10 theses about MBSE and PLM

Challenges and Benefits of Model Based Engineering (MBE)

Abstract

The complexity of innovative products is increasing through interaction and interdependency induced by mega-trends such as the “Internet of Things”, “Smart Manufacturing” and “Industrie 4.0”. Multiple engineering disciplines must be well coordinated to cope with the challenge; both organization and technology are affected. In this context, our goal is to establish a solid foundation for a lifecycle spanning development and manufacturing process, called Model-Based Engineering (MBE). Specifically the information management in the product conception phase, including the interoperability of the supporting IT infrastructure, are of prime importance for the whole product lifecycle.

This paper elaborates 10 theses about the necessity to integrate the currently isolated Product Lifecycle Management (PLM) and Model-Based Systems Engineering (MBSE) methods. Model-Based Engineering (MBE) is seen as the resulting concept of combining lifecycle spanning management of product data (PLM) and formal description of systems (MBSE).

Management Summary

Model-Based Engineering is considered to be one of the key technologies to resolve issues in the development and production of innovative and interconnected products. A holistic system model of the product under consideration (and consequently the ability to build it) is a prerequisite for effective collaboration of all contributing disciplines from product management to mechanical, electrical and software development and finally to production. It is widely accepted that completeness, consistency and feasibility of the resulting product concept can only be assured based on an integrated information management.

The following theses cover the challenges and benefits of integrating MBSE and PLM – leading to the concept of MBE.

Theses:

1. MBE is *the* enabler for the “Internet of Things” and “Industrie 4.0”.
2. Product liability and functional safety regulations are a driving factor for MBE.
3. Future PLM systems need a holistic view on a product as a multidisciplinary system.
4. Early design decisions must be logically and functionally validated using system models.
5. Results from MBSE must be made available over the whole product lifecycle.
6. MBE requires models with meaning.
7. The MBE tool chain must rely on technology independent standards.
8. Increasing complexity of products and production systems asks for new development processes, methods, and tools.
9. MBE requires changes in organization, methodology, technology and education.
10. Investments in MBE can deliver a ROI of 3:1.

1. MBE is the enabler for “Internet of Things” and “Industrie 4.0”

As of today, PLM and MBSE are two independent approaches to product development, evolved from different requirements. We consider Model-Based Engineering (MBE) to be the result when integrating PLM and MBSE.

Initially, Product Data Management (PDM) was introduced to handle product specifications like requirements documents and CAD files in terms of version management, change processes and product configuration. 2D CAD models were usually just printed out and the drawing were stored in a cabinet. With the introduction of 3D CAD there was the strong necessity to manage 3 dimensional CAD models virtually, which has become the realm of PDM systems.

PLM is the evolution of PDM to manage all information around a manufactured product throughout its lifecycle. In future, even more virtual models covering new engineering aspects need to be managed – requiring even stronger model management capabilities. Today, PLM systems follow a document-based approach as they are handling documents and correlating metadata. Future PLM must incorporate more structured information with meaning and follow a stronger model-based approach than today. In other words, new generation PLM must itself be model-based. This involves a major paradigm shift.

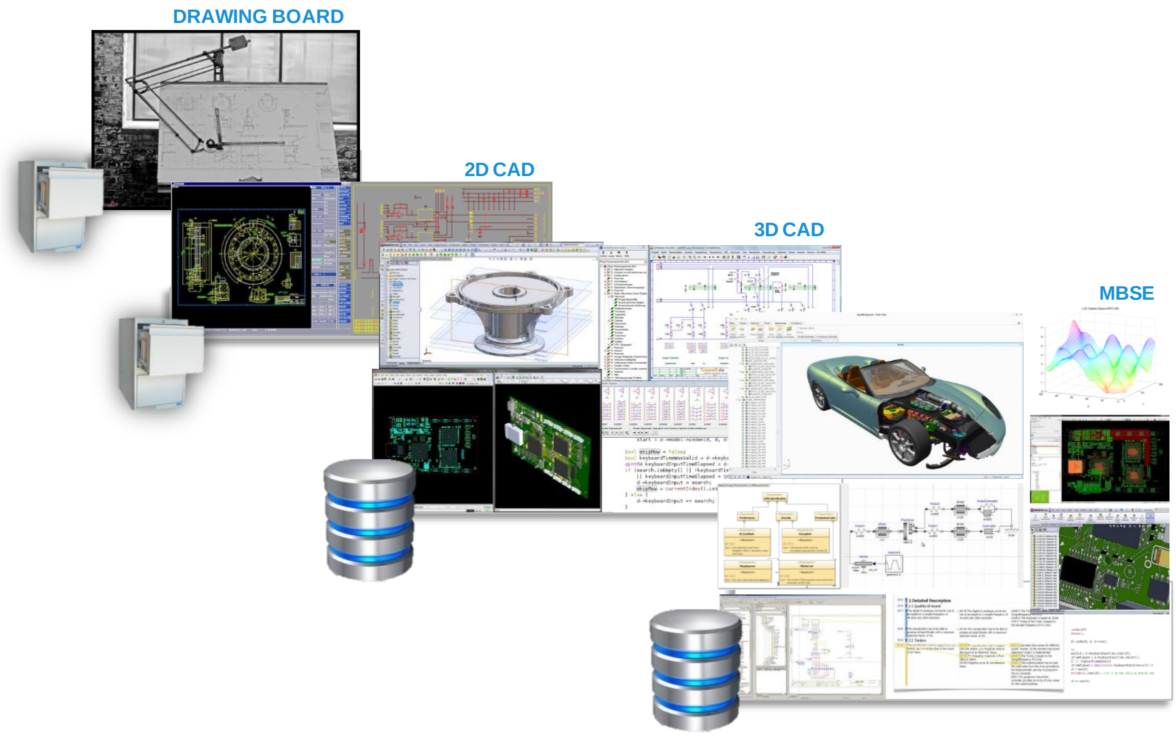

Fig.1: The evolution of product development from the drawing board to model-based engineering

On the other hand, Systems Engineering (SE) grew up from the need to conceptually understand a product and to ensure an adequate design early in the development phase. The overall objective of systems engineering is to produce a complete, consistent and feasible system specification upon which the product can be built. Due to the ever increasing complexity of the products under consideration, systems engineers have started to employ models to simulate structural stability, behavior, cost or other aspects. Functional verification and safety assessment is often a major objective.

Today, SE is mostly applied in the early stages of product development and is yet disconnected from PLM data in the subsequent stages of product development, manufacturing and maintenance.

But how can a complex interconnected product be developed, manufactured and maintained without the knowledge of its functional behavior and interconnection with the environment? Moreover, smart manufacturing plants ("Industrie 4.0") are itself complex systems calling for model-based development methodologies, as well.

We conclude that:

- PLM supports the management of document versions and variants, product configuration and change processes throughout the lifecycle, but lacks transportation of semantic, computer-interpretable information.
- MBSE works with semantic information of a product, but lacks managing it throughout the lifecycle.

2. Product liability and functional safety regulations are a driving factor for MBE

The European legislation with respect to product liability has imposed higher requirements in terms of product and process documentation. Before, consumers needed to prove that product defects have caused a damage to receive compensation. The new regulation is based on the "reversed onus of proof" where a manufacturer is now obliged to demonstrate that he has diligently developed and manufactured a product using state-of-the-art methods to successfully defend himself from legal claims. As a result, higher standards in quality assurance and in process documentation have been adopted. This is one of the reasons why PLM systems have been introduced in many industries to handle *all* product related information.

To further improve consumer protection, regulations regarding the functional safety of products have been released. For example the ISO 26262 is setting high requirements on automotive OEMs and suppliers to ensure functional safety. Obviously, more complex systems lead to more extensive verification and documentation. Functional modeling will help to ensure functional safety of systems. Models also support an accurate documentation. That's why MBSE is widely used for risk and safety assessments with *fault tree analysis* (FTA), *failure mode effect analysis* (FMEA) or other methods. Of course, documentation for functional safety will still be made available in a document – however the document will not be the written as such but drawn from a system model at a certain point in time.

3. Future PLM systems need a holistic view on a product as a multidisciplinary system

To meet the challenge, it is essential that all information created during a product's life cycle from the conception to recycling is properly managed in a common context. This is by no means true with respect to today's PLM, because disparate work products from mechanical, electrical and software engineering are handled without any notion of its logical interdependencies. PLM systems claim to be the interdisciplinary backbone of modern virtual product development, but in fact are missing support for non-mechanical domains or multidisciplinary development. The overarching concept remains informal and is rarely documented adequately.

However, complex products must be seen as multidisciplinary systems made of integrated and inter-connected work products of all involved disciplines. Therefore we propose to enable PLM systems to manage MBSE constructs and artefacts: A transition from document-centric to *atomic and logically interrelated artefacts* is postulated.

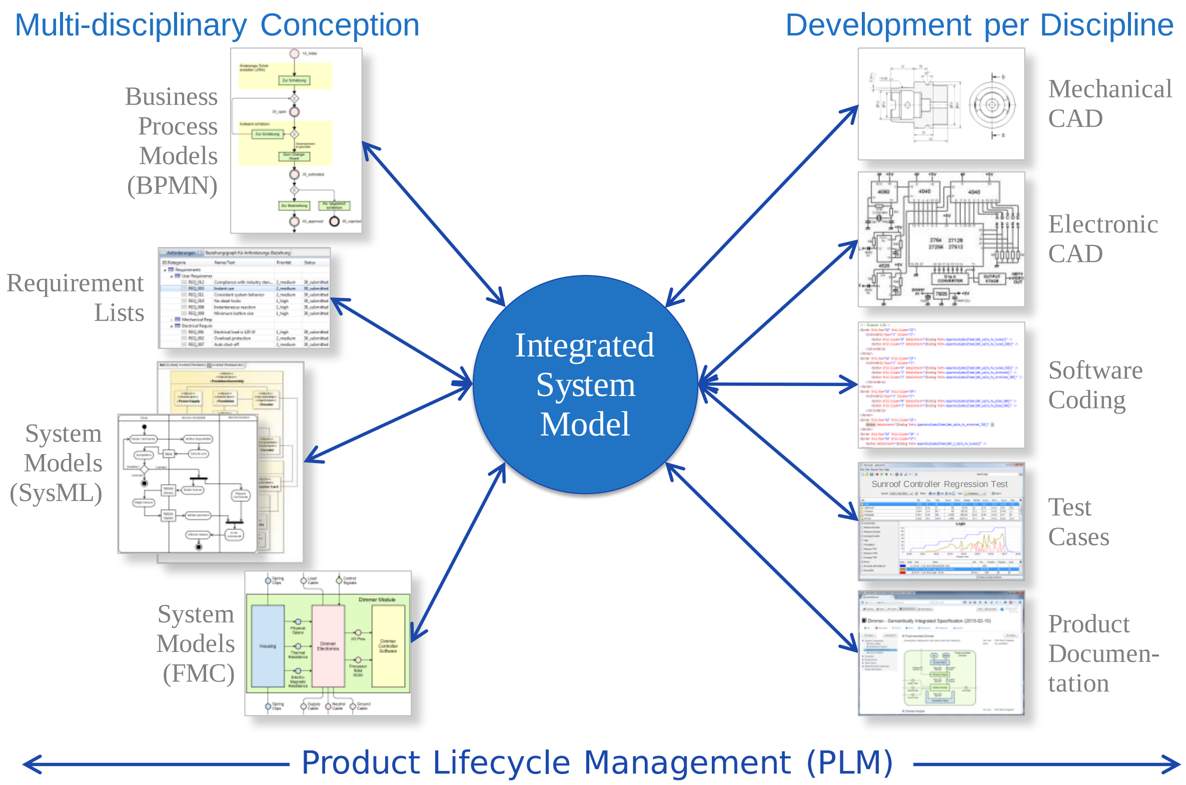

Fig.2: An integrated system model combines information from different sources and is a reference for development

MBSE interrelates the information elements pertaining to engineering, i.e. to weave a semantic net ('graph') of entities in a way that it can be interpreted by computers. It thus reduces the effort required to create, find, control, use, share and maintain information, making it an ideal basis for an efficient information life-cycle management system. This approach allows for effective traceability of and accountability for information. Continuous improvement of data quality is fostered by enabling automated, rule-based data governance. Recurring human effort is reduced and more attention can be given to the engineering design. Documents will still be used as official or legal records, but they are derived from system models – a major step towards consistency of design and documentation.

To toughen the Integrated System Model manageable for use in the whole system's lifecycle, certain objects in the system model must be put under configuration control: so called Model Configuration Items (MCI) [7]. Versioning and configuration control has been supported by PLM systems for a long time. The artefacts which were put under configuration control were documents containing a multitude of information items – in future the atomic elements of the integrated system model need to be put under configuration control themselves.

4. Early design decisions must be logically and functionally validated using system models

The cost of a design change rapidly increases the later it is made in the product lifecycle. Changes resulting from manufacturing issues are considered dramatic, only surpassed by the cost caused by product recalls. Design flaws can even lead to the failure of a whole mission such as space exploration. Thus, design decisions should be verified promptly and systematically throughout the product lifecycle. A good example is electronic design: Formal circuit descriptions (VHDL) are subject to automatic rule checking, so that the electronic industry has maintained an extremely high innovation pace for years – at an impressive level of product quality.

Thus, CAE activities like logical validation and model-based simulation must become an integral part of the systems engineering domain. As system design progresses, the validation addresses more and more concrete aspects long before the product is actually built and used.

5. Results from MBSE must be made available over the whole product lifecycle

In early project phases the development team extensively analyses the needs of end users and elaborates a product concept offering a solution to the user problems. This analysis results in knowledge about the new system and its environment. It is documented in process models, functional decompositions, requirements specification and numerous other artefacts. This information base sets the stage for the development teams and in the following phases of the product lifecycle.

As an example, let's consider reuse of the MBSE information in the product manufacturing, market introduction and product maintenance phases.

- Requirements and system component specifications are the natural reference for manufacturing *quality control* and *acceptance tests*.
- Documented user needs are a good base for developing a *marketing* strategy, while product data sheets may be derived from the technical product specification.
- Business process models, use cases and functional models may be used for preparation of the final *product documentation*, to be used throughout the maintenance phase.
- Training materials for the support personnel can include information from the System concept, architecture and user requirements documents. Detailed system models provide a good basis for specific training e.g. for service technicians and service engineers.
- Finally, requirements and the rationale for design decisions are extremely valuable for further development or follow-up products.

6. MBE requires models with meaning

As discussed before, the conception of new products is often quite informal: Results from different disciplines and sources are documented using office documents, requirement lists and sometimes system models. The content with considerable detail is conceptually interrelated and made consistent by personal communication, brain power and discipline; yet there is little tool support and lots of redundancy. The transition to concrete development is again informal and based on individual understanding. The first structuring of information with relevance is found in PLM systems, very often according to the 'Bill of Materials'.

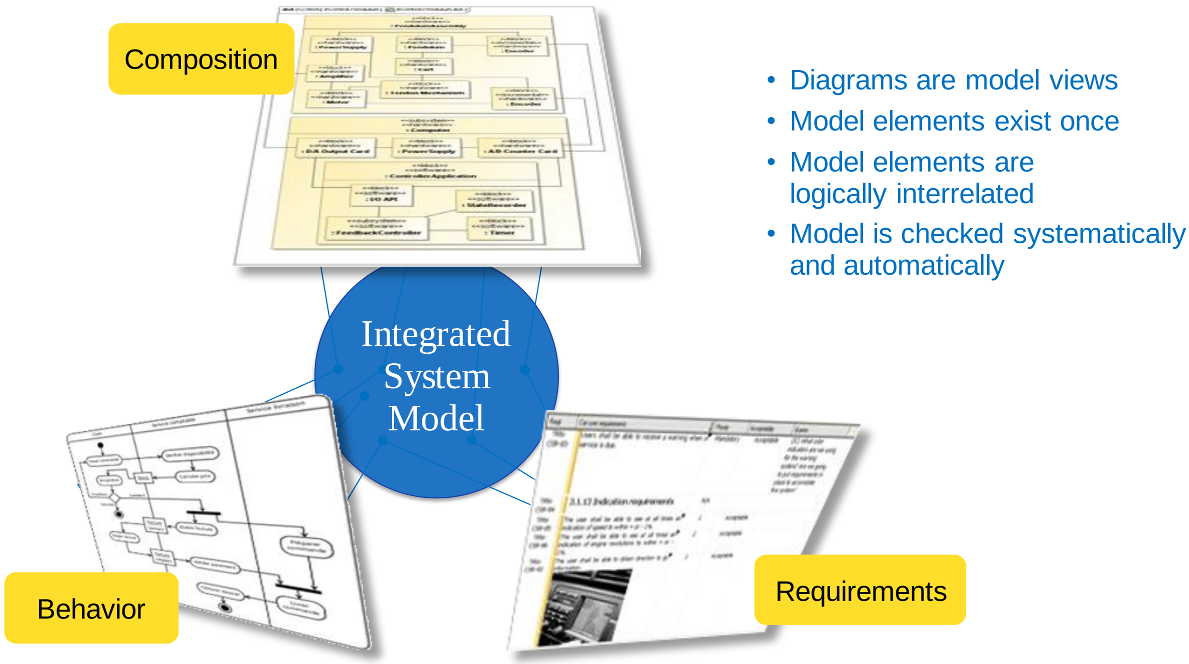

Fig.3: Diagrams and lists are views on the Integrated System Model

However, research and practical experience from various projects have shown that more formal rigor in the conception phase is beneficial. The work results are thus integrated in a common context, an overarching system model with the following characteristics:

- *Redundancy-free*: Every information element, such as user story, requirement, function or part exists once and is used in different model views for different communication purposes.
- *Interlinked*: Explicit logical relationships between information elements express meaning, such as 'A part satisfies a requirement' or 'A subassembly contains a part'.
- *Focused*: Different roles, such as 'Product Manager' or 'Electronic Engineer', see just the aspects they are interested in.

A 'Model with Meaning' is not only a collection of diagrams, but expresses the logical, i.e. semantic relations of the elements shown [6]. Some relations can be derived automatically from a diagram, such as a composition 'Gearbox contains Gear', some need to be added manually, such as a requirement affecting a part 'Gearbox satisfies Maximum Torque'. Hence, a 'Model with Meaning' is a semantic net of entities and relationships which can be used for navigating, searching, reasoning and rule-based checking.

Building such models by integrating individual work results from different disciplines has some prerequisites:

- A *vocabulary* identifies the information elements ('entities') and establishes a common understanding. With all likelihood there will be different vocabularies for different domains; important is that a vocabulary is agreed upon in the context of a given system. The contributions of different disciplines are mapped to the common vocabulary.
- An *abstraction* to fundamental information elements helps to integrate work results from different sources. Practical experience has shown that the fundamental entity types 'Actor', 'State', 'Event' and 'Requirement' can properly express the logical relevance of the mostly used information elements [5], which may have been created with informal, semi-formal and formal methods. For example, 'Actor' is the abstraction of 'Function', because it is an active entity, and 'State' is the abstraction of 'Form' or 'Data', because they are passive entities.
- A *language*, with both syntax and semantics, relates the entities and captures the meaning. Past research suggests that assertions as triples 'Subject predicate Object', being powerful and simple at the same time, are well suited for the purpose. A language proposes a sufficient number of assertion types with predicates such as 'satisfies' and 'contains' and restricts the entity types being eligible as subject and as object.
- Finally, a domain-specific *constraint-set* allows the verification of the integrated system model. Inference based on predicate logic and other techniques may be applied for this purpose.

We suggest to use existing work results from model-based system engineering and to prepare practical show-cases for the approach.

7. The MBE tool chain must rely on technology independent standards

The interoperability of the IT infrastructure for product development and production is a key concern of the ongoing digitization of manufacturing. To ascertain interoperability of tools and to avoid vendor lock-in, the tools must rely on accepted standards. For instance:

- Service-oriented architectures (SOA) have shown their potential to integrate partial IT solutions along the tool-chain. On one hand, data storage and business logic are modularized and made accessible by standard web-services. On the other hand, user interfaces and an orchestration of services support the identified business processes. As long as the general architecture is maintained, improvements can be planned and implemented independently for partial solutions without affecting the total system. International standard organizations such as W3C and OMG have established widely accepted standards, so that IT solutions of different technologies from different vendors can fairly easily be interfaced and successfully interoperate.
- The 'Open Services for Lifecycle Collaboration' (OSLC) is supported by many vendors with the goal to integrate their tools. Standards and conventions are being established to access the data where it is instead of copying it from one tool to the next.
- A catalog of principles and criteria is being promoted by ProSTEP iViP e.V., namely the 'Code of PLM Openness' (CPO): Tools must provide open and documented interfaces, all data must

be accessible using industry standard technologies and it must be possible to employ third-party organizations to extend or run a system, not only from a technical, but also from a practical and an organizational point of view.

- Besides PLM there are discipline-specific management systems in place, i.e. ALM in the software domain. PLM wants to be an interdisciplinary management backbone and does not want to replace discipline-specific management systems (so called TDM systems [8]). The interoperability of PLM and those TDM systems must be ensured. Both type of systems are trying to incorporate functionality of the other, there will be a co-existence of those IT-systems setting high requirements on interoperability. There is the strong need to support a federative system approach in heterogeneous IT landscapes. Meaning, it is better to let specialized IT solutions (e.g. PLM, ALM, ERP) communicate with each other properly instead of trying to make one solution obsolete by obtaining its functionality.

8. Increasing complexity of products and production systems asks for new development processes, methods, and tools

Innovation and customer´s need for individuality are drivers for more and more dynamic and complex products, as well as their processes of development and production.

Complexity *reduction*, *avoidance* and *mastering* are three main strategies for handling the growing complexity of systems. The first two strategies are preferred, if the complexity can be avoided by smarter designs without compromising system functionality and safety. Whereas third strategy is applicable when the complexity is inevitable. As new products and systems are more and more composed of advanced multi-disciplinary technologies, the importance of complexity mastering becomes even more apparent.

There are a number of proven methods and principles to master complexity. Well known are separation of concerns, hierarchical decomposition, abstraction, encapsulation, standardization and others. We are convinced that it is possible to put the work products of the contributing engineering disciplines into a common context. The meaning or 'semantics' can guide the interrelation of all information elements, as described before. We propose a collaborative effort to define the common context and to prepare the tools to support it.

For realization of this integration in development processes, there is a need for methods and (modeling) languages, which can describe a product and the domain in an interdisciplinary, holistic way through different abstraction layers. The discipline of model-based systems engineering (MBSE) pictures the interaction of methods, languages and tools and defines processes. The tools must have a gentle learning curve. Past experience shows that sophisticated modeling tools were not adopted because of their non-intuitive user interface and the elevated effort needed for training and education.

We suggest to support the development process from end to end with tools having a low entry-barrier:

- Tools must be readily available for all participants and easy to use.
- Practitioners must have the choice of the most appropriate methods in their domain and yet the results must be mapped to the integrated system model as described, before.
- Users must be able to find and reference information elements across tools; the copying of information from one tool to the next must be abandoned.

9. MBE requires changes in organization, methodology and tooling

To introduce MBE in an organization several things need to be done. Along with new tools and methods, MBE requires a profound organization change. Companies without systems engineers in their development teams not just need to hire people with these skills, but also need to define new roles and responsibilities. One good guideline is given in ISO/IEC 15288: So called *Integrated Product Development Teams* (IPDTs) are established, where a lead system architect is responsible for the overall system concept while moderating the contributions of all other development teams. The results are documented in an integrated model of the system under development, as discussed before [6].

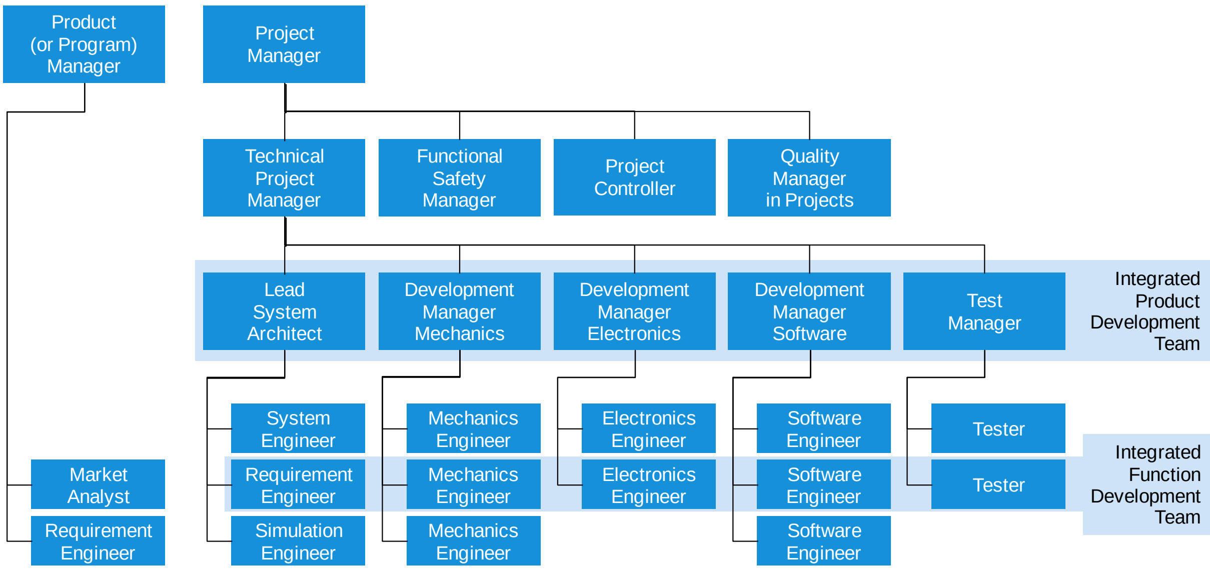

Fig.4: The Integrated Product Development Team is responsible for the overall system design

With respect to methods, all disciplines must collaborate right from the beginning: Conceptual alternatives may be implemented by different disciplines: An electric car's propulsion may rely on a common motor with transmission shafts or on separate motors per wheel, for example. There must be efficient methods to *conceive, assess* and *document* such overarching concepts; they are not covered adequately by any of the individual disciplines, alone.

Even though this may sound straight-forward, the transition from a traditional tayloristic organization to such collaborative system engineering is rather difficult, as much as any other fundamental organization change. It must be well prepared and supported; it usually takes a rather long time with a number of intermediate steps.

Organizations adopting Model-Based Engineering practices need to

- revise their product development process (PDP) with respect to interdisciplinary collaboration and define where Systems Engineering activities with their deliverables fit into the overall development milestones.
- introduce Integrated Product Development Teams (IPDTs) and reshape the roles and responsibilities in product development.
- consider interdisciplinary collaboration not only in development phases, but also during manufacturing and maintenance.
- revise their engineering methods and collaboration between engineers of complementary skills and different locations all over the world.
- define the information flow resp. referencing between different activities and tools, e.g. requirements, functions, behavior, geometry, software code, electrical/electronic layouts, test-cases etc.
- shed light on product data management: The versioning, variants, access policies, release management, change management, etc.
- create technical infrastructure to support team collaboration and concurrent engineering.

10. Investments in MBE can deliver a ROI of 3:1

Model-Based Systems Engineering is a powerful approach to improve and shorten systems development. It has been shown that the *time to market* is reduced and a competitive advantage can be achieved. There is a strong correlation between better systems engineering and shorter delivery times [1]. In fact, leading companies using MBSE reach their targets for quality, cost, *time to market* and sales in 84% of their development projects [2,4]. The ideal amount of effort for MBSE is about 14% of the project volume [1,3]. The invest pays off: A return on investment (ROI) of 3.5:1 was confirmed in a study of executed projects [1] and a tool vendor calculates a minimum ROI of 2.5:1 for his tools [3].

Outlook

The 10 theses presented here will be further refined in a subsequent and more detailed article which we will prepared in the weeks to come. Research results and practical knowledge will be collected. It is the goal to conduct concrete implementation projects with lead customers and to share approaches as well as experiences.

Literature

[1] B. Boehm, R. Valerdi, E. Honour: The ROI of Systems Engineering: Some Quantitative Results for Software-intensive Systems, Wiley Interscience DOI 10.1002/sys.20096,2007

[2] J.P. Elm: A Study of Systems Engineering Effectiveness: Building a Business Case, 2012, http://resources.sei.cmu.edu/asset_files/specialreport/2012_003_001_34067.pdf

[3] Z.B. Scott: ROI of Model-based System Engineering with CORE™, Vitech Corp, 2014

[4] D. Steffen, S.-O. Schulze, F. Gaupp: Systems Engineering: Produktentwicklung erfindet sich neu, UNITY AG, 2014, http://www.unity.de/studien/?file=1A1030B0

[5] A. Knöpfel, B. Gröne, P. Tabeling: Fundamental Modelling Concepts – Effective Communication of IT Systems. ISBN-13: 978-0-470-02710-3. John Wiley & Sons, Chichester, 2005

[6] O.v. Dungern: Übergreifende Konzeption von Geräten für die Gebäudeautomation – Methodik und Management. GfSE Tag des Systems Engineering, Bremen 12-13.11.2014, http://enso-managers.de/files/resources/enso-m/documents-de/TdSE-2014_Dungern_Uebergreifende-Systemkonzeption_(Text).pdf

[7] A. Fisher, M. Nolan, S. Friedenthal, M. Loeffler, M. Sampson, M. Bajaj, L. Van Zandt, K. Hovey, J. Palmer, L. Hart: Model Lifecycle Management for MBSE, http://www.omgwiki.org/MBSE/lib/exe/fetch.php?media=mbse:model_lifecycle_management_for_mbse_v4.pdf

[8] M. Eigner, R. Stelzer: Product Lifecycle Management: Ein Leitfaden für Product Development und Life Cycle Management. ISBN 978-3-540-68401-5. Springer Verlag, 2009.

Abbreviations

ALM	Application Lifecycle Management
BPMN	Business Process Model and Notation
CAD	Computer-Aided Design
CM	Configuration Management
CPO	Code of PLM Openness (ProSTEP iViP e.V.)
DMS	Document Management System
ECAD	Electrical Computer-Aided Design
EDA	Electronic Design Automat
EDM	Engineering Data Management
ERP	Enterprise Resource Planning
FMC	Fundamental Modeling Concept
FMEA	Failure Mode Effect Analysis
FTA	Fault Tree Analysis
IoT	Internet of Things
PDM	Product Data Management
PLM	Product-Lifecycle-Management
MBE	Model Based Engineering
MBSE	Model Based Systems Engineering
MCAD	mechanical Computer-Aided Design
OSLC	Open Services for Lifecycle Collaboration
RE	Requirements Engineering
SE	Systems Engineering
SysML	System Modeling Language
TDM	Team Data Management

About

The authors of this position paper are members of the GfSE/INCOSE working group PLM4MBSE, chartered by the Gesellschaft für Systems Engineering e.V. (GfSE), the German chapter of the International Council of Systems Engineering (INCOSE). The group can be contacted by E-Mail via **plm4mbse@gfse.de** and is awaiting comments to this position paper.

http://gfse.de/arbeitsgruppen-mainmenu-85/plm4mbse.html

Contributors / Authors

Alexander Adam	Siemens AG
Bastian Binder	ITI
Lukas Bretz	Fraunhofer IPT-EM
Marco DiMaio	projectglobe.com
Oskar von Dungern	enso managers GmbH
Yousef Hooshmand	University Duisburg-Essen
Uwe Kaufmann	ModelAlchemy Consulting
Christian Muggeo	VPE at Technical University of Kaiserslautern
Florian Munker	IPEK – Institute of Product Engineering at Karlsruhe Institute of Technology (KIT)
Michael Pfenning	XPLM
Siegmund Priglinger	dr.priglinger consulting GmbH
Philipp Pribbernow	T-Systems International GmbH
Andre Scholl	Consultant
Ralf Schuler	University of applied science Esslingen
Tim Weilkiens	oose Innovative Informatik
Robert Woll	Fraunhofer IPK