

# Product Based Interoperability – Approaches and Requirements

Håvard D. Jørgensen, Dag Karlsen, Frank Lillehagen

<sup>1</sup>Active Knowledge Modeling  
PO Box 376, N-1326 Lysaker, Norway  
h.jorgensen@akmodeling.com

**Abstract:** Product data, information and knowledge are the core ICT resources for collaborative design. This paper describes five different approaches to exchanging and sharing product data in collaborative engineering: 1) Document management, 2) Enterprise application integration, 3) Reference models and semantic web, 4) shared product data repository, and 5) Federated product knowledge architecture. State of the art in academic research and industrial practice is briefly assessed. The paper concludes by outlining challenges and directions towards realizing federate product knowledge architectures.

## 1 Introduction

Business analysts recognize that innovative design is the most important competitive factor for western manufacturing industries. In more and more industries, product platforms with dynamic modularization and configurable components are introduced to meet evolving diverse and conflicting customer, technology and business requirements. Conventional IT applications are built to support routine information processing, rather than creative design work. Analysts cry out that “IT doesn’t matter”, because IT does not extend the core capabilities of the business. In this context, product based interoperability should remove barriers to

- Interdisciplinary *knowledge* sharing and cooperative problem solving,
- *Information* logistics and quality management,
- *Data* exchange between companies and targeted applications for different disciplines.

This paper outlines five different approaches to product based interoperability, addressing challenges on the data, information, and knowledge layers. A state of the art analysis indicate that current product lifecycle management (PLM) systems tend to favor one business perspective at the expense of the others, leading to multiple “islands of automation”, as each function or discipline selects the tools that fit best their needs.

## 2 Product Interoperability Profiles

The ATHENA project [ATHENA] has developed a business-oriented approach towards the implementation of interoperability solutions. This work identifies typical, recurring business interoperability issues, and systemizes them into a set of Business Interoperability Profiles (BIP) [GLLW07]. We here outline five different BIPs from a product-oriented perspective, complementing the previously published process-oriented profiles [GLLW07]. The product oriented interoperability profiles are differentiated by how data, information and knowledge are shared and exchanged between partners, internally in a company, or cross-organizationally.

### 2.1 Product Document Exchange and Management

The most basic level of product based interoperability relies completely on manual interpretation and processing of the product information. It is captured in documents, including drawings and spreadsheets, and exchanged in a rather ad-hoc manner. Figure 1 below illustrates this profile. There are two variants of the profile

- Document exchange, using e.g. email to send documents, or
- Document sharing, using a repository that multiple partners have access to.



Figure 1. Product document exchange and management.

This interoperability scenario is dominant among SMEs. For larger corporations, it is also widely used, though most often in combination with more sophisticated automated information systems. Situations with a rapid pace of change or a high degree of uncertainty typically must be handled manually. Examples include early lifecycle phases, e.g. product concept development and target setting, and high level strategic decision making, e.g. in product portfolio management.

### 2.2 Product Data Exchange through Mapping

In this profile product data is stored in a structured format in different application systems. Most applications offer generic or specific import/export mechanisms, towards other major players in the same or related application markets, and towards standard formats such as XML, or higher level standards such as STEP EXPRESS etc. Application programming interfaces (API) such as web services are also commonplace. Through Enterprise Application Integration (EAI) these interfaces are typically mapped directly, linking data elements in one tool to corresponding elements in another tool, as illustrated in Figure 2. The figure shows that each partner or application has its own

internal database, with its own execution rules implemented in the application (as indicated by the lightning symbol).

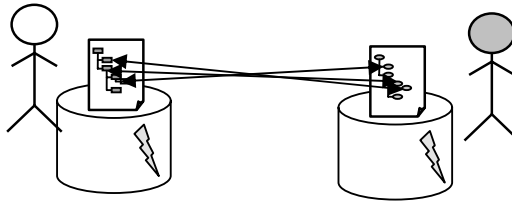


Figure 2. Product data exchange through data mapping.

In this profile, you typically end up making a new integration solution for each pair of applications or companies involved. Exports to word processors, spreadsheets, or graphics formats link data mapping to the product document profile of section 0.

### 2.3 Product Data Exchange Based on Reference Models and Semantic Mediation

The data level mapping approach outlined in section 0 is straightforward and direct. However, the large number of bi-directional links needed for each application pair, makes it less scalable when a large number of companies and engineering disciplines is involved in the supply network, like in the automotive or aerospace industries. The bi-directional links between data elements also become hard to maintain consistently when many different systems are involved in the life-cycle. Consequently, *global reference models* such as thesauri, taxonomies, ontologies, and product data dictionaries have been introduced. As shown in Figure 3, multiple local models, e.g. from different companies or applications, are linked to a central reference structure, and the links to this structure are used for automatically identifying the best semantic mapping between two concepts.

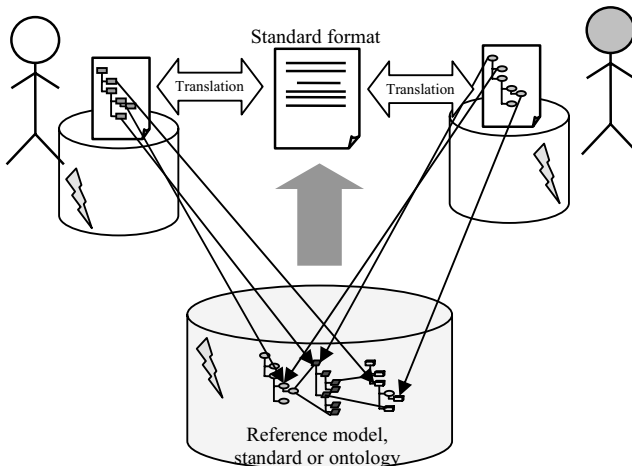


Figure 3. Product data exchange through semantic technologies.

As indicated in the figure, semantic web applications are commonly based on global reasoning according to formal rules defined as part of the reference model, e.g. for correctness verification and semantic reconciliation.

## 2.4 Shared Product Information Repository

For close collaboration, e.g. in collaborative concurrent engineering projects, the classification and generic part structure provided by an ontology or reference model is insufficient as a shared product architecture. Instead, the concrete product information being worked on must be available to all participants and stakeholders. This single global product data model, is found e.g. in Product Data Management (PDM) systems. As illustrated in Figure 4, the global data model now contains a lot of instance data, in addition to the categorization information included also in Figure 3 above.

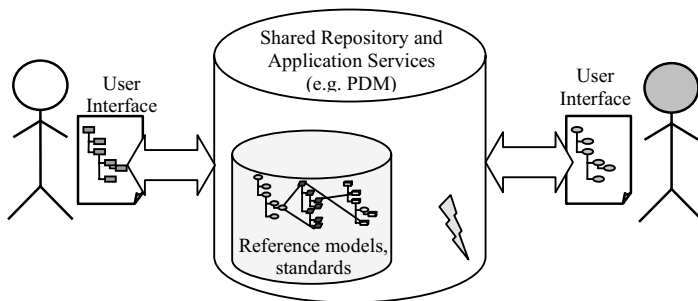


Figure 4. Product data management with a shared repository.

The figure also illustrates some important features of shared information repositories. Execution semantics is maintained at the global level, with a global rules engine, and typically global process management around the product information base.

## 2.5 Federated Product Knowledge Repository

The fifth and final product oriented business interoperability profile is illustrated in Figure 5. Like the two previous profiles, a shared product representation is the main coordination artifact. However, the architecture below is based on the recognition that different participants and stakeholders have different perspectives on the product information, and that different parts of the overall enterprise knowledge architecture will require different management approaches.

Each view or repository in Figure 5 may have its own metamodels, execution rules, task patterns, and other kinds of pragmatic design support functionality. The repositories may be logically and/or physically distributed, but the core element that makes this architecture a knowledge sharing architecture, is that the views are federated, so that each repository has some degree of local autonomy. The enterprise knowledge architecture (EKA) provides the means for negotiating between the views, and for coordinating changes and work across the repositories.

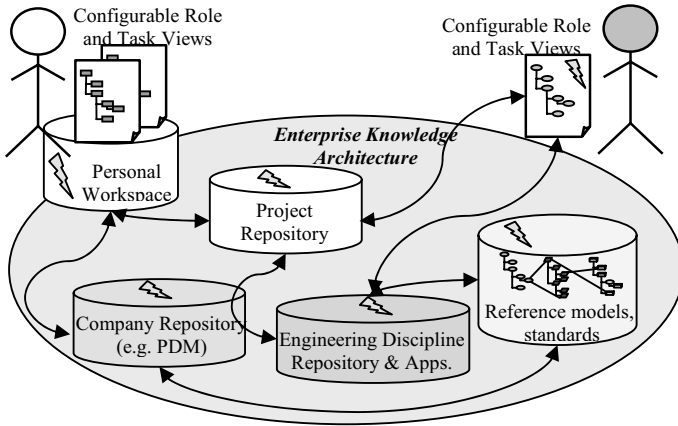


Figure 5. Product knowledge sharing with local views and pragmatic execution.

Local ownership to a view is crucial, because maintenance of single global models such as that of PDM systems Figure 4, quickly becomes too complex and bureaucratic. In PDM systems, autonomous local views typically are constructed ad hoc by someone taking data out of the system to perform some analysis, and then inputting the results (hopefully). The AKM (Active Knowledge Modeling) platform [JKL05] provides services for managing the federated knowledge architecture, including a metamodel and mechanisms for handling multiple views and customizable workplaces.

### 3 State of the Art and Requirements

This section gives a brief summary of the state of the art in product based infrastructures, highlighting the need for configurable solutions that can integrate and mediate between heterogeneous views on product data, held by different stakeholders and disciplines

#### 3.1 State of the Art

Information about products, their design and production, is the key source of knowledge for most companies. As indicated in Figure 6, this information reflects three main forces that shape product design:

- Voice of Customer (VoC), representing the needs and requirements of the market,
- Voice of Business (VoB), ensuring that the company is profitable, managing its resources and competences in the best way, following clear visions and strategies,
- Voice of Technology (VoT), representing the various disciplines who design and manufacture the products, technological constraints and opportunities.

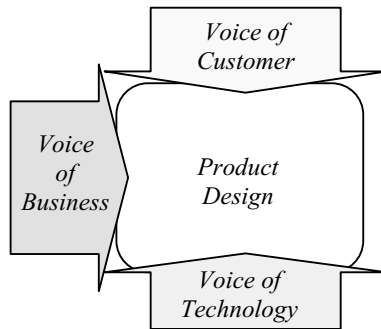


Figure 6. Forces shaping product design.

Different technologies have emerged to support each of these forces:

- Customer Relationship Management (CRM), business intelligence (BI), and requirements management systems aims to capture the voice of the customer, but lack services for product design engineers to capture life-cycle experiences, for manufacturing engineers to adapt to the design, and for life-cycle management engineers to influence both design and manufacturing.
- Enterprise Resource Planning (ERP) and Business Process Management (BPM) are applied to ensure that the voice of business managers are heard, that work is performed with maximum efficiency and according to established procedures, but these processes do not currently embrace any product engineering, customization or design activity. Conventional approaches to IT systems development enforce sequential peer-to-peer work processes and uni-directional information flows with poor support for collaboration and mutual learning and decision-making.
- Computer Aided Design (CAD), Manufacturing (CAM), and other engineering (CAE) tools are designed to represent general or specific technological domains, supporting the work of engineers from a specific discipline. No services to represent integrated product structures are currently provided, and most properties are poorly handled, forcing users to revert to parameter file versioning and sequential computation with personal backups to avoid overwriting of parameter values. Data management must thus be re-engineered to support concurrency.

System vendors from each of these applications areas have naturally extended their portfolio to target the central Product Lifecycle Management (PLM) and Product Data Management (PDM) challenge, as depicted in Figure 7. However, as they still fail to capture the core situated or work-generative knowledge, they fail to support the most critical knowledge harvesting and managing roles of creative work processes. To resolve this situation a new approach to holistic design must be introduced. This is the focus of the federated knowledge repository profile.

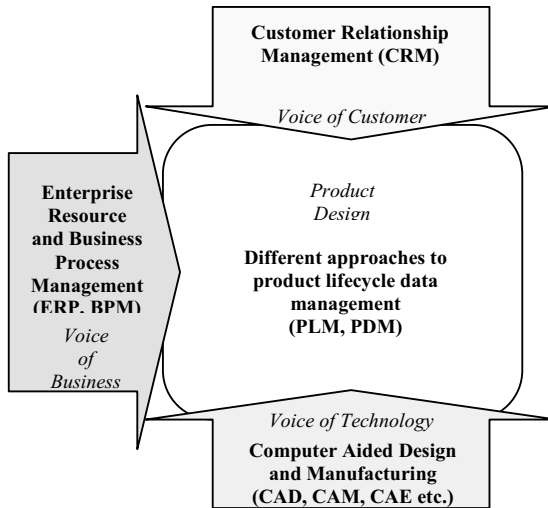


Figure 7. Inroads towards product lifecycle data management.

All the approaches in Figure 7 share one important shortcoming: They favor one perspective over the others. This is evident in

- The focus on business processes and document management in ERP-based PLM offerings,
- The focus on predefined structures and Cartesian geometry calculations and geometric data structures in CAD-based PDM.

Such perspective bias often leads to a lack of support for other voices and disciplines. If product information is mainly captured in documents, then the main design work occurs outside of the PLM system. The result is that the core information about knowledge discoveries and innovation is not structured so that it may be processed automatically or transformed into new views suitable for other perspectives. The knowledge architecture is predefined and the content is generic and static. Nor do the application services reflect work driven data and knowledge, and consequently they are much too general for designers to find adequate support for all their creative tasks.

CAD systems have problems in adequately representing robust geometry representations for life-cycle support, involving dimensions and tolerances across product assemblies. Even closely related domains such as static and dynamic mechanics calculations are poorly supported, and more remote disciplines such as chemistry, materials or electronics must be handled by disjoint tool sets, creating technical interoperability barriers between engineers.

The resulting product information infrastructure thus often consist of a large number of general, poorly integrated applications, ranging from established engineering tools to ad-hoc solutions in spreadsheets, document tables, drawings etc. Product design and lifecycle management becomes a poorly coordinated multi-disciplinary endeavor, and

inter-disciplinary collaborative engineering remains a distant vision. An integrating *product knowledge architecture*, configurable working environments, and effective role-oriented workplaces is lacking. To support design interoperability, we must go way beyond hub-and-spoke integration, towards supporting dynamic service-team roles and knowledge configured role-specific workplaces.

### 3.2 Product-Based Interoperability Approaches

Various research areas have developed singular approaches that embrace and extend established research strategies towards solving problems in product-based interoperability. Though many of these approaches show promising early results, they do not represent a holistic approach for capturing and nurturing enterprise knowledge:

- *Enterprise modeling* provides generic and user oriented means for capturing information about most aspects of the product lifecycle, but primarily from the business perspective. Customer input and market analysis is also commonly represented, but the messy technological details of product design, such as property embodiment and parameter handling, are typically outside the scope of enterprise modeling. However, with its focus on capturing multiple views of business knowledge, enterprise modeling is a promising starting point for a more powerful and configurable product interoperability infrastructure.
- *Cross-organizational business process management* deal with the automation of routine procedures, supporting some of the tasks in the product life-cycle, but not the most innovative and important ones, e.g. in the early phases of design. Product information is typically treated as black box artifacts and business documents being manipulated and exchanged during the business process. The content and structure of product information is outside of the BPM scope. To support design, a more configurable and user controlled process enactment approach is needed, working in concert with a business rule engine capable of capturing and managing product design rationale.
- *Ontology and semantic web research* is concerned with capturing and reasoning about product information. An ontology captures essential established facts about the product domains, and applies a global logic for reasoning about these facts, for transforming between different data representations etc. Ontology languages such as OWL and RDF are better equipped for representing e.g. product property structures, than conventional software engineering approaches such as UML. However, by demanding a formal, precise, and global representation, ontologies are not well equipped to capture local, heterogeneous product views from different disciplines, or unfolding, incomplete, and incoherent models reflecting the current state of product information during e.g. the early phases of design. Semantic approaches are designed to simplify automatic reasoning, but the critical problems of pragmatic information capture from users, inter-disciplinary sense-making and interpretation of product information, demands more interpretive flexibility and situated, user-controlled analysis and reasoning.
- *Service-oriented architectures (SOA)* aim to break up monolithic applications into reusable component services that can be put together in new ways to support emerging business needs. To be useful, this foundation does however require



business and user-level, product-oriented configuration and composition tools, and an integrating product knowledge architecture.

- *Model-driven architectures (MDA)* utilize modeling languages derived from object-oriented programming (e.g. UML and MOF) to build new software applications and to integrate existing applications. As a relatively young discipline, software engineering has not yet developed as sophisticated modeling approaches as other engineering disciplines. In particular, MDA has inadequate support for reflective models, instance modeling, multi-perspective, aspect-oriented, and multi-dimensional modeling. Some recent software engineering advances, e.g. Microsoft's approach to software factories, have started to learn from the experience of the manufacturing industries, advocating more configurable and domain-specific visual languages, but at the moment, mainstream MDA offers little support for product based interoperability.
- *Industry standards and reference models* exist in multitude. Typically, they are designed to support a concrete interoperability need, to bridge two particular application islands. The number of different combinations of disciplines, roles, applications, and processes in each industry sector, implies that the number of particular standardization needs is insurmountable. The standardization process, often ending up in consensus compromises that allow most competing approaches to coexist, further contributes to the ever-increasing complexity of industry standards. A simple, well-designed core product knowledge architecture, would be needed to ensure that a single family of standards, such as STEP or ebXML, does not become unmanageable. However, if there is such a core, it is generally based on ill-suited approaches such as MDA or semantic web. The result is unnecessarily complex and large standards that are too expensive for most companies to apply more than a small fraction of, which only support data exchange, and not interdisciplinary knowledge sharing and mediation.

Given the many shortcomings outlined above, product based interoperability may seem like an unsolvable problem. The main reason for this is the lack of a more holistic approach. We must find a way to relate data, information and knowledge created by computation to roles, and tasks performed by people.

As long as the core product knowledge is found in numerous specialized communities of practice, each with their own language and methods, interoperability barriers cannot be removed altogether. The best support we can offer is an approach that recognizes that design is multi-dimensional and driven by multiple role views. The situated, local, emergent, and pragmatic nature of product knowledge cannot be supported by off-the-shelf application systems. To enable knowledge workers to collaborate within the frame of product lifecycles to achieve sufficient degrees of knowledge sharing, interdisciplinary and inter-organizational learning and problem solving, a new approach to software engineering is needed.

### **3.3 Supporting Early Design Phases**

Throughout the product lifecycle, and in particular during the early design phase, the product information becomes increasingly well-structured and precise. Design decisions lead to the narrowing down of potential solution alternatives, and design elaboration

increases the amount of information which describes the future product. Later, engineering calculations and component testing helps specify the product properties and parameter values with greater accuracy and trustworthiness. It is only when interoperability problems in the handover between different disciplines or phases lead to information loss, that this incremental formalization process is temporarily interrupted.

The early phases of design are thus characterized by high degrees of uncertainty about most product aspects. This is also when the most important product design decisions are made. However, due to the lack of structured and reliable information, conventional IT tools poorly support early design. Instead, they are geared towards the later phases, when structured information is in abundance, and processes, methods, and data management can be fixed and coded into rigid application systems.

Consequently, while IT has enabled manufacturing industries to decrease the time to market by compressing downstream design activities, standardizing product family platforms etc., the early phases also consume an increasing fraction of the time and resources spent.

## **4 Conclusions and Further Work towards Product Knowledge Sharing**

The ad-hoc and knowledge-intensive nature of early design phases demand a product based interoperability approach, because the product information reflects the core content of the work to an extent that e.g. business process models reflecting standard bureaucratic procedures cannot. Moreover, such an approach should be

- Service-oriented and component-based, plugging in available IT tools in a need-driven manner,
- User-controlled, with semi-formal and interactive reasoning, because the key knowledge is the individuals' technical design skills,
- Collaborative, because most products are too large to be designed by a chief designer, and involve too many different engineering disciplines and other business roles,
- Business-oriented, because business resources, constraints, and requirements constitute the basic framework within which the design process takes place,
- Configurable in every aspect, allowing dynamic design languages, processes, systems and services,
- Configurable on every level, allowing, capturing, and learning from local deviations, exceptions, and innovations,
- Extensible, allowing new organizational roles, working practices, system services, etc. to be brought into the joint design arena when needed,
- Based on multiple views
  - Allowing each person to access the rich and complex product information structure through simplified role- and task-oriented workplaces,
  - Allowing heterogeneous and inconsistent views to coexist, enabling

- negotiation between perspectives and shared reality construction, not enforcing a global, shared model prematurely by only allowing consistent, already interoperable, views to be expressed,
  - Multi-dimensional, combining multiple type hierarchies, part structures, properties and parameter aspects, for different disciplines,
- Inter-organizational, because design increasingly requires core competence found outside of the company, among e.g. suppliers and consultants.

In summary, we must recognize the nature of innovative enterprise knowledge: The multi-disciplinary complexity of early design can only be effectively solved by developing what we call an Enterprise Knowledge Architecture. The approach is based on Active Knowledge Modeling (AKM), the foundation for an agile IT infrastructure. Through business centric modeling, end users and service providers can adapt, extend, compose, and configure services into complete solutions for innovative product design, knowledge management, and project management. The core of these models is a holistic reflection of the product structures being designed, developed, manufactured, maintained, improved etc. Other structures, such as the processes that constitute the product life cycle, the organizations and people involved, and the tools, applications and web services they use, are secondary. Product is the core dimension that reflects the content of the work, not just the supporting administrative structures

## References

- [JC99] Jørgensen, H. D. and Carlsen, S. (1999): Emergent Workflow: Integrated Planning and Performance of Process Instances, Workflow Management '99, Münster, Germany. University of Münster.
- [JKL05] Jørgensen, H.D., Karlsen, D., Lillehagen F. (2005): *Collaborative Modeling and Metamodeling with the Enterprise Knowledge Architecture*, Enterprise Modeling and Information Systems Architectures, An International Journal, Vol. 1, No. 1.
- [Jør01] Jørgensen, H. D. (2001): *Interaction as a Framework for Flexible Workflow Modelling*, ACM GROUP conference, Boulder, USA.
- [Jør04] Jørgensen H. D. (2004): *Interactive Process Models*. PhD thesis, Norwegian University of Science and Technology, Trondheim, Norway, <http://www.diva-portal.org/ntnu/theses/abstract.xsql?dbid=4&lang=en>.
- [GLLW07] Greiner, U., Legner, C., Lippe, S., Wende, K.: *Relating Business Interoperability Issues to Technical Interoperability Solutions*, Interoperability for Enterprise Software and Applications Conference, I-ESA 2007, Madeira.
- [ATHENA] Advanced Technologies for the interoperability of heterogeneous Enterprise Networks and Application, EU Integrated project 2004-2007, <http://www.athena-ip.org/>

