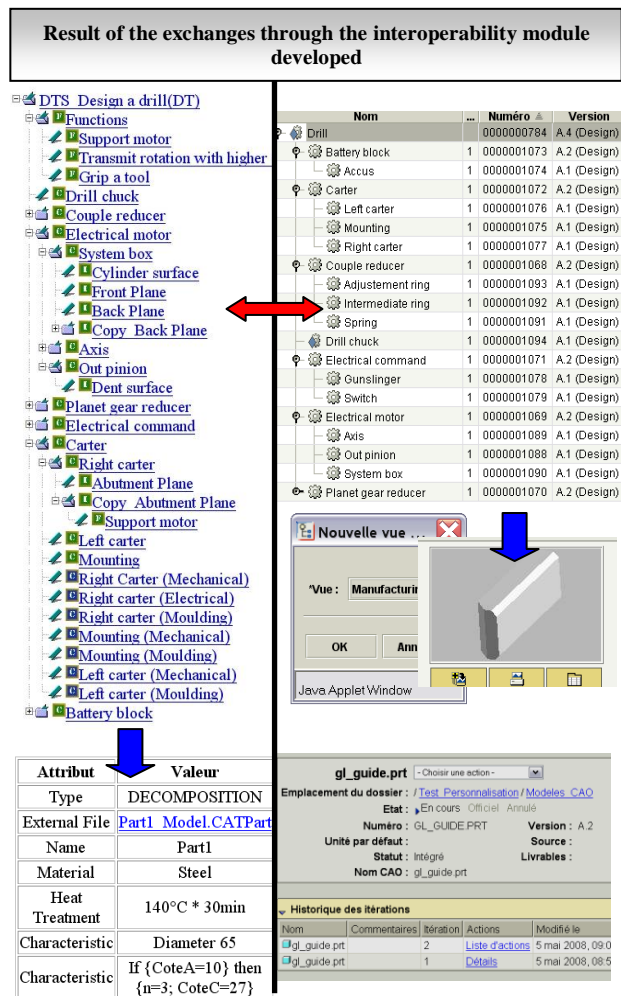


Figure 6 – Exchanges and managed information

Above is (Figure 6) an overview of the exchanges result for the objects given previous example. This figure shows the global breakdown result (like trees) and it shows that Windchill EBOM tree don't take in account information like functions, component interface, etc. as in PPO tree. The figure also show the difference of granularity of the information managed (Under the tree). In fact PPO kernel manage parameters (Ex: “Diameter 65”) and files that use the parameters (Here : “Part1_Model”) and Windchill manage only the file and metadata to manage the file (Version, Status, document name).

Now the imported product elements are integrated in the PLM system and then are reliable from the company's global point of view. The project leader creates a reference configuration for the product and manages the lifecycle of the product. The PLM system allows the partners to work on the product. For instance in our case, the drill chuck is supposed entirely made by a supplier. The supplier checks out the “Drill chuck” and the specification files to this part. When the PLM system is resynchronized with the PPO kernel, the drill chuck model is available for the different experts. After many iteration of synchronization between PLM/PPO, in the PLM system the project leader applies a workflow to the product that will be reviewed and then put in release state.

This paper has concretely shown the two different roles: One system (PPO) is in charged to link the product data (functions that come from the list of requirements, to physical phenomenon, to geometric shape, to parameters, etc.) and the other (PLM) is in charge to properly manage conforming to the companies' organization.

6. CONCLUSIONS

This paper has described how the actual PDM/PLM systems could be re positioned around a PPO kernel. The different functionalities between the two systems could be observed as well as how those systems have to be used in the product design process. A real added value in using both systems is highlighted. However the example has been, so far, quite succinct due to the effort spent on the correspondence development between the two systems. In the future, it would be interesting to develop this case more and integrate other systems.

This paper has shown the abilities of the PPO kernel to interoperate with CAx system without hardcode link but through mapping definition files in XSLT. In this case the exchanges were limited to product data exchanges and did not really explore the “services” exchanges. This will be a next part of our work. Services will also allow making the link between processes of the different systems.

This paper also shown that XSLT technology allows doing some work but there is still some missing element in the implementation due to the youthfulness of the technology. Moreover, the mapping has been written by ourselves, but further work is to integrate this technology to the concept of “user subscription” (The user choose himself the concepts to map as one goes along) could be interesting.

One difficulty with the PPO kernel is that most of systems have their own and private data model. In

consequence, it is really difficult extract this data model and there is still the problem of the syntax of the extract model.

REFERENCES

- Bianconi F., Conti, P., Di Angelo L., "Interoperability among CAD/CAM/CAE systems: A review of current research trends", *Geometric Modeling and Imaging New Trends*, 2006, art. no. 1648749, pp. 83-89.
- Sudarsan R., Fenves S., Sriram R., Wang F., "A product information modeling framework for product lifecycle management", *CAD Computer Aided Design*, 37 (13), 2005, pp. 1399-1411
- Ford T., Colombi J., Graham S., Jacques D., "A survey for interoperability measurement", 2007, 12th ICCRTS "Adapting C2 to the 21 st Century"
- IEEE, Institute of Electrical and Electronics Engineers, "IEEE Standard Computer Dictionary: A Compilation of IEEE Standard Computer Glossaries". New York, NY: 1990.
- DoD, United States Department of Defense, Joint Publication 1-02, "Department of Defence Dictionary of Military and Associated Terms", 12th April 2001 (amended 23rd March 2004); p270.
- ATHENA A4, "D.A4.2 Specification of Interoperability Framework and Profiles, Guidelines and Best Practices, Version 1.0", ATHENA IP, Deliverable D.A4.2, March 2007.
- IDEAS, Interoperability Development for Enterprise Application and Software, IST project, 2002). <<http://www.ideas-roadmap.net>>
- Yang J., Han S., Grau M., Mun D., "OpenPDM-based product data exchange among heterogeneous PDM systems in a distributed environment", *The International Journal of Advanced Manufacturing Technology*, Division of Industrial and Information System Engineering, Ajou University, San 5, Wonchun-dong, Yeongtong-gu, Suwon, 2008, pp 443-749, South Korea
- CIMDATA, "Eurostep's Share-A-space, Product Lifecycle Collaboration through Information Integration", White paper, October 2007.
- Gunpinar E., Hanmin L., Soonhung H., "Interfacing heterogeneous PDM systems by PLM services for design collaboration", *Computer Supported Cooperative Work in Design*, 2006, CSCWD '06. 10th International Conference on, 2006
- Lämmer L. and Bugow R., "PLM Services in Practice", the Future of Product Development Proceedings of the 17th CIRP (College International pour la Recherche en Productique) Design Conference, 2007
- Million O., *De l'intégration des métiers par les données techniques vers la maîtrise de la modélisation conceptuelle : la méthode V.I.M. (Viewpoints Information Modelling)*, Thèse de doctorat, Université Henri Poincaré Nancy I, 1998.
- Harik R., Lombard M., Derigent W., "Product model adjustment methodology to support its product lifecycle management", CPI'2005 (Conception et Production Intégrées), Casablanca, Morocco, 2005.
- Twigg D., "Managing the design/manufacturing interface across firms", *International journal of Integrated Manufacturing System*, 13(4): 221-221, 2002.
- Nguyen Van T., "System engineering for collaborative data management systems: Application to design / simulation loops", PhD Thesis of the ECP (Ecole Centrale Paris), December 2006
- Guyot E., "Gestion de données techniques et structuration des échanges au sein de partenariats de co-développement aéronautique », *Mémoire du Diplôme de Recherche Technologique de l'Université Bordeaux 1*, July 2007
- Van Wijk D., Eynard B., Roucoules L., "Towards collaborative design platform for the aeronautics industry: Improvement of data exchanges", PLM08, Product Life Cycle 2008, Seoul, 2008
- W3C, "XSL Transformations (XSLT) Version 1.0", W3C Recommendation 16 November 1999, Retrieved: 25/04/2008, <<http://www.w3.org/TR/xslt>>
- W3C, "XSL Transformations (XSLT) Version 2.0", W3C Recommendation 23 January 2007, Retrieved: 25/04/2008, <<http://www.w3.org/TR/xslt20/>>
- Noël F., Roucoules L. and Teissandier D., "Specification of product modeling concepts dedicated to information sharing in a collaborative design context", *Proc. IDMME 2004 (Integrated Design and Manufacturing in Mechanical Engineering)*, Bath, UK, 2004, pp. 1-12.
- Dufaure J. and Teissandier D., "Integration of a tolerance analysis tool (MECAMaster) in a Tolerancing framework based on IPPOP, an open collaborative platform", *IDMME 2006 2004 (Integrated Design and Manufacturing in Mechanical Engineering)*, Grenoble, France, 2006, pp.1-10.
- Etienne A., Guyot E., Cabannes G., Ducellier G. And Roucoules L., "Specification and developments of Interoperability solutions dedicated to multiple expertises Collaboration in a Design Framework", SKIMA08 (International Conference on Software, Knowledge, Information Management and Applications), March 2008.