

# **Open Standard, Open Source and Peer to Peer Tools and Methods for Collaborative Product Development**

Hayder Aziz and James Gao, Enterprise Integration, Cranfield University

Paul Maropoulos and Wai M. Cheung, School of Engineering, University of Durham

Keywords: PLM, knowledge management, peer-to-peer, STEP, collaborative product development

## **Abstract**

This paper reports on a collaborative product development and knowledge management platform for Small to Medium Enterprises. It has been recognised that current Product Lifecycle Management (PLM) implementations are document oriented, have a non-customisable data model and inter-enterprise integration difficulties. To overcome these, an ontological knowledge management methodology utilising the semantic web initiative data formats was added to a PLM and an open-source alternative. Shortcomings of centralised architectures are highlighted and a solution using a de-centralised architecture proposed. This is implementable at low cost, the scalability increases in line with user numbers. Ontologies, rules and workflows are reusable and extendable.

## **1. Introduction**

This paper builds on the initial work carried out by the team as per Gao (2003), Cheung (2003) and Aziz (2003). The topics of collaborative product development, virtual enterprises (VE) and the technologies associated with implementing them have

been the subject of widespread research. Some of these studies tackled problems within single enterprises, others looked at the problems associated with creating and managing of VE (Beckett 2003). Tools such as product lifecycle management (PLM) enable collaboration within and between enterprises.

### **1.1 The Lack of Support Tools for SMEs**

There are a number of facets that the authors consider to be basic flaws in current thinking on the subject. Large enterprises have invariably been the target of software vendors for development of such tools, resulting in centralised applications. These are invariably beyond the means of small to medium enterprises (SMEs). Even after these efforts had been made, large enterprises face numerous difficulties with PLM. Firstly, enterprises evolve, and an evolving enterprise needs an evolving data management system. With large applications, such configuration changes have to be made at the server level by dedicated staff. The second problem arises when enterprises wish to collaborate with a large number of suppliers and original equipment manufacturer (OEM) customers. Current applications enable collaboration using business-to-business (B2B) protocols. However, these do not take into account that disparate enterprises do not have unitary data models or workflows. This lack of standardisation results in a lot of time and money spent mapping the B2B links. Within the setting of a VE, where alliances can include dozens of enterprises and may often last for only a single project, these costs are prohibitive. This is a strong factor in reducing the abilities of large enterprises to participate in one-off VE partnerships, reducing agility and losing potential business. Other solutions include the use of application service providers (ASP) and also enabling a large enterprise's data system to function as an ASP to their collaborators. However the problem with this type of system is the management of intellectual property of the individual enterprises and whether enterprises can give up control of their information to a partner, who may also be

collaborating with their competitors simultaneously. Such fears have hindered the formation of VEs.

The enterprises' prime asset, its knowledge, is not managed coherently thus perpetuating an 'invisible' limit on the company's knowledge, based on the impulse of knowledge workers' recollections of previous experience. In addition, the problems for inter-enterprise collaboration are also inherent when enterprises with different level of detail and different nomenclature collaborate. Finally, the prime reason for this investigation is that SMEs have up to now been left out of developments in PLM, knowledge management and VE developments in spite of forming the majority of the world's engineering community. This negatively affects the ability of SMEs to manage their knowledge, reuse existing expertise, collaborate with other SMEs in the virtual enterprise settings and collaborate with the larger enterprises that form their customer base. This weakness affects not only the SMEs but also the large enterprises that utilise SMEs within their supply chain or VE network.

Smaller engineering enterprises lack the infrastructure and manpower for complex solutions. Additionally, it was found that employees simply do not have the time or the budgetary resources to learn the necessary application functionality and new concepts associated with their operations. It has been observed that rigid procedures and centralisation do not produce the desired results. In fact adherence to the procedures and workflows is generally considered to be a nuisance as knowledge workers, who follow procedures, are detached both from the process of drawing up these procedures and methodologies and from the running and "control" of these processes and tools. This study has focused on the experiences of automotive and discrete machining companies in integrating their product realisation cycles with their

supply chains and their customers in real-time, and also to enable the companies to make rapid appraisal and cost-estimation for their customers, from simple concept designs. The system has to be able to reuse the company's existing base of knowledge and to push the manufacturing knowledge higher up into the design chain to reduce the need for costly and time consuming reworks and engineering changes.

## **1.2 Specific Requirements of SMEs**

Based on the above set of problems facing industry, the authors have drawn up three core requirements to codify the needs of SMEs in managing their product and process information, i.e.,

- (i) Enabling project managers and all knowledge workers to have access to the applications needed to create and manage knowledge within their domain according to the agreed nomenclature and ontological representation;
- (ii) Information created has to be in a form that can be queried, reused and transformed into new representations through the use of rules and agents; and
- (iii) Enabling the real-time collaboration between SMEs, and larger partners, by facilitating the fast and cost-less construction of VEs;

To test the requirements and try to meet the objectives the authors took an evolutionary approach by assessing the current technology and methods, and then constructing three new example applications. The as-is technology used was a web-based commercial PLM system. The aim of the research is to solve the problems of the as-is technology via three test scenarios: (1) Modification of the PLM system for flexible and customisable data model; (2) Implementation of a functionally equivalent system based on open source tools and (3) The implementation of a peer-to-peer (P2P) based system.

### **1.3 Previous Work**

Lihui et al (2003) wrote an extensive survey of collaborative design systems, and highlighted eight areas as having scope for development including: System architecture for web-based collaborative design, Collaborative Conceptual Design Modelling and Data Sharing, Conceptual Design Selection, Knowledge Management in Collaborative Environments. Beckett (2003) discussed the topic of communication and understanding in VEs between unfamiliar participants. He discussed the best tools and standards to apply in VE settings for collaboration and knowledge management and has an overview of various methodologies and applications. Camarinha-Matos (2003) on a similar note, reviewed current trends in VE developments, and conclude that there is a need to develop a generalised framework for VEs, to enable harmonisation, international collaboration and rapid deployment.

Kim (2001) developed a 'Distributed open-intelligent PDM' system, which adopts ISO standard STEP, whilst offering standard PDM functions. A dynamic and flexible workflow model is implemented. This could greatly enhance the flexibility of the system. Zha (2001) proposed a STEP based application to manage the entire product lifecycle. The information that is not already defined in STEP is modelled in EXPRESS. The system is focused primarily on assembly mating features and does not consider the machining requirements of each component. Vasara (2003) proposes ARACHNE, the adaptive network strategy to enable integration between 87 enterprises, the authors highlight the benefits of peer to peer networks to achieve synergy between collaborators. Their developed methodology called RosettaStone enabled many-to-many integration between enterprises using a three-tier architecture.

## 2. Knowledge Representation Methodology

Figure 1 illustrates the data persistence problem in document oriented PLM. Documents and CAD models contain the company's knowledge asset with meta-tags to identify the basic purpose of the file. In the figure 'Kn' denotes an item of knowledge. Both documents A and B within the system contain unique knowledge. However, kn-1 is stored in both documents. In this example, a user viewing or modifying document A or B will need to be instinctively aware of such data duplication and update both documents kn-1 values. In most instances such 'simple' issues cause major problems for larger teams where the volume of documents contained in the vault is too large for individuals to be instinctively aware of it. Additionally the problem of finding knowledge, as opposed to documents, is intrinsic.

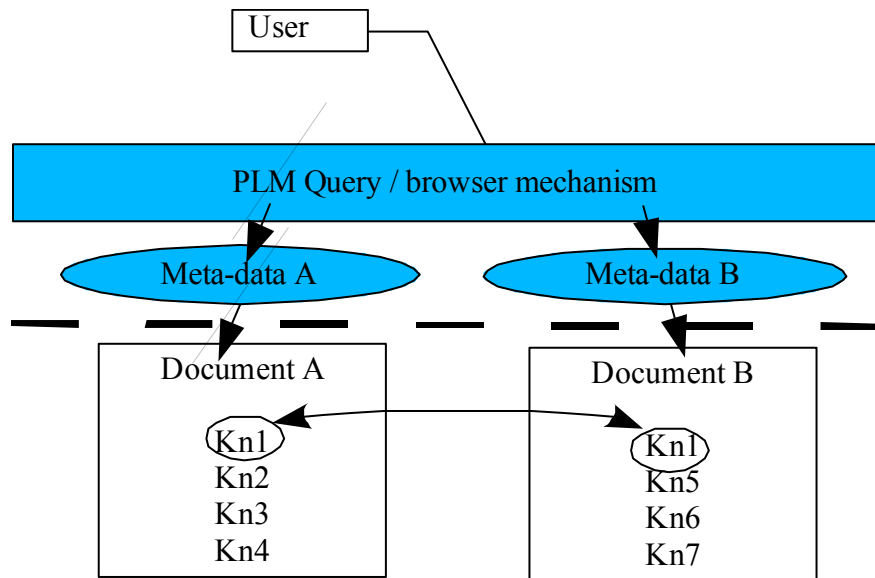


Figure 1: Data persistence in document oriented systems

The above problem can be solved by Knowledge representation in this methodology is split into two identifiable components. The first one is the encoding and nomenclature component with the second component defining the terminology of the knowledge to be represented. This distinction between the encoding and terminology

enables the deployment of flexible knowledge bases using different industry schemas whilst retaining the integration of the knowledge base. This has meant that standards such as STEP have to have their schemas translated into the encoding specifications created, as opposed to the original Express schemas. The authors define knowledge as the semantically complete definition of a domain's information that is both machine readable and interpretable. Thus a knowledge base contains an ontology which defines the classes and their relationships, instances or objects that form the domain 'data', meta-data that constrains the data within a particular domain (transforming it into information), and Universal Resource Identifiers (URI) that allow the global identification and contextual interpretation of the information. These schemas are shared within the collaborative environment of a project involving many parties.

The resource description framework [www.w3c.org](http://www.w3c.org) (RDF) has been used in this application as the format for encoding the knowledge base, as opposed to eXtensible mark-up Language (XML). The reasons for this choice are the extra flexibility and 'machine-understandable' format of RDF graph triple model as opposed to the simple 'machine-readable' XML based mark-up vocabularies. In effect, any RDF-parser can derive the semantics and context from the URI and metadata attached to every instance.

The STEP AP-224 feature models are defined in Protégé and use the Java expert system shell (JESS) to define the rules and functions of the standard. In addition some of the rules have been created using the Protégé Axiom Language (PAL) and the constraints these impose enable error-checking for the user during the definition of features. However, these constraints can only be applied to individual entities within the STEP model and cannot enforce any constraints between entities/features. Due to

the size of the standard, only a subset of AP-224 has been translated from the original Express schema. The definition of the Express (STEP) data types is also contained in the Clips interface. This is very flexible as it allows for the inclusion of STEP data types to other components within the ontology on a need basis without having to have any expertise in Express or any other programming language. The above enables the mixing of feature and meta-data information in the knowledge base, meaning that users can access the information stored in STEP models using queries and RDF parsers. This integration at low level between the geometric, feature and 'meta-data' within a single environment is intended to reduce repetition and errors, and also enables the reuse of all the data created during the conceptual design process. A STEP AP-224 based automatic process planner by Sharma (2003) generates plans from concept designs. The aggregate process planning tool by Cheung (2003) generates assembly plans from incomplete models to enable the optimization of the design for assembly and manufacturing from the early stages.

### **3. Collaboration on Distributed Designs**

One of the first problems of collaboration (both at system and at user level) is trying to understand what the others say and mean. The previous section showed the methods used to create the data models. This development is interrelated to the problem of collaboration and problems of collaborative design between enterprises of different sizes and complexity, that is, mapping between low and high content data models which results in irretrievable loss of information from the high content data model. This problem cannot be overcome traditionally by creating a mapping from one data model to another. Instead the authors have sought to create a project oriented ontology that can be created, shared and used by all parties collaborating in an enterprise in real-time. This eliminates the problems faced when low-end suppliers collaborating with advanced enterprises face integration issues. Using the building-



block ontologies and STEP standards to describe the terminology. There are various collaboration 'enabler' methods available with different strengths and weaknesses. Some are centralised, controlled by a third party or de-centralised.

### **3.1 B2B Integration and Application Service Providers**

Business to Business (B2B) integration is the traditional method for companies to collaborate and create VE networks. This area includes Electronic Data Interchange, XML based messaging and portal solutions. Integrating 15 companies together on a one-to-many basis where a single repository manages the project knowledge and workflows would need 14 separate mappings.

Application Service Providers (ASPs) can be set up in two ways, either by a large 'controlling' enterprise, or through independent third party hosting. These services intend to provide the same utilities as enterprise level systems but in a non-enterprise specific service. ASPs offer project and product data management vaults where the VE administrator can customise the third-party portal for their own use. It also offers opportunities to contact companies already listed on the ASP who might join the VE if they had required services to offer. The ASPs generally use software very similar to the PLM solutions like Enovia and Windchill. However, they offer them as low cost solutions to some customers with specific needs, which cannot be met by purchasing their own server software. These include: Reduced cost for the enterprise as maintenance and backup is delegated. Increased opportunities if customers and VE initiating enterprises seek out partners through the portal. This is a "Democratic" system where no one enterprise controls the VE server and data. It sets down de-facto standards for data exchange, to which other enterprises in the same domain will adhere to in order to join the network of enterprises.

There are of course some fundamental disadvantages to the use of ASPs, and other centralised systems, for product development, and these include: The bandwidth and server bottleneck problem associated with centralised services. The security fears of intellectual property rights being compromised. The potential risks of downtime and data losses in an “uncontrollable” environment and the liability issues associated with it. The difficulty of creating direct interfaces from the enterprise system to the ASPs portal. The exact functionality required for the VE may not be available from the “generic” ASP.

There are already some ASPs operating in the automotive and aeronautical sector enabling supply companies to interact and bid openly for contracts with OEMs and then manage the project/product information on the portal. However due to the disadvantages highlighted above, the authors sought to find a third way. Whilst traditional client-server systems can operate in a collaborative manner, for example over a LAN or Internet, they are not truly distributed as they are centralised.

### **3.2 Peer to Peer Systems**

Peer to Peer (P2P) applications address the needs of de-centralised organisations to collaborate and share knowledge regardless of geographical location. The principle of P2P has been around for a long time, and is today implemented in a number of applications such as instant messaging and file-sharing ([www.GNUtela.com](http://www.GNUtela.com)). There are already a number of P2P PLM in existence. Primarily aimed at the lower end of the market. The two commercial applications are AutoManager workflow from Cyco ([www.cyco.com](http://www.cyco.com)) and Columbus from Oasys Software (<http://www.oasys-software.com/>). The latter is available for free, and aimed at AutoCAD users within the construction sector. However they are crude solutions relying on the underlying file system and adding some "meta tags" to files for version control. There is no

workflow or process management implemented and access control is via the standard operating system access control functions. As an example of what can be achieved, Alibre is a P2P CAD/PLM and collaboration tool in one. It uses the STEP standard and combines low cost and fast configuration.

The advantages offered by P2P applications are (i) no single point of failure, the network is alive as long as one peer is on-line, (ii) distributed sharing of bandwidth storage and processing power, so the system becomes more powerful as more users attach, (iii) lower running cost due to the lack of servers or high bandwidth central nodes, as well as (iv) maintaining individual control of the shared knowledge. P2P groups can be used to create profiles of the peer, and also more importantly of the peer's list of contacts within different domains. These profiles can be used within the network to search for and assess people's competences, interests, and memberships of trusted groups, and can aid in the construction of new relationships based on commonalities and third party assessments. There have been a number of issues that reduce the performance of the system using pure P2P architecture. The lack of indexing and routing services in P2P degrades the peer discovery and query functions. In order to leverage the advantages of client/server systems with the independence and interoperability of P2P systems a hybrid system where "super peers" act as peers to the extended P2P network and as a server to the enterprise's internal peer network is used. In addition rendezvous peers can be assigned to manage some of the peer information assigned to particular peer nets or projects. This hybrid has been shown to have the best potential for high-performance de-centralised services.

### **3.3 Inter-enterprise Communication Architecture**

Since no two enterprises are the same, the idea of using XML based messaging for inter-enterprise collaboration is not easy as the two company schemas have to be

mapped to each other. In order to achieve the speedy interoperability a standard has to be set for basic messaging. There are a plethora of standards in development by various groups including RosettaNet, an endless list of XML schemas and data type definitions. These standards implement different levels of detail in the ontology. Some like STEP PDM attempt to implement the complete information structure of the engineering enterprise. Others like JuxtaPose (JXTA) implement only the “messaging” components. The choice of standards to use depends on the level of “standardisation” that all enterprises can adhere to. In the view of the authors only a very low level subset of all enterprises can be “standardised”. This small subset should exist only as a medium to enable communication, identification and access control management. All other aspects of the collaborative data environment are enterprise specific (although constructed from a subset of ontological components).

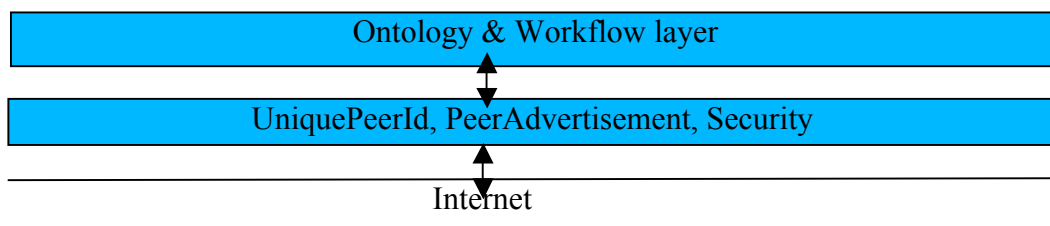


Figure 2: The 2-tier communication

This project uses the two-tier solution shown in Figure-2 which enables enterprises to establish connections in real-time between them. Speeding up of the integration process is further enhanced through the use of open standard semantic web enabled ontology formats for storing the information, plus the use of open-standards in addition to company ontologies to define those ontology components.

## **4. Case study**

### **4.1 Commercial PLM System**

Windchill is an out-of-the-box PLM system. It uses open source technology for some of its key components. A traditional document management tool, that stores all manner of data and make revision controls. It does not however have an intelligent method of containing and persisting information in an object oriented format. To alleviate this, the functionality of the system has been extended to include the management of knowledge in an ontology. The integration with Protégé was made using Windchill's bespoke but highly configurable workflow engine.

### **4.2 Open Standard and Open Source Tools**

The applications used in this project are the Protégé ontology editor and Sun's JXTA peer to peer protocol. Both of these tools offer the best usability, portability and cost/performance capability within their niches, and when allied to a scalable open source object/relational database such as SAP-DB, can offer enterprise level performance for zero capital expenditure, and low customisation cost. Moreover they free the customer from the clutches of software vendors and the vagaries of obsolescence. Other open-source tools used in this work include OPEN-CASCADE, which is a native STEP solid modeller and the open-flow tool that forms the foundation of the open-source PLM solution.

In the application (see Figure 3) the bottom layer consists of the database. This can be chosen from the range of relational databases available, and for this test case MySQL was utilised due to its stability, ease of use and close integration with the PHP

scripting language that the open source PLM was written in. On top of this layer the two open source gateways, one is Apache web server for serving static and PHP based web pages and the middle tier is the Tomcat servlet engine which serves the Java server pages (JSP) based applications. These three layers (database, web server and application server) form the server side of the system.

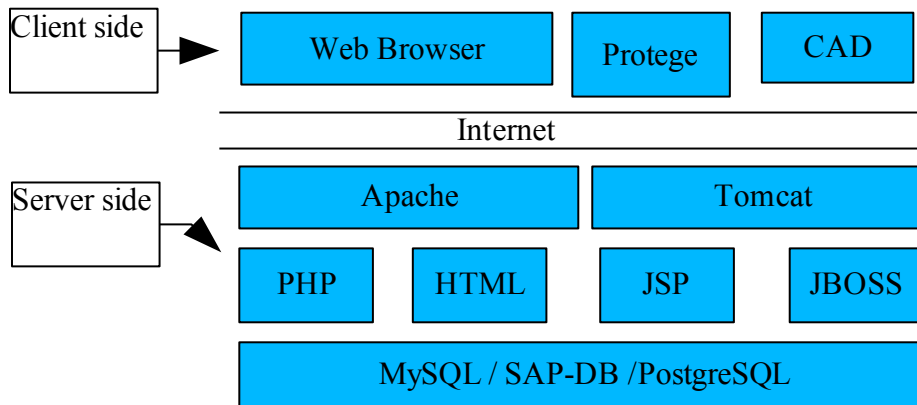


Figure 3, high-level view of open source server and user interface architecture

On the client side, the user has three main applications i.e. (i) the web browser through which interactions with the PLM system are carried out, (ii) the Protégé Java applet that allows the user to query and manage the knowledge base and (iii) a CAD system to enable the user to create and manipulate the STEP based models held in the PLM system. This user interface ensures ease of use for the user because the web-browser and its own cad system are already familiar to the user with the only 'new' tool for interaction being protégé. This extension also includes the management of STEP entities within the PLM system in an intelligent and object oriented manner. Knowledge is stored as objects within the modified PLM system. This was integrated with the lifecycle management and workflow functions offered by the PLM system.

### 4.3 Peer to Peer Implementation

The implementation was kept as simple as possible. The simplicity of the architecture is itself a bonus and enabled the researchers to concentrate on the value adding aspects of the project. As mentioned previously, the flexible Protégé ontology development and user interface environment is used. The back end consists of the open source SAP-DB database with the Java database connectivity (JDBC) connector to Protégé. The knowledge base ontology is defined in a Clips file and the instances are stored in the database. Connectivity is achieved using an open source implementation of JXTA open standard P2P network protocol ([www.jxta.org](http://www.jxta.org)). The choice was made because JXTA implements a unique but anonymous identification mechanism for peers and for rendezvous peers. As well as “advertisement” implemented for all peers that give information on the one peer to other peers. Rendezvous peers can act as managers for peer groups and store the peer advertisements for the group for distribution to other P2P networks. Implementation of JXTA is in Java 2 standard edition, an extension to enable RDF queries and ontologies to be shared over P2P is used to share the Protégé knowledge base. The second window shows the user interface from where queries, project management of collaborative groups and such widely used items as group chat and instant messaging are readily implemented by the JXTA protocol. The systems’ settings enable enterprises and users without static addresses to collaborate using dynamic addressing, and this flexibility as well as the users’ ability to work offline (that cannot be done with web based systems) empowers users in all possible circumstances.

## 4.4 Example data

As stated earlier, an automotive enterprise was used to demonstrate the application and collaboration methodology. When initiating a collaborative product development project, a list of tasks, users, roles, responsibilities are assigned. Initially this is very sketchy, however the template ontology (implementing the STEP-PDM entities) as well as the generic-product development ontology contains both the fuzzy early stage and accommodates the updates and refinements as they are entered. The enterprise's processes were split into lifecycle states and defined in workflow processes. Due to the complexity of the overall processes, only the conceptual design phase was modelled into the ontology and workflows. The sequence of processes for this phase are:

- Customer request for quotation is submitted including requirements and associated verification methods;
- Generation of specifications through analysis of earlier customer requests and testing against the new verification methods;
- Creating concept design options from the specifications by pattern matching against earlier concepts developed with similar specifications; and
- Selection of concept and utilising the verification methods.



User interaction through the interface as shown in Figure 4 was straightforward.

Workflow sequences acted as widgets guiding the user through different

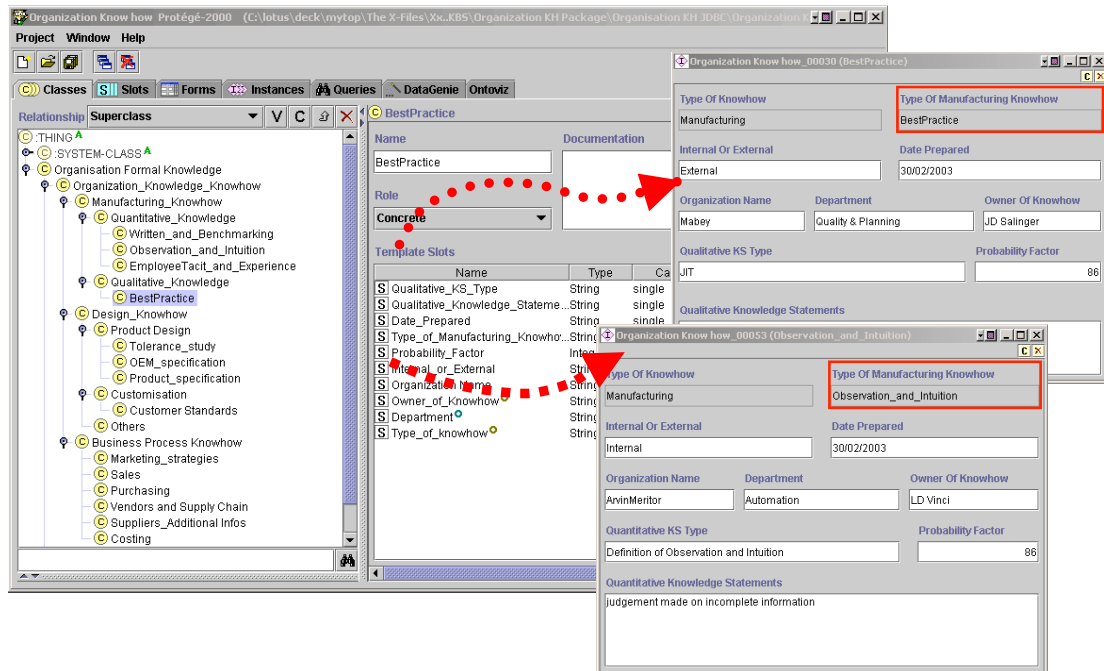


Figure 4: Protégé user interface for knowledge acquisition

configurations. The example in Figure 5 assigns budget and feasibility analysis to actors, manages release of the individual tasks, which are sent for review for approval / rejection.

The retention plate and four item assembly for the door latch (see Figure 6) was modelled for the entire lifecycle. In addition the full project data including over 300 parts were modelled into the Bill of Material. New components are entered interactively and work seamlessly throughout the lifecycle. The application proved very simple and intuitive to use. The collaboration tools use proven technologies by SUN and worked seamlessly for many-to-many connections.

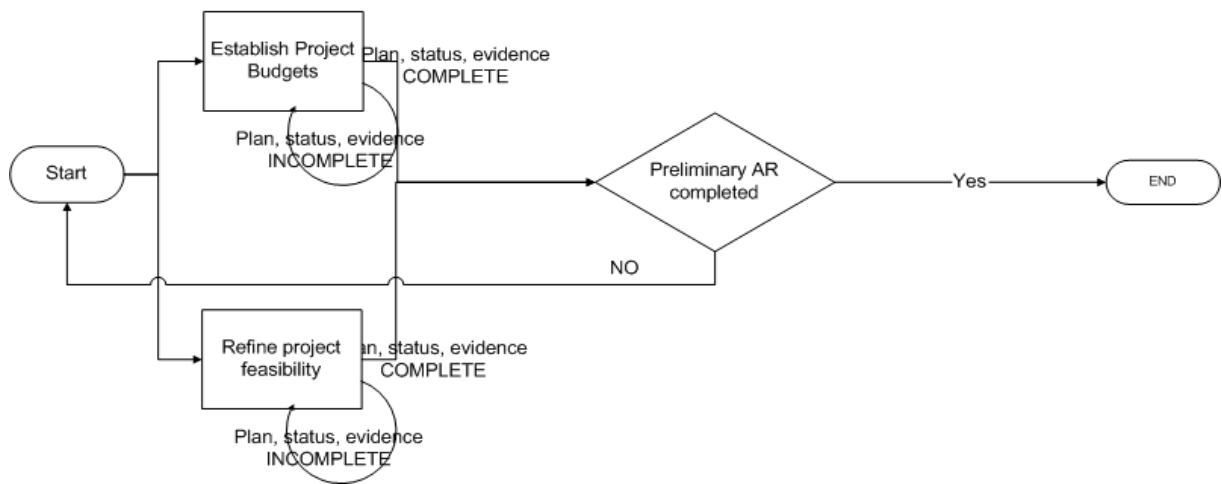


Figure 5, workflow process, visualized from xpdL,

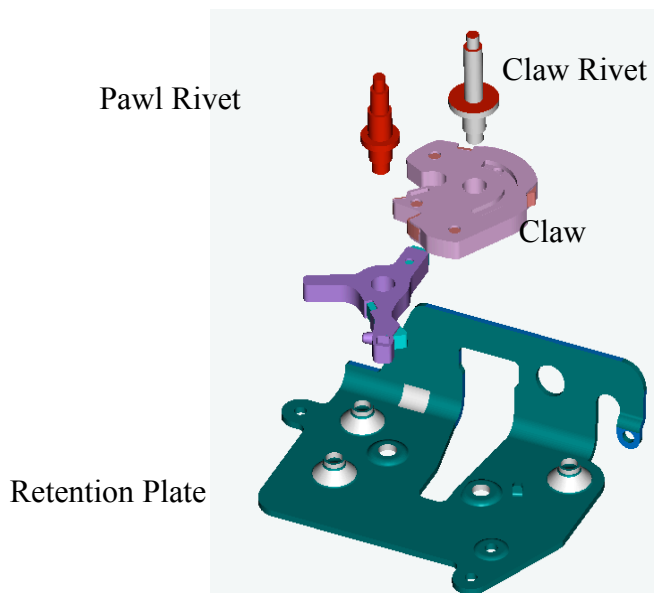


Figure 6, Latch assembly view. (Courtesy ArvinMeritor)

## 4.5 Results

The field for the results above were tested empirically using the data and a small inter-enterprise setup described previously. The server was located in Southern England, and users concurrently created the new project, generated new concepts and optimised the design gradually from two remote locations in addition to users at the main site.

Application	ontology	VE setup	Deployment time	Lifecycle management	Total cost of ownership
Windchill	RDFs ontology in document container, access via windchill exp	B2B integration via Info*engine (one-to-many integration), manual	1 month for small project.	Graphical workflow	Server & client licence, implementation and maintenance
Open PDM	RDFs ontology accessed via Protégé applet	B2B integration via XMLRPC (one-to-many integration), manual	1 week	PHP workflow	Implementation and maintenance
Pure P2P	RDFs ontology in java application	Peer to peer, automatic discovery	1 day	Version control and rule based system	Training on P2P and knowledge app
SuperPeer net	RDFs ontology in java application	SuperPeer automatic discovery	1 day	Version control and rule based system	Training on P2P and knowledge app

Table 1 Comparison of key features of tested system

Table 1 compares the key features requisite for a collaborative project environment, these were:

**Ontology:** how easy was it to enter, query, and reuse both the ontology and domain knowledge within the collaborative group? Since this is common to all the tools, the only difference was the way in which the ontology was stored within the different systems.

**VE setup:** how rapidly could the VE be set up and start to operate on all the remote sites?

**Deployment time:** The time to customise and deploy the entire project including the project management framework (access rules, task lists, workflows), project ontology (sub-ontology selection from the generic ontology), and communications setup.

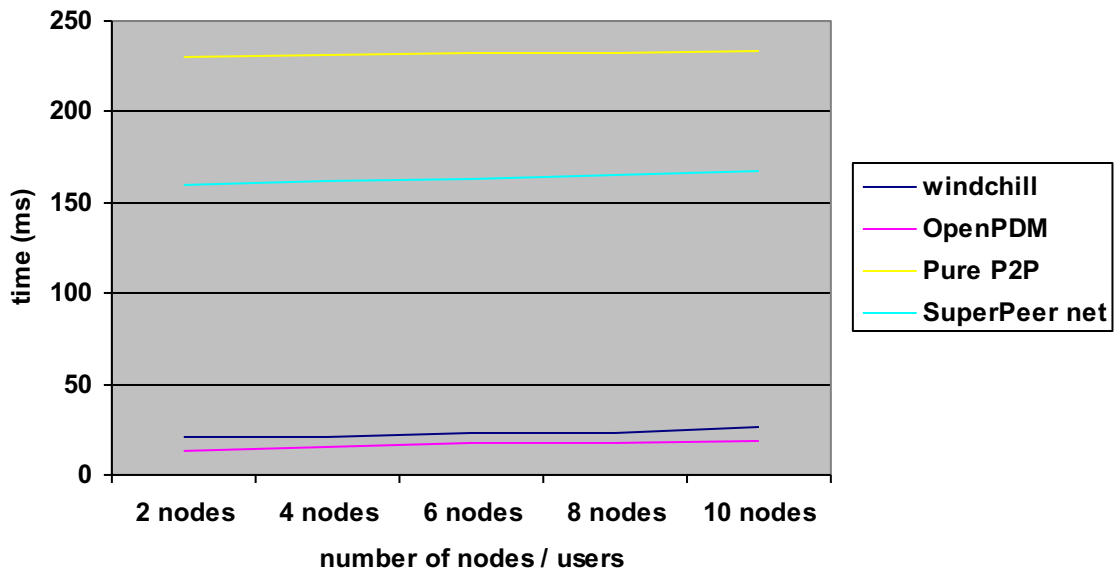
**Lifecycle Management:** The ability to control the state of a document, manage versioning and history of the data.

**Total Cost of Ownership:** The costs of licences, implementation, system integration, training and maintenance.

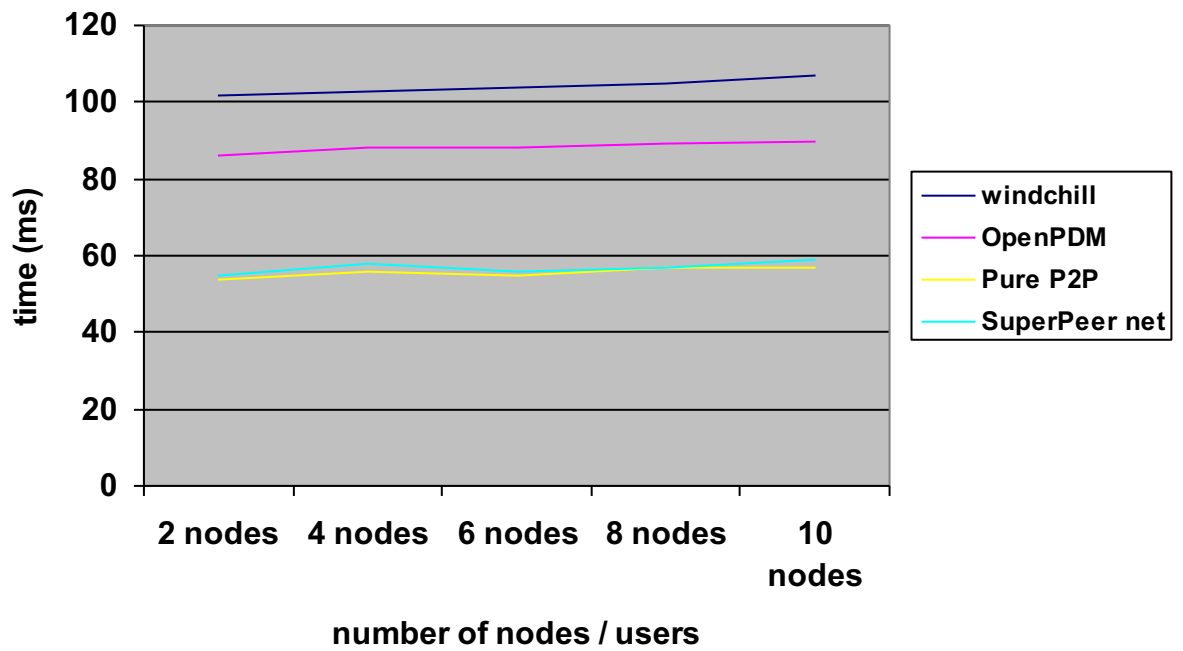
All ontologies deployed shared a common back end. The applications were tested on identical hardware with identical configurations and internet connections. The apparatus were Dell Xeon workstations with 1GB RAM, 15K rpm disks and 1mb internet connection, the operating system was Windows 2000 on all machines.

Due to the small scale of the test it is impossible to gauge anything but the most rudimentary performance benchmarking of the systems as high loads of queries / updates and multi-peer concurrent access could not be simulated with the available resources of users. However the test suffices for a small scale pilot. The test simulated a small scale VE setup through from initiating the VE to running the system within the collaborative network. This setup has a typical VE with a large number of small servers distributed within a common network, a small number of transactions are processed at each node.

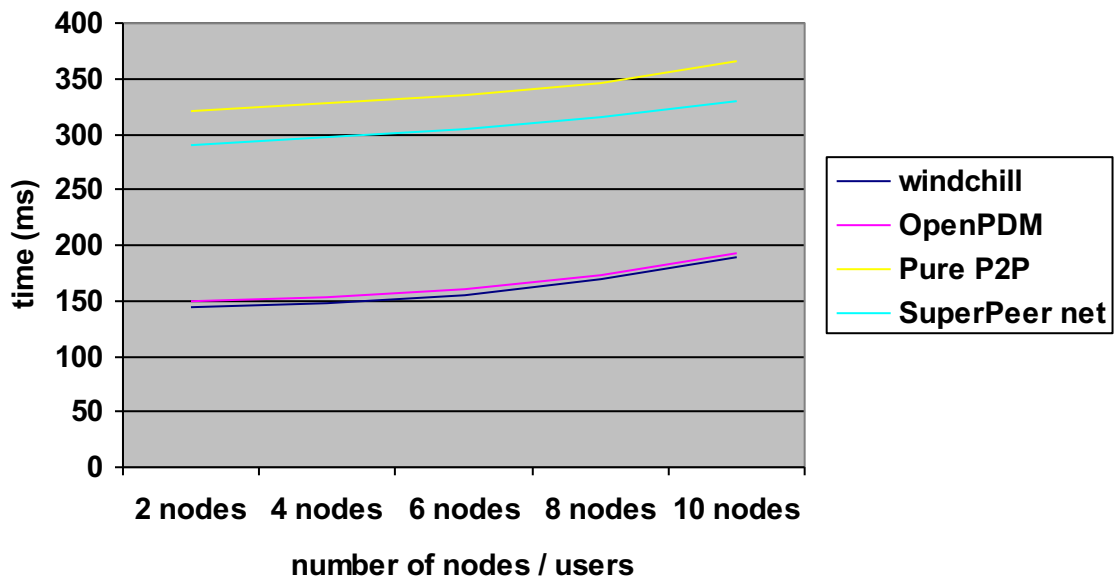
**Chart 1. Simple query (single term search over the entire KB)**



**Chart 2. Time to update instance**



**Chart 3. Complex Query (generating an inferred hierarchy of the KB before processing a result tree)**



Charts 1-3 show some performance characteristics of the systems in use, as stated earlier the number of users were not sufficient to fully test the scalability of the applications, however the distributed nature of the applications means that the number of concurrent transactions are much lower in P2P type applications than centralised programmes. The Peer based applications showed a base performance lag when compared to the client-server applications due to routing of peer based systems without an indexing service. Otherwise the performance of the systems in practical use showed no noticeable differences.

## **5. Discussion**

Windchill is inherently centralised, overly complex to set up and requires a long period of time for customisation. The system's main strengths are in the workflow tools. However, even here there is the problem of lock-in. The workflow and any customisation carried out cannot be reused on another system. Windchill has a sophisticated application layer for inter-enterprise communications, and inherent in its weakness is the client/server paradigm whereby every enterprise needs to have

custom integration between it and its extended enterprises' systems to function. This has been elaborated on in the previous sections. On another note, the RosettaNet standard, implemented in Windchill, can decrease the amount of customisation needed, but not eliminate it as RosettaNet only provides a subset of the messaging and data models of the inter-enterprise link. In addition, unlike the ontological 2-tier system, Windchill's data models and workflows are neither portable nor standardised.

Other issues encountered were the difficulty in modifying the data models. Windchill has a rather rigid data modelling interface based on the Unified Modelling Language (UML) to model the Java data model classes, and generate SQL scripts for the database. It can be modified by a professional who has to model, program, compile, update the database and integrate the code into Windchill before it can be used. This does not provide for the flexibility and ease of use for modelling ontologies that the project manager needs. The costs of implementation, licences and maintenance are very large by themselves and only practical when the enterprises involved can share the cost between each other over a long period of co-operation (as in an extended enterprise). However, the case with VEs, in which a partnership is valid for the relatively short period of a single project, this would be unacceptable.

The Windchill and open source systems are the evolution of product development management systems from document centric PLM systems to knowledge centric, intelligent systems of the future. However, it had been discovered during the course of the project that centralisation, by its very nature, is an inhibiting factor for one-off inter-enterprise collaboration as the only methods available for collaboration in such an environment are B2B schema mapping integration, centralised web-enabled system or use of an Application Service Provider. Portals in this instance would simply be

classified as B2B integrations as each data source has to be separately mapped into the portal.

The requirements set out in section 1.2 are fulfilled by the new methodology thus:

Requirement 1 is satisfied by the use of domain specific ontologies distributed to the clients (users) via the P2P network and open standards: Elimination of the interoperability issues for product and project knowledge, enabling small enterprises to implement application standards that were beyond their means, such as STEP-PDM, and easier set up for inter-enterprise collaboration. Requirement 2, is fulfilled through the use of RDF ontologies that can be queried by agent based systems and transformed by KIF based rules. The information contains the URI and meta-data eliminating the ambiguous context of the knowledge. Efficient query and retrieval mechanisms, intelligent agents to function on context aware information and provides an object oriented representation of an enterprise's data model.. The third requirement is met by de-centralisation: Elimination of centralised bottlenecks in bandwidth and resources, empowerment of collaborators within networks to “control” the knowledge they create, solving the management of intellectual property rights within VEs, elimination of centralised administration that results in reduction in cost and complexity and enable domain professionals to tailor the system. This, however, has a cost in terms of ease of maintenance and centralised control, as well as some query speed problems encountered due to the lack of a complete indexing service.

In addition the methodology described can be:

(i) Open source: Elimination of software licence costs, a solution to the problem of vendor lock-in in the long term, elimination of unnecessary complexity and freedom to modify the application. (ii) Platform and application independence: Enable the



enterprise to concentrate on its work and not be tied in to any vendor, the rapid and complete migration from the proposed system to future applications, empower the enterprise to leverage its existing investments.

## **6 Conclusion**

The methodology described is a suitable solution for collaborating enterprises especially SMEs, to create manage and reuse their knowledge, collaborate within VEs easily and without expense. In view of the author the proposed methodology can achieve the ubiquity required due to the following factors. It enables enterprises to implement STEP-APs. Easy to set up, modular, expandable and establishes rapid B2B connections. There is a lot of work in progress, and final conclusions can only be drawn once the knowledge and communication protocols are integrated more completely.

## **Acknowledgements**

The authors would like to thank the project sponsors without whom this research would not be possible. Firstly the UK Engineering and Physical Science Research Council (EPSRC) for its financial support, in addition to the industrial collaborators ArvinMeritor, LSC Group, Mabey and Johnson and PTC Corporation who provide training, software and valuable case-study material.

## **References**

Aziz, H; Gao, J; Maropoulos, P; Cheung, W, 2003, A design environment for product knowledge management and data exchange, Methods and Tools for Co-operative and Integrated Design, (ed) Serge Tichkiewitch and Daniel Brissaud (Kluwer Academic Publishers).

Beckett, R, 2003, Determining the anatomy of business systems for a virtual enterprise. *Computers in Industry* 51, 127-138.

Camarinha-Matos, L; Afsarmanesh, H, 2003, Elements of a base VE infrastructure, *Computers in Industry* 51, 139-163.

Cheung, W; Maropoulos, P; Gao, J; Aziz, H, 2003, Knowledge-enriched Product Data Management System to Support Aggregate Process Planning, 1<sup>st</sup> International Conference on Manufacturing Research, *Advances in Manufacturing Technology XVII*, Ed. Y Qin and N Juster, 253-258.

Gao, J; Aziz, H; Maropoulos, P; Cheung, W, 2003 Application of product data management technologies for enterprise integration, *International Journal of Computer Integrated Manufacturing* 16, 491-500.

Kim, Y; Kang, S-H; Lee, S-H; Yoo, S-B, 2001, A distributed, open, intelligent product data management system. *International Journal of Computer Integrated Manufacturing* 14, 224-235.

Wang, L; Shen, WM; Xie, H; Neelamkavil, J; Pardasani, A, 2002, Collaborative conceptual design - state of the art and future trends, *Computer-Aided Design*, 981-996.

Sharma, R; Gao J; Bowland, W, 2003, Implementation of STEP Application Protocol 224 in an automated manufacturing planning system, *Journal of Engineering Manufacture (Part B)*.

Vasara, P; Krebs, V; Peuhkuri, L; Eloranta, E, 2003, Arachne--adaptive network strategy in a business environment, *Computers in Industry* 50, 127-140.

Zha, X-F; Du, H, 2001, A PDES/STEP-based model and system for concurrent integrated design and assembly planning. *Computer-Aided Design*.