Chapter 16
Product Lifecycle Management

Lutz Lämmer and Mirko Theiss

Abstract Product lifecycle management (PLM) is widely understood as concept for the creation, storage, and retrieval of data, information and, ideally, knowledge throughout the lifecycle of a product from its conceptualization or inception to its disposal or recovery. PLM is seen in industry as one of the core concepts to fulfill a number of business requirements in the manufacturing industry with respect to completeness, high transparency, rapid accessibility, and high visibility of all product data during a product's lifecycle. Those requirements are related to financial aspects such as cost management and revenue growth; to the product itself like innovation, time to market, quality, and high productivity; and to regulatory aspects such as compliance and documentation. PLM is implemented by deploying IT systems such as product data management (PDM) systems and induces a high level of interoperability of related applications. With PLM, industrial companies attempt to gain advantages in shorter cycles, lower costs, better quality by avoiding errors, and misunderstanding. After reviewing basic concepts and building blocks of PLM, we provide empirical evidence of implementation scenarios and use case studies for different integrations to build up PLM solutions. We have evaluated applications in automotive, aerospace and consumer electronic industries focused on engineering design, change management, simulation data management integration and communication with partners. Emphasis is on the organizational and IT implications and the business benefit of the provided solutions.

Keywords Product lifecycle management · Product data management · Engineering release · Change management · Access management

L. Lämmer (✉) · M. Theiss
PROSTEP AG, Dolivostrasse 11, 64293 Darmstadt, Germany
e-mail: lutz.laemmer@prostep.com

M. Theiss
e-mail: mirko.theiss@prostep.com

J. Stjepandić et al. (eds.), *Concurrent Engineering in the 21st Century*,
DOI 10.1007/978-3-319-13776-6_16

16.1 Introduction

Product lifecycle management (PLM) is well established as a must in state of the art for large companies in manufacturing industries, especially in the world of automotive, aerospace, and increasingly of consumer electronics industry. This world is driven by the need for high transparency of all activities during the product creation process, clear, repeatable workflows, fast access to all product related data, distributed engineering and early supplier involvement. PLM as a concept focuses on the creation, storage, and retrieval of data, information and, ideally, knowledge throughout the lifecycle of a product from its conceptualization or inception to its disposal or recovery. Technological fundaments of PLM are product data management (PDM) systems which manage product data and knowledge during a product's lifecycle. Accompanied by powerful CAD tools, PDM builds comprehensive software systems which support all relevant roles in the product creation process, building an umbrella over all product-related activities [1].

Today's development process of attractive, complex products requires the contribution of large specialist networks (see Chap. 7). In this kind of collaboration, product data must be frequently exchanged between involved parties in digital form, with a high level of information security (see Chap. 18). Basically, PLM can also be understood as a concept for collaboration in the supply network and for managing product creation and lifecycle processes in today's networked world. Thus, PLM facilitates the acceleration of product creation process, by enabling communication on a regular basis between participants, preventing errors, and cutting the costs.

During serial production, engineering change management (ECM) is one of the key processes to be considered inside PLM. ECM is recognized as an issue that gains relatively little attention considering its importance. A particular issue is ECM within a supply chain. Suppliers are mostly loosely connected to their customers, and they are often not involved into an engineering change approval process. Many and especially late engineering changes induce high additional costs for any development project [2]. They consume one-third up to one half of the total engineering capacity and represent 20–50 % of total tool costs [3]. Thus, management of engineering change is a fundamental requirement for PLM.

After a short review on business requirements (Sect. 16.2) and related work (Sect. 16.3), we will describe the benefits of PLM (Sect. 16.4) in particular of the building blocks of PLM (Sect. 16.5). We will describe its processes and systems, its aspects, its software elements, its integrations, and its challenges. Those challenges are induced by the PLM concept due to its integrative nature. Consecutively, in Sect. 16.6, we describe empirical findings on different integration scenarios such as system integration, cross-domain integration, partner integration, and the special case of PDM system migration. Those findings are supported by the description of an integration tool in Sect. 16.7 and its application to a number of case studies in Sect. 16.8. This chapter is closed by conclusions and an outlook for further work (Sect. 16.9).

16.2 Business Requirements

PLM is seen as one of the core concepts to fulfill a number of business requirements in the manufacturing industry. Those requirements are related to financial aspects such as cost management and revenue growth; to the product itself like innovation, time-to-market, quality, and high productivity; and to regulatory aspects such as compliance and documentation. Industries demand a comprehensive concept which fulfills the following requirements [4]:

- The product is described completely and consistently in a global network composed of various systems.
- Results of real tests and experience from exploitation are part of the digital description.
- Each singular configurable product can be visualized and simulated in each version during the product lifecycle.
- The entire product creation process is accomplished in a network with suppliers and partners.
- The entire product creation process is conducted in a distributed, international environment.
- The complete information about product is available during the entire product lifecycle.

The user-friendly PDM system that realizes these requirements shall provide the following basic functionality:

- Fast, filterable access to all information which is relevant to the product creation process.
- Storage for all information which is generated by an application used in the product creation process.
- Forward of all information which is needed in the corresponding downstream processes.

Where financial aspects constitute a business case which needs to be identified for most PLM introduction projects that consume significant human effort and invested capital, PLM itself tends to increase cost transparency and therefore is an enabler of cost reduction applied to the product and its manufacturing process. Such cost reductions comprise improved communication with less effort, reliable sources of information, and standardization of processes. Spending more effort into consistent product documentation during the design phase has significant advantages in later lifecycle phases of the product such as manufacturing, change, and after sales.

PLM is implicitly a source of revenue growth which is achieved by accelerating product development and increased product variety under full cost control. PLM is seen as a preconditioning for flexible and agile development in mutually beneficial collaboration scenarios to create innovative and competitive products within shorter time, under full quality control and with a highly efficient production line. It is a competitive advantage to phase-in and phase-out partners in the supply chain and to establish working relationships fast and reliably. This is both true from a cost perspective by moving from single sourcing to competitive sourcing and from an innovation point of view by relying solely on the single source partner's capability to develop the new technology eventually or choosing partners capable of providing a sought for new technology.

As a single source of data PLM avoids erroneous manual data replication, supports traceability and manages dependencies. This opens the option for completely virtual and therefore faster and potentially cheaper product creation. Of course, this requires the virtual product creation methods to be stable, reliable and efficient. PLM integrates digital techniques and builds up a network of interconnected information. It connects data across system boundaries and is a precondition for accountability across the value chain. PLM has to become a reliable and constantly available source of information. Representing the lifecycle of a product and managing the lifecycle of its associated data is essential for cost efficient downstream processes in change management, after sales, and customer services.

PLM is a precondition for a seamless product documentation process to support all kinds of requests arising from regulatory definitions. PLM has to provide the data base for all kinds of documentation. Product documentation needs to be represented in different formats and with a well-defined status. This documentation must accompany product instances along their lifetime. This is essential for mission-critical products, like aircrafts and their engines, to keep track of the product instances themselves. Comparable requirements arise from regulations for providing spare parts or replacement instructions for the product lifetime and to provide documentation on proper end-of-life procedures.

16.3 Related Work

Eigner and Stelzer [1] define PLM solutions as "the functional and administrative backbone of IT solutions" which comprise a number of "IT systems and tools for CAD, CAM, CAE, simulation and visualization" and which were "derived in the 90s as an extension to PDM". Furthermore, they claim that PLM must be seen as a part of IT strategy to support the complete product development process and demands an integrated product data model, and technological and organizational preconditions in the enterprise.

Silcher [5] outlines differences in scoping the lifecycle by Germany-based researchers and others. Eigner [1], Schuh [6], and Vajna [7] focus, for example, on production phases but Stark [8] focuses on the product itself. He adds the integration of factory lifecycle management (FLM) and supply chain management (SCM) to the picture. The interface of the engineering and design phase with production is reviewed based on available empirical reports by Dekkers et al. [9]. He claims, for instance, the need for a more thorough understanding of the nature of collaboration through PLM between the involved disciplines and improved understanding of the networked structure of the lifecycle processes, especially the necessary feedback from production and maintenance to PLM instead of solely data consumption. In the following we will focus on the engineering, design and production planning phases of the product and the factory lifecycle and the supplier network management part of SCM.

Where frequent studies, by Abramovici [2] and follow-ups, underline the importance of PLM the actual level of PLM implementation varies. The key features of PLM systems are classified as [10]:

- backbone platform for all engineering data and engineering processes including process management and simulation tools,
- cross-domain and cross-enterprise management of systems, and
- management of virtual products and of representations of real instances during virtual try-out, for lifetime management or service purpose.

A PLM solution is based on a reference model which covers all lifecycle stages and all involved disciplines to represent a digital version of a product [11]. According to the industry wide accepted “Liebensteiner theses” (see [12]) PLM

- is a concept and not a system or a self-contained solution,
- is constituted of software modules like CAD, CAE, CAM, VR, PDM and other software tools supporting product development,
- provides interfaces to other application domains like ERP, SCM and CRM, and
- is supported by specialized service providers and software products to realize PLM concepts.

Representation of the product structure is central to PLM [13]. Consequently, PDM is seen as the core component of PLM. PLM, though, has a strong focus on products lifecycle processes. Corresponding authoring tools, workflow management capabilities and connected application areas are prerequisites of a successful realization of PLM throughout an enterprise.

16.4 Benefits of PLM

PLM becomes the challenging enabler for better, faster and innovative digitally based product development in increasingly complex enterprises. PLM manages product information from the earliest idea until the end of life of a manufactured product or a single item. Often seen as a concept to organize processes, PLM needs a strategy, dedicated organizations and well-selected software tools to allow collaborative product management along the product lifespan, across different domains to address all process requirements, and across different enterprises to allow for efficient and localized manufacturing. Business drivers for introducing advanced capabilities to manage product information are the demand for distributed development and manufacturing, service and maintenance in a globalized market both of customers and suppliers and more complex products in shorter time intervals. This is especially true for complex products like cars or airplanes but also for consumer goods which need locally and timely adaptations to regional demands.

PLM is not only an integration concept process-wise and system-wise, but mandatory for other methods and technologies such as requirements engineering, digital mock-up and variant management. By implementing a PLM solution a company is empowered to act internally as a virtual entity with an almost instant flow of information across the different departments but this methodology allows integration across enterprise borders and acting as an entity together with external partners in a so-called "Extended Enterprise" (see Chap. 7). Thus, PLM also enables supplier integration. Particular importance gets PLM in the case of a closer collaboration scheme or a company merger, when multiple internal lifecycle processes need to be integrated. With PLM the product relevant business processes gather a high level of transparency.

PLM requires the standardization of methods, interfaces and processes, and hence is a strong driver of standardization of the underlying information technology. Relevant standards in the context of PLM are hosted by international organizations like ISO, OMG, OASIS and a large number of national standard initiatives of the relevant industry associations like VDMA, VDA or VDI in Germany.

After discussing potential approaches to realize a company-wide PLM implementation we will focus on typical development challenges like choosing a suitable PLM architecture, selecting the right PLM tools, transforming processes and organizations, integrating PLM with legacy and supplier data to the expected advantages with respect to data availability, complexity management and increased flexibility in product creation and utilization of virtual techniques.

The case studies explain typical approaches of PLM in different industries like automotive, aerospace, and consumer goods to support data exchange and collaboration scenarios at different levels.

16.5 PLM Building Blocks

PLM comprises a number of concepts to manage all product related data from early idea until the end of life of the product.

16.5.1 Processes and Systems

From a process point of view PLM can be seen as a support for market analysis, product planning, product development, manufacturing, after-sales marketing, repair and services, and de-assembly or recycling. From a software point-of view it comprises originally disjoint software solutions like computer aided design, computer aided manufacturing, manufacturing engineering, project management, program management, team data management, product data management, material

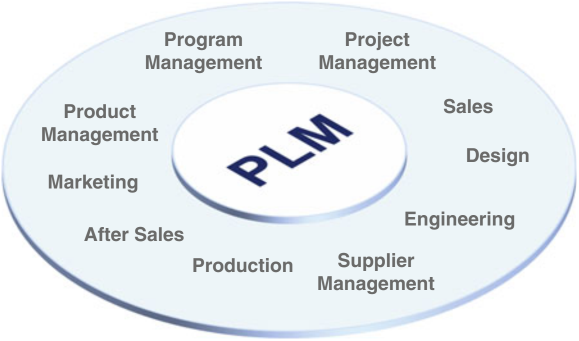


Fig. 16.1 PLM domains

resource planning and enterprise resource planning domains (see Fig. 16.1). With its interfaces PLM supports, for example, change management or reporting by a seamless data flow and acts as the data backbone of a digital enterprise. The PLM vision is to grant access, wherever necessary, to a single source of data with reliable state and to represent all stages of a product lifecycle in such a way, that the static description of the product and its components are sufficient to perform all necessary management tasks without replicating data input.

16.5.2 Aspects

All different concepts or methodologies are based on a shared data model which is build up from specific product related master data management, document management and status management aspects. The shared data model is furthermore used to build-up more complex aspects like bill of material for functional, assembly, management, or sales breakdown, and like configuration management, effectivity management, change management, project and process management.

Accompanied by specific searching and grouping concepts, potentially also geometry based, efficient browsing and data manipulating processes can be supported. Those aspects can be seen as PLM specific. They are accompanied by not specifically PLM related aspects like user management, transaction management, security management, and workflow management or file management, which are more common to general software systems.

16.5.3 Software Elements

From a software architectural point of view PLM systems are usually multi-tier and candidates for SOA enabled solutions. Based on a generic data model representation in a general purpose database system multiple layers of abstract data handling and manipulating routines are assembled to fulfill predefined business methods and orchestrated by embedded workflows. A communication layer allows the seamless integration with infrastructure components like a firewall or an enterprise service bus. The representation layer may consist of a thin web layer with less functionality or of more advanced solutions based on rich client technology. It is the software vendor's choice to implement general purpose PLM functionality or just industry related business logic into the client. The first solution tends to produce rich but complicated and not easy-to-learn interfaces while the latter may have a competitive edge in specific customer niches with a neat and dedicated approach. Customization or a proprietary software development may be the solution for a carefully chosen customer specific solution.

The end-user experience is build-up from tables, forms, hierarchical lists, spreadsheets and embedded visualization of status networks, schematic figures and geometry viewers or graphical editors.

16.5.4 CAD Integrations

PLM systems do not have CAD editing facilities themselves but provide the necessary support based on associated CAD tools. This functionality is usually better supported for tools of the same provider. The reason for this is the somewhat historically motivated moving boundary between CAD and PLM. Both systems handle to certain, but not clearly separable, extent versioning, structure and configuration. Quite a few PLM related attributes are easily manageable by CAD editors. Some attributes—like calculated weight—even have their source in CAD models. The biggest challenge arises from handling references of CAD data which need to be taken into account for building corresponding PLM relationships. Such relationships need to be synchronized throughout the editing process. This task becomes even more complex for relationships between multiple CAD documents representing assemblies or derived information like drawings.

The interactions between a CAD workspace during the ongoing design process and the necessary versioning or updating methods in the PLM data store require carefully tuned integration tools which have access to internal know-how and state of both CAD and PLM data. Some PLM vendors resolve this challenge by coupling the CAD workspace to the PLM data base directly. Others just use internal application programmer's interfaces to build this bridge.

16.5.5 Challenges in PLM

Most PLM products gained an increasing complexity over the last years. This is driven by the demand to provide users more comfort and a new usage experience—especially for products of the automotive, aerospace and consumer goods industries.

Additionally, new regulations and laws drive the need to monitor environmental compatibility from early conception of a product until the recycling phase.

PLM is a methodology to manage all data during the lifecycle of a product, but PLM is not a single application. PLM is always to be seen as a network of many applications supporting different special domains during the lifecycle of a product. In result, there is a strong need to exchange and share PLM data between different applications and domains as well as within partner networks and supply chains. Consequently, these requirements lead to a number of challenges in PLM:

- Integration with legacy systems (cross-technology and cross-system)
- Integration with other applications (cross-domain)
- Collaboration with external partners (cross-company).

16.5.6 Integration

A PLM solution depends heavily on data availability and quality. Primary sources of data are authoring systems or legacy data stored in databases, spreadsheets or archives. Data need to be identified, clustered, categorized, and cleaned-up according to a chosen terminology to be revised and maintained in a PLM system. IT tools to inspect, transform and load data via well-defined interfaces will help to prevent erroneous interactive replication of data. While it is sufficient to achieve interoperability of the utilized tools, data models of the tools need to be aligned and mapped. Even if there is no need to integrate all tools into a single solution, interdependency of data models leads to a necessarily contextual integration.

If data is stored in legacy systems, the preferred solution to integrate them into a single PLM solution would be to identify use cases, map them to PLM functionality and migrate the data in a single step after applying carefully all necessary measures to ensure data quality. Often it is not possible to eliminate existing processes based on legacy systems. In such cases an attempt to restrict legacy data to read-only usage may be an option. Especially processes modifying PLM data, like versioning or editing attributes, status or associated secondary content, need to be transferred to the PLM solution. If this is not feasible, uni-directional integration is not sufficient and needs to be bi-directional to allow for an update of the legacy data from the PLM solution. Organizational measures to make the old data source obsolete, for instance by not initiating new projects in the old database, will eventually lead to a significantly longer but definitely finite life of the legacy system. Well-defined interfaces to technology and systems help to build such integrations.

16.5.7 Collaboration

The traditional way to extract a subset of data in well-known office spread-sheets and send this information to a partner is not efficient anymore. The manual effort to prepare the data before sending and to use them after sending it is too high and, of course, dramatically error-prone. Furthermore, regulatory and contractual obligations require an instantly repeatable and secure exchange process. Cross-enterprise communication is bound to even more legally binding and reliable means of communication. Coupling processes supported in different software systems demand data transformation. Securing intellectual property needs careful filtering of data to be exchanged. Round-trip scenarios ask for correlation of input and output data flows.

Today's requirements for an efficient PLM-based collaboration are the demand for a close integration of data exchange methods into the PLM systems itself. Typically, a PLM system supports data exchange with other PLM systems of the same vendor, but this needs to be adjusted to the customization of the PLM solution. Data exchange with a PLM system of a different vendor is in all cases not supported directly. Such integration comprises the following steps which differ in complexity depending on the scale of integration (Fig. 16.2):

- Collecting PLM data to be sent to a partner in the PLM system.
- Exporting selected data automatically out of the PLM system and track the transfer of the data.
- Conversion of CAD data during export to a preselected format to ensure intellectual property protection.
- Package data and sent them to a selected partner.
- Re-importing data sent back from the partner into the originally sending PLM system.

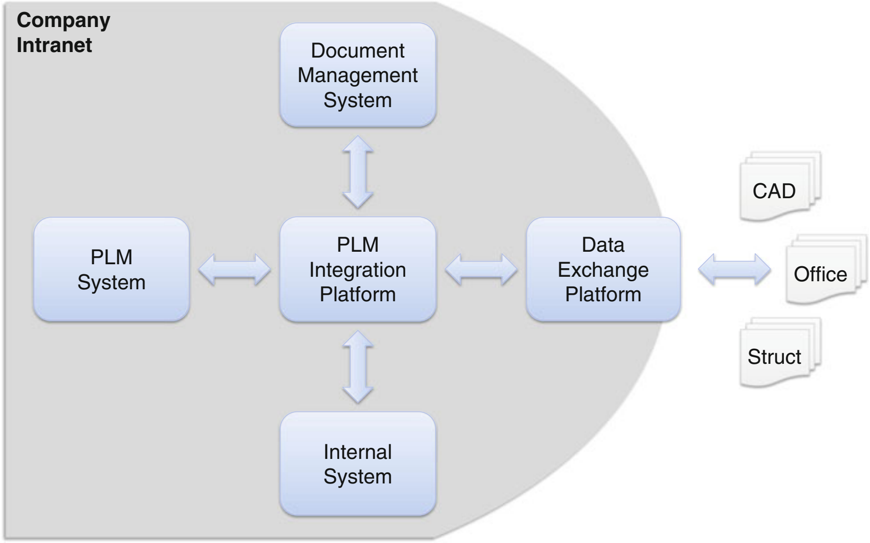


Fig. 16.2 PLM system integration and data exchange

16.6 Integration Scenarios

Already in 2005 Abramovici et al. [2] stated: “advanced PLM users will prefer integrated PLM/ERP solutions”. Such integrated solutions have been detailed by Mechlinski in [14] and we will refine the business drivers and features of the identified four levels of integration below (compare overview in Table 16.1):

- TDM solution
- Integrated approach
- Best-of-breed approach
- Point-to-point integration.

Implementing commercial-off-the-shelf (COTS) PLM software as a so-called team data management system appears to be very useful if some or optimally all of the following assumptions are true:

- A single department or a small number of departments with some employees is working with product data. The data authors and the data consumers share their perspective on the product data. There are no external consumers for product related data.
- All working processes are stable and shared between the process owners. The processes are homogeneous and consistent. They are well understood and common practice. They will not change in the future.
- Accompanying processes are not based on IT support. No significant value is gained from other IT tools supporting the product development process.
- Only one single CAD system is in use. It will not be replaced in the future.
- Legacy management tools like document management or classification schemes are well understood and are easy to migrate to the new data base.
- The product management solution resides in a single network domain. There is no need for a bridge to a remote location or to external partners.

COTS solutions for integrated CAD and PDM with the right CAD tool integration of the identical software vendor are optimal in such cases. The implementation can be optimized with respect to customization efforts if it is applied out-of-the-box (OOTB). This allows a seamless migration path if new software versions evolve and additional functionality is desired. Optimal business value is offered by special pre-packaged solutions of system vendors. A *TDM solution* will come with the basic support functionality to manage CAD document based workflows for sharing design artifacts within small work groups. All users have a quite similar set of functionality available. The graphical user interface is capable of simple check-in and check-out workflows and to manage comparisons of database content within report and list views and allows spreadsheet like actions for search, replace, modify and update.

Table 16.1 Integration scenarios and their corresponding assumptions

	TDM solution	Integrated approach	Best-of-breed approach	Point-to-point integration
Involved departments	Single department within same enterprise	Cross-department solution	Cross-department solution	Cross-enterprise solution
COTS	Feasible	Feasible	N/A	N/A
CAD system	Single CAD	Single CAD	Multiple CAD	Single CAD
PDM system vendor	Identical with CAD vendor	Identical with CAD vendor	Arbitrary	Arbitrary
Partner integration	Data exchange solution based on check-in-check-out only	Portal or direct access, additional data exchange solution	Portal or direct access, additional data exchange solution	Targeted at distributed scenario with shared data
Network	Local only	Local only	Distributed locations	Targeted at distributed locations
Customization effort	Small	Moderate	Challenging	N/A

Nevertheless, the criteria mentioned above will not apply to all situations. A number of PDM integration scenarios exist to overcome problems while some or potentially all of the criteria are not met. An example for such a solution is the *integrated approach* which addresses the following assumptions:

- Many departments collaborate with identical requirements on working with product data. Data authors and data consumers share their perspective on the product data. There are some external consumers for product related data.
- All working processes are stable and shared between the process owners. The processes are homogeneous and consistent. Interfaces are well defined and understood. There exists a common practice of revising processes and the related IT tools. There will be small changes only in the future.
- Accompanying processes are based on some IT support. Their requirements will be covered by the integrated solution itself, if not, then by some customization or by data exchange via an existing interface.
- Only one single CAD system is in use with no plans to replace it in the future.
- Legacy management tools like document management or classification schemes are well understood and are easy to migrate to the new data base.
- The product management solution resides in a single network domain. The need for a bridge to a remote location or to external partners is covered by the COTS PLM solution, preferably by an integrated portal solution.

A single highly integrated PLM system—with a unique CAD system by the same PLM vendor will meet these assumptions. It depends on the ability of the enterprise to adapt to the features of the PLM system how much customization and legacy implementation, especially for interfacing with legacy, is required. More customization is associated to constantly high efforts for implementing, maintaining, testing and additionally revising the software maintenance efforts spent by the software vendor and paid for by every customer. The major advantage of such a solution is the single system appearance, which allows a high degree of standard management and training procedures out of the box as long as the customization of the software is moderate. Software updates are feasible but need special considerations for all interfaces and potentially locked-in customizations.

The PLM system needs to be highly flexible to accommodate all requirements, at least to an acceptable extent, especially for the following building blocks:

- The data model needs to be flexible and extendable. This includes, but is not restricted to, additional attributes for standard elements, new lists of values to control the population of attributes, inheritance mechanisms to derive specialized custom element types from standard elements, which need to be recognized in workflows, status and access control.
- The workflow needs to be highly flexible with respect to triggered data manipulation actions, state control and other automated processes.
- The graphical user interface must allow role-centric subsets according to specialized user profiles, like CAD engineer, PLM data manager, controller, project manager. A one-size-fits-all approach tends to be too heavyweight for the majority of the users and will reduce the number of potential or satisfied users.

Nevertheless, depending on a single vendor and his single vision on PLM, or on a monolithic software infrastructure, may have its disadvantage. If cutting edge PLM functionality and individual processes becomes a competitive asset the *best-of-breed approach* may be more appropriate. This approach addresses the following requirements:

- A large number of departments collaborate with differing requirements on working with product data. The data authors and the data consumers have different views on the product data. There are a large number of external consumers for product related data, for example in after sales and maintenance.
- The working processes are mature but not shared between the process owners. Interfaces exist, but they are not established at all levels.
- There exists a common practice of revising processes and the related IT tools. There will be constant changes in the future.
- Support processes are heavily based on IT tools. Their requirements are not covered by the PLM solution itself, but by purpose made, legacy applications which exchange data via an existing interface.
- Multiple CAD systems are in use or a change will take place in the future.
- Legacy management tools like change management, document management or classification or configuration schemes need to be supported as they are. There are no plans to integrate them into the PLM solution.

- The product management solution resides in multiple network domains to accommodate multiple sites around the world. The need for a bridge to a remote location or to external partners is covered by a specialized portal solution.

The best-of-breed approach supports individual processes optimally. Custom made solutions fit to the specific requirements of the users. Specialized CAD- and CAE-interfaces allow seamless access to required databases for the engineers and designers. Nevertheless, all solutions need to share common design models, design product structures and documents, standard parts libraries, configuration, and effectivity control. This asks for a company data model with deeply integrated interface solutions. Setting up the company-wide data model is an additional challenge. Those interfaces cause system dependencies which need to be managed by a common versioning process.

To avoid strongly coupled solutions a service-oriented architecture approach based on a neutral communication model and implemented with an appropriate middleware technology like web services is advised. This approach is called loosely coupled integration and offers a high degree of flexibility on utilizing the best-of-breed components to build the enterprise PLM solution. But even if such a solution may be called loosely coupled from an implementation point of view it requires the management of strongly related data models. Even if the underlying technical interface solution may survive a data model change, related processes need to be adapted. Additional attributes arising within one system and exchanged by loosely coupled interfaces need to be represented and understood by processes in the other system.

Collaborative and cross-enterprise PLM processes, where a centrally managed infrastructure is not available or not desirable, are characterized by the following aspects:

- A small number of participants collaborate in ad hoc manner without forming a permanent partner network.
- The isolated working processes are mature but not shared between the process owners. Interfaces exist, and they are established at all necessary levels.
- No changes through the lifetime of the collaboration are allowed.
- A single CAD system is in use at least for sharing.
- Legacy management tools like change management, document management or classification or configuration schemes need to be supported as they are. Every partner uses them on their own.
- No shared product management solution resides anywhere, elementary versioning and document tagging is in effect.

This approach is named *point-to-point integration* and is currently subject of research and investigation to support collaborative product development [15]. Without a central PLM solution and consequently without a single point of failure this solution is very flexible and robust. It is very easy to enter or to leave the design team. This approach suits the ad hoc nature of small-scale collaboration. This approach is accompanied by tagging technologies or utilizing ontologies as a common vocabulary to perform inter-project communication and to link digital product representation artifacts. Realization on top of peer-to-peer technology is an option but currently not supported by industry software applications.

16.6.1 PDM System Integration

16.6.1.1 Synchronous Integration

A synchronous PDM integration updates referenced PDM data more or less immediately between source and target systems. Whereby the actual update operation may consume significant computing resources it is expected to be very fast. In reality, synchronous update must be performed within several seconds. From an execution point of view a synchronous integration may cause the initiator of the update operation to stall until the operation has succeeded. But this depends on the implementation techniques used.

Common to all implementation approaches is the definition of a so-called trigger initiating the necessary update operation at the occurrence of a specified event. Current PLM systems provide extension points to define such triggers for instance in the database operation, within workflow or other business process operations, or as callback in the graphical user interface.

Whereby immediate update may be a requirement of the business for continuous flow of information, it may cause a large number of small changes to be executed. Within network based integrations this is not the optimum. Problems arise from the following scenarios:

- a significant number of changes is caused by automated processes like propagating date changes in effectivity control,
- interactive maintenance of PLM data may actually flip attributes,
- usage of wizards may cause interactive data cleaning,
- data maintenance may trigger other data updates which are handled better in combination, or
- volume of a single data set may be too large for a synchronous update to be performed without breaking existing timeout limits for blocking synchronous operations.

Transferring changes as fast as possible induces unnecessary load to mirror all changes in coupled systems. If history is not of concern, transferring state at defined times or after performing a well understood chain of interactive changes may reduce the data management effort. Furthermore, combining a large number of changes into a compound update data set may improve performance by reducing consumption of computing resources like network bandwidth and database updates. This is achieved by switching to asynchronous integration.

16.6.1.2 Asynchronous Integration

An asynchronous PDM integration updates referenced PDM data under the control of a third instance like an external timer or an external event signaling data availability. The actual update operation is controlled by the receiving system. Data to be processed by the update operation is already extracted from the source system and does not consume source system or network transport resources. The performance of the update in the target systems depends on the performance of the messaging and the receiving system only. Asynchronous integration decouples the system resources of source and target. This approach allows the aggregation of consecutive changes into a single change to the final stage if history is no concern. Keeping the order of changes the aggregated update can be performed much more efficiently by grouping update operations into a single target system operation.

For updates of large data sets like CAD data, asynchronous integration may be the only solution. If data sets comprise structure data that need to be efficiently processed, asynchronous integration may be performed practically online and the end user experience will not be different from synchronous integration but with better utilization of computing resources.

16.6.2 Cross-Domain Integration

The cross-domain integration addresses the challenges connecting different engineering disciplines like electrical, mechanical and software engineering. Whereby the disciplines do not differ in their notion of the general development process, the cycle schedule, the deliveries and their impact control, and the frequency of changes differ significantly. An integrated solution with a single system approach will fail in this situation. The integration needs to take into account.

- Isolated synchronization points only when a release or change exists during the product development and even during the whole lifespan of the product.
- Integrity of the solution is guaranteed by fulfilling interface contracts.
- Traceability (why a decision was taken) across domains is necessary to control the development.

The challenge of cross-domain integration is the replacement of the aspect of common data storage in a single database by the notion of information association across system and domain boundaries. The association between related information becomes additional external information. It is not sufficient to build up this relationship once and for ever. Whereby read access seems straightforward it is required to manage this association as an additional constraint, which needs to be taken into account for data access operations like update and delete. This additional reference information is used to realize the further use cases like where-used search, impact analysis, release management, which are illustrated below with their inherently increasing complexity. These functions make it easier to master the product creation process and avoid errors. This results in shorter development times and satisfies legal and customer requirements regarding traceability.

16.6.2.1 Where-Used Search

The where-used search identifies references to an asset in question by searching for all occurrences of a particular reference to that asset. The search may be narrowed to a subset of source item types. Narrowing the search to a particular reference depth may speed up the operation but will not fit in general to the use case which asks for instance like: "Given an item *X* in product *A* identify all other products using this item *X*." This search is expensive in a single system context but it becomes extremely costly when executed across domains using different software systems. In this case is not sufficient to maintain cross-domain references in an additional data base but it is necessary to build up corresponding reference indices in the related software systems to support this use case efficiently. Depending on the chosen solution architecture the search may involve traversals in up to three databases, e.g. source, target and reference database, to succeed. Optimizing this search requires adaptions in the systems involved.

16.6.2.2 Impact Analysis

An impact analysis tries to find answers to questions like: "What impact will the changes needed to satisfy a requirement involve and which materials or software modules will be affected?" Other examples are: "Is there a test case for each requirement?", "Is there a requirement for each material?" or "Is there a requirement and a test for each specified software function?" These questions are part of an audit support use case. Such use cases call-out for a traceability report which selects objects with an indication of all their dependencies and contexts of potentially different concepts which are not stored in the same database in general. These use cases are part of general methodologies like SPICE and CMMI and require the cross-domain references.

In contrast to the where-used use cases, it is not sufficient to traverse a product structure and find referenced artifacts of the same quality, e.g. item, product, but it is necessary to built-up and maintain associations between otherwise unrelated entities like parts, solutions, products, CAD files and requirements or software artifacts. This approach is easily applicable to the relevant consistence checks as mentioned above. The associations need to be volatile but revision-proof and may be built-up by classification attributes or semantic means.

16.6.2.3 Release Management

Mastering impact analysis is the precondition for automated maturity check on all modules and from all domains with the linking of the release level, requirements baselines, test results and much more. Whereby impact analysis may involve human interaction to select affected entities, it is desirable to select all necessary entities for release repeatedly, secure and automatically. Release management systems may maintain entire sets of affected entities to get hold of all induced changes. Whereby capturing this information in a single system is a well understood feature of PLM systems in conjunction with status management and workflow, integration with an external release management system involves the management of references across domains. The static part of this reference management is usually implemented with an external link database. Consistency and integrity, as well as workflow are challenges which need to be addressed by an integration solution.

16.6.2.4 Example: Integration of PDM with SDM/CAE

A special cross-domain integration scenario is to bridge the gap between design and simulation. Whereby some of the market leading PLM systems come with their simulation support inbuilt, like Siemens PLM Teamcenter and Dassault Systèmes V6, a number of proprietary simulation data management (SDM) solutions need external cross domain integration. Such SDM solutions like MSC SimManager, Altair Data Manager or ANSYS Engineering Knowledge Manager have similar functionalities like PDM systems to manage product information. A number of challenges [16] arise from the following requirements:

- Simulation data needs to be referable
  Simulation data are inherently redundantly used to evaluate alternatives and needs to be provided for the different simulation tools in their proprietary data and file format representation.
- Simulation data need to be organized within a product context
  Analysis is not restricted to a single part but always carried out under certain conditions for a purpose. PLM provides this context.
- Relationships with other domains need to be kept
  This challenge relates to the previous one because iterative changes in the design process in one domain cause respective changes in the other dependent domain which need to be evaluated and negotiated with respect to the validity of the assumptions.
- Simulation data needs to be managed during the product lifecycle
  Development in the interrelated domains is carried out concurrently. Synchronization occurs at predefined milestones or maturity gates. The corresponding lifecycle-state information needs to be managed.

With an integration platform it is possible to create an automated end-to-end process between the PLM world and such simulation data management systems

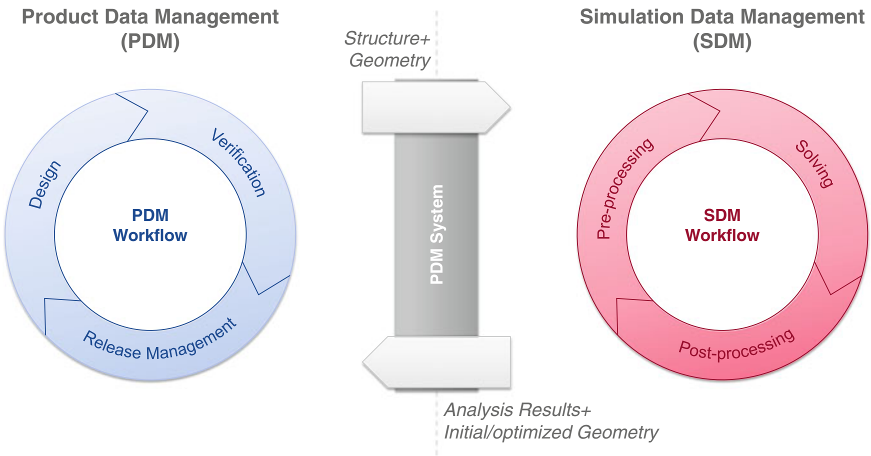


Fig. 16.3 General workflow of PDM-SDM integration

(Fig. 16.3). This process keeps track of changes on both sides, synchronizes shared data models and avoids additional manual work by making sure that all changes to relevant geometry, structure and connectivity are transferred between PDM and SDM systems. Furthermore, incremental update may reduce the volume of data so that it is perfectly suited for the simulation project. This requires filter mechanisms to configure the structure such that it is relevant to the SDM.

The integration platform supports the robust, fast and flexible integration of simulation and PLM processes by the following generic data transfer use cases from the PDM system to the SDM system:

- initial transfer of parts and assemblies including attributes and corresponding files (CAD and others),
- update of previously transferred data, and
- check for update in the SDM system, if changes to PDM data exist.
  This use cases supports the push communication model.

The following generic data transfer use cases are supported in the reverse direction from the SDM system to the PDM system:

- transfer of the data changed in the SDM system, e.g., CAD files or analysis results, back into the PDM system, and
- manually controlled data transfer from the SDM system to the PDM system.

Interactive filter use cases are necessary to reduce the transfer volume of data for a certain simulation project to just the parts required for that particular project. The PLM-related classification, configuration and status control mechanisms with their corresponding selection functionality are necessary in both domains. If they are not shared, a powerful mapping mechanism is essential.

16.6.3 Partner Integration

Especially for OEM another key question is: “How does collaboration with development and manufacturing partners fit into the PLM picture?” The level of integration into the PLM solution increases depending on the level of integration of the development partner from a part supplier up to a tier-1/2 supplier, which contributes with systems or to be completed products.

The partner integration can be operated in both a synchronous and an asynchronous way. The synchronous approach requires online access to data sources and systems and usually involves manual handling of data. This may be achieved by remote access to the PLM solution. This incurs security and infrastructure measurements to guarantee a safe and reliable mode of operations. A portal with associated data storage providing online access to data packages may be a cheaper option for online access.

A more automated, and therefore more reliable and repeatable mode of operation is offered by an asynchronous approach utilizing a data exchange tool managing data packages. Depending on the functionality this exchange tool may comprise basic data transfer functionality like FTP and ENGDAT, or more advanced features for data package handling, temporary storage and provisioning in a portal, or full-fledged PDM functionality to keep track of the exchange project separately.

A data exchange portal is well suited for distributing product data from the OEM to the development partner and re-integrating their deliverables. It is not always necessary, to distribute originally authored data. Simplified data, which do avoid an unnecessarily high data volume and ensure intellectual property protection of parameterized CAD data with full development history will be sufficient in most cases (see Sect. 18.6).

16.6.4 PDM System Migration

Introducing a PDM solution from scratch is the absolute exception. After more than 20 years of IT support some areas of PDM are already implemented with the help of IT technology. It does not matter whether the solution is based on a product solution or the solution is a legacy-based system. The data in this installation are of value for the enterprise and need to be migrated to the newly-chosen PDM solution. This requires a PDM system migration. Although migration can be seen as a singular activity synchronized with the PDM introduction process it often yields a situation in which migration forces a specific integration scenario.

In general, a system migration project follows the usual system introduction approach but source and target systems are usually known in advance. Nevertheless, several challenges exist. Those challenges arise from a process and from a source and a target system point of view at minimum.

16.6.4.1 Process Challenges

Introducing a new PDM system needs a business justification and usually addresses several shortages of the old solution. Improving the process support by a new IT solution requires adjustments to the original process at a minimum to leverage the PDM functionality to the actual processes. PDM system vendors try to minimize the impact of introducing their solutions by introducing a mature set of functionality. It is difficult to differentiate the market leading products by their functionality. Furthermore, PDM systems may be customized according to customer needs. Customization means implementing additional functionality in the PDM system. This may actually leverage the gap between the process in focus and the newly introduced solution. Unfortunately, this involves increased development costs, time, and resources at minimum, but lack of portability to new product versions at worst. The most difficult-to-handle problem is the lack of understanding how the new function would look like in the newly designed process. The potential end-user acceptance depends on the match between expectations on how the process will be supported by the new PDM solution while at the same time the process itself undergoes a transformation process. Therefore, it is necessary to implement a very sharp and visionary requirements management process which leads to a workable PDM solution and the same time leads the PDM customization process to reduce the negative impacts on portability and flexibility. PDM system vendors are usually not of much help in this situation. Their business driver is the product placement. Independent consultants tend to increase the customization efforts. It is the duty of the process owner himself to develop a workable vision of the processes in the new system.

From a process point of view this situation introduces two key questions to the migration: “Which data from the old solution will be required to drive the new process based on the new PDM solution?” And: “How do we need to restructure the existing data to accommodate the requirements of the newly introduced and customized product?” It will be clear from the statements above, that the answer to these questions controls the efforts needed in the system migration.

16.6.4.2 Source System Challenges

The source system serves two purposes during the migration: It is the source for feeding existing, well-known application data into the new system and it supports downstream processes, which consume the product data stored so far in the old system. Data representation in the source system must be fully understood and analyzed before the migration begins. This analysis is based on process and system know-how and needs to take into account implicit knowledge on history, legacy, and quality. The amount of data must be estimated. It might be an indicator for the migration costs, but is definitely a denominator for the business requirements of the migration schedule. Data clean-up and filtering before the actual transfer are necessary. It is in general not a good idea to propagate workarounds with respect to data quality or data consistency to the newly introduced system. Preparing the source system for migration is a perfect point in time to introduce a consistent terminology, a streamlined classification system, reduce ambiguities, and resolve historically motivated duplications or redundancies. The target system data representation capabilities will have an impact on data cleansing and on required data quality. This applies not only to product data but to CAD data in particular.

Even more complex is the management of downstream processes. If they are solely based on the original source system they need to be part of the PDM introductions process and they will undergo the same process transformation steps as described above. If downstream processes maintain their own data management and consume some of their data from the source system in question it is worth to justify the existence of this special solution. It might be an additional business driver for the PDM introduction process to incorporate this downstream process into the new target system to reduce the number of interfaces. At a minimum, downstream processes need read-only support until all necessary data are available in the newly introduced target system. This approach easily applies to read-only processes. It becomes more difficult, if downstream processes are not only based on but update the data source. Such processes need to be migrated in parallel to the PDM system migration. Their interfaces will evolve from existing interfaces to the source system and migrate to interfaces to the target system. These interfaces need to be developed in parallel. The business process will break and cause additional efforts, if they are not in place and functional by the first time of migration. Requirements for backward migration will be the consequence. This might cause the extension from a unidirectional migration scenario to a bidirectional integration scenario. The migration approach needs to be flexible to this requirement. This is especially the case, if the migration is part of a transformation process and the impact of all business drivers is not known in advance.

16.6.4.3 Target System Challenges

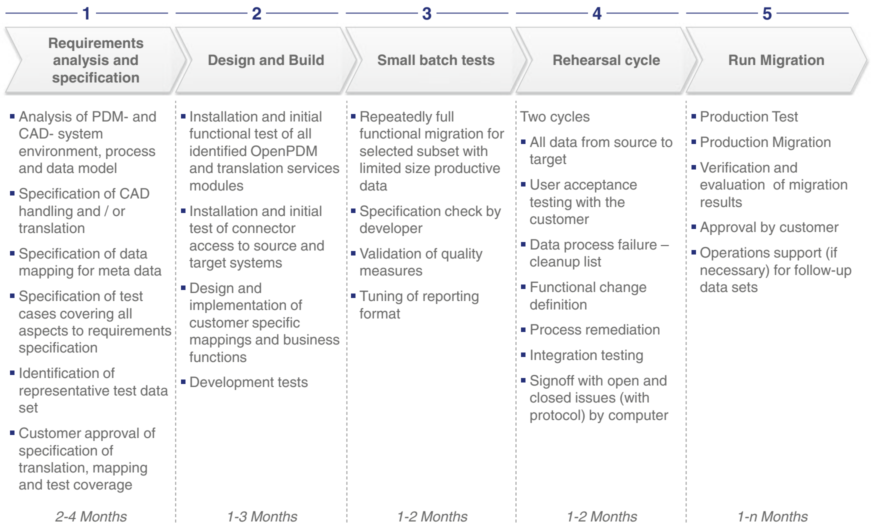
Ideally, for a successful migration the target system is up and running stable for a while. Access control is established. The end users are trained in the new system and they are aware of the new features of the new data model and the processes implemented within the target system. This approach is the perfect setup for a migration project to begin with. The target is well-understood and ready to use.

In practice, this will be the case for a small fraction of the migration projects. Usually, the target system is under development, the data structures are still changing and even more important, customer requirements are changing and customer expectations will develop as soon as customer data are visible in the context of the target system. The approach becomes more complicated as soon as customer acceptance depends on availability of live customer data in the target system. Migration needs to be prepared and performed at least partially in parallel to the overall PDM introduction process. This causes high pressure on the flexibility and speed of the migration approach.

All know-how aspects of the data model which apply to the source system as described above apply to the target system as well with one exception—history and implicit know-how are reduced to a minimum. It is a challenge to define future-proof stable mappings between source and target data models. We find no difference with respect to complexity, completeness, consistency and integrity between mappings for integration and migration. Of course, a clear understanding and expectation on data quality and data cleaning will help to reduce the effort. Migration needs to maintain data quality.

16.6.4.4 Migration Project Approach

The migration project is divided into the well-known phases of software development: requirements analysis and specification, design and build, acceptance tests and productive go-live (Fig. 16.4). The acceptance test phase differs from the common software development cycle. It splits into a phase of small batch tests which proofs the implementation to be compliant with specification and requirements and a test phase, which may be performed repeatedly or which may iterate over improvements of the specification, implementation and test phases. The tests try to identify the impact of the implemented migration solution on the complete source data set considered for migration. This approach requires a full dump of the source data to a separate stage solely dedicated to the test phase. This step proofs the assumption of the requirements phase to be correct and helps to identify anomalies or unexpected data in the source database. Thus, all data considered for productive migration need to be taken into account. The user acceptance test is



1	2	3	4	5
Requirements analysis and specification	Design and Build	Small batch tests	Rehearsal cycle	Run Migration
▪ Analysis of PDM- and CAD- system environment, process and data model ▪ Specification of CAD handling and / or translation ▪ Specification of data mapping for meta data ▪ Specification of test cases covering all aspects to requirements specification ▪ Identification of representative test data set ▪ Customer approval of specification of translation, mapping and test coverage	▪ Installation and initial functional test of all identified OpenPDM and translation services modules ▪ Installation and initial test of connector access to source and target systems ▪ Design and implementation of customer specific mappings and business functions ▪ Development tests	▪ Repeatedly full functional migration for selected subset with limited size productive data ▪ Specification check by developer ▪ Validation of quality measures ▪ Tuning of reporting format	Two cycles ▪ All data from source to target ▪ User acceptance testing with the customer ▪ Data process failure – cleanup list ▪ Functional change definition ▪ Process remediation ▪ Integration testing ▪ Signoff with open and closed issues (with protocol) by computer	▪ Production Test ▪ Production Migration ▪ Verification and evaluation of migration results ▪ Approval by customer ▪ Operations support (if necessary) for follow-up data sets
2-4 Months	*1-3 Months*	*1-2 Months*	*1-2 Months*	*1-n Months*

Fig. 16.4 Migration project plan

performed on the complete data set in advance and not just on a well-chosen subset. The data processing failures are logged in a cleanup list, which will be evaluated from a business perspective. Not all defects necessarily need to be fixed. Depending on the business value of the identified problems user acceptance may be achieved simply by ignoring a particular error condition, fixing the problem in the source data or excluding data from migration. The expected outcome of this phase is that there is no doubt on the result of the migration. Risks are reduced to a minimum and are isolated to runtime problems like system and resource availability.

Furthermore, the test phase gives an indication of the results to be expected from a performance and a reporting point of view. Both aspects need to be tuned to the customer's expectations before the productive migration takes place. Especially the performance needs to be managed because it drives the scheduling of the migration process and proves an optional splitting approach to be valid or not.

Depending on the nature of the migration, the go-live may be a singular event, which is a very rare case, or it may take several steps due to the chosen migration strategy. To reduce the risk associated with the migration or to split the overall effort in manageable sub tasks a step-by-step approach is more appropriate. The steps shall be derived from business drivers like products, programs, or subdivision. It needs to consider basic preparatory work, common data like standard parts or libraries, and will handle structural data separately from mass CAD data.

Depending on the requirements of the processes based on the source system and the chosen splitting approach a partial re-synchronization may be necessary (Fig. 16.5). This involves an additional data flow back from target to source. The data flow back asks for an automated solution which tends to increase the complexity of the uni-directional migration to a bi-directional integration scenario.

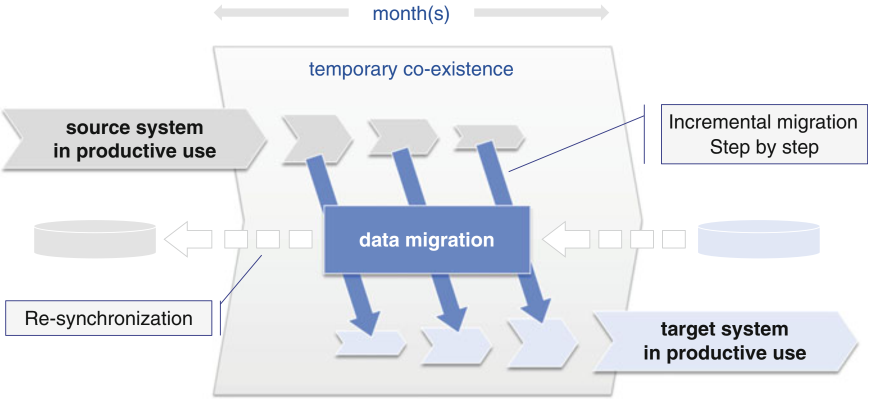

month(s)
temporary co-existence
source system in productive use
Incremental migration Step by step
data migration
Re-synchronization
target system in productive use

Fig. 16.5 Migration with temporary co-existence

16.7 The PLM Integration Platform

A PLM integration platform provides a common, easily available access mechanism to all necessary PLM information for collaborative engineering in a heterogeneous system world. Standardized interfaces to the majority of the required PDM functionality allow short set-up times for establishing working partnerships. Such a PLM integration platform is a product designed for the exchange of PLM information between different in-house systems, cross-domain integrations as well as partner data exchange. All components and processes are optimized for PLM structure handling. The platform enables to access the PLM data located in different systems via a standardized interface. Those interfaces with a wide range of PLM systems are realized in specialized connectors. As a base functionality these connectors can read and write objects, relations, files and of course all corresponding attributes (Fig. 16.6).

The PLM integration platform build more coarse-grain functionality by orchestrating and combining the logic to read and write PLM structured data by using the generic fine granular interface of the PLM system connectors. As a result, the export and the import of complete PLM structures together with the referenced secondary content like CAD, which is the primary data for the design engineer in almost all cases, are managed by such an integration platform.

The user interface, if necessary, is provided by a direct PLM system integration. This integration is controlled directly within the graphical user interface of the PLM system and gives the user the opportunity to select and filter information without changing the user interface. As a result, the direct integration communicates with the PLM system on one side to collect the data and with the integration platform on the other side to collect the necessary receiver information. The receiver information is needed for routing purposes but as well for filtering or transformation

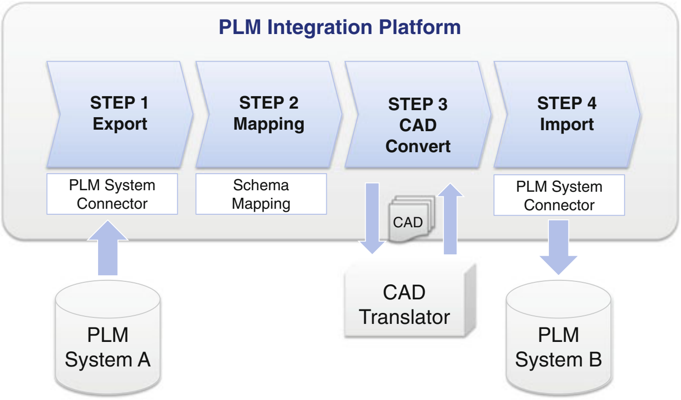


Fig. 16.6 Integration platform with workflow ⇒ PLM system connector ⇒ PLM system

control. Finally, the sending PLM system usually manages a record for all data to be exchanged to allow efficient retry or control, and for bookkeeping.

16.7.1 Cross-System Integration Scenario

A PLM integration platform is able to implement aggregated simultaneous views on multiple PLM systems. Sharing a common data model as defined by the specification and merging of multiple data sources into such a common view is an obvious, but complex application. The integration platform provides not only a shared and system-independent view on PLM data, but defines the semantics of access functions and the necessary answers of the connected PLM systems to form valid PLM data. The aggregated view on multiple PLM systems is in reality a one system-like image on distributed data. Users are no longer forced to use multiple user interfaces and multiple terminologies to access similar data from different sources. This shared representation requires structural, semantic and timely transformations. The platform provides such transformations which are based on a well-established transformation chain from PLM system specific data to generalized PLM representation, and to the special purpose end user representation.

Furthermore, the integration platform with its generalized data model allows an easy replacement of product data viewers to meet the requirements of different user groups. Based on the generalized system independent PLM data representation specialized viewing and processing solutions will support designers and other end users directly. If the cross-system scenario is based on a synchronous data access solution and all data access are realized online the need for erroneous data replication is minimized and applications may work on original data. This will allow optimistic change strategies to be implemented but may cause complicated interlocking scenarios which are not easy to manage or will cause inefficient data locking behaviors when multiple changes compete in the process.

16.7.2 Cross-Technology Integration Scenario

Instead of binding the integration solution directly to an implementation data model of one of the involved PLM vendors, a neutral representation based on the model-driven architecture (MDA [17]) approach is more suitable. Within the generalized MDA approach the necessary transformation steps from the vendor specific representation to the neutral, potentially standardized representation are defined. Utilizing well-defined data models (UML [18], XMI [19]) XML representations can be easily derived. Such examples are the IBM PLM framework infrastructure [20], the OSLC [21] infrastructure or the PLM Services [22] set of specifications. Bindings to programming languages exist as standards or quasi-standard industry solutions. Thus, a seamless transformation between an XML message comprising a PLM data set of information and a specific language or technology binding exists. The MDA nature of the data representation specification would allow the extension of the data exchange capabilities to new implementation techniques by defining a platforms specific mapping without losing the semantic meaning, but gaining the flexibility of multiple language or representation bindings.

One important way of data exchange in PLM is the STEP file representation. The PLM integration platform is suitable to implement integration scenarios with existing STEP file-based infrastructures. A complete STEP Part21 representation suitable for processing by industry strength STEP processor is a must. A lot of processes, especially in aerospace and defense, depend on such implementations. Due to the normative mappings between AP214 [23] AIM, which is the data model exchanged in STEP files, and the XML representation, exchanged by the Web Services defined by the integration platform interfaces, a complete round-trip and unambiguous data transfer scenario becomes feasible.

16.7.3 Cross-Domain Integration Scenario

The PLM integration platform is suited to implement cross-domain scenarios interconnecting PLM, CAD, ERP and other planning system data sources. The structural information provided by product or document information given in the PLM system provides a comprehensive access methodology to complete product definition data including support measurements for manufacturing, process data, logistics, change management and others.

16.7.4 Cross-Company Collaboration Scenario

Highly complex manufacturing goods like the products of the automotive, aerospace or consumer goods industry today need a close collaboration between the original equipment manufacturer (OEM) and its suppliers within an integrated context (Fig. 16.7). There is a strong need to collaborate during the design phase of a product and exchange information between partners (see also Chaps. 7 and 8).

Organizational data are usually within the scope of the specification of the connected systems. All PLM information may be associated with their corresponding creator and owner information, although it is not obvious for external resources. They need special consideration mainly for two reasons: (1) management of external resource information requires special efforts, and (2) external resources demand competitor safety, so that multiple external resources need to be granted access to carefully chosen subsets of information to avoid conflicts with non-disclosure agreements. Thus, the utilization of alias names suitable for cross-company referencing of parts, documents and others might be a solution. A better approach is the separation of all external bodies to prevent unnecessary cross-company access or

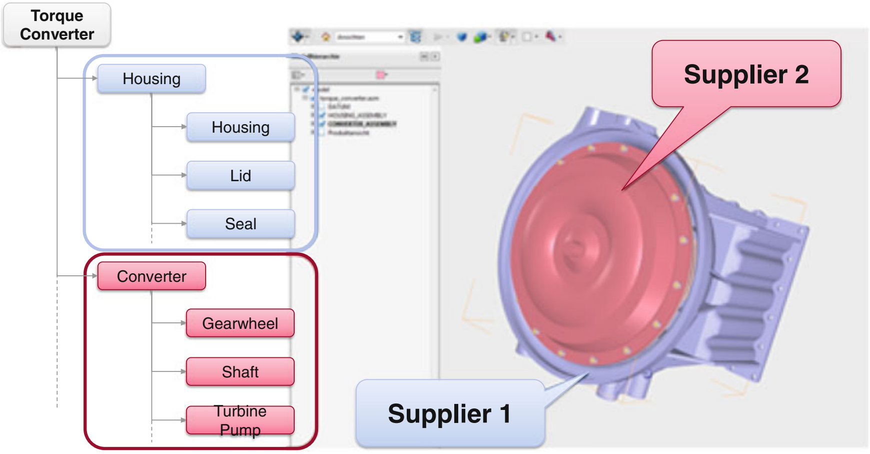


Fig. 16.7 Assembly with parts highlighted which are designed by suppliers

communication and the delegation of resource management to the external partner directly. With the corresponding legal and information management scheme in place a collaboration partner is able to allow its employees access to data and to guarantee that access control information is current and accurate.

The scenario is realized by providing a definition for a common generalized neutral data model and access methods based on common web technology, easily accessible for a large number of automotive companies. It is state-of-the-art to use a web-based transport mechanism which delegates aspects of authentication and authorization as well as encryption and other security measurements to already existing technologies and infrastructures. The PLM integration platform just utilizes this framework to access valid PLM data within user or system context. The framework comprises the aspects of generalized neutral data representation, data mapping, and atomic data handling functions. The data handling can be easily combined into more complex activities, which are controlled by a workflow mechanism and a framework for access control and process management. The framework solution is accompanied with extended logging and monitoring facilities. These services are highly customizable to meet a wide range of customer demands and to be integrated in the management and reporting services. This allows the build-up of efficient interactive scenarios.

16.8 Case Studies

The following case studies comprise realizations of the use cases described above. They gather our experiences from real-life applications built with the software products described in Sect. 16.7. We introduce the details and aspects of those solutions to the extent of understandability and comprehension.

16.8.1 Automotive

The central PLM concept in the automotive industry is the platform approach which is currently extended into a modular approach. Several car lines of an automotive OEM have an identical platform or are built based on the same module set sharing parts, components, concepts, manufacturing principles, supply chains, maintenance procedures and so on. For the OEM this leads to a reduction of costs on the one hand due to a standardization and re-use. On the other hand its modularization is the key to an increasing variety of products. This variety needs to be managed, approved and documented. This also leads to a very complex PLM concept due to the necessary modularization of PLM structures and the management of product changes over a larger number of car derivatives (see Sect. 21.2).

Since several years a cross-OEM partnership for the design and the production of several vehicle types is established. From the PLM point of view two types of partnership are established: collaboration and asynchronous integration. The collaboration approach is typically used for long term partnerships or will be reused for several project scenarios. The asynchronous integration is well known for ad hoc partnerships and short term or single project partnerships.

16.8.1.1 Collaboration

Focus of a collaborative partnership is the development of a cross-company platform concept. Both partners build this platform contributing their product expertise. To support this collaboration with PLM concepts, typically a central PLM system is used to manage data of both partners. There is no need for all partners to internally run the same PLM solution as the so-called Partner Hub. It is an independent solution with a clear separation from the internal systems. Editing PLM data is typically only carried out in the internal PLM system. This avoids the situation that a designer has to work in two different PLM environments. The Partner Hub mirrors only the PLM data that are partner relevant.

From a PLM process perspective the challenge is to provide in the Partner Hub only such data that are needed to fulfill the collaboration with the partner. In addition, data need to be synchronized with the local PLM system in an automatic way to avoid replicated work and version mismatches. To establish such collaboration, internal data are typically pushed to the Partner Hub initiated by triggers. The trigger is fired at well-defined stages of the design process in the internal PLM system to synchronize the PLM data from with the Partner Hub. This connection is typically established by using a synchronization framework with connectors to the internal PLM system and to the Partner Hub.

Technically this synchronization acts like the PLM integration described in Sect. 16.6.1, but of course with a much more complex logic to secure the process and to avoid the exchange of unauthorized data. The most significant difference is the location of the Partner Hub in the demilitarized network zone of one partner or on an external server. After the trigger has initiated the data transfer from the internal PLM system to the Partner Hub, the synchronization framework exports all related data out of the internal system and maps the data structure to the scheme of the Partner Hub. CAD data need a special focus before the transfer to the Partner Hub is started, because not all content of the native CAD data should be published to partners. Sometimes just neutral formats like JT will be provided or native CAD data are prepared as described in Sect. 18.6.3.2. Only these processed CAD data are transferred to the Partner Hub.

From usage perspective there are two concepts to view and work with data published in the Partner Hub. The simplest way is to use the standard PLM system client to access the Partner Hub to view or download the data interactively. The second way is to download and import the data automatically into the own PLM system. With the help of effective PLM selection operations and lightweight viewing facilities capable of web-based geometry inspection the design engineer identifies the relevant changes in the structure and defines the geometry models that have to be exchanged. After that the engineer initiates a data exchange process to transfer the identified data to the own internal PLM system. The exchange process keeps track of all the interactions, converts CAD models according to the guidelines of the company exchange processes and imports the data in the internal PLM system according to the defined data mappings. The partner data can now be used in the standard internal PLM processes.

This real-life example demonstrates the capability of the integration platform. It allows the interaction of several tools on a neutral and open foundation, namely the PDM system providing the necessary structure information, the exchange infrastructure providing the data converting and messaging capabilities and the web browser technology providing easy access to cross-company and cross-domain functionality. Every single building block of this tool set is exchangeable with another one compliant to the standard interfaces. The product specification is capable to support the required functionality under productive conditions.

16.8.1.2 Asynchronous Partner Integration

Another common approach for partner data exchange in the automotive industry is the Asynchronous Partner Integration. This approach uses a data exchange tool to transfer data from one partner to the other and does not define a central collaboration management tool. This approach defines high demands on the data format and the package sequence during the exchange.

Different to a collaboration platform in which a central tool provides information to all partners and data exchange is supported by a synchronization framework, a neutral data model is a central aspect for asynchronous data exchange. In the automotive industry typically standards like STEP [23] or JT [24] are used for data exchange (see Sects. 21.6.2 and 21.7). It is the responsibility of all partners to provide and accept data in the defined neutral format. PLM integration platforms are typically used to support the data import and export scenarios.

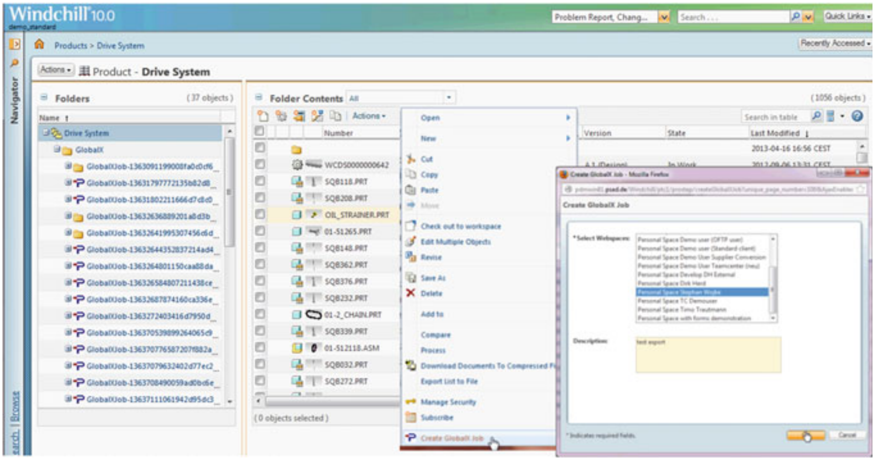


Fig. 16.8 Collaboration platform realized with Windchill

Our implementation with OpenPDM and OpenDXM GlobalX provides a new callback in the graphical user interface in the PLM system. A user interactively selects product data by choosing a root element and applying some filtered traversal and selection operations. Then, he starts in the PLM system the data transfer. Figure 16.8 shows the data selection and the partner data transfer dialog fired from the graphical user interface of the PLM platform Windchill and communicating with the data exchange platform.

The integration platform starts the background process to export the selected data and maps the data to a neutral format. It yields a package of metadata with linked CAD data in the neutral formats. This package will be handed over to the data exchange tool, which performs the transfer to the partner. The partner system recognizes the new package and starts an import process with its own integration tools. To avoid corrupt data in the target system, the import engineer uses tools to perform a dry run import. Based on visual inspection, he can decide to import the data or to decline this data delivery. A central aspect is to guarantee the sequence of data package imports in case of huge volume. The export, transfer and import of the PLM data from one partner to the other can take more than one day. In result it is possible that small packages can overtake big packages in the data transfer process. If the package sequence is not recognized during the import the loss of data can occur. The data exchange tool prevents the lost update in that scenario.

16.8.2 Aerospace

The aerospace industry heavily depends on stringent documentation of their product along the whole lifespan. Where the number of products is significantly smaller than in automotive, the sheer volume of product data, e.g. number of parts, number of documents, variety of configuration options, is much larger. Even the lifespan of the product is longer and usually challenges the lifespan of IT systems that manage the design process [25]. The documentation needs to store information on the complete product structure "as-designed", "as-defined", "as-planned" and "as-maintained" and not only as reference but as individual representation of every single aircraft identifiable by its serial number.

16.8.2.1 PDM-ERP Integration

Both PDM and ERP systems manage product-related data. Common denominators are part master data, configuration and effectivity, but product structure differs. PDM systems manage highly detailed design structures. This view on product structure is the engineering bill-of-material and feeds the bill-of-material in the ERP view. ERP stores the planning bill-of-material which manages manufacturing or maintenance planning. Its structure supports assembly processes, logistics, and maintenance or spare part management in after-sales processes. This structure reflects not only the design hierarchy, but the product management hierarchy (Fig. 16.9).

The PDM view evolves during the development and change phases of the product. This view will change if design principles change. Their lifespan is significantly shorter than the manufacturing and maintenance phase of the ERP structure. The planning structure is also significantly flatter than the design structure.

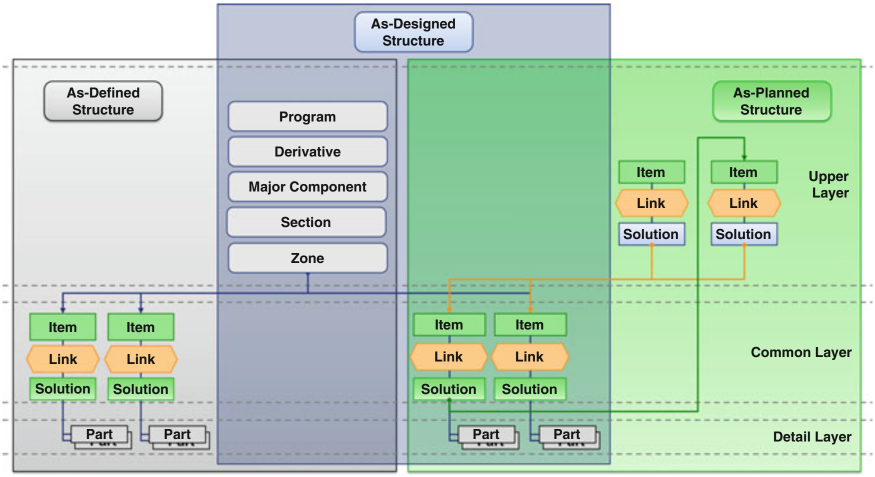


Fig. 16.9 Elements of definition, design and planning product structure layers

16.8.3 Consumer Electronics

The consumer electronics market has an extremely short time-to-market. The lifespan of devices is significantly shorter than in automotive or aerospace. The number of players is larger compared to the automotive or aerospace industry. Customer expectations are targeted on usability, energy efficiency and increasingly on sustainability. Success in such a mass market strongly depends on a vendor's ability to react fast on trends, to be cost efficient at the same time and to deliver well-designed gadgets. New trends in mobile and connected solutions demand constantly innovations. Simulation of mechanical and electro-magnetic properties is an essential part of the development process and a must to choose alternatives for a constantly changing product design. The integration of PLM with simulation data management systems is an application for cross-domain integration.

16.8.3.1 PDM-SDM Integration

With OpenPDM an integration of PLM with SDM was built to allow an extremely short lead time to incorporate simulation analysts into a team of designers and to work efficiently co-located. Thus, it was planned to setup an additional project environment within the PLM system which stores all necessary data of the original product structure as shallow copy. This project specific workspace is referenced from the SDM system. Analysis is performed on propagated information of the reference data set. The functionality of the integration comprises these use cases:

- Setup of a reference project environment
- Initial load of a selected reference substructure into the SDM system
- Refresh the initially loaded substructure

The refresh is carried out on the main assembly and all its children to propagate structural changes and releases, and to monitor release state and design changes in the SDM system workspace (Fig. 16.10).

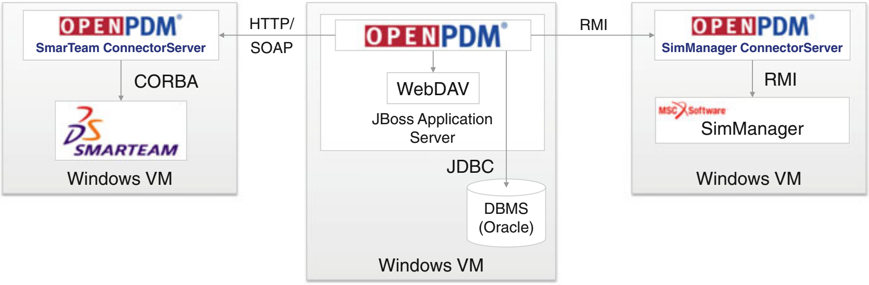


Fig. 16.10 PDM-SDM integration architecture

The refresh functionality was supported both by push and pull mechanisms. The PLM system could trigger the refresh function to push changes to the SDM system. The SDM system is capable to pull changes from the PLM system. This was bound to a user interaction. The cross-domain communication can be initially supported by those basic building blocks. But this elementary approach leaves some potential for further improvement like

- Feeding back of simulation results to the PLM project
- Documentation of design quality checks
- Induced model changes in the simulation need to be reflected in design domain.
- Handling of variant and alternative design data.

16.9 Conclusions and Outlook

PLM is a complex concept to deal with the challenges in state-of-the-art product development. The PLM IT solution is just a single constituent. The IT solution itself consists of a number of integrated tools. The integration scenarios show common patterns but differ in detail and level of interaction.

Central to PLM integration is a common set of product structure information. Every lifecycle stage utilizes its special aspects of product structure. The representations as-designed, as-planned, as-build are just examples. Their functional relationship is defined by the use cases they are applied to. The use cases have their real-life counterpart in business processes. The business processes are specific to the industry. To some extent they are differentiators of the market players. The implemented PLM solution is not a decisive advantage of competitors. Its level of business process support definitely is. It is questionable, if COTS solutions "one size fits all" will meet all process requirements. More generic template approaches addressing the needs of standardized processes will ease adaptation but will never avoid customization. Furthermore, it is not clear, if harmonization and standardization of processes is desirable across an industry to accommodate COTS utilization. Today, specific software solutions support the product development process in practice. It is questionable, if all lifecycle stages of a product shall adapt to the same business processes which has proven to be successful in a single business.

The level of business support may be judged by simple evaluation of the product lifecycle solution and its supporting measures. Taking into account, that PLM solutions are usually integrated solutions, the biggest advantage of an implemented PLM concept arises from the seamless flow of information across all boundaries: system, domain, technology and enterprise. We have illustrated how a specialized integration tool may help to bridge those boundaries. With the presented toolset a wide range of requirements was accommodated. New PLM components and application can be easily integrated without breaking existing integrations. The integration approach is customizable along the development and customization path of the connected components. The integration framework specialized to PLM concepts has a significant added value in comparison with enterprise service bus solutions or EAI solutions. The PLM integration framework is specialized to PLM concepts with modules to create, read, update the product structure together with well-designed building blocks to support PLM patterns like versioning, check-in and check-out, and configuration. The general purpose of enterprise service bus solutions is bridging technological gaps, while EAI solutions focusing on the exchange of tabular data, which are a subset only of the structured product data.

New challenges arise from the demand for better support of the PLM-CAD integration and from new technologies like mobile devices or computer graphic technologies asking for new metaphors to handle PLM data, not by browsing data columns but by exploring graphical representations of the product with a selected set of PLM data as overlay, potentially in an immersive graphical environment.

References

1. Eigner M, Stelzer R (2009) Product lifecycle management. Springer, Dordrecht, Heidelberg, London, New York
2. Abramovici M, Schulte S, Leszinski C (2005) Best practice strategien für die einführung von product lifecycle management. Ind Manage 21(2):47–50
3. Tavcar J, Potocnik U, Duhovnik J (2013) PLM used as a backbone for concurrent engineering in supply chain. In: Stjepandic J, Rock G, Bil C (eds) Concurrent engineering approaches for sustainable product development in a multi-disciplinary environment. Springer, London, pp 681–692
4. Katzenbach A, Steiert H (2011) Engineering IT in der Automobilindustrie—Wege in die Zukunft. Informatik-Spektrum 34(1):7–19
5. Silcher S, Seeberg B, Zahn E, Mitschang B (2013) A holistic management model for manufacturing companies and related IT support. Proc CIRP 7:175–180
6. Schuh G (2006) Produktionsplanung und -steuerung: Grundlagen, Gestaltung und Konzepte. Springer, Berlin, Heidelberg
7. Vajna S, Weber C, Bley H, Zeman K (2009) Integrated design engineering. Springer, Berlin
8. Stark J (2011) Product lifecycle management. Springer, London
9. Dekkers R, Chang C, Kreutzfeldt J (2013) The interface between "product design and engineering" and manufacturing: a review of the literature and empirical evidence. Int J Prod Econ 144:316–333
10. Abramovici M, Bellalouna F, Neubach M (2010) Delphi-Studie PLM 2020. Ind Manage 26 (3):47–50
11. Abramovici M, Bellalouna F, Flohr M (2008) PLM für individuelle reale Produkte. Ind Manage 24(3):41–44
12. Sendler U (2009) Das PLM-Kompendium: Referenzbuch des Produkt-Lebenszyklus-Managements. Springer, Berlin, Heidelberg
13. Schuh G, Assmus D, Zancul E (2006) Product structuring—the core discipline of product lifecycle management. In: 13th CIRP international conference on lifecycle engineering, pp 393–398
14. Mechlinski T (2007) Aktuelle Konzepte für die PDM-Integration. Produkt Daten Journal 14 (1):34–37
15. Stiefel P (2010) Eine dezentrale Informations- und Kollaborationsarchitektur für die unternehmensübergreifende Produktentwicklung. PhD thesis, Technische Universität Clausthal, Clausthal-Zellerfeld

16. Grau M, Trautmann T (2007) Vendor-independent integration of CAD and CAE processes based on OMG PLM Services. NAFEMS 2006
17. OMG (2003) Model driven architecture guide. http://www.omg.org/cgi-bin/doc?omg/03-06-01. Accessed 23 Aug 2014
18. OMG (2004) Unified modeling language (UML), ISO/IEC 19501. http://www.omg.org/cgi-bin/doc?formal/2004-04-04. Accessed 23 Aug 2014
19. OMG (2005) XML metadata interchange (XMI). http://www.omg.org/cgi-bin/doc?formal/2005-05-01. Accessed 23 Aug 2014
20. Credle R (2008) SOA approach to enterprise integration for product lifecycle management. IBM, International Technical Support Organization, Raleigh
21. OSLC (2013) Open Services for Lifecycle Collaboration Change Management Specification Version 2.0, http://open-services.net/bin/view/Main/CmSpecificationV2. Accessed 23 Aug 2014
22. OMG (2011) Product lifecycle management (PLM) services. http://www.omg.org/spec/PLM/2.1. Accessed 23 Aug 2014
23. ISO (2003) STEP 10303:214 core data for automotive mechanical design processes
24. ISO (2012) 14306: Industrial automation systems and integration—JT file format specification for 3D visualization
25. Mas F, Menendez J, Oliva M, Rios J (2013) Collaborative engineering: an airbus case study. In: The manufacturing engineering society international conference, MESIC. Elsevier, pp 336–345